

# МИКРО- ПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ

6 | 1988

ISSN 0233-4844

**Автономное устройство подготовки и отладки программ — РАСТР** на основе микропроцессорного комплекта БИС КР580 для задач разработки и проектирования прикладного программного обеспечения однокристального цифрового процессора КМ1813ВЕ1

**Интегрированная графическая система декларативного программирования** позволяет описывать асинхронные процессы с использованием графического представления входной и выходной информации

**Сколько стоит «бесплатный» программный продукт:** свойства информации как товара особого рода

**Стандарты 896, 1014, 1196, 1296** — 32-разрядные системные магистрали для объединительных панелей функционально - модульных систем

**Измерители частотно-временных параметров сигналов на базе ОЭВМ К1816**



## «МАЭСТРО»

— программное средство для специалистов, работающих в области рекламы, разработки графического интерфейса, системных и прикладных (в том числе учебных и игровых) программ на компьютерах MSX и MSX2.



## КОМПЛЕКС АВТОМАТИЗИРОВАННОЙ ОЦЕНКИ УРОВНЯ ФИЗИЧЕСКОЙ ПОДГОТОВЛЕННОСТИ ЧЕЛОВЕКА НА БАЗЕ ПЭВМ «МИКРОША»

Методологической основой комплекса является анализ показателей, характеризующих антропометрические данные человека и состояние его сердечно-сосудистой системы.

Уровень физической подготовленности оценивается компьютерной диагностической системой по суммированию баллов, полученных от измерения: массы тела (ее отклонения от оптимальной), степени стрессового напряжения (индекс напряжения по Р. М. Баевскому), ортостатической пробы, функциональной пробы (укороченная проба С. П. Летунова), времени восстановления частоты сердечных сокращений,

максимального потребления кислорода.

Итоговая оценка от 1 до 5 баллов дифференцируется в зависимости от пола и возраста. Результаты автоматизированной диагностики отображаются на экране видеотерминала (телевизора) в реальном времени.

Комплекс может быть использован в медицине (массовой скрининг, паспортизация данных о здоровье, физиологические исследования, оценка уровня физической подготовленности), физической культуре (индивидуальный тренинг с использованием биологической обратной связи, учебно-педагогический контроль).

Ориентировочная стоимость комплекса — 6,0 тыс. руб.

Водоватов Ф. Ф.

ОРГАН  
ГОСУДАРСТВЕННОГО  
КОМИТЕТА СССР  
ПО ВЫЧИСЛИТЕЛЬНОЙ  
ТЕХНИКЕ  
И ИНФОРМАТИКЕ

Издается с 1984 года

# ММП МИКРО ПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ

ВЫХОДИТ ШЕСТЬ РАЗ В ГОД

НАУЧНО-ТЕХНИЧЕСКИЙ И ПРОИЗВОДСТВЕННЫЙ ЖУРНАЛ 6 / 1988 МОСКВА

## СОДЕРЖАНИЕ

Публицистика,  
философия, экономика,  
компьютеризация  
МИКРОПРОЦЕССОРНАЯ  
ТЕХНИКА

ПЕРСОНАЛЬНЫЕ  
КОМПЬЮТЕРЫ  
ПРОГРАММНОЕ  
ОБЕСПЕЧЕНИЕ  
Языковые средства  
микроЭВМ

Машинная графика

ПРИМЕНЕНИЕ  
МИКРОПРОЦЕССОРНЫХ  
СРЕДСТВ

Устройства связи  
микроЭВМ  
с объектами

Медицинские  
приложения

Справочная  
информация

Черкасский В. С., Черных Д. В.—К экономическому механизму информационной сферы народного хозяйства . . . . .	3
Беляков Б. Н.—Анализ и синтез экономики и информатики (отклик) . . . . .	5
Шереметьевский Н. Н., Долкарт В. М.—Стандартизация 32-разрядных функционально-модульных систем для промышленных применений . . . . .	8
Артюхов В. Г., Глухенький В. Б., Лернер Р. А., Макеенок А. Н., Носенко Л. В., Страутманис Г. Ф.—Цифровой процессор KM1813BE1: подготовка и отладка программ . . . . .	14
Лопатин В. С., Матвеев В. М., Юрочкин А. Г.—МикроЭВМ MC 0102 с кеш-ОЗУ . . . . .	16
Пройдаков Э. М.—Второе поколение персональных ЭВМ фирмы IBM . . . . .	18
Муттик И. Г., Самарин Н. А.—Один полезный алгоритм . . . . .	23
Пройдаков Э. М.—Использование макроопределений в языке ПЛ/М . . . . .	25
Таюпов А. Р.—Методы программирования аппаратуры КАМАК на языке Паскаль . . . . .	27
Власенко В. А., Коханов А. Б.—Получение твердых копий графических и полутоновых изображений . . . . .	29
Тепляков Л. Г.—Программа получения твердой копии графического дисплея ДВКЗМ . . . . .	31
Адушкин Л. А., Богданов М. В., Земцов А. И., Хамидулин Ю. М.—Интегрированная графическая система декларативного программирования . . . . .	33
Кацман В. Е., Саталкин А. В.—Система визуального вывода полутоновых изображений на базе ПЭВМ «Искра 226» . . . . .	34
Шпанько В. И.—Драйвер графического дисплея для ДВКЗ . . . . .	35
Лигская А. В.—Проблемы защиты авторских прав на программное обеспечение в США . . . . .	39
Жуков Ю. И., Кармалин П. Н., Боровиков А. В.—Программирование ПЗУ в модуле 15УЗПП-16КХ16 в конструктиве «Электроника 60» под ОС РАФОС . . . . .	44
Чернявский В. Е., Маханек М. М.—Устройства приоритета в базе ПЛМ — компактность и быстродействие . . . . .	45
Михуткин А. В.—Контроллер KP580BH59 в устройствах приоритетного прерывания микроЭВМ «Электроника 60» . . . . .	48
Банников С. Ю., Марфенко К. С., Подлепецкий Б. И., Торубаров С. В.—Программно-аппаратный комплекс КПС-1813 для проектирования биомедицинских устройств с цифровым процессором сигналов KM1813BE1 . . . . .	55
Азин С. Н., Муравьев Н. И.—Модуль ввода-вывода видеосигнала в память микроЭВМ «Электроника 60» . . . . .	62
Ройзен М. И., Карпушов А. Н., Бургасов А. Ю., Белый А. Г.—Система организации прямого цифрового управления на базе микроЭВМ «Электроника 60» . . . . .	66
Груздев А. В., Федонов Е. Н.—Устройство ввода-вывода аналоговых сигналов микроЭВМ . . . . .	71
Водоватов Ф. Ф., Панов В. В., Сорокин А. В.—Применение персонального компьютера в массовой диагностике и тренинге ритма сердца . . . . .	76
Барановский А. Ю., Неверова А. Ю., Филинов В. Н., Цветкова Т. Л.—Программное обеспечение прогноза течения и исходов язвенной болезни желудка . . . . .	78
Глушченко К. А., Кирианки Н. В., Котыло О. Б., Юриш С. Ю.—Измерители частотно-временных параметров сигналов на базе ОЭВМ серии K1816 . . . . .	80
Белов Л. А., Бухаров Д. Г., Демидов В. М., Фомичев М. Ю.—Широкодиапазонный микропроцессорный частотомер . . . . .	82
Интерфейсные БИС микропроцессорного комплекта K1801: K1801BP1-034 . . . . .	85

**Редакционная коллегия**

**А. Г. Алексенко**  
**В. М. Брябри**  
**А. А. Васенков**  
(зам. главного редактора)  
**И. В. Вельбицкий**  
**А. Б. Венгеров**  
**Г. Р. Громов**  
**В. П. Иванников**  
**М. Б. Игнатев**  
**А. В. Каляев**  
**И. З. Карась**  
**В. П. Куприянов**  
**С. С. Лавров**  
**В. В. Липаев**  
**К. А. Меликян, И. А. Мизин**  
**С. М. Пеленов**  
(зам. главного редактора)  
**А. К. Платонов**  
**А. А. Попов**  
**Д. А. Поспелов**  
**Б. И. Рамсев**  
**О. Л. Смирнов**  
**А. Л. Стегний**  
**М. К. Сулим**  
**Н. М. Шаруненко**

**Редакционный совет:**

**И. В. Бабынин**  
**С. Н. Бушев**  
**Е. П. Велихов**  
**Н. Н. Говоруи**  
**В. В. Корчагин**  
**В. П. Макаревич**  
**А. Р. Назарьян**  
**Ю. Е. Нестерихин**  
**А. Л. Нефедкин**  
**И. В. Прангишвили**  
**Л. Н. Преснухин**  
**В. В. Пржиялковский**  
**Н. Л. Прохоров, Г. Г. Рябов**  
**В. И. Хохлов**  
**Н. Н. Шереметьевский**  
**В. В. Шильдин, А. В. Яковлев**  
**Э. А. Якубайтис**

**Номер подготовили:**

**Е. И. Бабич, Г. Г. Глушкова,**  
**В. М. Ларионова, С. С. Матвеев**

**Технический редактор**

**Г. И. Колосова**

**Корректор Е. М. Кучерявенко**

**Адрес редакции журнала:**

**103051, Москва, Малый Сухареvский пер., д. 9А**  
**Телефоны: 208-73-23, 208-19-94**

Сдано в набор 30.10.88. Т-18577.  
Подписано к печати 14.12.88.  
Формат 84X108/16. Бумага № 2.  
Высокая печать. Усл. печ. л. 10,08.  
Уч.-изд. л. 14,6. Тираж 100 660.  
Заказ № 236. Цена 1 р. 10 к.  
Орган Государственного комитета СССР по вычислительной технике и информатике  
Московская типография № 13  
ПО «Периодика» ВО «Союзполиграфпром» Госкомиздата СССР  
107005, Москва, Денисовский пер., дом 30

**Рисунки на первой странице обложки и развороте вкладки — художницы Е. Родионовой, слайды — О. Гаспарянца**



**АКАДЕМИК**

**АНДРЕЙ**

**ПЕТРОВИЧ**

**ЕРШОВ**

Эта фотография впервые была опубликована пять лет назад в первом номере «МП» за 1984 год рядом с обращением главного редактора к читателям вновь созданного журнала:

**Уважаемый читатель!**

*В Ваших руках первый выпуск нового журнала Государственного комитета СССР по науке и технике «Микропроцессорные средства и системы». Организация этого журнала является выражением той огромной роли, которую призвана сыграть микропроцессорная техника в научно-техническом прогрессе ближайших десятилетий...*

*Появление микропроцессорных средств радикально изменило характер использования и внедрения вычислительной техники. Главное здесь — массовость и необычайная широта применения.*

*Редакционный совет, редакционная коллегия и сотрудники редакции хотят сделать журнал полезным каждому, кто связан с разработкой и применением микропроцессорных средств и систем. Задача состоит в том, чтобы реальные научно-технические достижения и удачные новшества сделать достоянием как можно большего числа разработчиков и потребителей. Мы хотим, чтобы журнал давал не только общую ориентацию в направлениях развития микроэлектронной вычислительной техники и ее компонент, но и был бы полезен в повседневной практической работе...*

Первый и пока единственный в стране массовый журнал, практически необходимый более чем 100 тысячам специалистов в области микропроцессоров, микроЭВМ и программирования — это лишь один из многих социально значимых плодов многогранной подвижной научно-педагогической и общественной деятельности А. П. Ершова.

Андрей Петрович Ершов родился 19 апреля 1931 года в семье служащих. После окончания аспирантуры механико-математического факультета Московского университета в 1957 году возглавил отдел теоретического программирования Вычислительного центра АН СССР. С 1959 года он работает в Сибирском отделении Академии наук СССР сначала в Институте математики, а с 1964 года — в Вычислительном центре СО АН СССР. С апреля 1987 года А. П. Ершов был председателем Научного совета АН СССР по комплексной проблеме «Кибернетика».

А. П. Ершов — ученый с мировым именем и широким диапазоном научных интересов, внесший большой вклад в фундаментальные и прикладные исследования в области теоретического, системного и программного программирования. Ему принадлежит концепция всеобщей компьютерной грамотности и научное обоснование государственной программы компьютеризации общества.

Мы надеемся, что традиции научной добротности и практической полезности, заложенные А. П. Ершовым в основу первых пяти лет развития журнала «МП», и в будущем останутся прочным основанием для его развития.

Вместе с Вами, читатели, мы склоняем голову перед портретом человека, который успел так много сделать для своей страны, своего народа, и многого не успел из-за необъяснимо скупо отмеренного ему времени...

УДК 681.3.06

В. С. Черкасский, Д. В. Черных

## К ЭКОНОМИЧЕСКОМУ МЕХАНИЗМУ ИНФОРМАЦИОННОЙ СФЕРЫ НАРОДНОГО ХОЗЯЙСТВА

Статья Б. Н. Белякова и его коллег [5] заслуживает одобрения прежде всего за привлечение внимания к экономическим проблемам информатики. Конечно, такие основополагающие утверждения авторов, как плодотворность объяснения известного (законов товарного производства) через неизвестное (законы информационной производства), наличие «опосредованно-общественной» формы собственности или возможность математического выражения потребительной стоимости, не выдерживают критики. Наименее приемлемой представляется попытка выяснить производственные отношения в информационной сфере, не выходя за ее рамки. В настоящей статье решение актуальных экономических проблем информационной сферы ищется исходя из общей полиэкономической модели товарного производства.

**1. Информационное производство и его производительные силы.** Информационной сферой народного хозяйства мы называем совокупность деятельностей по производству, распределению и обмену информацией. Основным потребителем информационной продукции является сфера материального производства (см. рисунок), которая, в свою очередь, обеспечивает информационную сферу техникой и материальными носителями информации. Аналогичная кооперация имеет место и в сфере образования, которая обеспечивает народное хозяйство квалифицированной рабочей силой.

Обособление информационной сферы в составе народного хозяйства — результат современного этапа развития производительных сил материального производства, для которого обновление продукции, технологии и производственного аппарата (в том числе путем автоматизации) стало постоянным элементом процесса воспроизводства и, следовательно, научно-техническая информация является неотъемлемым средством труда [6].

Разумеется, материальное производство потребляет информационные продукты лишь в силу их полезности. Полезность информации обусловлена, во-первых, ее новизной. Это не совсем тавтология: с момента обособления производства информации возникает принципиальная возможность выпуска таких информационных продуктов, которые не содержат информации для потребителей. Во-вторых, удобной для потребления товарной формой. И, в-третьих (в главных), той экономией живого и прошлого труда, которая может быть получена в производстве при производительном потреблении информации\*. Эта экономия и является оправданием существования и развития информационной сферы, хотя значительная часть производимой информации потребляется в ней самой.

Специфика производительных сил самого информационного производства (в отличие от материального) заключена не в субъекте и даже не в средствах труда. Производством информации заняты, как правило, кадры высшей квалификации, но то же имеет место и в наукоемких отраслях материального производства: без этого они не смогли бы потреблять ин-

\* Закон экономии труда — это всеобщий закон развития производительных сил [4]. Любое обновление производства осуществляется для экономии труда. Поэтому мы вправе считать, что применение информации в производстве экономит труд даже в тех случаях, когда мы не умеем подсчитать эту экономию. Другое дело, что для реализации возможной экономии одной информации недостаточно.



Место информационной сферы в народном хозяйстве

формацию. Средства труда как в информационном, так и в материальном производстве представляют собой сочетания технических и информационных средств. Мы не видим оснований считать, что различия в применяемых средствах труда выходят за рамки различных видов конкретного труда. Таким образом, специфика заключена в предмете, процессе и продукте труда.

«...в процессе труда деятельность человека при помощи средства труда вызывает заранее намеченное изменение предмета труда» [1, с. 191]. Так в материальном производстве. Предмет труда, утрачивая свою материальную форму, переходит в продукт. Но информация как предмет труда не теряет ни своего содержания, ни формы и может использоваться неоднократно, т. е. идет себя как средство труда. Информация как средство труда, в свою очередь, отличается тем, что не подвержена физическому износу (в отличие от ее материальных носителей). Причина — в специфике процесса труда: информация воспринимается либо человеком, либо специальными техническими средствами (ЭВМ), но в последнем случае предварительно переносится на специальные носители.

Наконец, информация как продукт труда отличается свойством тиражируемости. Если для производства очередного экземпляра материального продукта нужно приложить (при прочих равных условиях) тот же конкретный труд, что и для первого экземпляра, то тиражирование информационного продукта требует совсем не той деятельности, нежели его создание. Процесс тиражирования, между прочим, зависит не от содержания информации, а от типа материальных носителей.

Специфика производительных сил информационной сферы отражается на производственных отношениях.

## 2. Парадокс бесплатности

Бурный рост информационной сферы и ее влияния на материальное производство (обусловленный прежде всего автоматизацией информационного производства на базе ЭВМ [6]) до последнего времени не находил у нас адекватного отражения в системе производственных отношений в этой сфере. По традиции считалось, что только продукт материального производства может быть товаром. Информационная сфера, как и сфера образования, относилась (наряду с здравоохранением и культурой) к так называемой непродуцирующей сфере, в рамках которой товарные отношения существовать не могут. Информационный продукт рассматривался как бесплатный (за исключением затрат на тиражирование) и принадлежащий всему обществу в целом. В информационной сфере возникло классическое противоречие между уровнем развития производительных сил и характером производственных отношений, выразившееся в ряде парадоксальных явлений, в конечном счете приведших к торможению развития производительных сил как в самой информационной сфере, так и в материальном производстве.

Применительно к такой прогрессивной отрасли информационной сферы, как производство программного продукта, парадокс состоял в том, что значительная (может быть, подавляющая) часть прикладных программистов работали, по сути дела, входостую. Поскольку программа — не товар, и продать ее невозможно, программы создавались, как правило, лишь для нужд тех предприятий и организаций, где работают программисты, и очень редко доходили до стадии программного продукта. Обмен программами по этой же причине ограничивался натуральной формой, при которой, как известно, «...каждый товаровладелец хочет сбыть свой товар лишь в обмен на такие товары, потребительная стоимость которых удовлетворяет его потребности» [1, с. 96]. В результате прикладные программисты в большинстве случаев вместо создания

новых программ тратят силы либо на разработку программ, где-то уже созданных, но им не доступных, либо на поиск таких владельцев нужных им программ, которых можно заинтересовать в натуральном обмене, либо, наконец, на освоение полученных по обмену программ, не имеющих товарной формы (о различии программы и программного продукта см. [7, с. 11—12]).

Второй, более серьезный парадокс связан с проблемой информационных ресурсов, подробно рассмотренной Г. Р. Грозовым в [6]. Смысл термина «ресурсы» применительно к информации, в частности, в том, что для широкого использования в народном хозяйстве эти ресурсы должны быть предварительно «освоены». Иначе говоря, потенциальный потребитель должен быстро и дешево получить не только справку о наличии в составе информационных ресурсов нужного ему продукта, но и сам этот продукт. Сегодня это уже невозможно сделать без создания автоматизированных баз и банков данных (а в ближайшем будущем — баз знаний). Но, затраты на их создание (не считая даже стоимости технического и программного обеспечения) сопоставимы с затратами на освоение природных ресурсов. А так как окупить эти затраты продажей информационных услуг нет возможности, то баз данных общего пользования в нашей стране — считанное число, они только библиографические и охватывают весьма узкий круг источников. В результате не только ранее накопленная, но и вновь создаваемая информация в значительной мере пополняет неосвоенную часть информационных ресурсов «залежи»: бесплатно — получается себе дороже.

В принципе возникшее противоречие должно найти свое разрешение в ходе идущей в стране радикальной экономической реформы. В постановлении ЦК КПСС и Совета Министров СССР «О переводе научных организаций на полный хозяйственный расчет и самофинансирование» [3] указано, что «...научные организации наряду с производственными предприятиями являются социалистическими товаропроизводителями...». Но на основе этого принципиального решения еще предстоит создать эффективный экономический механизм информационной сферы. Требуют теоретической проработки прежде всего проблемы отношений собственности на информационный продукт, ценообразования на него и оплаты труда его производителей.

### 3. Собственность на информацию

В общественном мнении укоренилось представление о собственности при социализме как только общественной категории, поэтому сошлемся на авторитетный источник. В законе о предприятии [2, с. 4—5] недвусмысленно указано, что предприятие «обладает обособленной частью общественной собственности» (на средства производства — прим. авт.) и использует ее «как хозяин». Собственность на средства производства по законам политэкономии влечет за собой и собственность на продукцию. А так как действие закона о предприятии распространяется, согласно [3], на научные организации, следовательно, право собственности на информационный продукт принадлежит организации, которая его произвела. Это относится и к программному продукту\*, несмотря на то, что программа, входящая в его состав, обычно имеет индивидуального автора (работающего в этой организации). Здесь мы солидаризируемся с Ф. П. Бруксом мл. [7, с. 12].

Однако признания права собственности еще не достаточно, необходим юридический механизм его охраны. А в случае информационного продукта этого сделать не просто из-за его тиражируемости. Практиче-

скую сложность этой проблемы хорошо иллюстрирует статья А. И. Ракитова и Ю. М. Батурина [8], где дав также обзор состояния дел в этом вопросе за рубежом.

Несанкционированное копирование программного продукта, пример которого приводится в [8], действительно не подпадает под понятие хищения, но, тем не менее, наносит производителю ущерб, равный стоимости реализации этого продукта в порядке тиражирования. Юридическую основу для возмещения этого ущерба, на наш взгляд, можно было бы обеспечить по аналогии с международным патентно-лицензионным правом. Иначе говоря, следует признать право собственности на информационный продукт, включая право тиражирования, за его производителем, а за отдельным потребителем признать лишь лицензионное право использования. Это, конечно, означает, что производитель вправе не только сам тиражировать созданный им информационный продукт, но и уступать это право (полностью или в ограниченном размере) другим организациям. В случае же производства информационного продукта по индивидуальному заказу заказчик может обусловить в договоре передачу ему права тиражирования с компенсацией ожидаемого ущерба производителя.

Для реализации своего права собственности организации-производителю, видимо, будет необходим соответствующий охраняющий документ, аналогичный патенту. Для программных продуктов таким документом может стать регистрационное удостоверение, выдаваемое ВФАП. Но необходимо оговорить, что регистрация сама по себе не означает передачи ВФАП права тиражирования, как это понимается сегодня.

Что же до авторских прав создателей отдельных алгоритмов и программ, их экономический смысл, по-видимому, должен быть аналогичен смыслу современных авторских свидетельств на изобретения: право на получение вознаграждения при использовании. При этом факт использования должен отражаться в регистрационном удостоверении на программный продукт.

Изложенная концепция собственности на информацию представляется соответствующей современному этапу развития информационной сферы, но ее реализация требует еще большой работы юристов, экономистов и специалистов по информатике.

### 4. Стоимость информации

Поскольку информация — товар, ее производство есть одновременно производством стоимости, а ее производительное потребление, безразлично — в сфере материального производства или в самой информационной сфере, есть перенос ее стоимости на продукт. До настоящего времени на эту сторону дела практически не обращали внимания. Необходимую информацию потребители получали либо бесплатно (через сеть государственных информационных служб, финансируемых из госбюджета), либо с оплатой лишь издержек тиражирования, а в тех случаях, когда информационный продукт создавался по индивидуальному заказу, — с оплатой лишь прямых издержек производства, да и то далеко не всех. Между тем, судя по зарубежным аналогам [6], стоимость информационной продукции весьма велика. С учетом стоимости потребляемой информации издержки производства продукции «научоемких» отраслей материального производства могут вырасти в два и более раз по сравнению с сегодняшним уровнем. Это обстоятельство необходимо учесть в ходе проводящейся перестройки системы цен на промышленную продукцию.

Стоимость потребленной информации должна учитываться также в издержках производства самих информационных продуктов. Определенную специфичность здесь имеет создание автоматизированных баз и банков данных. Поскольку стоимость неразрывно связана с потребительной стоимостью, товар, не имеющий последней, не имеет и первой. Неформализованная информация, образующая «ресурсы», вообще говоря, не имеет потребительной стоимости, ибо потребителям практически недоступна — в этом и состоит сущность информационного кризиса. Поэтому она не имеет и стоимости. Зато затраты на извлечение этой информации из «ресурсов» для включения в базу данных огромны, ибо вся работа ведется вручную.

С другой стороны, при создании экспертных систем крупным источником информации являются знания экспертов. Этот товар как раз обладает и потребительной стоимостью, и стоимостью, и в [8] правильно ставится вопрос об оплате не только услуг экспертов, но и передаваемых ими знаний. Ведь в этом случае эксперт фактически передает системе право тиражирования своих знаний, и при этом сам теряет эту воз-

\* Конечно, не исключается производство программ (особенно для персональных компьютеров) в порядке индивидуальной трудовой деятельности.

возможность. Между прочим, такая же ситуация будет в определенных случаях иметь место и при создании индивидуальных баз данных на персональных компьютерах, которые Г. Р. Громов в [6] удачно назвал средствами автоматизации профессиональных знаний. Хотя пользователь ПК и не имеет в виду превращение формализованных им знаний в товар, но при несанкционированном копировании он потерпит ущерб, который нужно оценивать стоимостью этих знаний (страховой ценой).

Независимо от того покрывается информационный товар в самой информационной сфере или в материальном производстве, потребляется он как средство или предмет труда, стоимость его переносится из продукта в общем случае не полностью, ибо, как отмечалось в п. 1, один и тот же информационный продукт может использоваться в производстве продукции неоднократно, и срок его использования ограничен лишь моральным износом. Необходимость практического учета стоимости информации в издержках производства продукции, следовательно, ставит задачу разработки своеобразных норм амортизации (морального старения) информации, для чего, в свою очередь, необходимо предварительно разработать классификатор видов информационных продуктов.

Переходя к определению стоимости информационного продукта, необходимо прежде всего указать, что она не может быть определена через рынок. В отличие, скажем, от сапога (типичный модельный объект К. Маркса в [1]) информация индивидуальна: различные информационные продукты удовлетворяют различные потребности, поэтому средняя цена реализации здесь смысла не имеет. Стоимость информации следует определять через более общую категорию — общественно необходимые затраты труда (ОНЗТ). Эти последние, в свою очередь, могут быть оценены с двух сторон.

Если потребность в данном информационном товаре налична (и, следовательно, он имеет потребительскую стоимость), ОНЗТ на этот товар определяются издержками его расширенного производства. На языке конкретной экономики, цена информации должна обеспечивать уровень рентабельности, достаточный для самофинансирования научно-технического и социального развития организации-производителя. Эта цена, вместе с тем, гораздо выше той, которую привыкли платить заказчики по договорам за создание информационных продуктов. Помимо прямых издержек производства, обычно учитываемых в сметах, она включает амортизацию основных фондов, стоимость потребленной информации (также в форме амортизационных отчислений, о чем говорилось выше), нормативную прибыль (которая и используется на самофинансирование), а также затраты, учитывающие связь производства данного продукта с производством всех остальных продуктов. Природу этих затрат для продуктов материального производства подробно исследовал В. В. Новожилов в [4]. В осуществляемой экономической реформе эти затраты нашли отражение в форме платы за фонды, трудовые и природные ресурсы [2].

Описанная оценка ОНЗТ является минимальной. Продажа информационного товара по цене ниже минимальных ОНЗТ нецелесообразна не только для производителя, но и для общества, ибо не обеспечивает производителю условий самофинансирования и тем самым вынуждает общество финансировать его развитие. Что, собственно, и имело место.

Максимальная оценка ОНЗТ определяется той экономией труда, которую обеспечивает информационный продукт при его производственном потреблении и которая, как отмечалось в начале настоящей статьи, является оправданием информационной сферы вообще. Оплата информационных продуктов по цене выше максимальных ОНЗТ убыточна для потребителей, следовательно, нецелесообразна для общества, то есть затраты труда выше максимальных ОНЗТ не являются общественно необходимыми. Как видим, максимальные ОНЗТ не зависят от минимальных и могут быть как намного выше, так и равны и даже ниже них. Последний случай означает, что на информационный товар отсутствует платежеспособный спрос, хотя и есть потребность, следовательно, его производство нецелесообразно, пока не будет найден способ снизить издержки производства.

Наличие двух методологически различных и численно несопоставимых оценок ОНЗТ представляет значительный теоретический интерес, потому что на первый взгляд противоречит трудовой теории стоимости. Оно имеет место не только для информационных продуктов, но и для наукоемких продуктов материального производства, прежде всего так называемой «новой техники». На наш взгляд, причиной является то, что в производстве таких продуктов целенаправленно применяется специфическая производительная сила — наука, в том числе информатика. Методом практического ценообразования на информационные продукты, учитывающим существование двух оценок ОНЗТ, является договорная цена, что и отражено в [3].

Заканчивая этот весьма предварительный обзор проблематики стоимости информации, считаем необходимым отметить следующие практические выводы из факта существования двух оценок ОНЗТ для информационных продуктов. Во-первых, строго говоря, максимальная оценка существует лишь для тех информационных продуктов, которые потребляются в сфере материального производства либо в самой информационной сфере, но не в сфере образования. Так, программы для школьных компьютеров имеют потребительскую стоимость с момента принятия решения о преподавании основ информатики в школе, и их цены должны формироваться на основе минимальных ОНЗТ. Во-вторых, даже в производственной сфере максимальную ОНЗТ не всегда можно получить. Примером может служить такой типичный информационный продукт, как справка, выданная автоматизированной базой дан-

ных. И, наконец, как максимальные, так и минимальные ОНЗТ представляют собой оценки, ибо определяются не в результате обмена уже произведенных товаров, а до их производства.

Вывод. Последовательное использование опыта хозяйственных производственных отношений, накопленного в сфере материального производства (особенно производства средств производства), оказывается достаточно плодотворным для решения экономических проблем информационной сферы, которые лишь на взгляд авторов [5] представляются принципиально новыми и требующими пересмотра всех категорий политической экономии.

Телефон 32-25-24, Новосибирск

## ЛИТЕРАТУРА

1. Маркс К. Капитал. Т. 1.— М.: Изд. полит. лит-ры, 1983.— 905 с.
2. Закон СССР о государственном предприятии (объединении).— М.: Известия, 1987.— 64 с.
3. О переводе научных организаций на полный хозяйственный расчет и самофинансирование. Постановление ЦК КПСС и Совета Министров СССР // Экономическая газета.— 1987.— № 42.— С. 2, 13.
4. Новожилов В. В. Проблема измерения затрат и результатов при оптимальном планировании.— М.: Экономика, 1967.— 376 с.
5. Беляков Б. Н., Зюляркин Д. Д., Подвальный Л. М., Чернышев В. А. Производительные силы и производственные отношения в сфере производства программного продукта // Микропроцессорные средства и системы.— 1987.— № 2.— С. 3—6.
6. Громов Г. Р. Национальные информационные ресурсы.— М.: Наука, 1984.— 237 с.
7. Брукс мл. Ф. П. Как проектируются и создаются программные комплексы (Мифической человеко-месяц).— М.: Наука, 1979.— 150 с.
8. Ракилов А. И., Батуринов Ю. М. Компьютер перед лицом закона // В бюлл. ИТР — проблемы и решения.— 1987.— № 19 (58).

Статья поступила 18.01.88

## ОТКЛИК

Б. Н. Беляков

## АНАЛИЗ И СИНТЕЗ ЭКОНОМИКИ И ИНФОРМАТИКИ

Статья В. С. Черкасского и Д. В. Черных посвящена, безусловно, актуальной теме, обсуждение которой на страницах данного журнала нужно только приветствовать. Экономика информатики — комплексная проблема, находящаяся на стыке наук: информатики, теории информации, экономики, политической экономии, социологии и, возможно, психологии и др. Поэтому она требует комплексного рассмотрения, глубокого синтеза знаний, в первую очередь — синтеза категорий теории стоимости и категорий теории информации.

Это означает, что теория экономики информатики должна содержать в себе анализ и синтез двух углов зрения. Первый состоит в анализе товарного производства как некоего информационного процесса, в котором носителем информации является товар, и в процессе обмена товарами происходит не только обмен стоимостями, но и информацией; второй — в противоположном подходе: информационный обмен рассматривается как товарный, а информационное производство как товарное. Синтез этих двух подходов и должен дать то, что мы сейчас пытаемся обозначить как экономико-информатику.

В дискуссиях на страницах журналов по поводу экономических проблем информатики многие авторы выступали преимущественно с позиций одного из отме-

ченных подходов. Этот этап научного обсуждения проблемы был необходим и дал свои результаты, которые позволяют двинуться дальше: от анализа к синтезу, от дифференциации к интеграции знаний о информатике и экономике.

Большого от этого первого этапа научного обсуждения, по-видимому, ожидать не следует. Он свои задачи выполнил. Этому есть свои приметы: все чаще в статьях по экономическим проблемам информатики встречаются повторения ранее сказанного, взаимные ссылки друг на друга. Дискуссия начинает гложуть и терять научную продуктивность.

Пора от одностороннего анализа информационного производства как товарного обратить внимание и на само товарное производство как информационное (например, попытаться определить, почему, несмотря ни на что, продолжает существовать и развиваться так называемое символическое производство и потребление, когда вещь производится и потребляется не ради ее материальных потребительских свойств, а ради символов, которые за ней закреплены), а затем попытаться сопоставить эти два подхода. Экономика информатики не может развиваться иначе, чем путем одновременного и взаимного развития как экономической теории, так и теории информатики.

В. С. Черкасский и Д. В. Черных, к сожалению, предлагают уже достаточно исчерпанный и ставший односторонним подход «исходя из общей (по их выражению) политэкономической модели товарного производства», отбрасывая информационные аспекты самого товарного производства, так сказать, информационную модель товарного производства.

В первой части «Информационное производство и его производительные силы» авторы заостряют внимание на «специфике производительных сил информационной сферы». Безусловно, чтобы проанализировать сущность и состав производительных сил информационной сферы, необходимо проанализировать и их специфику. Но не только. Так мы решим лишь задачи дифференциации наук, их размежевания. Задачи же интеграции, синтеза, которые сейчас более актуальны, остаются за бортом. Анализ экономической специфики информационной сферы должен быть дополнен анализом информационной специфики: производительных сил материальной сферы для возможности последующего выхода на синтез представлений о производительных силах материальной сферы с представлениями о производительных силах информационной сферы. Иначе, под вопросом вся постановка проблем экономики информатики.

У авторов же присутствует только один подход, что характерно для большей части экономистов, занимающихся проблемами информатики. Как следствие — ошибочность выводов и односторонность тезисов. В частности, это выразилось в ряде положений их статьи.

С первых страниц идет речь об особенностях информационной сферы в составе народного хозяйства без упоминания об относительности этого обособления и необходимости более широкой интеграции информатики со сферами материального и духовного производства. В настоящее время именно вопросы интеграции информатики с народным хозяйством и образованием являются действительно актуальными, а не вопросы обособления информатики.

Обособление информационной сферы — это действительно объективный процесс и действительно вызван развитием производительных сил материального производства. Но упоминание о том, что для интеграции информационной сферы со сферами народного хозяйства, образования, культуры и науки существуют объективные причины, делает анализ несколько односторонним.

Кроме того, не только развитие производительных сил послужило предпосылкой обособления информационной сферы. Вполне определенное развитие про-

изводственных отношений, хозяйственного механизма, бюрократизация, ведомственность в управлении тоже оказали свое влияние на развитие и обособление информационной сферы. Так, например, вопросы обособления информатики решались с ведомственных позиций, что повлияло на распределение технических средств информатики по отраслям и регионам, на техническую политику развития вычислительных средств в пользу больших машин. Обособление осуществлялось не для развития информационной сферы, а с целью укрупнения и выделения ведомств, каждое из которых стремилось реализовать свою техническую политику, основываясь не на принципах экономической соревновательности, а на принципах бюрократического давления.

С другой стороны, процессы объективного выделения информационной сферы использовались и используются для обоснования необходимости ведомственного обособления в ущерб процессам ее интеграции с производством, культурой, наукой и образованием.

Таким образом, называть процесс обособления информационной сферы результатом современного этапа развития только производительных сил сферы материального производства в отрыве от развития производственных отношений и надстройки, в частности духовной сферы, мне представляется недостаточно полным. Можно было бы согласиться с такой позицией авторов, если бы они представили развитие производительных сил как развитие средств труда и рабочей силы (поскольку производительные силы — это совокупность средств труда, предметов труда и рабочей силы). Из их же утверждения следует представление о производительных силах только как о средствах: «развитие производительных сил материального производства, для которого обновление продукции, технологии и производственного аппарата (в том числе путем автоматизации) стало постоянным элементом производственного процесса...» без упоминания о рабочей силе и человеческом интеллекте как одной из основных составляющих производительных сил.

Современный этап развития производительных сил материального производства в его тесной связи с НТР поверхностно представляется процессом, происходящим только в средствах труда. Более правильной все же следует считать трактовку, согласно которой обособление информационной сферы в составе народного хозяйства объясняется результатом современного этапа развития интеллектуального потенциала материального производства, а затем развитием остальных его производительных сил. Иначе предложенный авторами подход выглядит излишне технократическим, некогда многообещающим, но, как оказалось, односторонним, нежизненным и уже потерявшим основную массу своих сторонников.

Хочется обратить внимание еще на некоторые неточности в первой части статьи. Они также вызваны настойчивым подчеркиванием только специфической стороны информационной производства.

Информация, по Черкасскому и Черных, — предмет труда и в то же время «ведет себя как средство труда». Общий итог — информация в информационной сфере предметом труда в традиционном понимании не является. Это подтверждается и рисунком. Согласно ему в информационную сферу входят только кадры и материальные средства производства, а выходит информация. Информационные же ресурсы, или, так сказать, информационное сырье в информационной сфере не потребляются, т. е. предмета труда в ней в виде информации нет. Это также следует из первого раздела: «Предмет труда, утрачивая свою материальную форму, переходит в продукт. Но информация как предмет труда не теряет ни своего содержания, ни формы и может использоваться неоднократно, т. е. ведет себя как средство труда».

Я не согласен с этой формулой. Информация дей-



ствительно имеет свою специфику в качестве предмета труда, но это не значит, что она не может быть предметом труда в традиционном понимании и терять при обработке материальной формы. Для информации материальной формой является ее информационная форма. Она существует в виде вполне материальных сигналов, которые при обработке многократно преобразуются в электромагнитные, электрические, электрохимические, механические и др. Это и есть изменение формы информации. Часть этих сигналов может быть использована только однократно и утратить свою материальную форму. Так, например, в АСУ ТП оперативная информация о состоянии техпроцесса используется однократно. Возьмем плавку металла. При повторении плавки поступает уже новая информация. Это типичный случай классического предмета труда. Информация здесь утрачивает свою электрохимическую форму и приобретает электромагнитную знаковую форму.

Правда, если оперативная информация, используемая в качестве предмета труда, накапливается в некий банк данных и корректируется последующей оперативной информацией, то этот банк данных может служить в качестве информационного средства труда и использоваться многократно. Это также типичный случай и для материального производства, когда, к примеру, машина во время ее ремонта является предметом труда, а во время использования — средством труда.

Не все, сказанное авторами, справедливо в отношении способности информации к тиражированию. Они утверждают: «Если для производства очередного экземпляра материального продукта нужно приложить (при прочих равных условиях) тот же конкретный труд, что и для первого экземпляра, то тиражирование информационного продукта требует совсем не той деятельности, нежели его создание».

Это не так. И в материальном производстве характерна ситуация (для массовой продукции), когда для первого экземпляра требуется совсем иная деятельность, нежели для последующих образцов. Это касается опытного производства и последующего перехода от опытного образца к массовому производству. Разница между себестоимостью опытного образца изделия и его себестоимостью в процессе массового производства достигает, как и в случае с информационным продуктом, сотен раз. В случае с тиражированием информации эта разница более типична. Но и только.

Точно такое же отличие в технологии изготовления опытного образца в лабораторных или опытных условиях от технологии тиражирования этого образца в условиях массового производства. Особенно, если это происходит в условиях ГАП или роторных линий. Эту разницу можно сравнить разве что с разработкой программ и последующим их тиражированием.

Второй и третий разделы статьи возражений не вызывают. Здесь помещены рассуждения, известные по публикациям в популярной и научной литературе, в том числе на страницах журнала «МП». Изложенная в третьем разделе концепция собственности на информацию общеизвестна и не является открытием авторов.

Ошибки, сделанные в первом разделе статьи, по сути не ошибки, а элементы логики авторов, необходимые для выхода на концепцию общественно необходимых затрат труда (ОНЗТ), сторонниками которой они, по-видимому, являются.

Одностороннее подчеркивание «специфики» информационного продукта и информационной сферы авторами делается насильственно для того, чтобы обосновать главный тезис своей статьи и концепции ОНЗТ. Он состоит в следующем: «Переходя к определению стоимости информационного продукта, необходимо прежде всего указать, что она не может быть определена через рынок», а должна определяться «через

более общую категорию: ОНЗТ», — читаем мы у авторов.

Здесь же В. С. Черкасский и Д. В. Черных обосновывают это положение совершенно, с нашей точки зрения, непонятно: «информация индивидуальна: различные информационные продукты удовлетворяют различные информационные потребности».

Что значит «информация индивидуальна»? Ну и что, если информационные продукты удовлетворяют разные потребности. Все продукты удовлетворяют разные потребности, а не только информационные продукты. Даже сапоги, когда они приобретаются для верховой езды, полевых работ, для зимы, осени, для танцевальных коллективов и, в конце концов, для спекуляции (кстати, типичным модельным объектом К. Маркса являются не только сапоги, а сукно, золото, железо, зерно, сюртук, холст и т. д.).

Опять надуманные поиски «специфики».

Отношение к концепции ОНЗТ можно выразить словами д. э. н. В. Гальперина (Коммунист, 1988, № 3, с. 74): «Бесконечно длящиеся дискуссии о природе ОНЗТ как непосредственной, ближайшей основе ценообразования — разве не уводят они нас в сторону от исследования законов рынка, механизма спроса и предложения, посредством которых только и возможно соизмерение затрат и результатов, а на этой основе и объективизации ценообразования — маскируют отсутствие научной теории цены».

Ошибочное представление информационного продукта как только средства труда приводит авторов к другому ошибочному и невыполнимому требованию «разработки своеобразных норм амортизации (морального старения) информации, для чего, в свою очередь, необходимо предварительно разработать классификатор видов информационных продуктов». Это предложение бессмысленно, в частности, для оперативной информации и информационных продуктов, срок жизни (морального старения) которых длится от нескольких секунд до года.

В заключение хочу обратиться к редакции журнала с просьбой продолжить обсуждение экономических проблем информатики.

*Телефон 39-93-90, Челябинск*

## Внимание: конкурс

### В ЛЮБОМ СЛУЧАЕ — БЕЗ ПРОИГРЫША!

Всесоюзный кооперативный центр по научно-технической информации «Спутник» предлагает желающим принять участие в конкурсе на лучшую разработку по товарам народного потребления. Принимаются заявки от учреждений, предприятий, кооперативов, центров НТТМ, самостоятельных коллективов и отдельных лиц.

Главное условие — наличие действующего образца и полной технической документации. Авторские свидетельства на изобретения, дипломы на промышленный образец и удостоверения на рацпредложения не обязательны. Не принимаются работы на серийно выпускаемые товары, а также документация на изделия, авторы которых заключили трудовые соглашения до 01.11.88 г. с кооперативом «Спутник».

Желающие принять участие в конкурсе должны прислать в Центр фотографии изделия, краткое описание, сведения о внедрении и наличии технической документации. По отобранным заявкам Центр вышлет авторам договоры (см. журнал ИП, 9, 88, с. 48). Заявки принимаются до 01.02.89, итоги конкурса по наибольшему числу запросов предприятий на любое из предложенных изделий подводятся 01.04.89. Победителей ждут: первая премия — 10 тыс. руб., две вторые — по 7 тыс., три третьи — по 5 тыс. и двадцать поощрительных по 1000 рублей. Материалы присылать по адресу: 246004, Гомель, ул. Рабочая, 22, Всесоюзный кооперативный центр «Спутник» с пометкой: «На конкурс».

УДК 681.326

Н. Н. Шереметьевский, В. М. Долкарт

## СТАНДАРТИЗАЦИЯ 32-РАЗРЯДНЫХ ФУНКЦИОНАЛЬНО-МОДУЛЬНЫХ СИСТЕМ ДЛЯ ПРОМЫШЛЕННЫХ ПРИМЕНЕНИЙ

Наша промышленность выпускает различные 8- и 16-разрядные управляющие вычислительные средства, которые, хотя и используют одни и те же микропроцессоры, но аппаратно и программно плохо совместимы между собой.

Учитывая предстоящий переход к 32-разрядным управляющим вычислительным средствам и широкое внедрение таких средств в промышленных автоматизированных системах, необходимо ускорить создание общесоюзного стандарта на управляющие вычислительные средства, обеспечивающего их полную совместимость. В основе такого стандарта должен лежать соответствующий международный стандарт. Это гарантирует высокое качество и широкое применение управляющих вычислительных средств, разработанных в соответствии со стандартом.

Полученный в 60-х и начале 70-х годов опыт создания сложных промышленных автоматизированных систем управления оборудованием и процессами показал необходимость разработки для них легко развиваемых специализированных заказных управляющих вычислительных комплексов (УВК), различающихся производительностью, набором УСО и распределенностью в пространстве.

Потребность в заказных УВК чрезвычайно велика и удовлетворить ее можно лишь на основе использования наиболее перспективных изделий электронной техники — однокристалльных микропроцессоров (МП) и однокристалльных ЭВМ (ОЭВМ). Это привело к созданию в конце 70-х годов управляющих вычислительных МП средств на основе архитектурной концепции иерархической, распределенной, многоуровневой, гетерогенной мультипроцессорности с переменной структурой процессоров и комбинентарностью магистральных связей.

В рамках данной архитектуры предполагалось создать заказные УВК с глубокой дифференциацией заказных характеристик на различных уровнях функциональной иерархии.

Эта концепция была реализована в виде набора стандартных одноплатных функциональных МП модулей, объединяемых в микросистему общей системной параллельной магистралью с унифицированным интерфейсом. Изменяемый по желанию пользователя состав входящих в микросистему и также перестраиваемых пользователем одноплатных функциональных модулей позволяет создавать заказные микросистемы, которые, в свою очередь, могут объединяться локальными вычислительными сетями (ЛВС) в заказной УВК.

Заказные характеристики УВК в соответствии с требованиями автоматизируемого объекта дифференцируются аппаратно-программным способом на уровне модуля, микросистемы и комплекса. В первых управляющих вычислительных средствах, реализующих функционально-модульную структуру, использованы 8- и 16-разрядные однокристалльные МП, а в качестве парал-

лельной системной магистрали чаще всего применялась 16-разрядная системная магистраль по стандарту IEEE 796 (Multibus 1) или ее варианты [1].

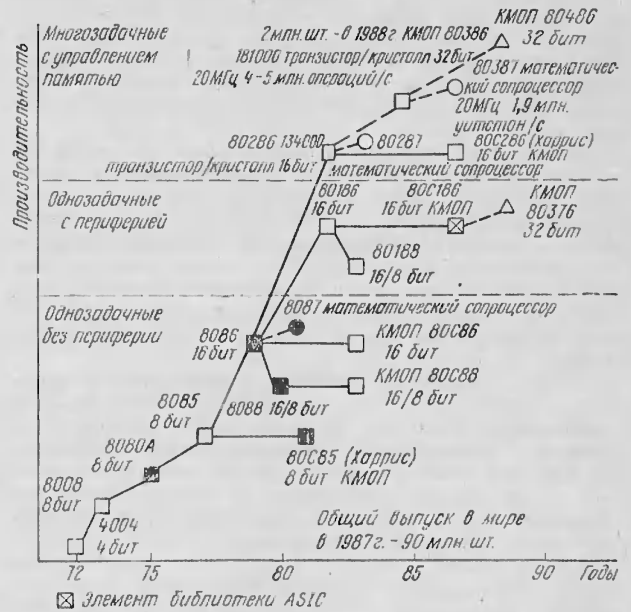


Рис. 1. Тенденции развития однокристалльных микропроцессоров (МП) фирмы Интел

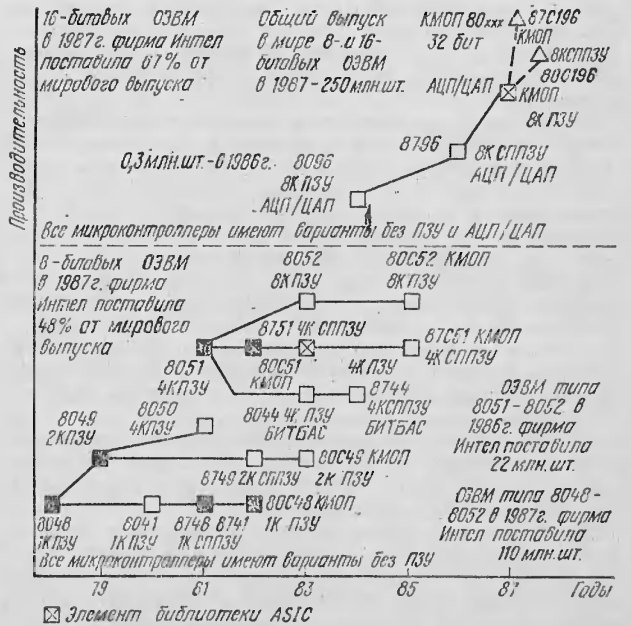


Рис. 2. Тенденции развития однокристалльных ЭВМ (СЭВМ) фирмы Интел (xxx — 960)

Необходимая производительность функционально-модульных систем достигается за счет параллельной работы множества модулей одноплатных микроЭВМ, использующих однокристальные МП. Однако работающие в реальном времени (РВ) системы управления новейшим сложным оборудованием (роботами с системами машинного зрения, автоматическими системами управления АЭС, автоматизированными системами управления группами станков и участками в системах комплексной автоматизации производства) требуют существенно повысить производительность.

Это необходимо и при использовании в системах автоматизации экспертных систем, работающих в РВ, и гибридных экспертных систем, применяющих математические и логико-лингвистические модели. Для удовлетворения всех этих потребностей необходимо применить новые поколения высокопроизводительных МП и в первую очередь 32-разрядных.

Однокристальные МП и ОЭВМ фирмы Интел — основа электронной техники, используемой в современных функционально-модульных системах (рис. 1, 2). Отметим, что каждое следующее поколение показанных на рисунках МП и ОЭВМ превосходит по производительности предыдущее в 1,5—2 раза и что каждый тип МП имеет несколько вариантов исполнения с различной производительностью и стоимостью.

Развитие происходит в направлении увеличения разрядности и функциональных возможностей кристаллов при одновременном снижении удельной потребляемой мощности за счет переходов на КМОП-технологии. В классе высокопроизводительных многозадачных МП, содержащих на кристалле схемы управления памятью, выпускается 32-разрядный МП 80386 (20 МГц), выполняющий 4...5 млн. операций/с, а с математическим сопроцессором 80387 — до 1,9 млн. операций/с. Для этого МП разработаны контроллер ПДП 82380 и контроллер кэш-памяти 82385. МП 80386 — основной процессор новейших модулей одноплатных микроЭВМ. 16-битовый однозадачный МП с расширенной периферией на кристалле типа 80186 и КМОП-типа 80С186 — основные МП для различных модулей управления УСО и периферийными устройствами. Все МП, выпускаемые фирмой Интел, обеспечены ПО, широким набором работающих совместно с ними сопроцессоров и периферийных СБИС, а также средствами отладки с внутрисхемной эмуляцией.

ОЭВМ фирмы Интел (8- и 16-разрядные) применяются в одноплатных модулях УСО и в различных встроенных в оборудование системах управления, непосредственно взаимодействующих с функционально-модульными системами. Перепрограммируемые пользователем 8-разрядные ОЭВМ типа 8751 и КМОП 87С51, содержащие СПЗУ (4 Кбайт), используются в модулях функционально-модульных систем в качестве процессоров самоконтроля. Ожидается, что развитие 32-разрядных многозадачных МП фирмы Интел будет продолжено и в скором времени появится мощный МП типа 80486, в 2—3 раза более производительный, чем МП 80386. Ожидается также появление 32-разрядных однозадачных МП с расширенной периферией и 32-разрядных ОЭВМ.

Непрерывно возрастающие требования к повышению производительности функционально-модульных систем вызвали появление целого ряда 32-разрядных функционально-модульных систем для применения в системах измерений, автоматизации и САПР. Однако отсутствие международных стандартов, обеспечивающих стабильность и открытость таких систем, препятствовало их широкому внедрению.

Большой шаг — утверждение в 1987 г. компьютерным обществом IEEE четырех стандартов [2, 3] на 32-разрядные системные магистрали для объединительных панелей функционально-модульных систем (табл. 1).

Главное различие этих стандартов состоит в том, что предложенные в 1979 г. и в 1981 г. стандарты 896 и

1014 имеют асинхронные протоколы работы, а предложенные в 1983 г. стандарты 1196 и 1296 — синхронные протоколы работы с частотой 10 МГц. Кроме того, только протокол 1014 работает без мультиплексирования, когда одновременно возбуждается большое число сигнальных линий и требуется увеличенное число возбуждателей.

В случае синхронного протокола работа начинается по срезу синхроимпульса и в течение первых 35 нс (первого временного окна) сигналы на шинах магистрали переключаются. В течение следующих 30 нс сигналы в шинах не изменяются, а помехи и сигналы импульсного сдвига уровня земли в модулях из-за индуктивности выводов разъемов затухают. Это временное окно установления сигналов.

Во время третьего временного окна (35 нс) включаются приемники и принимаются сигналы с шин (сигналы остаются неизменными). Контроль по четности байтов данных, адреса и сигналов управления определяет правильность сигналов на шинах во время третьего окна, дополнительно защищая системы от передачи неправильных данных и команд. Контроль по четности обязателен для стандарта 1296 (для стандартов 896 и 1196 он вспомогательный).

Стандарт 896 обеспечивает наибольшую пропускную способность в ограниченных условиях, остальные — примерно одинаковую пропускную способность (порядка 40 Мбайт/с) по всей длине панели, для любого расположения модулей.

Для мультипроцессорной многомашиной работы многих модулей по одной параллельной системной магистрали существенна система арбитража. Три стандарта используют распределенную параллельную и только стандарт IEEE 1014 — централизованную параллельно-последовательную систему арбитража. Такая система может привести к асинхронной метастабильности, когда в магистрали асинхронно сталкиваются почти одновременно поданные из модулей запросы магистрали и слишком быстродействующая схема арбитража, попадая в метастабильное состояние, позволяет выйти на захват магистрали сразу двум модулям (такие случаи характерны для стандарта 1014; см. в работах [4, 5]). Для стандартов с синхронным протоколом работы нет проблемы асинхронной метастабильности в магистралях. Во всех стандартах эта проблема остается только в локальных магистралях на уровне модулей, где, однако, меньше влияние переменной нагрузки магистрали при разном числе подключенных модулей, изменения задержки распространения сигналов при изменении расположения модулей, отражений сигналов в линиях магистрали, импульсного сдвига уровней земли и помех.

Передача сообщений в пространстве сообщений предусмотрена лишь в стандарте 1296 (в дальнейшем будет и в стандарте 896); она придает стандарту 1296 системный характер, поддерживая организацию совместной работы в РВ множества модулей в мультипроцессорной многозадачной системе (как в локальной вычислительной сети).

Аппаратная поддержка передачи сообщений обеспечивается СБИС сопроцессора передачи сообщений (СПС) типа VL82С389 (1986 г.).

В стандартах 896, 1196 и 1296 имеются или предполагаются в дальнейшем виртуальные прерывания. В стандарте 1296 — 255 адресов прерываний, т. е. уровень прерываний, передаваемых как неожиданные сообщения прерывания, содержащих идентифицирующие прерывания данные.

Хотя, на первый взгляд, традиционные, назначаемые для конкретных целей линии прерывания (стандарт 1014) обеспечивают большее быстродействие, однако для многомашиной многозадачной операционной системы (ОС) виртуальные прерывания в целом ускоряют и облегчают межадачную связь. Ограничение в стандарте 1296 длины передаваемого пакета данных

Утвержденные в 1987 г. институтом по электротехнике и радиоэлектронике (ИИРЭ)  
международные стандарты на 32-разрядные системные магистрали

Стандарт	IEEE 896.1	IEEE 1014	IEEE 1196	IEEE 1296
Разработчик	IEEE	Motorola	Texas Instruments	Intel
Фирменное название Название стандарта	Future bus, Future bus Backplane Bus	VME-bus Versatill Bus (VME) bus VME	Nu-bus Simple 32-bit Backplane Bus Backplane Bus	Multibus Full-Future, 32-bit Backplane Bus
Протокол	Асинхронный с мультиплексированием	Асинхронный немультимплексированный	Синхронный с мультиплексированием 10 МГц	Синхронный с мультиплексированием 10 МГц
Пропускная способность, Мбайт/с Арбитраж	117,6 (соседние модули) Распределенный, параллельный, равноправный, приоритетный	20 ... 57  Централизованный, параллельно-последовательный	37,5  Распределенный, параллельный, равноправный	40  Распределенный, параллельный, равноправный, с двумя уровнями приоритетности
Число уровней параллельного арбитража	32	Возможна асинхронная нестабильность	16	21
Передача сообщений	Будет предусмотрена в 896.2	—	—	Предусмотрена, имеется сопроцессор передачи сообщений, обеспечивающий совместимость модулей различных изготовителей
Контроль по четности	Дополнительный	—	Дополнительный	Обязательный для адреса, данных и управления
Прерывания	Виртуальные предусмотрены в 896.2	Назначаемые, 7 линий	Виртуальные по отдельному документу	Виртуальные, 255 адресов
Географическая адресация по месту на панели	Проводится	—	Проводится	Проводится
Программная автоконфигурация	—	—	»	»
Встроенный самоконтроль модулей системы	—	—	—	»
Размеры основной европлаты	366,7×280 мм 1027 см <sup>2</sup>	233,25×160 мм 373 см <sup>2</sup>	366,7×280 мм 1027 см <sup>2</sup>	233,35×220 мм 513 см <sup>2</sup>
Количество используемых разъемов	1	2	1	1
Количество активных сигнальных линий	67	107	46	62
Отношение числа возбуждаемых сигнальных линий к числу линий питания и земли	—	32 бит 64/17, 16 бит 40/13; возможен импульсный сдвиг уровня земли в платах	—	46/30

32 байтами не загружает магистраль более чем на 1 мкс и не вносит заметной задержки передачи виртуальных прерываний, приоритетно занимающих магистраль. Имеющаяся в стандартах 896, 1196 и 1296 географическая адресация и программная автоконфигурация существенно уменьшают использование в модулях перемычек и тумблеров. Это позволяет конфигурировать модуль в начале работы программно, используя географические адреса модулей. По географическому адресу можно идентифицировать каждый модуль, определить его тип изготовителя и аппаратную и программную версии.

Встроенный самоконтроль модулей (стандарт 1296) проводится после включения питания и на любом этапе работы. Данные о самоконтроле модулей собираются

центральным служебным модулем для контроля всей системы. Возможна дистанционная диагностика системы.

Конструктивная реализация основных модулей во всех четырех стандартах основана на использовании стандартных европлат, однако в стандартах 896 и 1196 это 3-разъемные платы, а в стандартах 1014 и 1296 — 2-разъемные. При этом для магистрали в трех стандартах использовано по одному разъему, а в стандарте 1014, не использующем мультиплексирование, — два. Поскольку выводы применяемого 96-выводного разъема DIN 41612 имеют последовательную индуктивность около 40 нГн, то быстрое переключение множества сигналов одновременно приводит к искажению выходящих из нее сигналов и к импульсному сдвигу

уровня земли в плате (тем выше, чем выше отношение одновременно возбуждаемых выходных сигнальных линий разьема к числу линий питания и земли). Стандарт 1296 имеет здесь заметное преимущество (см. табл. 1) по сравнению со стандартом 1014 [6].

Положение дополнительно улучшается при использовании синхронной работы с задержкой окна приема информации относительно окна переключения выходных сигналов (стандарт 1296).

В настоящее время в промышленных автоматизированных системах и автоматизированных системах для научных исследований наиболее распространены в мире стандарт IEEE 1014 (поддерживается фирмой Motorola) и стандарт IEEE 1296 (поддерживается фирмой Интел). В 1986 г. поставки модулей по стандарту 1296 достигли всего 21 млн. долл. против 312 млн. долл. по стандарту 1014. Однако ежегодный прирост поставок продукции по стандарту 1296 составляет 86,5 % против 16,7 % по стандарту 1014, так что в 1991 г. ожидается продажа модулей по стандарту 1296 уже на 448 млн. долл., а по стандарту 1014 — на 676 млн. долл. Ожидается, что в середине 90-х годов будет наиболее распространен стандарт 1296 [7].

Причина того, что стандарт 1014 уступает стандарту 1296 при мультипроцессорной работе с 32-разрядными МП (особенно при большом числе одновременно работающих в микросистеме одноплатных микроЭВМ), — наличие аппаратных трудностей, вызываемых возможной асинхронной метастабильностью и возможным импульсным сдвигом уровня земли в модулях, а также системные трудности организации совместной работы большого числа одноплатных микроЭВМ. Кроме того, синхронный стандарт 1296 существенно облегчает эксплуатацию оборудования, сделанного по этому стандарту. В заводских условиях обслуживающему персоналу с низкой квалификацией достаточно иметь осциллограф и логический анализатор.

Отметим, что наличие сопроцессора передачи сообщений в виде одной СБИС облегчает широкую разработку новых модулей по стандарту 1296 и гарантирует их полную совместимость. Большая (по сравнению со стандартом 1014) полезная площадь платы облегчает разработку модулей по стандарту 1296.

Центральный служебный модуль стандарта 1296, используемый в функционально-модульных системах, обеспечивает связь с функционально-модульными си-

стемами по стандарту 796 (Малтибас I), широко распространенным по всему миру в различных вариантах. Вот почему стандарт 1296 завоевывает все большее внимание в качестве кандидата на общепромышленный стандарт.

В стандарте 1296 к традиционным для функционально-модульных систем адресным пространствам памяти и ввода-вывода добавлены адресные пространства межсоединений и сообщений (табл. 2).

Адресное пространство межсоединений позволяет обращаться к модулям по их географическим адресам места на панели и идентифицировать тип модулей и их конфигурацию с возможностью программного изменения конфигурации. Регистры пространства межсоединений обычно размещаются в микроконтроллере 8751, используемом в модулях для выполнения теста самоконтроля. Через магистраль результаты самоконтроля передаются в центральный служебный модуль для диагностики.

Адресное пространство сообщений обеспечивает (под управлением сопроцессора передачи сообщений) междомодульную, межпроцессорную связь с передачей сообщений о прерываниях, т. е. виртуальных прерываний, и сообщений, содержащих данные.

Благодаря аппаратной поддержке передачи сообщений задача обмена сообщениями между ведущими модулями изымается из ведения модульного МП и передается сопроцессору передачи сообщений.

К числу незапрашиваемых, неожиданных сообщений (рис. 3) относятся и виртуальные прерывания, имеющие код типа сообщения 00H (для их приема в модуле есть четыре буфера). Если модульный МП не успел обработать сообщения из буферов и все они полны, то модуль, пославший сообщение, получает ответ с индикацией о невозможности приема (из-за того, что нет пустого буфера). Сообщение передается до тех пор, пока оно не будет принято (до 16 неудачных попыток, выполняемых через увеличивающиеся интервалы времени).

Пакет неожиданного сообщения имеет заголовок с указанием адресов получателя и источника, типа сообщения и дополнительной информации, определяющей тип сообщения. Широковещательное сообщение передается сразу всем получателям.

В стандарте 1296 МП связь основана не на передаче данных ссылкой, когда два МП имеют общие данные в

Таблица 2

Адресные пространства в модулях с 32-разрядной параллельной системной магистралью IEEE 1296

Характеристики	Тип адресного пространства			
	межсоединений	памяти	ввода-вывода	сообщений
Объем	2 <sup>5</sup> модулей в каждом 2 <sup>9</sup> регистров	2 <sup>32</sup> байт (4 Гбайт)	2 <sup>16</sup> устройств	2 <sup>8</sup> модулей
Тип доступа	Чтение-запись	Чтение-запись	Чтение-запись	Только запись
Разрядность передаваемых данных, бит	8	8, 16, 24, 32	8, 16, 24, 32	32
Возможность групповой последовательной передачи	Нет	Выполняется с приращением адресов	Выполняется без приращения адресов	Выполняется без приращения адресов
Число принимающих модулей	1	1	1	1 или все
Назначение адресных пространств	Идентификация типов и конфигураций плат. Обеспечение тестирования и диагностики	Обеспечение доступа к устройствам памяти для запоминания и выдачи данных и кодов	Обеспечение доступа к периферийным устройствам	Обеспечение междомодульной, межпроцессорной связи с передачей сообщений о прерываниях и сообщений, пересылающих данные

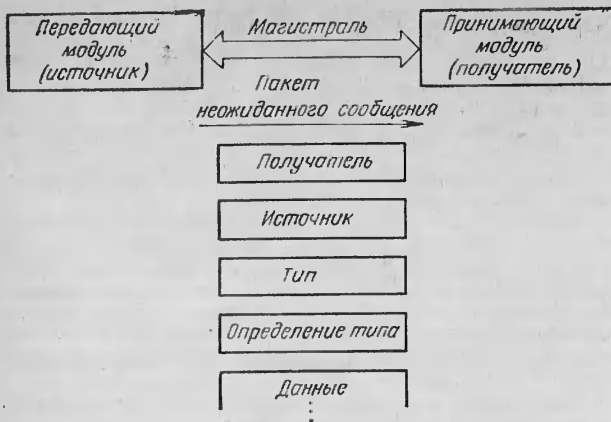


Рис. 3. Передача неожиданного сообщения в параллельной системной магистрали IEEE 1296

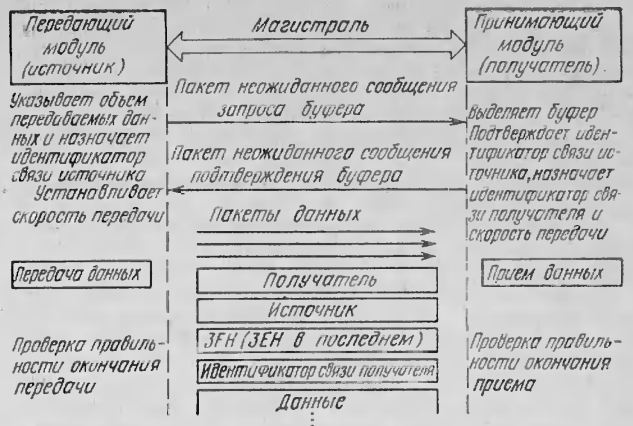


Рис. 4. Передача ожидаемого сообщения в параллельной системной магистрали IEEE 1296

двухпортовой памяти и ссылаются на них, хотя это и возможно, а на передаче значением, когда копии данных фактически передаются из одного модуля в другой. Такая связь облегчает совместную работу большого количества разнотипных МП и хорошо защищает память от постороннего вмешательства.

Массивы данных (до 16 Мбайт), сформированные в пакеты (до 32 байт), передаются ожидаемым сообщением (рис. 4), подготовляемым двумя неожиданными сообщениями запроса и подтверждения буфера в принимающем модуле. После этого данные передаются без вмешательства центральных МП модулей (как в распределенной системе с ПДП). Данные передаются со скоростью, указанной в неожиданном сообщении подтверждения буфера, в соответствии с возможностями

принимающего модуля. Сообщения передаются 32-рядными кодами достаточно быстро. Поэтому аппаратно поддерживаемая сопроцессором передача сообщений передача значением обеспечивает получение характеристик, аналогичных характеристикам систем с передачей ссылкой.

Протоколы стандарта 1296 не только регламентируют правила формирования и передачи сигналов. Используемая в нем передача сообщений (реализованная аппаратно в БИС сопроцессора передачи сообщений) носит системный, сетевой характер. Она может быть описана с помощью уровней эталонной сетевой модели взаимосвязи открытых систем (ВОС), предложенной международной организацией по стандартизации (МОС),

Таблица 3

Реализация стандартом IEEE 1296 и его предстоящим расширением уровней эталонной сетевой модели ВОС, предложенной международной организацией по стандартизации

Уровни модели ВОС	Стандарт IEEE 1296				
7. Прикладной	Прикладные задачи разработчика системы				
6. Представительный	Интерфейс * периферийного контроллера	Интерфейс * связного контроллера	Интерфейс * контроллера ЛВС	Интерфейс * межпроцессорного обмена	Специальные протоколы межпроцессорной и междоулевой связи
5. Сеансовый					
4. Транспортный	Интерфейс транспортного протокола *, спецификация 1296, аппаратно реализованная СПС				
3. Сетевой	Отсутствует				
2. Канальный	Спецификация 1296, аппаратно реализованная СПС				
1. Физический	Спецификация 1296, аппаратно реализованная СПС				

Примечания: \* — предполагаемое расширение стандарта 1296; СПС — БИС сопроцессора передачи сообщений

В настоящее время готовится стандартизация не только формы, но и содержания сообщений. Вопросы сетевого системного характера будут решаться для обеспечения стандартной системы связи между одноплатами микроЭВМ и модулями интеллектуальных периферийных контроллеров, а также между самими одноплатами ЭВМ (табл. 3).

Если сейчас стандарт 1296 охватывает четыре нижних уровня модели ВОС, то после расширения стандарт распространится на шесть уровней. Верхние уровни протокола передачи сообщений будут включать в себя протоколы интерфейсов межпроцессорного обмена; периферийного и связанного контроллеров, контроллера ЛВС. Предполагается, что это будут универсальные протоколы, по желанию пользователя допускающие расширение и включение специальных команд [8].

Особенности параллельной системной магистрали 1296 (см. табл. 1): есть развитое мультиплексирование сигналов; разрядность данных — 8, 16, 24, 32; сигнализация ведется сообщениями (пространство сообщений), межпроцессорный обмен — также сообщениями; обслуживание по стандарту 1296 — централизованное, т. е. в центральном служебном модуле проводятся синхронизация, тайм-аут магистрали, идентификация и реконфигурация модулей, а также дистанционная диагностика модулей.

Отметим, что в стандарте 1296 регистрируются ошибки по четности при передачах, фиксируемые принимающими модулями, и ошибки длительного занятия магистрали — «тайм-аута», фиксируемые центральным служебным модулем. Появление этих ошибок прекращает работу магистрали и вызывает исполнение цикла исключения, во время которого модули принимают один из сигналов ошибки магистрали и переходят к фазе восстановления цикла исключения. Во время фазы восстановления вся работа по магистрали прекращается на время трех синхроимпульсов, после чего работа магистрали возобновляется, начинаясь с нового цикла арбитража.

На рис. 5 представлен типичный модуль 32-разрядной одноплатной ЭВМ. В настоящее время такие ЭВМ выпускаются уже несколькими фирмами. В основном они применяют МП 80386, но есть модули и на других 32-битовых МП, в частности 68020.

Наиболее сложная одноплатная ЭВМ фирмы Интел типа SBC 386/120—M16 использует 20-мегагерцевые кристаллы МП 80386—20, математического сопроцессора 80387—20 и ОЗУ (16 Мбайт).

В настоящее время эта микроЭВМ работает с ОС RMX 286, обеспечивающей 16-мегабайтную адресацию и многозадачную работу. Это позволяет эффективно применять ее в сложных системах автоматизации производства.

Операционная система RMX 286 фирмы Интел опережает все известные в мире ОС РВ по своим функциональным возможностям. Так, только в этой ОС есть развитая система обработки многоуровневых прерываний с многократной буферизацией, три различные системы синхронизации параллельных вычислений, многоуровневая система порождения и прекращения параллельных задач и т. д.

Последняя редакция ОС RMX 286 учитывает особенности работы в магистрали по стандарту 1296 со свойствами географической адресации и самодиагностики.

Операционная система реального времени «Ядро 386» — дальнейшее развитие ОС RMX 286.

Основные разработчики и изготовители модулей по стандарту 1296 — фирмы Интел (США), Сименс (Западная Европа) и Тошиба (Япония).

Фирма Сименс, например, выпускает серию из 16 типов модулей OSM—B (Open Siemens Multibus). Количество типов модулей, выпускаемых в мире по стан-

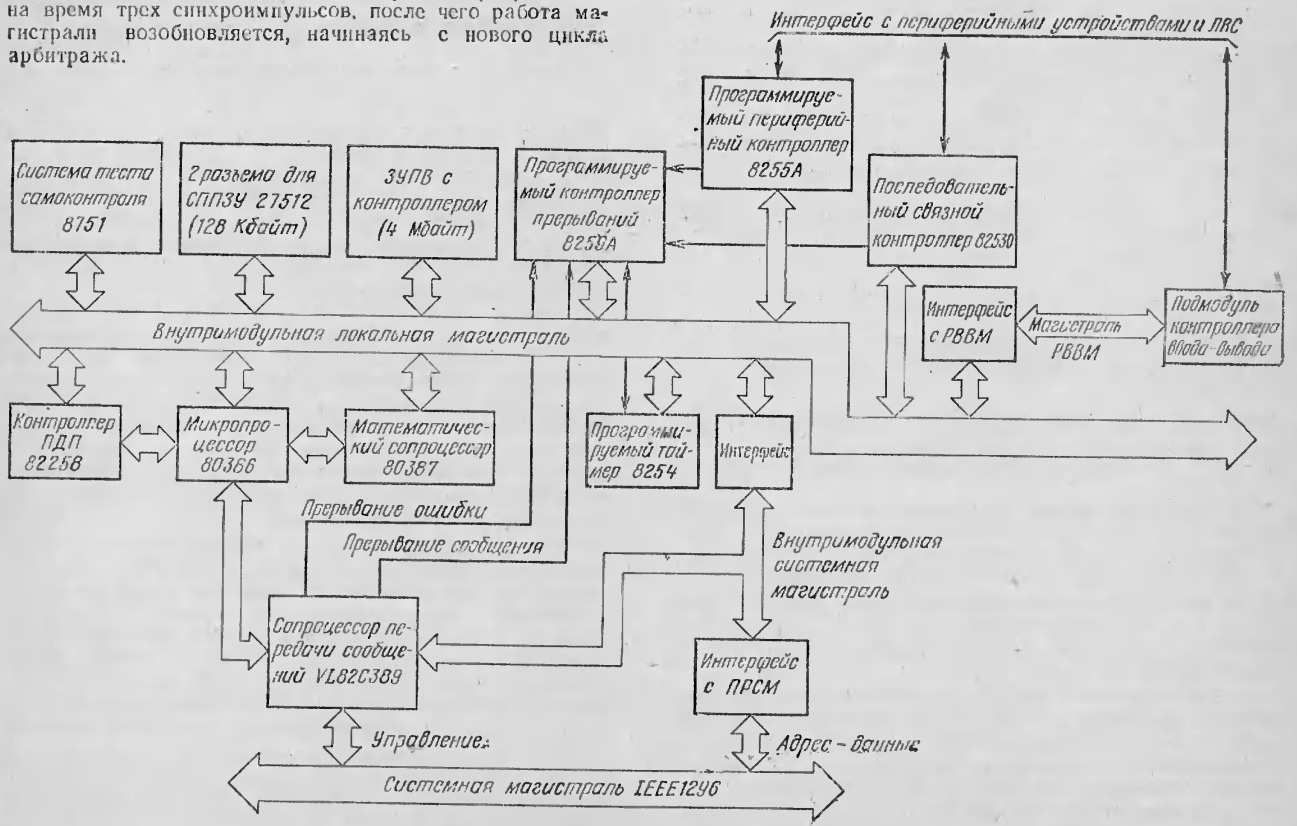


Рис. 5. Модуль одноплатной ЭВМ

дарту 1296, достигает нескольких десятков и быстро увеличивается.

Применение 32-разрядных функционально-модульных систем продолжает расширяться, распространяясь, в частности, и на ПЭВМ. Так, фирма Apple Computer использует в своей ПЭВМ Macintosh II 32-разрядную магистраль по варианту стандарта IEEE 1196. Фирма IBM разработала для ПЭВМ Personal System/2 Model 80 (выпуск начал в 1987 г.) 32-битовую магистраль Micro Channel, обеспечивающую параллельную работу нескольких ведущих модулей.

Что же касается систем автоматизации производства, то стандарт 1296 должен и в нашей стране стать основным промышленным стандартом на 32-разрядную магистраль для функционально-модульных систем промышленной автоматизации.

## ЛИТЕРАТУРА

1. Шереметьевский Н. Н., Долкерт В. М. Магистрально-модульные микросредства управляющей вычислительной техники // Микропроцессорные средства и системы.— 1984.— № 2.— С. 24—27.
2. Smolin M. More on IEEE Standards generation // IEEE Micro.— 1987.— Vol. 7. N 5.— P. 92—94.
3. Borril P. L. Objective Comparison of 32-bit Buses // Microprocessors and Microsystems.— 1986.— Vol. 10. N 2.— P. 94—100.
4. Marrin K. Metastability hants VMEbus and Multibus II System Designers // Computer Design.— 1985.— Vol. 24, N 9.— P. 29, 30, 32.
5. Beaton J., Tetrick R. S. Designers Confront Metastability in Boards and Buses // Computer Design.— 1986.— Vol. 25. N 5.— P. 67—71.
6. De Bock R. M. Simple solution cures glitches on high-speed buses // EDN.— 1987.— Vol. 32. N 9.— P. 173—178, 180.
7. Newcomez Multibus II predicted to gain rapidly on VMEbus // Mini-Micro Systems.— 1987.— Vol. XX. N 4.— P. 52.
8. Lieberman D. VMEbus and Multibus II mature for multiprocessing applications // Computer Design.— 1987.— Vol. 26. N 20.— P. 25—28, 30—32.

Статья поступила 19 апреля 1987.

УДК 681.325.5

В. Г. Артюхов, В. Б. Глухенький, Р. А. Лернер, А. Н. Макеев, Л. В. Носенко, Г. Ф. Страутманс

## ЦИФРОВОЙ ПРОЦЕССОР КМ1813ВЕ1: ПОДГОТОВКА И ОТЛАДКА ПРОГРАММ

Цифровой процессор обработки сигналов (ЦПОС) КМ1813ВЕ1 представляет собой однокристалльную ЭВМ с аналоговыми устройствами ввода-вывода [1], архитектура и система команд которой ориентированы на реализацию алгоритмов цифровой обработки сигналов (ЦОС), и предназначенную в основном для замены аналоговых узлов в системах обработки сигналов и управления.

Среди возможных типов средств автоматизации разработки программ ЦПОС [2] заслуживают внимания автономные специализированные устройства, оснащенные средствами микропроцессорной техники. Рассматриваемое автономное, функционально законченное специализированное устройство РАСТР [3] обеспечивает ввод, редактирование и трансляцию программ; диалоговую отладку программ на программной модели; запись программ во внутреннее ПЗУ; отладку СБИС на

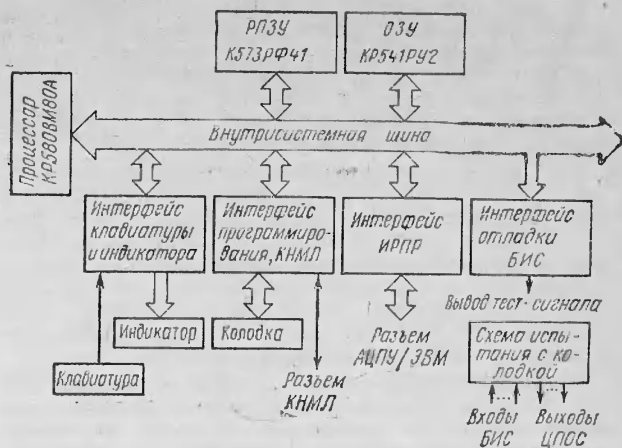


Рис. 1. Структурная схема РАСТРа



Рис. 2. Клавиатура РАСТРа

тестовых сигналах; комплексную отладку СБИС в реальном масштабе времени на реальных сигналах. РАСТР представляет собой микроЭВМ, выполненную на основе микропроцессорного комплекта КР142ЕН2 на основе микропроцессорного комплекта КР142ЕН2. Устройство содержит средства, обеспечивающие диалог с оператором, программирование и комплексную отладку ЦПОС, подключение к КНМЛ, АЦПУ и универсальной ЭВМ (рис. 1).

Оператор управляет устройством с помощью клавиатуры и индикатора. Каждая функциональная клавиша (рис. 2) имеет два значения, причем таких, что в любой момент времени только одно из них имеет смысл. Устройство само выбирает значение, отвечающее контексту. Очист клавиатуры, подавление дребезга контактов, а также «авторепетитор» клавиши, находящейся в нажатом состоянии более чем 0,8 с, осуществляются программно. Основные параметры РАСТРа приведены на четвертой странице обложки.

Обмен информацией между ОЗУ устройства и ПЗУ ЦПОС осуществляется с помощью интерфейса программирования, построенного на трех ИС К155ТМ8 и управляемом формирователе напряжения КР142ЕН2. Один из разрядов интерфейса служит для обмена информацией в последовательном коде с КНМЛ, в качестве которого используется бытовой кассетный магнитофон. Запись на ленту КНМЛ ведется двухчастотным методом с частотами 2400 Гц для записи единицы и 1200 Гц для записи нуля. Способ и формат записи соответствуют принятым в МСУВТ В7/В9 [4]. Соединение с АЦПУ и универсальной ЭВМ обеспечивается через интерфейс ИРПР.

Интерфейс комплексной отладки предназначен для формирования тестовых сигналов, подаваемых на входы



запрограммированного ЦПОС, подключенного к схеме испытания через колодку, установленную на передней панели устройства. Схема испытания содержит фильтры питания, генератор тактовых импульсов, источник опорного напряжения, накопительную емкость и разъемы, обеспечивающие подключение к испытываемому процессору контрольно-измерительной аппаратуры.

Временные диаграммы всех интерфейсных сигналов в устройстве РАСТР формируются программным путем. Оператор в диалоге с устройством выбирает необходимый режим работы (рис. 3). Вводить программу целесообразно в форме мнемонических обозначений полей. Поля команд программы ЦПОС вводятся последовательно, в порядке их записи. Трансляция программ в объектные коды выполняется устройством автоматически после ввода каждой команды. Ранее разработанные и оттранслированные программы с известным кодом можно вводить в 16-ричном коде.

Программу в процессе ввода в ОЗУ можно редактировать, изменяя отдельные поля ранее введенных команд, исключая и добавляя новые команды.

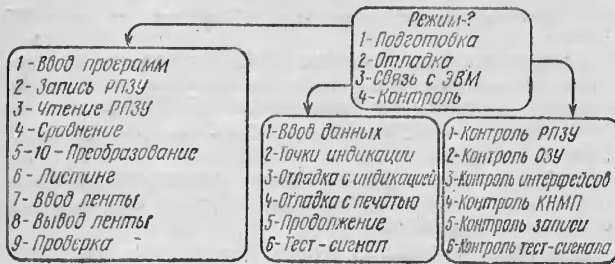


Рис. 3. Режимы работы РАСТРА

В подрежимах ЗАПИСЬ РПЗУ, ЧТЕНИЕ РПЗУ и СРАВНЕНИЕ перед выполнением соответствующих действий программно проверяется правильность установки ЦПОС в колодку программирования и выдается сообщение на экран. Двоично-десятичные преобразования выполняются над числами, целая часть которых не превышает 44 двоичных разряда, а дробная 32 двоичных разряда.

Отладка запрограммированной БИС ЦПОС осуществляется путем подачи на ее входы тестовых сигналов, генерируемых непосредственно устройством РАСТР в подрежиме ТЕСТ-СИГНАЛ на основании отсчетов, заданных для входного канала 0 ЦПОС в подрежиме ВВОД ДАННЫХ.

Период следования моделируемого сигнала  $T = 15T_0 - 85$  мкс, где  $T_0$  — параметр, задаваемый с клавиатуры устройства и принимающий значения 0..65535<sub>10</sub>.

ЦАП использует тот же источник опорного напряжения, что и ЦПОС, установленный в колодку ИСПЫТАНИЕ (это обеспечивает согласование уровней сигналов). Комплексная отладка запрограммированной БИС ЦПОС в реальном масштабе времени проводится в режиме ИСПЫТАНИЕ путем подачи на ее входы реальных сигналов и анализа результатов их обработки ЦПОС при помощи внешней контрольно-измерительной аппаратуры.

Режим СВЯЗЬ С ЭВМ используется при работе устройства в составе систем автоматизированного проектирования ППО ЦПОС. В этом режиме устройство подключается к внешней ЭВМ по интерфейсу ИРПР и работает как интеллектуальное периферийное устройство. Управление с клавиатуры при этом исключается: действует только клавиша СБРОС, которая переводит устройство в автономный режим работы. Функции подготовки и отладки программ ЦПОС в этом режиме возлагаются на ЭВМ.

Режим КОНТРОЛЬ предназначен для самодиагностики устройства. Контролируются ОЗУ и ПЗУ уст-

ройства, а также все его интерфейсы (ИРПР, КНМЛ, программирования и испытания СБИС ЦПОС). Сообщения о результатах проверки выдаются на экран индикатора. Неисправности в блоках ОЗУ и ПЗУ обнаруживаются с точностью до корпуса БИС. Для контроля интерфейсов ИРПР и КНМЛ их входные и выходные каналы соединяются между собой специальными внешними разъемами. В подрежиме КОНТРОЛЬ ЗАПИСИ осуществляется проверка исправности цепей, управляющих чтением и записью ППЗУ ЦПОС, а также формированием тестового сигнала (при помощи осциллографа). При контроле всех интерфейсов возможные неисправности распознаются с точностью до одного сигнала.

Телефон 441-96-20, 430-51-02, Киев

## ЛИТЕРАТУРА

1. Усовершенствованная СБИС цифрового процессора обработки сигналов с аналоговыми устройствами ввода-вывода / Вольперт Л. А., Туревич М. Х., Кузнецов А. А., Страутманн Г. Ф. // Тез. докл. на конф. «Методы и микроэлектронные средства цифрового преобразования и обработки сигналов». — Рига, 1983.
2. Артюхов В. Г., Макеев А. Н., Глухенький В. В. Средства автоматизации проектирования программного обеспечения процессоров сигналов // Изв. вузов МВ и ССО СССР. Сер. Радиоэлектроника, — 1985. — № 8. — С. 74—78.
3. А. с. № 4097988. Артюхов В. Г., Макеев А. Н., Глухенький В. В. и др. Огубл. 1987. Бюл. № 39.
4. Шереметьевский Н. Н., Долкарт В. М. Магистрально-модульные микросредства управляющей вычислительной техники // Микропроцессорные средства и системы. — 1984. — № 2. — С. 24—27.

Статья поступила 31.03.88

## КРАТКОЕ СООБЩЕНИЕ

УДК 681.3.07—681.327

А. А. Связзов

## МОДУЛЬ СОПРЯЖЕНИЯ ТЕЛЕТАЙПА С МАГНИТОЛЕНТОЧНЫМ УСТРОЙСТВОМ ЕС-9004

Информация, поступающая и передаваемая по теле-тайпу с помощью модуля, записывается на МЛ (ЕС-9004). Телетайп и ЕС-9004 работают параллельно; функции связи и управления передачей в канале выполняются телетайпом.

Сообщения абонентов записываются на МЛ блоками по 80 или 160 символов в коде ДКОИ. Преобразование кодов исключает специфические кодовые комбинации МТК-2 (ПС-перевод строки, ВК — возврат каретки, признак регистра). Для сохранения структуры принимаемых сообщений комбинация ВК вызывает заполнение блока информации до конца кодами символа «Пробел».

Модуль сопряжения выполнен в виде двух плат стандартного формата, устанавливаемых в ЕС-9004 в блок 05 на места 14 и 30. Можно работать по прямым, выделенным и коммутируемым каналам сети абонентского телеграфа. Телетайп и ЕС-9004, объединенные в систему, сохраняют возможность выполнения всех своих функций по отдельности. В такой системе принимаемая (передаваемая) информация одновременно с записью на МЛ визуализируется на экране монитора, а при необходимости заносится на перфолену и выводится на печать.

Телефон 23-75-96, Таллин

Сообщение поступило 08.02.88

## МИКРОЭВМ МС 0102 С КЕШ-ОЗУ

Развитие технологии интегральных компонентов позволяет последовательно совершенствовать модели программно совместимых микроЭВМ семейства «Электроника».

Заканчивается подготовка серийного производства новой 16-разрядной микроЭВМ МС 0102 с кеш-ОЗУ с двухкристаллическим процессором К1831, реализующим систему команд мини-ЭВМ «Электроника 79».

## Основные технические характеристики:

Разрядность, бит	15
Быстродействие на коротких операциях сложения типа «регистр — регистр», оп./с, не менее	2,5 · 10 <sup>6</sup>
Область физических адресов, Мбайт	4
Емкость ОЗУ, Кбайт	512
Емкость сверхоперативного ОЗУ, Кбайт	8
Число команд (обеспечивается программная совместимость с мини-ЭВМ «Электроника 79»)	140
Емкость НГМД, Кбайт	800
Емкость НМД, Мбайт	10
Число каналов последовательной связи с внешними устройствами	5
Потребляемая мощность, В · А, не более	550
Удаление терминальных устройств от ЭВМ, м	до 15
Возможность организации многомашиной работы	Имеется, с использованием дополнительных устройств связи
Площадь размещения, м <sup>2</sup>	0,5
Габаритные размеры, мм	730 × 260 × 630
Масса, кг, не более	42

Архитектура процессора обеспечивает работу в трех режимах: основном, пользователя и супервизора. Такой подход и динамическое распределение памяти позволяют более эффективно использовать ОЗУ при организации многопользовательских систем, где защита областей памяти, выделяемых пользователю, программ-монитору и программам ввода-вывода, особенно актуальна.

Для эффективного управления процессами вычислений, обработки прерываний и анализа ошибок в процессоре существует широкий набор регистров.

Прерывания, возникающие в процессе выполнения программ, обрабатываются в области ОЗУ, которая начинается с нулевого адреса, а верхний предел задается программно (выборка информации из этой области осуществляется по принципу стека).

Информация нижних пределов области, где расположены адреса-векторы прерываний устройств, защищается аппаратно. При уменьшении глубины стека до адреса 400 (желтая зона) в процессоре происходит прерывание выполнения текущей команды, до адреса 2 (красная зона) — отказ. В каждом режиме процессора свой регистр указателя стека (УСОП, УСС, УСП). Стек основного режима имеет защиту от переполнения, которая вызывает прерывание или отказ при нарушении желтой или красной зоны соответственно.

Наличие трех наборов регистров адреса страницы и описания страницы (РАС и РОС, по одному набору для каждого режима) позволяет проводить динамичное перераспределение ОЗУ в области до 4 Мбайт минимальными программными средствами. Регистры адреса страницы содержат поле активной страницы и перемещения внутри нее; описания страницы, информация о длине страницы, порядке доступа и направлении ее расширения. Доступ к регистрам адреса и описания страницы, как правило, производится программой-монитором.

Четыре регистра (РСДП0, РСДП1, РСДП2, РСДП3) управляют памятью. С помощью двух наборов (R0...R5, R'0...R'5) регистров общего назначения сокращается

число обращений к оперативной памяти (увеличивается число операций типа регистр-регистр).

Регистр состояния процессора (РССП) обеспечивает переключение режимов работы, выбор одной из двух групп регистров общего назначения, контролирует результат выполнения операций над данными (за исключением команд с плавающей запятой), устанавливает приоритетный уровень прерываний для процессора.

Регистр ошибок (РОШ) фиксирует коды ошибок и отказов, возникающие в ходе выполнения программы, регистр ошибок системной памяти (РОСП) — ошибки обращения к основному ОЗУ и кеш-ОЗУ.

Регистры «ЕСТЬ-НЕТ» и управления кеш-ОЗУ (РУ кеш) обслуживают кеш-ОЗУ (фиксируются обращения и обеспечивается исключение из системы и диагностика кеш-ОЗУ).

С помощью служебного регистра (СР) устанавливается адрес начального загрузчика и фиксируются сигналы состояния блока питания.

Для контроля выполнения операций над числами с плавающей запятой в процессоре реализованы регистр состояния операций с плавающей запятой (РСОПЗ) и два регистра исключений (РИА — адреса и РИК — кода).

Регистр линейного таймера (РЛТ) управляет процессами прерывания программы по внешнему событию. Регистр программных прерываний (РПП) следит за очередностью обслуживания прерывания (по приоритету уровня запроса).

Набор из шести 64-разрядных регистров-аккумуляторов обеспечивает выполнение операций над числами с плавающей запятой.

Кеш-ОЗУ емкостью 8 Кбайт, размещенное на плате процессора, уменьшает число обращений к основному ОЗУ.

## Обмен информацией между ОЗУ и процессором

При чтении процессор обращается в кеш-ОЗУ. По ассоциативному признаку адреса осуществляется поиск, и в МП поступает сигнал о наличии данных в кеш-ОЗУ, при их отсутствии логика процессора реализует обращение к основному ОЗУ через магистраль. По указанному адресу данные записываются одновременно в кеш-ОЗУ и основное ОЗУ.

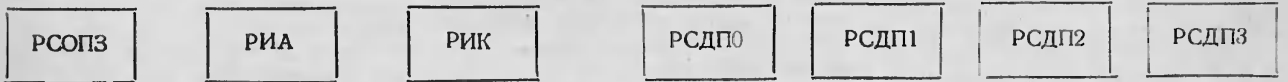
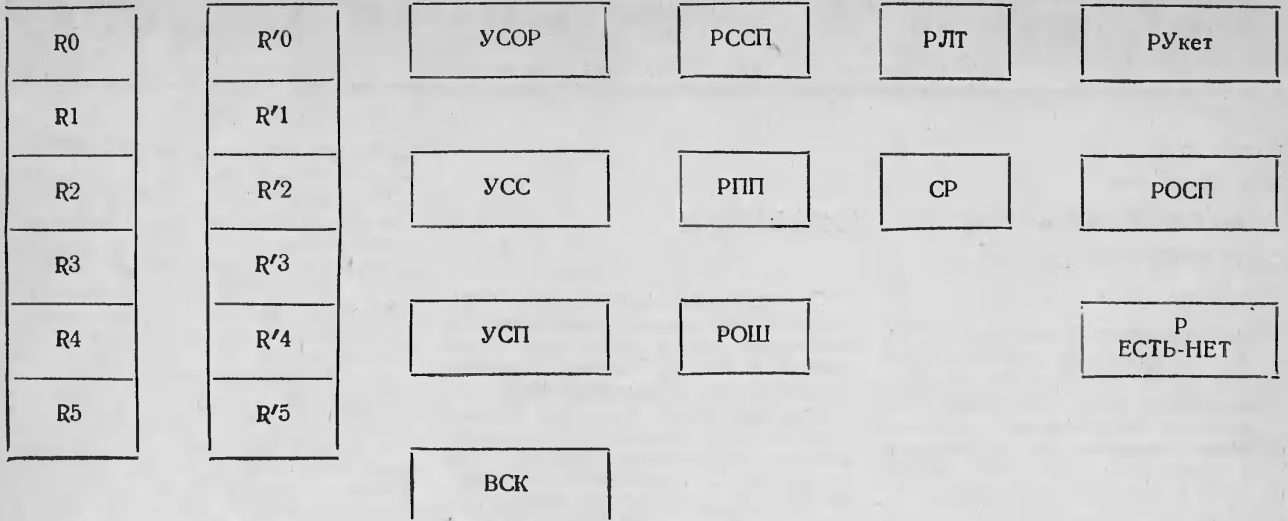
Программы в большинстве случаев имеют законченные структурные блоки в пределах 4 Кбайт, поэтому использование кеш-ОЗУ позволяет значительно сократить время выборки.

Пользователь может объявить данные в кеш-ОЗУ недействительными, установив соответствующий разряд в РУ кеш. Заполнение кеш-ОЗУ производится в процессе чтения-записи данных из ОЗУ. При отсутствии данных в кеш-ОЗУ или при объявлении этих данных недействительными, данные, поступающие из основного ОЗУ в МП, одновременно записываются в кеш-ОЗУ, где происходит контроль четности информации.

Для ускорения процессов обработки информации при последовательной выборке и обработке команд, учитывая, что в архитектуре процессора присутствует одна магистраль приема-передачи данных и команд, применяли принцип конвейеризации.

Конвейер процессора имеет четыре цикла: загрузка виртуального счетчика команд (ВСК); преобразование виртуального адреса с помощью диспетчера памяти (ДП) в физический (результат загружается в физический программный счетчик команд (ФПСК), содержимое виртуального счетчика команд увеличивается на 2);

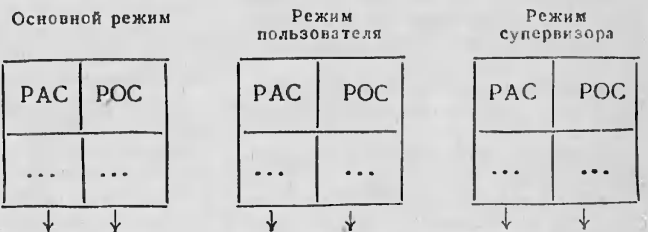
выборка команды из памяти в буферный регистр (БР); содержимое виртуального счетчика команд наращивается на 2, содержимое физического программного счетчика преобразуется;



Регистры-аккумуляторы



Регистры диспетчера памяти



Набор регистров

переход команды из буферного регистра в регистр команд (РК); буферный регистр, виртуальный счетчик команд, физический программный счетчик загружаются новым содержимым, содержимое виртуального счетчика команд увеличивается на 2.

Описанные действия показаны на схеме:

- |  |  |
|--|--|
| <b>1-й цикл</b>                        | <b>2-й цикл</b>                        |
| ВСК ← СК                               | ФПСК ← ДП                              |
|  | ВСК <sub>1</sub> ← ВСК+2               |
| <b>3-й цикл</b>                        | <b>4-й цикл</b>                        |
| БР ← данные ОЗУ                        | РК ← БР                                |
| ФПСК <sub>1</sub> ← ДП <sub>1</sub>    | БР <sub>1</sub> ← данные ОЗУ           |
| ВСК <sub>2</sub> ← ВСК <sub>1</sub> +2 | ФПСК <sub>2</sub> ← ДП <sub>2</sub>    |
|  | ВСК <sub>3</sub> ← ВСК <sub>2</sub> +2 |

За четыре цикла конвейер заполняется, обработка команд ускоряется. Короткие команды (сложение, пересылка, очистка и т. д.) выполняются за один цикл процессора, так как одновременно реализуется текущая команда, декодирование предыдущей и выборка сле-

дующей. Однако команда ветвления может разрушить синхронную работу конвейера. В этом случае необходимо предварительное заполнение конвейера новым содержимым. С помощью кеш-ОЗУ и конвейера достигается быстродействие коротких команд более 3 млн./с.

Кеш-ОЗУ реализована на микросхемах К132РУ5, К132РУ12, управление кеш и логики взаимодействия с МПИ выполнены на двух матричных БИС.

Телефон 23-16-22, Воронеж

Статья поступила 14.09.88

УДК 681.03

Э. М. Пройдаков

## ВТОРОЕ ПОКОЛЕНИЕ ПЕРСОНАЛЬНЫХ ЭВМ ФИРМЫ IBM

### Введение

Анонсированное 2 апреля 1987 г. второе поколение персональных ЭВМ фирмы IBM обладает интересными техническими характеристиками, отражающими современные схематехнические тенденции и направления технической и коммерческой политики IBM. В них явно выражено намерение за счет применения новых технологий подтянуть начинающие уступать конкурентам по техническому уровню изделия и тем самым улучшить положение фирмы на рынке.

Доходы IBM в 1986 г. упали на 27%, в том числе в 4-м квартале на 48% [1, 2]. Это вызвано:

увеличением спроса на суперЭВМ (где IBM не лидирует) и его уменьшением на большие ЭВМ из-за насыщенности рынка;

увеличением спроса на супермини-ЭВМ, где заметно усилилась фирма DEC;

временной неудачей IBM на рынке АРМ (IBM PC RT);

потерей позиций IBM на рынке настольных издательских систем (70% от общего объема продаж принадлежит фирме APPLE);

появлением класса ЭВМ, названного «супермикро» (в 1986 г. в нем доминировали системы на базе микропроцессора MC 68020);

ростом конкуренции со стороны производителей так называемых совместимых компьютеров (1,5 млн. ПЭВМ в 1987 г.);

рядом неудачных разработок в развитии линии IBM PC (табл. 1).

**Основные технические характеристики**

Во втором поколении ПЭВМ — «Персональная система 2» (Personal System 2, PS/2, для Японии — PS/55), объявлены четыре модели, каждая из которых имеет несколько модификаций. Номер модификации указывает на объем встроенных дисковых накопителей (табл. 2).

В предлагаемых моделях учтен опыт, накопленный фирмой при разработке, выпуске и маркетинге ПЭВМ IBM PC, PC XT, PC AT (табл. 1) и других, именуемых далее IBM PC. Это нашло отражение в организационных, архитектурных, технологических, схематехнических и программных решениях. Ниже приведены важнейшие из них.

Снятие ограничений на объем ОЗУ.

Адаптеры, находившиеся ранее на отдельных платах расширения и являвшиеся в PC опцией, сделаны встроенными: двунаправленный параллельный порт; последовательный порт; адаптер устройств типа «мышь»; адаптер графического дисплея, 30-я модель MCGA (Multi Color Graphics Array), видеопамять 64 Кбайт; остальные модели — VGA (Video Graphics Array), видеопамять 256 Кбайт; адаптер накопителей на гибких магнитных дисках (НГМД); адаптер клавиатуры; часы-календарь с батарейным питанием.

Увеличение в 2, 3 раза объемов дисковой памяти.

Использование 8 каналов прямого доступа в память (ДМА), из которых четыре могут работать одновременно (начиная с 50-й модели).

Появление единого графического адаптера (VGA) для всех моделей (начиная с 50-й) и повышение качества изображения.

Введение мультипроцессорирования позволяет, например, в 50-й модели

добавить две платы на МП 80286, а в 80-й модели — четыре такие платы. Эти платы могут работать в режиме ведущего и ведомого (подчиненного) процессоров. (Однако эта рекламируемая потенциальная возможность, требующая сильной программной поддержки, пока не реализована.)

Обеспечение совместности с IBM PC: во всех моделях PS/2 могут быть внешние (находящиеся в отдельном корпусе) НГМД 5.25", а к IBM PC XT/AT могут быть добавлены внешние НГМД 3.5" емкостью 720К, которые поддерживаются MS DOS, начиная с версии 3.20; графические адаптеры VGA и MCGA эмулируют адаптеры CGA и EGA; имеются технические и программные средства для подключения ПЭВМ IBM PC предыдущего поколения с помощью двухметрового кабеля для устройства печати с интерфейсом типа CENTRONICS к двунаправленному параллельному порту ПЭВМ PS/2 (IBM Data Migration Facility). С помощью нульмодема и соответствующего ПО расстояние между ПЭВМ можно увеличить до 15 м; файлы с IBM PC могут быть перекачаны в IBM PS/2 через общую локальную сеть или через хост-ЭВМ. Отметим, что этими способами не может переоснаститься ПО для IBM PC, поставляемое на дисках, защищенных от копирования.

Таблица 1

Хронологические сведения о ПЭВМ фирмы IBM

Тип	Месяц и год	Продано, млн. шт.	Степень успеха
<b>Первое поколение</b>			
IBM PC	Август 1981	3	+++
IBM PC XT	Март 1983	2	+++
IBM PC XT 370	Октябрь 1983	125 тыс.	—
IBM PC JR	Ноябрь 1983	—	Снята в марте 1985 г.
IBM 3270 PC	—	—	—
IBM PC AT	Август 1984	1	+++
IBM PC Portable	Весна 1985	—	—
IBM PC RT	Январь	—	—
IBM PC Convertible	Апрель 1986	—	—
IBM PC XT286	Август 1986	15 тыс.	—
IBM 3270 PC AT	—	—	—
IBM PC RT модель 115, 125, B25	Апрель 1987	—	—
<b>Второе поколение</b>			
IBM PS/2:		(в 1987 г.)	
Мод. 30, 50, 60, 80	Апрель 1987	1	+++
Мод. 25, 30.311	Август 1987	—	—

## Технические характеристики ПЭВМ семейства PS/2

Модель	Модификация	Процессор	Частота, МГц	ОС	ОЗУ, Мбайт	ПЗУ, Кбайт	НГМД, Мбайт	Емкость дисков, Мбайт
25		8086	8	DOS 3.3	512К	64	1×720К	—
30	002	8086-2	8	DOS 3.3	640К	64	2×720К	—
	021	+8087-8/2			640К	64	720К	20
50	021	80286	10	DOS 3.3	1	128	1,44	20
		+80287		OS/2				
60	041	80286	10	DOS 3.3	1	128	1,44	44
	071	+80287	10	OS/2	1	128	1,44	70
80	041	80386	16	DOS 3.3	1	128	1,44	44
	071	+80387	16	OS/2	2	128	1,44	70
	111		20	OS/2	2	128	1,44	115
	311	80386	20	OS/2	2	128	1,44	314

Продолжение таблицы 2

Модель	Емкость доп. ленточных дисков, Мбайт	Потребляемая мощность, Вт	Мас. ОЗУ	Слоты расширения	Габаритные размеры (В×Ш×Г)	Масса, кг	Цена, долл.
25	720К	—	0,64	2	38,1×33,0×38,1	—	1350
30	20	70	2,64	2			2010
	20		2,64	3	10,2×40,6×39,2	8,0	2834
50	20	94	7	3	14,0×36,0×42,0	10,5	4838
60	70	207	15	7	59,7×16,5×48,3	23,6	6291
	70/115	225	15	7			7173
80	44	207	16	7			6995
	70/115	225	16	7	59,7×16,5×48,3	23,6	8495
	70/115	225	16	7			10995
	314		16	7			13935

Примечание. Цены даны ориентировочные по предложениям в СССР на 05.88 г.

Использование для ведения архива данных внешнего накопителя на оптических дисках со съемными кассетами 5.25" емкостью 200 Мбайт с разовой записью и многократным считыванием со скоростью 171 Кбит/с (WORM — Write — Once — Read — Many Times).

Применение новой безлюдной роботизированной технологии производства ПЭВМ, позволяющей производить одну ПЭВМ за 20 с вместо 45 с для IBM PC (15 с у фирмы Apple Computer). Эта технология обеспечена планарными CMOS микросхемами — двусторонними платами без сверления отверстий, с поверхностным монтажом микросхем (SMF — Surface Mount Technology) вместо обычных печатных плат (PCB — Printed Circuit Board). Производством заняты пять заводов, расположенных в США, Австралии, Мексике и Шотландии и выпускающих 9000 ПЭВМ в день [3].

Высота плат расширения системы уменьшена, что не позволяет в тех

же габаритных размерах воспроизвести на россыпи схемы, эквивалентные новым БИС.

Предлагаются две высокоуровневые архитектуры:

SAA (System Application Architecture) — содержит спецификации избранных программных интерфейсов, соглашений и протоколов и служит общей платформой при разработке прикладных систем. Уже сейчас SAA поддерживается языками Си/2 и Фортран/2 (обозначение «/2» свидетельствует только о принадлежности к PS/2, а не о новом языке программирования). Пакеты прикладных программ, разработанные на этих языках, должны легко переноситься на большие системы фирмы IBM;

MCA (Micro Channel Architecture) — разработана [4] в 1974 г. для мини-ЭВМ, позволяет увеличить скорость передачи данных внутри системы до 8 Мбайт/с, применяется начиная с 50-й модели (в 30-й модели шина IBM PC XT). Идентификация плат расширения осуществляется посылкой се-

рии программных кодов, которая для гарантии работоспособности должна быть согласована с фирмой IBM.

В августе 1987 г. IBM были объявлены две крайние в семействе PS/2 модели: модель 25 для игр, обучения, домашних систем и терминалов для связи с мини-ЭВМ System 36/38, а также модификация 80.311, выполняющая функции файл-сервера в локальной сети ПЭВМ (дискковая память до 600 Мбайт).

В июле 1988 года появились две модели: 50.061, которая скорее всего заменит резко критикуемую модель 50.021, и самая производимая в семействе — модель 70 (модификации 70.F61, 70.121 и 70.A21). Эти модели имеют настольное исполнение, более современную схемотехнику и элементную базу, и поэтому нарушают порядок возрастания производительности от младших моделей к старшим. Наибольший интерес представляет модификация 70.A21 (A — Advanced), с контроллером кэш-памяти Intel 82385, объемом кэш-памяти 64 Кбайт, с временем выборки из нее 30 нс, с рабочей частотой 25 МГц, арифметическим сопроцессором 80387 и винчестером объемом 120 Мбайт (аналог ПЭВМ DeskPro 386/25). Модель 70 — лучшая в семействе PS/2. Недостаток — всего 3 слота расширения, один из них предназначен для видеoadapterа 8514/A.

Для всех моделей PS/2 единая клавиатура — 101 клавиша (102 — в вариантах для других стран), из них 12 функциональных (световая индикация для клавиш Caps, Num, Scroll Lock). Габаритные размеры 5,8×49, 2×21,0, масса 2,25 кг.

Вариант с кириллицей отсутствует. В IBM PC XT клавиатура насчитывает 83 клавиши, в IBM PC AT — 84 (поддерживается МП Интел 8042), в 25-й модели укороченная клавиатура (40,6×19).

## Диски

Во всех моделях PS/2 используются 3,5-дюймовые дискеты, закрытые в твердый пластиковый конверт. Дискетод в 30-й модели может форматировать, считывать и записывать на дискету емкостью 720 Кбайт, а дискетоды в 50-й и 60-й моделях — 720 Кбайт и 1,44 Мбайт. Встроенный адаптер НГМД поддерживает два дискетода. Для обмена данными с диском типа винчестер в канале DMA используется специальный импульсный режим (BURST MODE), позволяющий увеличить скорость обмена в три раза по сравнению с ПЭВМ IBM PC AT (табл. 3).

В модели 50 трудно заменить медленный винчестер на более быстрый, так как используется нестандартный 50-штырьковый разъем.

Поставщиком оптических дисков для PS/2 является японская фирма

Таблица 3

## Характеристики дисковых накопителей

Емкость, Мбайт	Интерфейс	Время доступа, Мс	Диаметр, дюйм	Цена устройства, долл.
0,72	8272A	—	3,5	200
1,44	8272A	—	3,5	311
20	ST 412	80	3,5	604
44	ESDI	40	5,25	1823
70	ESDI	30	5,25	2756
115	ESDI	28	5,25	4027
314	ESDI	—	5,25	6495

Примечание. ESDI-Enhanced Small Device Interface.

Таблица 4

## Технические характеристики дисплеев

Параметры	8503	8512	8513	8514
Тип	Монохромный	Цветной	Цветной	Цветной
Размер экрана, см	30,5	35,5	30,5	40,6
Размер точки, мм	—	0,41	0,28	0,32
Число строк на экране	25/30	25/42	25/43	25/51
Число знаков на экране	80×30	80×43	80×43	146×53
Матрица знака, точек	8×16	9×16	9×16	9×16
Растр экрана, мм	640×480	640×480	640×480	1024×768
Цена, долл.	403	1018	1178	1550
Частота кадровой развертки (при адресации 640×480), Гц	60	60	60	60
Частота строчной развертки, кГц	31,5	31,5	31,5	43,5
Вес, кг	8,5	12,5	10,5	18,0
Габариты (Ш×Г), см	32,1×36,1	35,5×39,4	32,1×36,1	41,5×40,0
Высота (с подставкой), см	27,0 (31,5)	30,4 (37,0)	27,0 (31,5)	32,0
Потребляемая мощность, Вт	55	61	80	92

Примечание. Дисплей 8514 имеет независимое питание.

Matsushita. Стандартная емкость оптических дисков в настоящее время 400 Мбайт, а не 200, поэтому в PS/2 можно ожидать переход на другой тип оптических дисков (важное обстоятельство — отсутствие стандарта на формат записи этих дисков, так как уже сейчас существует пять основных форматов).

Работа с дисками большой емкости в PS/2 поддерживается пока только различными вариантами ОС типа UNIX.

## Дисплеи

Дисплеи (табл. 4) были наиболее слабой стороной IBM PC. В IBM PS/2 их характеристики «подтянуты» до средних для ПЭВМ AMIGA (Commodore) и MACINTOSH (APPLE COMPUTER).

В конфигурациях PS/2 предлагаются четыре новых дисплея. Они при-

соединены без применения переключателей, а только за счет обеспечения интегрированной поддержки в контроллерах VGA и MCGA (30-я модель). В контроллере MCGA по примеру ПЭВМ MACINTOSH используются квадратные пиксели, улучшающие качество графики (вместо обычных круглых).

Высокая частота кадровой развертки исключает появление на экране теней и мелькания. Применение аналоговой обработки сигналов изображения позволяет получить 256 цветов из палитры 262144 оттенка (табл. 5) на цветных дисплеях и 64 градации серого на монохромном дисплее, а также снижает уровень электромагнитных помех.

Цветные дисплеи обеспечивают два режима работы: стабильного четкого изображения (для САПР и полиграфии) и качественных фотоснимков

(полностью использует цветную палитру).

В модели 25 монитор конструктивно объединен с ЭВМ в одном корпусе.

Более высокие характеристики достигаются при наличии платы адаптера 8514/A (табл. 5).

В качестве новых устройств печати предлагаются более быстрый QUIETWRITER, лазерный принтер PAGEPRINTER и три модели PROPRINTER: IBM PROPRINTER II — многоскоростной, матрично-графический, имеет девять игл, загрузаемый буфер знаков, буфер печати (12 Кбайт), три режима печати по качеству, скорость (в зависимости от качества печати) 40...200 зн./с, ленту на 3 млн. знаков, формат бумаги A4, 240×144 точки на дюйм в графическом режиме, пропорциональный шрифт; IBM PROPRINTER XL имеет дополнительно 24 иглы, буфер печати 16 Кбайт, 360×180 точек на дюйм в графическом режиме; IBM PROPRINTER XL24 — аналогично PROPRINTER XL, формат бумаги A3.

Дополнительные адаптеры выполняются на отдельных платах расширения и используются для расширения конфигурации ПЭВМ: 8514/A (адаптер дисплея) заменяет адаптер для профессиональной графики, 3117 SCANNER ADAPTER/A служит для ввода текстовой и графической информации с бумаги и других материалов, HIGH SPEED ADAPTER/A — для сканера IBM 3118, компактного настольного прибора, позволяющего вводить одну страницу с текстом или изображением за 12 с с разрешением 94,5×94,5 точек на 1 см. Наличие двух шин MC и PC XT в семействе PS/2 требует для подключения одного и того же периферийного устройства двух различных плат адаптеров. В стадии разработки находится около 200 плат расширений для PS/2.

Адаптеры для локальных сетей: 3270 CONNECTION TOKEN — RING NETWORK ADAPTER/A, PC NETWORK ADAPTER/A, PC NETWORK BASEBAND ADAPTER II/A, DUAL ASYNCHRONOUS ADAPTER/A, MULTI-PROTOCOL ADAPTER/A для SDLC, HDLC, FSC и асинхронного протокола.

Новые БИС, разработанные специально для PS/2, имеют емкость интеграции до 10000 вентилей на корпусе (VGA — 12500): 72x7377 — контроллер прямого доступа (IBM DMA), восемь каналов, два из них могут поддерживать работу ведомых (SLAVE) DMA (50, 60 и 80); 72x8287 — контроллер видеографический VGA (50, 60 и 80); 72x8203 — контроллер видеографический MCGA (30); 72x7385 — БИС поддержки процессора (50 и 60); 72x8299 — контроллер ввода-вывода (50, 60 и 80).

Режимы работы дисплеев, обеспечиваемые адаптером VGA

Растр экрана, точек	Тип	Число пикселей	Число оттенков серого	Число знаков на экран	Матрица знака, точек	Совместимость		
						EGA	MCGA	CGA
320×400	АЛ/Ц	16	16	40×25	8×16	+	+	+
640×400	То же	16	16	80×25	8×16	+	+	+
320×200	Графический	4	4	40×25	8×8	+	+	+
640×200	То же	2	2	80×25	8×8	+	+	+
640×480	То же	2	2	80×30	8×16	+	+	+
320×200	То же	256	64	40×25	8×8		+	
320×350	АЛ/Ц	16	16	40×25	8×14	+		
640×360	То же	16	16	80×25	8×14	+		
720×350	То же	2	2	80×25	9×14	+		
320×200	Графический	16	16	40×25	8×8	+		
640×200	То же	16	16	80×25	8×8	+		
640×350	То же	16	16	80×25	8×14	+		
640×350	АЛ/Ц	16	16	80×25	8×8	+		
720×400	То же	16	16	80×25	9×16			
360×400	То же	16	16	40×25	9×16			
640×480	Графический	16	16	80×25	8×16			

Дополнительно для адаптера 8514/A

640×480	Графический	256	64	80×34	8×14			
1024×768	То же	256	64	85×38	12×20			
1024×768	То же	256	64	146×51	7×15			

Примечание. В графическом режиме адресуется каждая точка экрана.

Таблица 6

Основные характеристики микропроцессоров

Тип	Частота, МГц	Степень интеграции, тыс. транзисторов	Адресуемая память, Мбайт	Быстродействие, млн. операций
8086	4-10	29	1	0,8...2,5
80286	6-16	175	16	1,2...3,0
80386	12-25	275	4096	3,0...5,5
80486	20-25	1000		ОКР 1989 г.

Фирмами Western Digital, LSI Logic, VLSI Technology, Zimos, Faraday Electronics и другими выпускаются аналоги этих БИС.

Микропроцессоры, применяемые в PS/2, разработаны фирмой INTEL [5]. Они программно совместимы снизу-вверх (табл. 6). Фирме IBM принадлежало в разное время от 12 до 35 % акций фирмы INTEL.

Во второй половине 1988 г. ожидается анонсирование фирмой INTEL самого мощного из этого ряда МП 80486, который также будет полностью совместимым с МП 80386. В 1990 г. предполагается появление первых ПЭВМ с МП 80486. У конкурирующего направления уже появился и поставляется потребителям новый МП фирмы Motorola 68030 такого же класса, на который ориенти-

руется, в частности, Apple Computer; разрабатывается 68040.

ППЗУ выполнено на Интел 27256 (32К×8) и содержит в 30-й модели BIOS, аналогичный IBM PC XT, с добавлением поддержки новых графических возможностей и 3,5-дюймовых дискет.

В 50-й и 60-й моделях: CBIOS — для совместимости с IBM PC AT (Compatibility BIOS); A BIOS — для поддержки OS/2; BASIC — интерпретатор языка БЕЙСИК 3.3; POST — начальный тест, выполняемый при включении PS/2 (POWER ON SELF TEST). В ППЗУ также находится программа проверки правильности пароля, вводимого при включении ПЭВМ.

Динамическое ОЗУ выполнено на микросхемах фирмы Hitachi,

Время выборки в младших моделях 150 нс, в старших 80 нс. Во время начальной загрузки проверяются первые 512 Кбайт памяти; при обнаружении ошибки они переназначаются блоком в 1 Мбайт.

Программное обеспечение

Фирма IBM придает особое значение расширению работ по программному обеспечению. Несмотря на трудности, в 1986 г. на работу в фирму было принято дополнительно 2500 программистов и организовано новое отделение прикладных программных систем. Резко выросла доля доходов от продажи ПО. ПЭВМ серии PS/2 могут работать с несколькими ОС, основные — MS DOS 3.3 [6] и OS/2 [7, 8].

Возможности MS DOS хорошо известны; имеется ряд советских аналогов этой системы (МДОС1810, АДОС и др.). Версия 3.3 этой системы включает поддержку новых флоппи-дисков 720К и 1,44 Мбайт. IBM предложила новую кодовую таблицу для отображения национальных алфавитов, названную CODE PAGING SWITCHING (таблица 850), которая работает только на адаптерах EGA, MCGA, VGA и поддерживается устройствами печати IBM PROPRINTER, IBM QUIETWRITER. Первые 128 символов в ней совпадают с кодовой таблицей для IBM PC XT (таблица 840), а во второй половине изменено 54 символа. Как и прежняя кодовая таблица, таблица 850 не поддерживает кириллицу. Для этой системы кодирования в PC введены новые команды — CHCP и NLSFUNK, усилены GRAPTAB и MODE, объединены файлы для команд KEYB в единый файл KEYBOARD.SYS. Новая команда APPEND расширяет действие PATH на неисполняемые файлы. Команда FASTOPEN позволяет создать таблицу емкостью до 999 имен файлов для быстрого их открытия. Изменены команды FDISK, BACKUP, RESTORE, ATTRIB. Введено много мелких улучшений в программы и в документацию. В новой версии MS DOS поддерживаются четыре последовательных порта (в OS/2 предполагается восемь).

Новая операционная система OS/2, разрабатываемая IBM совместно с фирмой MicroSoft по соглашению, подписанному в августе 1985 г., имеет четыре варианта [9], два из которых уже названы: базовый (стандартный) и расширенный. Стандартный вариант OS/2, версия 1.0 (собственность MicroSoft, поставка с декабря 1987 г.) поддерживает мультипрограммирование, интерфейс прикладных программ (API), работу с 1 Гбайт виртуальной памяти и до 16 Мбайт физической памяти. В версии 1.1 (появление ожидается к октябрю 1988 г.) добавится поддержка многооконного интерфейса (Windows/Presentation Manager), цена

325 долл. Расширенный вариант OS/2, версия 1.0 (собственность IBM, поставка с июля 1987 г.) имеет дополнительно встроенную реляционную СУБД, совместимую с СУБД DB2 для IBM 370, диспетчер локальной сети (Communication Manager), эмулятор терминалов: IBM 3270, 3101 и DES VT100. В версии 1.1 (поставка с ноября 1988) появится Windows/PM, поддержка сетей (LAN Server) IBM Token Ring и PC Network, цена 795 долл.

Остальные два варианта: в 1989 г. ожидается появление OS/2, поддерживающей связь PC с другими ЭВМ фирмы IBM; в 1990 г. — OS/2, полностью поддерживающая возможности МП 80386.

Во всех версиях OS/2 используются трансляторы с языков программирования: БЕЙСИК/2, Фортран/2, Паскаль/2, Си/2, Кобол/2, Макроассемблер/2.

OS/2 может выполняться на всех ПЭВМ фирмы IBM, работающих на INTEL 80286, в том числе на IBM PC AT и XT-286. Для выполнения программ, написанных под MS-DOS, в OS/2 имеется так называемый «блок совместимости» (DOS Compatibility Box или кратко — 3.x Box), под который жестко резервируется 640 Кбайт ОЗУ. Стандартный вариант 1.0 требует для своей работы при наличии блока совместимости 2 Мбайт ОЗУ, без него — 1,5 Мбайт. Размер ядра OS/2 от 90 до 150 Кбайт в зависимости от конфигурации системы. Так как базовые варианты PS/2 имеют меньшую память, то необходима установка дополнительных плат расширения памяти. На XT с акселераторными платами на МП 80286 OS/2 требует значительной адаптации, и пока реализаций нет.

OS/2 обеспечивает возможность одновременной работы с 12 прикладными системами в защищенном режиме (12 сессий или сеансов), причем каждая из систем имеет виртуальный экран в ОЗУ, который может быть отображен на физический экран монитора. После нажатия клавиши Alt-Esc осуществляется переход от одного сеанса к другому, при этом сеанс получает физический экран. Кроме того, один сеанс (13-й) возможен в блоке совместимости в действительном режиме [10].

Существующие версии OS/2 построены на основе файловой системы MS DOS и поэтому не избавлены от основного недостатка этой системы — наличия ограничения в 32 Мбайт на логический размер диска.

Работы над OS/2 ведутся указанными фирмами параллельно по единым спецификациям, но программные продукты отличаются не только названиями, но и рядом улучшений, которые IBM независимо вносит в свою версию.

OS/2 является ключевым элементом SAA, системы протоколов и стандартов IBM.

Состав SAA:

API — около 220 системных вызовов OS/2. Подмножество API, называемое «Family API», содержит около 70 системных вызовов, имеющих аналоги в MS DOS, и обеспечивает мобильность ПО, транслируемого новыми компиляторами («/2») в MS DOS;

SQL — язык определения и манипулирования данными, ориентированный на реляционную модель данных. С 1986 г. является стандартом в США [11];

стандартизированные средства поддержки локальных сетей.

Возможность с помощью SAA переносить ПО между ЭВМ разных классов IBM рекламируется, но учитывая, что эти машины имеют сугубо различную архитектуру, такая вертикальная мобильность [12] требует больших доработок системного ПО и принципиально неосуществима в полной мере.

Фирмой MicroSoft уже выпущен пакет инструментальных средств поддержки разработки программ для OS/2 (MS OS/2 Developer's Tools Kit); аналогичный пакет разрабатывается фирмой IBM.

Для модели 80 [13] самой фирмой IBM совместно с фирмой Locus Computing Corp разрабатывается ОС AIX (Advanced Interactive Executive) (готовность в начале 1988 г.). AIX позволит объединить MS DOS и UNIX, как это сделано в PC RT [14].

Версия ОС XENIX (XENIX System V/386), ориентированная на МП 80386, разрабатывается фирмой Interactive Systems по соглашению с фирмой MicroSoft (готовность — 1 квартал 1988 г.).

Фирмы ASHTON-TATE, BORLAND INTERNATIONAL, LOTUS DEVELOPMENT Corp., MICRORIM и другие объявили о совместимости своих основных программных продуктов с PS/2.

Отметим, что реализация SAA, как и всего системного ПО для PS/2, требует больших усилий и времени. Поэтому до конца 1989 г. и даже середины 1990 г. будет чувствоваться отсутствие ПО, реализующего все возможности PS/2.

Защита информации обеспечивается системой паролей в OS/2. Пароль (1..7 знаков) записывается в энергонезависимое (CMOS) ОЗУ и проверяется при включении питания после прохождения начальных тестов. Однако если содержимое CMOS ОЗУ потеряно (например, выщута аккумуляторная батарейка), то защищенные паролями файлы становятся доступными. Чтобы этого не произошло, а также для блокирования клавиатуры имеется механический ключ, который

необходимо вставить для включения ПЭВМ и снятия верхней панели. Аналоговый вывод на экран дисплея не позволяет считывать с него информацию.

### Заклучение

«Новое поколение» ПЭВМ фирмы IBM — воплощение глобального проекта. Его появление, безусловно, яркое событие в истории ПЭВМ. Многие его черты были реализованы значительно раньше в ПЭВМ AMIGA фирмы COMMODORE и ПЭВМ MACINTOSH фирмы APPLE, в ранее анонсированных ПЭВМ на микропроцессоре INTEL 80386 фирмы COMPAQ COMPUTER CORP. Роль PS/2 заключается в том, что благодаря системному подходу, использованному в этом проекте, и массовому производству, она устанавливает некоторый средний уровень (своего рода стандарт), по которому сейчас оцениваются все разработки в этой области.

Стремление воплотить в одном проекте одновременно большое число новых возможностей характерно для IBM. Например, IBM 360/370, ПЛ/1 и OS/360. Ситуация с разработкой ПО аналогична ситуации, сложившейся к концу 60-х годов: появление большого числа нестыкующихся между собой версий, реализация которых громоздка и неочевидна. Как и в OS/360, где затраты на системные нужды достигают 60%, в OS/2 существует опасность такого роста системных затрат. На архитектуру семейства PS/2 оказали слишком большое влияние конъюнктурные соображения (это касается в первую очередь MCA), что должно вызвать при его развитии накопление неадекватных решений либо, что более вероятно, почти полную смену моделей.

Неоценимо косвенное влияние появления семейства PS/2 на разработчиков новых ПЭВМ. Массовый выпуск ПЭВМ, совместимых с PS/2, начал с июля 1988 г.; объявлено об альтернативных архитектурах, например, фирма Fujitsu в своих ПЭВМ G Series предлагает архитектуру SIA (System Integration Architecture). В США рынок ПЭВМ уже насыщен (более 25 млн. ПЭВМ). Поэтому появились два варианта перехода на МП 80386 модификацией уже существующих машин типа IBM PC XT/AT:

заменой основной системной платы с сохранением шины XT/AT; установкой платы акселератора (ускорителя), например для AT плата INBOARD 386 фирмы Интел, которая содержит МП 80386-16, 80387 и до 2 Мбайт ОЗУ. Стоимость такой модификации почти в три раза ниже покупки новой ПЭВМ.

Продолжение см. на с. 80



УДК 681.3.06

И. Г. Муттик, Н. А. Самарин

## ОДИН ПОЛЕЗНЫЙ АЛГОРИТМ

Наибольшее распространение для персональных компьютеров, построенных на базе микропроцессоров КР1810ВМ86 и КР1810ВМ88, получили операционные системы (ОС), совместимые с ОС MS-DOS, разработанной фирмой Microsoft. К числу наиболее популярных пакетов прикладных программ для этих ОС относятся пакеты LOTUS 1—2—3 и SYMPHONY. Они особенно выделяются гибкостью средств выдачи графики на экран персонального компьютера и его матричный принтер.

Несмотря на все достоинства этих программ, каждый пользователь рано или поздно ощутит их оторванность от вычислительных средств языков высокого уровня. Встает проблема передачи числовой и символьной информации из программы на алгоритмическом языке в программы, реализующие высококачественное представление данных в графическом виде.

Наиболее простой способ передачи данных из программ на языке высокого уровня — формирование файла в формате, совместимом с представлением, который использует программа LOTUS 1—2—3. Алгоритм программы, реализующий такой вариант передачи информации и написанный на языке БЕЙСИК, представлен на рис. 1. Исходной информацией для программы передачи (строки 1000...4010) служит двумерный массив строковых констант В\$. Если строковая переменная есть — символьное представление числа, то производится преобразование этого числа. Иначе — передается вся символьная последовательность (см. строку 1220). Это сделано с целью демонстрации алгоритма, а для реального использования предпочтительнее непосредственно обращаться к подпрограмме занесения в файл строковой переменной А\$ (строки 2000...2100) и к подпрограмме занесения в файл числа X (строки 3000...3220). Кроме того, можно не использовать массив временного хранения В\$.

Работа программы возможна и в режиме CHAIN. Тогда из текста программы следует удалить строки 130...270, а параметры (массив В\$, число строк таблицы NR, число колонок NC, имя файла FILE\$) передавать оператором COMMON.

Программа может быть использована как в интерпретаторе языка БЕЙСИК, так и после обработки компилятором (никаких изменений для этого в программу вносить не требуется). Запись в файл 1000 действительных чисел занимает 340 с при работе интерпретатора и 35 с при работе скомпилированной программы. При этом создается файл (типа .WKS) объемом 17 Кбайт. Программа LOTUS 1—2—3 прочитывает его за 9 с.

Дополнительные комментарии требуются к подпрограмме записи чисел. Дело в том, что LOTUS 1—2—3 использует представление действительного числа в восьми байтах, несовместимое с представлением числа в языке БЕЙСИК. Представление числа в программе LOTUS 1—2—3 изображено на рис. 2. Процесс приведения числа к этому виду требует введения дополнительного массива ВТ (8), в котором хранятся байты результатов, записываемые потом в файл. Байтам ВТ (1), ВС (2) всегда присваивается нулевое значение, поскольку точность чисел в языке БЕЙСИК (семь зна-

```

40 '*****
20 '
30 ' ПЕРСОНАЛЬНЫЙ КОМПЬЮТЕР : ЕС1840 * ИСКРА 1130.11 * IBM PC
40 ' ОПЕРАЦИОННАЯ СИСТЕМА : MS-DOS * АЛЬФА-ДЖОС * DOS-16
50 ' ЯЗЫКИ ПРОГРАММИРОВАНИЯ : BASIC (ИНТЕРПРЕТАТОР И КОМПИЛЯТОР)
60 '
70 '
80 '
90 ' СОЗДАНИЕ МАССИВА В$ ДЛЯ ПОСЛЕДУЮЩЕЙ ОБРАБОТКИ
100 '
110 DEF SEG : DEFINT A-V
120 DIM B$(20,500),BT(8)
130 INPUT " ROWS : ",NR
140 INPUT " COLUMNS : ",NC
150 FOR I=1 TO NC
160 PRINT : PRINT "====( ENTER COLUMN *I* )====" : PRINT
170 FOR J=1 TO NR
180 PRINT "SPREADSHEET(*I*,*J*):LINE INPUT ")*",B$(I-1,J-1)
190 NEXT J,I
200 PRINT CHR$(7) : INPUT "INPUT FILENAME : ",FILE$
210 '
220 ' ИМЯ ФАЙЛА ДЛЯ ПРОГРАММ SYMPHONY И LOTUS 1-2-3
230 ' ДОЛЖНО ИМЕТЬ РАСШИРЕНИЕ .WKS
240 '
250 I=INSTR(FILE$,".") : IF I=0 THEN FILE$=FILE$+".WKS" : GOTO 1080
260 FILE$=LEFT$(FILE$,I)+".WKS"
270 '
3000 '*****
3010 ' ЭТА ЧАСТЬ ПРОГРАММЫ ЗАПИСЫВАЕТ МАССИВ В$ В ФАЙЛ С ИМЕНЕМ
3020 ' FILE$ * ФОРМАТ ЗАПИСИ СОВМЕСТИМ С ПАКЕТАМИ ПРОГРАММ
3030 ' SYMPHONY И LOTUS 1-2-3 (VER.1A).
3040 '*****
3050 '
3060 ' ОТКРЫТИЕ ФАЙЛА FILE$. ОПРЕДЕЛЕНИЕ АДРЕСА БУФЕРА ФАЙЛА
3070 '
3080 OPEN "B":#1:FILE$
3090 FBADR=VARPTR(0)+160 : A=FBADR
3100 DATA 0,0,2,0,4,4,6,0,8,0,0,0,0,0
3110 FOR I=0 TO 13 : READ L : GOSUB 4000 : NEXT I *ЗАПИСЬ ЗАГОЛОВКА ФАЙЛА
3120 '
3130 ' ЗАПИСЬ РАЗМЕРОВ ДВУМЕРНОГО МАССИВА В$ В БУФЕР ФАЙЛА
3140 '
3150 L=(NC-1) MOD 256 : GOSUB 4000 : L=(NC-1)*256 : GOSUB 4000
3160 L=(NR-1) MOD 256 : GOSUB 4000 : L=(NR-1)*256 : GOSUB 4000
3180 '
3190 FOR K=0 TO NC-1
3200 FOR J=0 TO NR-1
3210 A$=B$(K,J) : X=VAL(A$)
3220 IF X=0 AND A$<>"0" THEN GOSUB 2030 ELSE GOSUB 3030
3230 NEXT J,K
3240 '
3250 L=1 : GOSUB 4000 : L=0 : GOSUB 4000 : GOSUB 4000 : GOSUB 4000
3260 PUT #1 : CLOSE #1 : END
3270 '
3280 '
3290 ' ПОДПРОГРАММА ЗАПИСИ СТРОКОВОЙ ПЕРЕМЕННОЙ А$ В БУФЕР ФАЙЛА
3300 '
3310 L=15 : GOSUB 4000 : L=0 : GOSUB 4000 : L=LEN(A$)+7 : GOSUB 4000
3320 L=0 : GOSUB 4000 : L=255 : GOSUB 4000 : L=(K MOD 256) : GOSUB 4000
3330 L=K*256 : GOSUB 4000 : L=(J MOD 256) : GOSUB 4000 : L=J*256 : GOSUB 4000
3340 L=34 : GOSUB 4000 * ЗАБЕСИТЬ CHR$(34) - ПРИБИЗНАК СТРОКОВОЙ ПЕРЕМЕННОЙ
3350 '
3360 FOR I=1 TO LEN(A$) : L=ASC(MID$(A$,I,1)) : GOSUB 4000 : NEXT I
3370 L=0 : GOSUB 4000
3380 RETURN
3390 '
3400 ' ПОДПРОГРАММА ЗАПИСИ ЧИСЛА X В БУФЕР ФАЙЛА
3410 '
3420 L=14 : GOSUB 4000 : L=0 : GOSUB 4000 : L=13 : GOSUB 4000
3430 L=0 : GOSUB 4000 : L=255 : GOSUB 4000 : L=(K MOD 256) : GOSUB 4000
3440 L=K*256 : GOSUB 4000 : L=(J MOD 256) : GOSUB 4000 : L=J*256 : GOSUB 4000
3450 '
3460 BT(1)=0 : BT(2)=0
3470 IF X=0 THEN FOR I=3 TO 8 : BT(I)=0 : NEXT I : GOTO 3210 * СЛУЧАЙ X=0
3480 IF X<0 THEN SIGN=-1 : X=-X ELSE SIGN=1
3490 DEG=INT((COS(X)*1.426951)+X*X/4*(-DEG))
3500 IF X=2 THEN X=X/2 : DEG=DEG/2 : GOTO 3110
3510 IF X<1 THEN X=X/2 : DEG=DEG/2 : GOTO 3120
3520 X=X-1
3530 DT=INT(X*16) : X=X*16-DT
3540 BT(6)=INT(X*256) : X=X*256-BT(6)
3550 BT(5)=INT(X*256) : X=X*256-BT(5)
3560 BT(4)=INT(X*256) : X=X*256-BT(4)
3570 BT(3)=INT(X*256) : X=X*256-BT(3)
3580 DEG=DEG*1023 : BT(8)=INT(DEG/16) : BT(7)=(DEG-BT(8)*16)*16+BT
3590 IF SIGN<0 THEN BT(3)+BT(8)+128
3600 FOR I=1 TO 8 : L=BT(I) : GOSUB 4000 : NEXT I
3610 RETURN
3620 '
3630 '
3640 ' ЗАПИСЬ ОДНОГО БАЙТА В БУФЕР ФАЙЛА FILE$ И
3650 ' СБРОС БУФЕРА (ДЛИНА=128 БАЙТ) ПРИ НАПОЛНЕНИИ
3660 '
3670 POKE A,L : A=A+1 : IF A>FBADR+128 THEN RETURN
3680 A=FBADR : PUT #1 : RETURN

```

Рис. 1

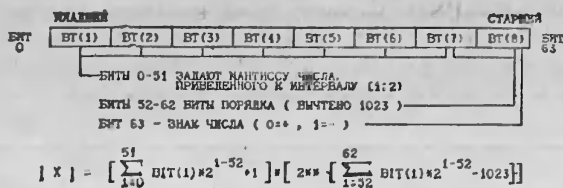


Рис. 2

чащих цифр) ниже, чем в программе LOTUS 1—2—3 (10 значащих цифр). Если требуется передавать больше значащих цифр, то в программе нужно перейти на оперирование числом X с двойной точностью и заполнять байты BT (1) и BT (2) аналогично байтам BT (6)—BT (3).

Телефон 939-11-47, Москва

Статья поступила 04.12.86

УДК 681.3

Ю. М. Быков, Г. А. Жемчугов, Ю. Л. Левин, Г. С. Мингалеев

## МНОГОКРИТЕРИАЛЬНАЯ СТРУКТУРНО-ПАРАМЕТРИЧЕСКАЯ ОПТИМИЗАЦИЯ СЛОЖНЫХ СИСТЕМ

Для ПЭВМ средней производительности типа ЕС1840/41, «Искра 1030/31», РС/ХТ, Правец 16 и другие во ВНИИЭМ разработан пакет прикладных программ для структурно-параметрической оптимизации сложных технических систем на этапе предварительного проектирования (в крупных блоках) СТРОП.

СТРОП выполняет следующие функции:

- создает и поддерживает базы данных решаемых на нем задач;
- организует ввод, коррекцию и контроль справочных и исходных данных задач (в диалоговом режиме);
- выполняет многовариантные многокритериальные оптимизационные расчеты (в интерактивном и автоматическом режимах);
- выполняет анализ решений и организует выдачу печатных материалов;
- ведет «оживленный» диалог с пользователем;
- представляет развернутые инструкции и пояснения в каждой диалоговой ситуации;
- обучает пользованию;
- организует различные режимы взаимодействия с внешними программами и ДОС.

Множество допустимых структур изделия задается структурным графом, каждому узлу которого соответствует вектор в пространстве «цен». Вектор «цен» изделия в данной версии пакета определяется как сумма векторов «цен» реализаций компонентов. Компоненты векторов «цен», в свою очередь, являются функциями в некотором пространстве параметров (фазовом пространстве).

В данной версии СТРОП рассчитан на дискретную многокритериальную оптимизацию изделия: число компонентов — до 32, число реализаций каждого компонента — до 16, число критериев — до 5, пространство параметров может быть 10-мерным, каждая из десяти осей может содержать до 255 координатных точек, общее число фазовых точек — от нескольких тысяч до гибких дисков до десятков тысяч для жестких дисков.

Вычисление оптимальной структуры размером  $32 \times 16$  в одной фазовой точке занимает 10...15 с; подготовительные операции и запись результатов на диск — 5...10 с. Объем оперативной памяти — 210 Кбайт. Потребности в дисковой памяти ограничиваются одним гибким диском для программы пакета и банка инструкций и пояснений и одним гибким диском на базу данных для 2—3 задач.

Программно-математическое обеспечение СТРОПа может быть использовано в качестве ядра программного обеспечения для разработки пользовательских версий, содержащих дополнительные сервисные и аналитические функции для решения конкретного класса задач пользователя.

Использование принципов организации программ, принятых в СТРОПе и библиотеке его процедур, существенно ускоряют разработку новых задач, обеспечивает их взаимную стыковку и повышает технико-эксплуатационные характеристики проблемно-ориентированных пакетов за счет унификации программных и диалоговых средств.

Оригинальный отечественный пакет СТРОП по своим эксплуатационным характеристикам не уступает зарубежным пакетам подобного класса. Он в большей степени рассчитан на конечного пользователя — предметного специалиста, не знакомого с программированием. Телефон 923-39-36, Москва, Мингалев Г. С.

Сообщение поступило 28.06.88

### УВАЖАЕМАЯ РЕДАКЦИЯ!

В заметке «Из цикла писем читателей: Васик, УНИКС и др.» («МП». — 1988 — № 2. — С. 93) рассказывается о «странной» операционной системе УНИКС, странной ее документации и более чем странном названии. Пожалуйста разработчикам, ответственным за все эти странности, удовлетворить любопытство заинтригованных читателей и ответить на вопросы, заданные в заметке.

Аббревиатура УНИКС — это общее название разработанного нами унифицированного инструментального комплекса мобильных программных средств, куда входит и склоняемая в заметке операционная система для ПЭВМ «Искра 226».

Рискуя вызвать удивление, все же сообщим непреложный факт: «Искра 226» не имеет архитектурных аналогов, ни отечественных, ни зарубежных. Распространенное мнение о происхождении ее от Wang-2200 ошибочно. Основным средством программирования на «Искре 226» был язык БЕЙСИК (чтобы не путать его с «Васиком, другой Ириши», прочтите заметку), сходный с БЕЙСИКОМ для Wang-2200, но отличный от реализаций этого языка для СМ, ЕС и современных персональных ЭВМ.

Как следствие изолированности «Искры 226», обмен программами был невозможен, а сохранение наработок — более чем проблематично. Это вызывало вполне обоснованную озабоченность пользователей. Но если аппаратная совместимость недостижима, то программную можно обеспечить. Так возникла идея создания операционной системы УНИКС-226 по спецификациям мобильной ОС UNIX.

Инициаторами проекта были специалисты Госплана СССР. Разработка велась в сотрудничестве с коллективами, усилиями которых реализованы мобильные операционные системы для СМ и ЕС ЭВМ. Большую помощь оказали создатели аппаратных средств ПЭВМ «Искра 226». Наконец, неизменной была поддержка и конструктивная критика пользователей системы.

Что же странного замечено в нормальной практике проектирования? То, что заказчиками работы были организации, имеющие большой и разнородный парк вычислительной техники и заинтересованные в эффективном его использовании? То, что учитывались требования самых разных категорий пользователей, невзирая на степень их ведомственного «родства» с разработчиком? Или то, что разработчиками являются специалисты института «Тяжиромавтоматика», успешно проектирующего программные комплексы и автоматизированные системы всего-навсего три десятка лет? О каких пикантных «плодах сотрудничества» говорится в связи с системой, внедренной в промышленную эксплуатацию на заводах, вычислительных центрах и конструкторских бюро (не на «плодотворной ниве» диссертаций и отчетов)?

Системный программист из Куйбышева т. Еременко едко замечает, что «пухлый том» нашей документации содержит сведения, которые «можно найти в любой книге по UNIX». Признаемся, что не один, а одиннадцать томов документации (около 800 листов) включают наряду с информацией, сугубо специальной, также и общезвестную. И такую же вопиющую картину можно увидеть, «сопоставляя» сотни томов руководства всех без исключения операционных систем как отечественных, так и зарубежных. Разумно ли поступают разработчики, затрачивая время и труд на «пухлые тома»? Не проще ли писать коротенькие обзоры со ссылками на популярные книги? А что делать, если книг еще нет? Ведь наша документация выпущена в 1983 г., а первая публикация на русском языке по упомянутой теме датирована 1985 г. Вправе ли мы были отсылать пользователей к «любимым книгам» на иностранных языках?

Кстати, т. Еременко могла бы обратить внимание на дату выпуска проекта (читатели заметки лишены этой возможности, так как в дословно процитированном тексте именно дата опущена). Вместо этого внимание обращается на фразу «пользователи, получившие систему неофициальным путем, обслуживанию не подлежат», она даже выделена жирным шрифтом. Что в ней странного? Путей что ли таких не бывает? Но ведь получила же редакция «вполне официально» (?) наши распечатки из города Куйбышева, где мы никогда и ничего не внедряли.

Возможно, неясен термин «обслуживание». В последних выпусках нашей документации он уточняется: «разработчики гарантируют пользователям бессрочное сопровождение и поставку всех усовершенствований и новых версий системы УНИКС-226».

Мы можем понять «особенно сильное впечатление», произведенное на т. Еременко вызывающим названием УНИКС. Такое созвучие с UNIX! Недопустимый промах именно для «системного программиста», чья работа зачастую сводится к переименованию программ.

Спешим успокоить бдительность т. Еременко. Поскольку архитектура «Искры 226» настолько своеобразна, что нет даже похожих реализаций ОС UNIX, пришлось нам систему программировать «с нуля», да добрый кусок ее — в машинных кодах, ибо Си-компилятор также делали сами, параллельно. Ну а если шапка на голове не горит, почему бы не подчеркнуть в названии причастность скромной разработки к замечательному семейству программных средств, приобретающих ныне статус мирового стандарта?

Вероятно, сопоставление в одном ряду Васик — Basic

и УНИКС — UNIX возникло из-за недостатка информации о нашей системе. Но мы не засекречивали свою разработку; значительную часть приведенных выше сведений специалисты могли бы почерпнуть из сообщения в «МП». — 1987. — № 5, — С. 41.

Пользуясь случаем, отметим новые возможности ОС УНИКС-226, не отраженные в сообщении.

Реализованы средства режима реального времени и асинхронного ввода-вывода, возможность работы с любым (в том числе и нестандартным) периферийным оборудованием по произвольным протоколам обмена. В комплект поставки включены кросстрасляторы подмножества языка Си и ассемблера МП серии К580 для программирования БИФ телекоммуникационного и последовательного интерфейсов. С целью ускорения программ значительная часть системной библиотеки языка Си переписана в машинных кодах. Усовершенствован терминальный интерфейс, даны средства программирования графического канала, упомянутый в заметке редактор ed (но не ЕД — команды систем типа UNIX принято именовать строчными латинскими, но не прописными русскими буквами) работает теперь и в экранном режиме.

Использование УНИКС-226 в десятках организаций и в самых разных приложениях (от учрежденческой АРМ до управления контрольной-измерительной аппаратурой) привело к значительному расширению фонда прикладного программного обеспечения. Сейчас на «Искре 226» доступны ИПС и СУБД, пакеты программ численного анализа, математической статистики и прогнозирования, машинной графики и текстового документирования. В быстром наращивании программного фонда «повинна» мобильность операционной среды: Си-программы из ИНМОС, ДЕМОС и даже из ОС РВ и MS-DOS переносятся в УНИКС-226 с минимальными трудозатратами. Имеется кросс-система программирования, позволяющая получать выполнимые модули для «Искры 226» на СМ ЭВМ.

В заключение разрешите обратиться к пользователям ПЭВМ «Искра 226». Как известно, машина эта снимается с производства. Возникает угроза потери громадного фонда прикладных программ на БЕЙСИКе. Мы сделали проработку (на уровне эскизного проекта) конвертора БЕЙСИК-Си, который позволил бы автоматизировать до 80 % работ по переводу программ. Актуальна ли эта тема, и если да, то не возьмется ли кто финансировать рабочий проект? Наши телефоны: (8-044) 290-82-56, 290-84-09, Киев.

В. М. Арнаксыд, А. Е. Дорфман, И. Б. Володарский

## ЯЗЫКОВЫЕ СРЕДСТВА МИКРОЭВМ

УДК 681.3

Э. М. ПРОЙДАКОВ

### ИСПОЛЬЗОВАНИЕ МАКРООПРЕДЕЛЕНИЙ В ЯЗЫКЕ ПЛ/М

Язык ПЛ/М [1—10] — это семейство машинно-ориентированных языков (ПЛ/М-80, ПЛ/М-86, ПЛ/М-286, ПЛ/М-386, ПЛ/М-51 и ПЛ/М-96) высокого уровня для программирования однокристальных ЭВМ, микропроцессорных контроллеров и систем. Цифры в названиях языков указывают на базовый тип поддерживаемого микропроцессора. Как правило, существует несколько реализаций каждого из перечисленных языков, включая кроссовый вариант. Языки семейства ПЛ/М имеют блочную структуру, строго типизированные данные и внешний набор библиотек ввода-вы-

вода. При всеобщем увлечении языком Си следует отметить, что ПЛ/М проще, надежнее, читабельней, лучше учитывает специфику программирования перечисленных выше систем и

архитектурные особенности микропроцессоров, для которых он реализован. Перечислим операционные системы, в которых реализованы языки семейства ПЛ/М:

ОС	Язык
CP/M-80, MP/M-80 МИКРОС-80, МикроDOS TS1S-II, TS1S-III, DOS1800, DOS1810 RMX-86, DOS1810, RMX-286	ПЛ/М-80
CP/M-86, МИКРОС-86, MP/M-86, ССР/М-86 VAX VMS MSDOS, МДОС1810	ПЛ/М-80, ПЛ/М-86, ПЛ/М-51 ПЛ/М-86, ПЛ/М-286 ПЛ/М-86, ПЛ/М-80 ПЛ/М-86, ПЛ/М-286, ПЛ/М-386 ПЛ/М-86, ПЛ/М-80, ПЛ/М-51

Макроопределение в языке ПЛ/М позволяет присваивать имя (идентификатор) произвольной строке символов и в исходном тексте программы писать его вместо этой строки.

С помощью макроопределений программист может создать себе некоторый диалект языка, обеспечивающий психологический микроклимат для работы. При умелом использовании макроопределений читаемость программ улучшается: четче виден творческий почерк и индивидуальность автора программы. Для программистов, работавших ранее на других языках, некоторый комфорт может быть создан аналогиями с помощью макроопределений [11].

Имя строке символов присваивается оператором DECLARE:

```
DECLARE <идентификатор>
LITERALLY <строка>;
```

где <> — метасимволы, в реальных операторах не пишутся.

При трансляции программы компилятор всюду, где встретится идентификатор с атрибутом LITERALLY, заменяет его на строку, заданную после LITERALLY. В качестве строки символов может быть использован произвольный текст, не противоречащий синтаксису ПЛ/М.

Для удобства в первую очередь следует переопределить некоторые ключевые слова самого языка ПЛ/М и прежде всего само слово LITERALLY (что, кстати, всегда и делают программисты фирмы Интел, где язык ПЛ/М используется наиболее широко). Например, один из вариантов:

```
LIT — сокращение слова LITERALLY;
```

```
IS или AS для получения англо-читаемого макроопределения:
```

```
DECLARE AS LITERALLY 'LITERALLY'; Это позволяет вместо объявления
```

```
DECLARE FNMSG(*) BYTE INITIAL
(CR,LF, 'ИМЯ ФАЙЛА: ');
```

записать более короткое и ясное

```
DCL FNMSG STRING (CR,LF, 'ИМЯ ФАЙЛА: ');
```

(Психологические аспекты выбора имен команд рассматриваются в статье [11].)

Для неизменяемых строк текста тип определяется как

```
DCL TEXT AS '(*) BYTE DATA';
```

Для работы с символами по аналогии с языком Си можно определить тип CHAR:

```
DCL CHAR AS 'BYTE';
```

можно сразу использовать в следующем макроопределении или операторе языка. Имеется тенденция к применению следующих макроопределений (см. предыдущую программу).

Предлагаемый далее набор макроопределений вводит в язык ПЛ/М несколько типов данных, которые в нем отсутствовали.

1. Логические данные. Они представлены в памяти одним байтом на каждую переменную или элементом одномерного логического массива и принимают значения 0FFH для состояния «истина» и 00H — для состояния «ложь»:

```
DCL BOOLEAN AS 'BYTE',
TRUE AS '0FFH',
FALSE AS 'NOT TRUE';
```

Эти макроопределения позволяют записывать операторы вида

```
DCL LEVELFLAG BOOLEAN;
...
LEVELFLAG=TRUE;
```

2. Строковые данные. Строки символов — важный объект в языке программирования. Однако в языке ПЛ/М первоначально стандарта на представление строк не было, он появился начиная с языка ПЛ/М-86. В ПЛ/М-80 каждый программист имеет возможность создать собственный набор процедур для работы со строками. Следующее макроопределение делает более удобным объявление строки символов:

```
DCL STRING AS '(*) BYTE INITIAL;
```

3. Указатели. В ПЛ/М-80 этот тип данных неявно вводится при использовании дот-операции; в языке ПЛ/М-86 он встроенный и определяется с помощью атрибута POINTER в операторе DECLARE. В ПЛ/М-80 указатели можно ввести для улучшения читаемости программ:

```
DCL POINTER AS 'ADDRESS';
```

Кроме того, к именам переменных, используемых в качестве указателей, рекомендуется добавлять окончания \$P или \$PTR, например: CHR\$PTR.

Константы в программе можно определить с помощью

```
DCL CONST AS 'BYTE DATA';
(ИЛИ 'WORD DATA');
```

Язык ПЛ/М имеет блочную структуру и, как это принято в современных языках, только один тип скобки, закрывающей блок (END;). Поэтому при написании программ с большим числом вложенных блоков, даже при аккуратном соблюдении отступов, не всегда сразу видно, какой «END;» к какому блоку относится. Типична такая запись

```
END;
END;
END;
```

Если именовать закрывающую скобку блока в соответствии с его типом, то читаемость программы заметно улучшится:

```
DCL ENDDO AS 'END';
ENDCASE AS 'END';
ENDIF AS 'END';
ENDPROC AS 'END';
ENDREPEAT AS 'END';
ENDWHILE AS 'END';
```

Вышеизложенное в полной мере относится и к используемым в языке Си фигурным скобкам, которые к тому же на многих 8-разрядных ПЭВМ отображаются буквами Ш и Ш.

При определении имен констант, портов ввода-вывода и управляющих символов употребление макроопределений аналогично по своей сути псевдокоманде EQU в Макроассемблере:

```
DCL SUM AS 'A+B+C';
DCL LINE AS '-A-E-X-N-E-E-V-N-E-';
DCL BUFFER AS 'STRUCTURE
(LEN BYTE,
ACTUALLEN BYTE,
SECTOR(128) CHAR)';
```

Набор объявлений, облегчающий работу с управляющими символами

Так как ПЛ/М является языком с явно определяемыми типами переменных, определений в программе пишется достаточно много, поэтому целесообразно заменить ключевые слова DECLARE и PROCEDURE на принятые для них сокращения:

```
DECLARE DCL AS 'DECLARE';
DCL PROC AS 'PROCEDURE';
```

Как видно из этих макроопределений, объявленный идентификатор

```
DCL EXT AS 'EXTERNAL';
```

```
REPEAT AS 'DO WHILE';
FOREVER AS 'WHILE 1';
PUBLIC AS 'COMMON';
THEN AS 'THEN DO';
ELSE AS 'END; ELSE DO;';
ELIF AS 'END; ELSE IF';
WORD AS 'ADDRESS';
DWORD AS '(2) WORD';
```

и часто встречающимися константами:

```
DCL CTRLRC AS '03H',
  CTRLRQ AS '11H',
  CTRLRS AS '13H',
  CTRLRX AS '16H';
DCL CR AS '0DH', /* ВОЗВРАТ КАРЕТКИ */
  LF AS '0AH', /* ПЕРЕВОД СТРОКИ */
  CRLF AS 'CR,LF', /* НОВАЯ СТРОКА */
  BELL AS '07H', /* ЗВОНОК */
  TAB AS '09H', /* ТАБУЛЯЦИЯ */
  BACKSPACE AS '0BH',
  SPACE AS '20H', /* ПРОБЕЛ */
  NULL AS '00H',
  NUL AS '0000H', /* НЕСУЩЕСТВУЮЩИЙ
  УКАЗАТЕЛЬ */
  MAXINT AS '0FFFFH'; /* МАКСИМАЛЬНОЕ
  ЧИСЛО */
```

и т. д. в зависимости от задачи, устройств и других обстоятельств.

Посредством макроопределений удобно записывать константы для программирования БИС, например для последовательного интерфейса КР580ВВ51А:

```
DCL BUFSIZE AS '4096',
  DATA31PORT AS '0F6H',
  STATUS31PORT AS '0F7H',
  WRTSTATUS AS '00000001B',
  RDSTATUS AS '10B',
  RESETB1 AS '01000000B',
  MODEB1 AS '110001110B',
  CMD1 AS '00100101B';
```

Пример записи процедуры с использованием введенных в статью макроопределений:

```
PRINT: PROC (PTR); /*ЭТА ПРОЦЕДУРА ВЫВОДИТ
НА ЭКРАН СООБЩЕНИЕ,
ПРИЗНАКОМ КОНЦА
КОТОРОГО ЯВЛЯЕТСЯ 03. */
DCL PTR POINTER,
  CHR BASED PTR CHAR;
REPEAT CHR<NULL;
  CALL CD(CHR);
  PTR=PTR+1;
ENDREPEAT;
ENDPROC PRINT;
```

Длина макроопределений (до 255 символов) позволяет использовать их и при определении структур. Кроме того, макроопределения могут быть

вложенными (глубина вложенности неограничена).

Все необходимые макроопределения рекомендуется сформировать в одном или нескольких файлах (обычно расширение имени файла .LIT) и включать их в исходный текст программы с помощью директивы

```
$INCLUDE (<имя — файла>),
```

где имя файла — имя файла с макроопределениями.

Более широкие возможности конструирования диалектов над языком ПЛ/М появляются при создании отдельных препроцессоров, например с использованием универсального макропроцессора STAGE2.

Макроопределения не влияют на объем объектного кода программы. Для улучшения качества объектного кода рекомендуется изучить получаемые листинги трансляции, в которых при задании директивы \$CODE видно, как реализуется тот или иной оператор. Проведенный автором подсчет коэффициента расширения для языка ПЛ/М-80 (версия 3.1) показал, что в среднем для системных программ один исполняемый оператор языка ПЛ/М-80 расширяется в шесть машинных команд. Это означает, что программа на ПЛ/М пишется от трех до шести раз быстрее, чем на ассемблере.

Для ряда программ (например, технологических) существенно программирование их на языке проблемной области пользователя. С помощью макроопределений можно заменить ключевые слова ПЛ/М на русские и использовать русские идентификаторы так, чтобы они несли смысловую нагрузку в рамках решаемой задачи. При этом нет потери пресловутой совместимости.

В число других важных вопросов техники программирования входят структура объектного файла, стыковка программ на разных языках и

мониторы микроЭВМ. Эти вопросы будут освещены в подготавливаемых статьях.

Телефон 283-97-64, Москва

## ЛИТЕРАТУРА

1. Мыскин А. В., Торгашов В. А. Инструкция по программированию на PL/M-80. — Л.: ЛНИВЦ АН СССР, 1980.— 63 с.
2. Каган Б. М., Сташин В. В. Микропроцессоры в цифровых системах.— М.: Энергия, 1979.— 193 с.
3. Микропроцессоры: системы программирования и отладки / Под ред. В. А. Мясникова, М. Б. Игнатъева. — М.: Энергоатомиздат, 1985.— 272 с.
4. Гребенщиков Л. К., Летник Л. А. Программирование микропроцессорных систем на языке ПЛ/М.— М.: Финансы и статистика, 1986.— 160 с.
5. McStachen D. D. A Guide to PL/M Programming for Microcomputer Application.— SI, Addison-Wisley, 1978.— 262 p.
6. PL/M-80 Programming Manual.— Intel Corp.— USA, 1977.— 106 p.
7. PL/M-86 Programming Manual.— Intel Corp., 980466A.— USA, 1978.— 110 p.
8. PL/M-86 User's Guide.— Intel Corp., 121636.— USA, 1982.— 250 p.
9. PL/M-51 User's Guide.— Intel Corp., 121966.— USA, 1983.— 150 p.
10. PL/M-286 User's Guide.— Intel Corp., 121945.— USA, 1982.— 239 p.
11. Кертис Б., Солоуэй Э. М., Брукс Р. Е. и др. Психология программных систем // ТИИЭР.— 1986.— № 8.— Т. 74.— С. 42—60.

Статья поступила 09.06.87

УДК 681.3.06

А. Р. Таюпов

## МЕТОДЫ ПРОГРАММИРОВАНИЯ АППАРАТУРЫ КАМАК НА ЯЗЫКЕ ПАСКАЛЬ

Существует множество вариантов программирования аппаратуры КАМАК. Если нужно написать небольшую (300...400 операторов) программу обслуживания нескольких модулей КАМАК, можно воспользоваться языком Паскаль [1]. Среди многих достоинств этого языка — «прозрачность» текста и жесткий контроль над типами данных (многие самые коварные ошибки определяются на этапе трансляции).

Паскаль, реализованный в среде операционной системы РАФОС, имеет дополнительные расширения. При

программировании аппаратуры КАМАК мы получаем возможность

например

```
CSR: origin 166000B; (* регистр состояния и управления*);
```

```
DNR: origin 166002B; (* регистр маски и прерываний *);
```

```
DNR: origin 166006B; (* регистр старшего байта *).
```

адресации к «физической» ячейке памяти. Специально объявленной переменной становится в соответствие содержимое конкретного слова в карте памяти ЭВМ [2].

Пример описания регистров контроллера 106:

Ячейка памяти 166000В (восьмеричный код) соответствует переменной CSR.

Простые пересылки чисел между

программой и регистрами модулей выполняются просто:

```

r
CSR:=0;
x:=real(INPUT);(*переменная INPUT должна быть описана*);
writeln(DNR)

```

Доступ к отдельным битам осуществить труднее.

Паскаль, реализованный в операционной системе РАФОС, позволяет выполнять побитовые логические операции (И, ИЛИ, НЕ) над соответствующими битами целых операндов:

```

(53 or 15) равно 59;
(5 and 3) равно 5;
Q= 1 , если CSR < 0;
Q= 0 , если CSR > 0.

```

Сигнал Q—16 бит регистра состояния и управления.

Предположим, необходимо прочесть содержимое бита D в регистре состояния и управления. Этот седьмой бит индицирует наличие прерываний от модулей КАМАК. Введем в программу «маску» — целое 16-разрядное число с 1 в 7-м бите и 0 во всех остальных битах:

```

const
    MASK11=4000B.

```

Произведя логическое побитное умножение, получим:

```

D= 1 , если (CSR AND MASKD)=128;
D= 0 , если (CSR AND MASKD)=0.

```

Определяя маски с 1 в разных позициях, можно одновременно получать информацию о содержимом тестируемых битов.

Аналогично можно селективно устанавливать биты регистра управления в желаемое состояние. Например, необходимо установить линию 1 в состояние 1. Это 5-й бит регистра состояния и управления. Определив маску с 1 в 5-м бите и с 0 во всех остальных, в нужном месте программы производим логическое суммирование.

```

const
    MASK11=2048;
    .....
    .....
    CSR:=(CSR OR MASK11).

```

Для определения значений масок удобно пользоваться восьмеричным

представлением чисел.

```

const
    MASKD=128.

```

Рассмотрим решение конкретной задачи. Возникла проблема ввода подготовленных на ЭВМ «Мега 60» управляющих файлов ЧПУ. Проще всего эту задачу решить, используя ввод информации из модуля КАМАК в разъем фотосчитывателя в стойке ЧПУ. В нашем модуле (см. рисунок) через регистр с субадресом A0

```

const
    MASKL=256; (*маска для тестирования 9 бита регистра DNR*);
var
    CSR origin 166000B, (*описание регистра состояния и управления*);
    CN5A0 origin 166240B, (*описание регистра ввода в ЧПУ*);
    DNR origin 166002B:integer; (*описание регистра прерывания*);
    MS:char; (*переменная для ввода символа в ЧПУ*);
    F:file of char; (**"внутреннее" имя файла данных*);
begin
    reset(F,'FILE','DAT'); (*открытие "внешнего" файла FILE.DAT*);
    while not eof(F) do
        if (MASKL and DNR)=256 then (*тест 9 бита DNR*);
            begin
                read(F,MS); (*чтение очередного символа*);
                CN5A0:=ord(MS); (*ввод кода символа в ЧПУ*);
                CN5A0:=ord(MS)+256; (*синхроимпульс 9 бит регистра*);
            end;
    end.

```

Пример упрощенной программы ввода данных в ЧПУ

ASCII код символа передается в ЧПУ. Восьмой разряд четности генерируется аппаратно. Сигнал «Готовность ЧПУ» инициирует линию L. Модуль размещается в порте N5. Линия L задействована на 9-й бит регистра прерывания DNR.

Приведенные примеры показывают,

что при программировании аппаратуры КАМАК на языке Паскаль простота и наглядность языка высокого уровня сочетаются с возможностями ассемблера. Для решения задач автоматизации технологических процессов это позволило уменьшить время разработки ПО.

Недостатки метода: применяется только в случае, если при работе с аппаратурой КАМАК не используются прерывания и не ведется работа в мультипрограммном режиме с одновременным доступом к одному и тому же крейту из нескольких программ.

450001, Уфа, ул. Ст. Халтурина, 39, ИИЦМ АН СССР, лаб. «Информатика»

## ЛИТЕРАТУРА

1. Перминов О. Язык программирования Паскаль. — М.: Радио и связь, 1983.

2. Операционная система СМ ЭВМ РАФОС: Справочник / Под ред. В. П. Семика. — М.: Финансы и статистика, 1984.

Статья поступила 24.09.87

УДК 621.397

В. А. Власенко, А. Б. Коханов

**ПОЛУЧЕНИЕ ТВЕРДЫХ КОПИЙ ГРАФИЧЕСКИХ И ПОЛУТОНОВЫХ ИЗОБРАЖЕНИЙ**

В цифровых устройствах обработки изображений [1] на экране видеоконтрольных устройств (ВКУ) и полутонных дисплеях изображения представляются в виде набора элементов (другие названия элемента — пэл, пиксель). Каждый элемент имеет свой уровень яркости, представленный в цифровом коде. На печатающих устройствах (АЦПУ) можно получить только два уровня яркости: белый (или другой цвет), соответствующий цвету бумаги, и черный (или другой цвет) — цвету красителя.

Существуют методы [2], позволяющие на таких строго двухуровневых (бинарных) устройствах получать дополнительные уровни интенсивности. В газетной печати применяют разрешение 20...30 точек/см, в то время как в журнальной и книжной печати используются до 60 точек/см. Устройства графического вывода позволяют аппроксимировать пятна переменной площади, с помощью которых строятся изображения. Идея заключается в том, чтобы заполнить область элемента изображения площадью зачернения (забеления) пропорционально интенсивности элемента. За счет ухудшения пространственного разрешения можно добиться улучшения разрешения по интенсивности (т. е. один полутонный элемент составляют из некоторого числа бинарных элементов).

С помощью процедуры (подпрограммы) управления АЦПУ типа Роботрон СМ 6329.02-М можно получать полутонные твердые копии изображения. В цифровом телевидении [3] изображения раскладываются на элементы (пиксели) и хранятся либо в специализированных ОЗУ [1], либо на внешних долговременных запоминающих устройствах (магнитные диски, ленты и т. п.). Формат стандартного телевизионного изображения (по ГОСТ 7845—79 — отношение 4:3) определяется сторонами прямоугольного экрана. Число элементов в строке и столбце выбирают обычно равным 512, 256, 128 и др. Такое отношение сторон реального изображения при равномерном числе элементов в строке и столбце требует выполнения этих же соотношений и для одного элемента изображения. При синтезированных в ЭВМ изображениях соотношение сторон элемента равно 1:1 [2].

Клип из 128 элементов и 32 строк имеет 16 градаций яркости. Поло-

сы Маха (перепады яркости) просматриваются отчетливо, что свидетельствует о нормальном психофизиологическом восприятии градиционного клина.

Печать выводится на АЦПУ Роботрон СМ 6329.02-М в режиме 960 точек/8 дюймов. Формирование растрового элемента для каждого уровня яркости изображения и управление режимами устройства печати осуществлялись с помощью подпрограммы PICTUR.

Оформлена эта подпрограмма в виде внешней процедуры на языке Паскаль (версия РАФОС, ОС ДВК, ЭЦВМ — ДЕКЗ). Подпрограмма обеспечивает вывод на печать одной строки длиной не более 128 элементов изображения, каждый из которых представляется набором полутонных элементов. Построчный вывод позволяет выводить на печать необходимое пользователю число строк. Управляющей программой можно менять длину строки.

При вызове процедуры необходимо передавать ей из управляющей программы всего два параметра: длину строки (число элементов) N, массив A [1]. Тип массива в управляющей программе определяется следующим образом:

```
TYPE
MAS : ARRAY [1...128] OF INTEGER;
VAR
A : MAS;
```

На рис. 1 приведена блок-схема программы управления АЦПУ Роботрон СМ 6329.02-М в точечном режиме при формировании растра. Параметры передаются (блок 1, 2) управляющей программой. Блок 3 задает интервал между строками (3/216 дюйма). Один элемент изображения формируется тремя точками по вертикали и семью точками по горизонтали. Такой выбор матрицы элемента растра позволяет получить размеры одного элемента (1,2 мм по горизонтали и 0,8 мм по вертикали). При увеличении яркости элемента изображения уменьшается число точек в растре, что приводит к изменению площади зачернения. В изображениях на рис. 1, 2 максимальное значение яркости элемента массива равно 63 (пять двоичных разрядов). При другом динамическом диапазоне необходимо в строке D := A[I] DIV 4; вместо делителя четыре поставить целое число, которое обеспечит из-

менение параметра D от 0 до 15 в заданном динамическом диапазоне.

Блок 5 обеспечивает печать нужного числа точек в строке. Так как каждый элемент представлен семью точками, то общее число точек в строке будет N\*7.

Блок 6 выводит элементы на устройство печати с использованием оператора вариантов CASE. Каждому из 16 значений параметра соответствует вариант матрицы элемента (7\*3 элемента). Эти значения в регистр данных устройства печати передаются на физическом уровне с помощью оператора ORIGIN. При этом необходимо установить соответствие двух переменных и регистров АЦПУ:

```
CONST READY=200 B;
ESC=33 B;
```

```
VAR
LPCSR ORIGIN 177514 B;
LPDAT ORIGIN 177516 B;
INTEGER;
```

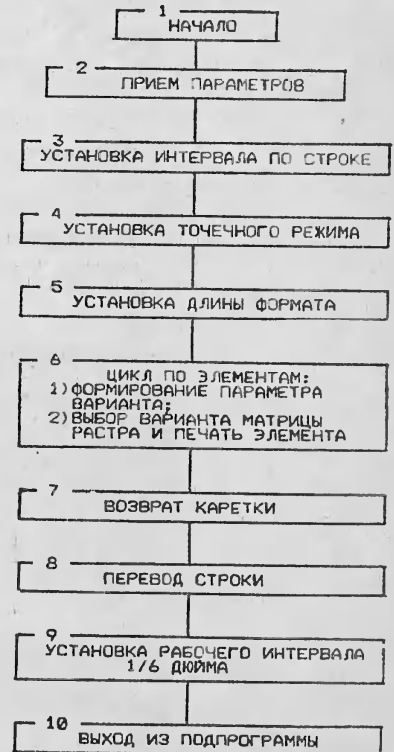


Рис. 1. Блок-схема процедуры PICTUR

```

CONST
  HB=128;
TYPE
  MAS=ARRAY(1..HB) OF INTEGER;
PROCEDURE PICTUR(VAR N,I: INTEGER; VAR A: MAS);
CONST
  READY=200B;
  ESC=33B;
VAR
  LPCSR ORIGIN 177514B;
  LPDAT ORIGIN 177516B: INTEGER;
  N1,N2,D,B,J: INTEGER;
PROCEDURE PRILINE(A1,A2,A3,A4,A5,A6,A7: INTEGER);
BEGIN
  WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
  LPDAT:=A1;
  WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
  LPDAT:=A2;
  WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
  LPDAT:=A3;
  WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
  LPDAT:=A4;
  WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
  LPDAT:=A5;
  WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
  LPDAT:=A6;
  WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
  LPDAT:=A7;
END;
BEGIN
  (*1*)
  IF N>128 THEN
  BEGIN
    WRITELN('Я ТАКОЕ НЕ ПЕЧАТАЮ.СМЕНИТЕ ФОРМАТ.ЗНАЧЕНИЕ N>128. ');
    (*УСТАНОВКА ИНТЕРВАЛА СТРОКИ 1/6 ДЮЙМА*)
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=27;
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=ORD('0');
  END
  ELSE
  BEGIN
    (*УСТАНОВКА ИНТЕРВАЛА ПО СТРОКЕ 3/72*);
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=ESC;
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=ORD('A');
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=3;
    (*УСТАНОВКА ТОЧЕЧНОГО РЕЖИМА 960 ТОЧЕК/8 ДЮЙМОВ*)
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=ESC;
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=ORD('I');
    (*УСТАНОВКА ДЛИНЫ ФОРМАТА*)
    N1:=(N*7) MOD 256;
    N2:=(N*7) DIV 256;
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=N1;
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=N2;
    (*ВЫВОД СТРОКИ НА ПЕЧАТЬ*)
    FOR I:=1 TO N DO
    BEGIN
      D:=ACIJ DIV 4;
      CASE D OF
        15: PRILINE(0,0,0,0,0,0,0);
        14: PRILINE(0,0,0,64,0,0,0);
        13: PRILINE(0,128,0,0,32,0,0);
        12: PRILINE(0,64,0,32,0,0,128);
        11: PRILINE(128,0,32,0,128,0,32);
        10: PRILINE(128,32,64,0,128,64,0);
        9: PRILINE(128,32,64,128,64,32,64);
        8: PRILINE(128,32,192,64,160,128,32);
        7: PRILINE(64,160,160,192,160,192,192);
        6: PRILINE(64,224,160,192,160,192,192);
        5: PRILINE(224,160,224,92,160,192,224);
        4: PRILINE(224,160,224,92,160,192,224);
        3: PRILINE(224,224,224,160,192,224,224);
        2: PRILINE(224,224,224,160,192,224,224);
        1: PRILINE(224,224,224,160,224,224,224);
        0: PRILINE(224,224,224,224,224,224,224);
      END
    END;
    (*2*)
    (*ВОЗВРАТ КАРЕТКИ*)
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=13;
    (*ПЕРЕВОД СТРОКИ*)
    WHILE (LPCSR AND READY)=0 DO (*ЖДАТЬ*);
    LPDAT:=10;
  END
  END;
  (*1*)
END;
BEGIN
END.

```

Константа ESC=33B (27 десятичное) указывает на прием управляющих символов. Командные символы приведены в описании АЦПУ Роботрон СМ 6329.02-М.

Все данные передаются в АЦПУ: WHILE (LPCSR AND READY) = 0 DO (\*ЖДАТЬ\*) LPDAT :=

т. е. перед каждой посылкой данных необходимо проверить с помощью оператора WHILE... (\*ЖДАТЬ\*); установлен ли разряд готовности.

После завершения вывода строки устройства печати процедура выполняет возврат каретки, перевод строки и установку рабочего интервала между строками 1/6 дюйма.

Изображения, приведенные на рис. 1, 2, выводились подпрограммой считывания файла с магнитного диска и построено преобразовывались процедуры PICTUR на АЦПУ (рис. 4). Возможно у этой подпрограммы найдутся и другие применения. Желающим воспользоваться процедурой PICTUR необходимо помнить, что транслировать ее надо с ключом E, т. е. PICTUR, PICTUR=PICTUR/E.

Телефон 288-764, Одесса

## ЛИТЕРАТУРА

1. Коханов А. Б. Специализированные видеотерминалы для моделирования систем обработки изображений // Приборы, средства автоматизации и системы управления.— 1986.— № 6, 7, 8.— С. 39.
2. Дж. Фоли, А. ван Дем. Основы интерактивной машинной графики.— М.: МИР, 1985.— 2 кн.
3. Узилевский В. А. Передача, обработка и воспроизведение цветных изображений.— М.: Радио и связь, 1981.
4. Роботрон СМ 6329.02-М. Инструкция по эксплуатации 69—260—0729—0. Системное описание.

Статья поступила 15.03.88



# ПРОГРАММА ПОЛУЧЕНИЯ ТВЕРДОЙ КОПИИ ГРАФИЧЕСКОГО ДИСПЛЕЯ ДВКЗМ

Простейший способ получения твердой копии на печатающем устройстве (ПУ) — перенести все видимые точки с экрана на бумагу без каких-либо геометрических преобразований. Для этого необходимо перекодировать образ графического экрана, хранимый в ОЗУ КГД [3], к виду, пригодному для вывода на ПУ. Байты из ОЗУ КГД отображаются на экране горизонтальными цепочками точек (по 50 байтов в строке), подвергаясь при этом зеркальному отражению (младший бит оказывается слева). На ПУ байты выводятся вертикальными столбцами точек, так что младший бит оказывается внизу. В обоих случаях точке соответствует установленный бит. Выделяя «кирпичик» изображения из первых байтов восьми первых строк в ОЗУ КГД, копируем его в блок из восьми байтов в оперативной памяти. Затем с помощью циклических сдвигов через бит «С» переносим столбец младших битов в байт, выводимый на печать (см. рисунок). Процесс повторяем до тех пор, пока не будут распечатаны все столбцы блока. После этого копируем в блок «кирпичик» из вторых байтов восьми первых строк. Когда первые восемь строк будут таким образом распечатаны, переходим к следующим восьми и т. д. Так как число строк изображения на экране не кратно восьми, при выводе последних шести строк два байта блока не используются. В программе предусмотрено их обнуление.

Для сохранения изображения на магнитном диске (подпрограмма WPICT) достаточно побайтно перенести информацию из ОЗУ КГД сначала в буфер ввода-вывода, а затем на диск. Для восстановления изображения на экране следует проделать это в обратном порядке.

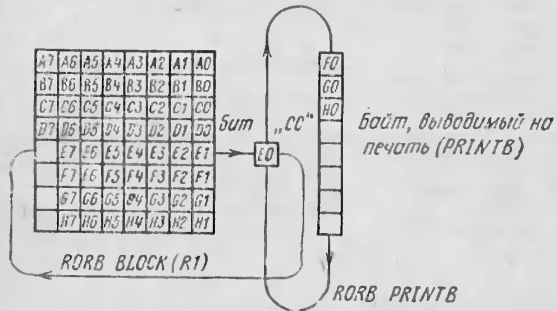


Схема перекодировки изображения:

буквы А — Н нумеруют байты в «кирпичике» изображения; цифры — биты в каждом байте. Кирпичик копируется в блок BLOCK. Циклическими сдвигами элементов BLOCK и байта PRINTB столбцы блока переносятся в PRINTB, который выводится на печать. На рисунке изображен момент, когда произведен сдвиг байта Е блока BLOCK. Следующим шагом будет сдвиг байта PRINTB. в результате чего элемент Е0 из бита «С» будет помещен в PRINTB

Все программы написаны на Макроассемблере и оформлены в виде подпрограмм. Их вызывает основная программа PIC, которая распознает соответствующую систему команд, вводимых с терминала (см. приложение 1). Спецификация файла при записи (восстановлении) изображения задается в формате ОС. Для реализации этой функции составлена программа FILCOD, переводящая символьную строку со спецификацией файла в код RAD50.

Настоящие программы могут оказаться полезными и сами по себе. Поэтому они оформлены в виде, пригодном для вызова из программ на Фортране, и занесены в библиотеку объектных модулей. Текст программы получения копии изображения на бумаге HDCOPY приведен в приложении 2. В приложении 3 описаны параметры и правила вызова упоминавшихся подпрограмм из программы, написанной на Фортране.

## Приложение 1

### Система команд программы PIC

- H — подсказка.
- P — вывод твердой копии на бумагу.
- R [FILE] — чтение файла FILE с диска на бумагу.
- W [FILE] — запись изображения с экрана в файл FILE.
- G — перевод дисплея в графический режим.
- S — перевод дисплея в символьный режим.
- N — получение негатива от изображения на дисплее.
- K — конец работы.

## Приложение 2

### Исходные тексты основной программы и программы получения копии на бумаге

```

.ENABL LC
;АССЕМБЛЕР MACRO VS.O2B
;*****
;*
;* ОСНОВНАЯ ПРОГРАММА PIC. РАСПОЗНАЕТ КОМАНДЫ С ТЕРМИНАЛА, ОСУЩЕСТВЛЯЕТ
;* ВЫЗОВ СООТВЕТСТВУЮЩИХ ПОДПРОГРАММ
;*
; ИСПОЛЬЗУЕМЫЕ РЕГИСТРЫ: R0,R1,R2,R3,R4,R5
;*****
.TITLE PIC
.GLOBAL TYPE,HDCOPY,RPIC,WPIC,FILCOD,NEG
.MCALL .EXIT,.TTYIN,.DBLK
KBDU=176640
PIC: MOV #MESS1,R1
JGR PC,TYPE
START: MOV #INVIT,R1
JGR PC,TYPE
MOV #CONSTR,R1
CLR R2
ACCEPT: .TTYIN ; ВВОД КОМАНДНОЙ СТРОКИ С ТЕРМИНАЛА
CMP R0,#12
BEQ ANA1
MOV# R0,(R1)+
INC R2
BR ACCEPT
ANA1: CMP R2,#17. ; ПРОВЕРКА ДЛИНЫ КОМАНДНОЙ СТРОКИ
BMT ANALRW
MOV #MESS2,R1
JGR PC,TYPE
JMP OUT
ANALRW: CMP# CONSTR,4*R ; РАСПОЗНАВАНИЕ КОМАНД R И W
BEQ CODEF
CAF# CONSTR,#W
BEQ CODEF
ANALH
CODEF: CMP# CONSTR+1,#40 ; ЕСЛИ ПОСЛЕ R ИЛИ W СЛЕДУЕТ ПРОБЕЛ, ТО
; ; ПЕРЕКОДИРОВКА ПОСЛЕДУЮЩИХ СИМВОЛОВ
MOV #CONSTR+2,BLFLC+2 ; В НИЖЕ ФАЙЛА В КОДЕ RAD50
MOV #FILNAM,BLFLC+4
MOV #RTCOD,BLFLC+6
MOV #BLFILC,R3
JGR PC,FILCOD
ADD #6,R5
TST E(R5)
BEQ 1H
MOV #MESS5,R1
JGR PC,TYPE
JMP OUT
1E: MOV R0,FILNAM
MOV R1,FILNAM+2
MOV R2,FILNAM+4
MOV R3,FILNAM+6
3R: MOV #FILNAM,BLKPRW+2
MOV #RTCOD,BLKPRW+4
MOV #BLKPRW,R5
CMP# CONSTR,#R
BNE 2H
JGR PC,RPIC ; ЧТЕНИЕ
BR ANERR
2K: JGR PC,WPIC ; ЗАПИСЬ
    
```

```

          .ENABL LC
          ; АССЕМБЛЕР MACRO V3.02B
          ; *****
          ;* ПРОГРАММА HDCOPY ПРЕДНАЗНАЧЕНА ДЛЯ ПОЛУЧЕНИЯ ТВЕРДОЙ КОПИИ
          ;* ИЗОБРАЖЕНИЯ ГРАФИЧЕСКОГО ДИСПЛЕЯ ДВКЗМ
          ;* НА ПЕЧАТАЮЩЕМ УСТРОЙСТВЕ РОБОТРОМ СМ6329.02-А
          ;*
          ; *****
          ; ИСПОЛЬЗУЕМЫЕ РЕГИСТРЫ: R0,R1,R2,R3,R4
          .TITLE HDCOPY
          .GLOBAL HDCOPY
          .MCALL .EXIT
          KSDA=176644
          KSDB=176642
          LPS=177514
          LPB=177516
          HDCOPY: CLR ADR0
          MOV @INITLP,R1
          JSR PC,BENDLP ; ИНИЦИАЛИЗАЦИЯ ПУ
          MOV @INTERV,R2 ; УСТАНОВКА ИНТЕРВАЛА МЕЖДУ СТРОКАМИ В/72"
          JSR PC,BENDLP ; R3 СОДЕРЖИТ СЧЕТЧИК ЧИСЛА БАЙТОВ НА СТРАНИЦЕ
          MOV @R3,R0 ; *** НАЧАЛО ЦИКЛА ВЫВОДА ГРАФИЧЕСКОЙ СТРАНИЦЫ
          PAGE: MOV @GRFCOD,R1
          JSR PC,BENDLP ; ПЕРЕВОД ПУ В ГРАФИЧЕСКИЙ РЕЖИМ
          MOV @R1,R4 ; R4 СОДЕРЖИТ СЧЕТЧИК "КИРПИЧКОВ" В ГРАФИЧЕСКОЙ
          ; СТРОКЕ
          ; *** НАЧАЛО ЦИКЛА ВЫВОДА ГРАФИЧЕСКОЙ СТРОКИ
          STRNG: MOV ADR0,R3 ; R3 СОДЕРЖИТ АДРЕС ПЕРВОГО БАЙТА "КИРПИЧКА"
          ; В КГД
          MOV @BLOCK,R2 ; В R2-АДРЕС БЛОКА, КУДА СКОПИРУЕТСЯ "КИРПИЧКИ"
          MOV @R1,R1 ; R1 СОДЕРЖИТ СЧЕТЧИК ЧИСЛА БАЙТОВ В БЛОКЕ
          TST R0 ; ВЫВОДИТСЯ ПОСЛЕДНЯЯ СТРОКА ПУ ?
          BNE COPY ; НЕТ
          CLR BLOCK+6 ; ДА, ОЧИСТКА ДВУХ ПОСЛЕДНИХ БАЙТОВ
          MOV @R1,R1 ; В R1-ЧИСЛО ЗНАЧАЩИХ БАЙТОВ В БЛОКЕ
          COPY: MOV R3,KSDA ; *** ЦИКЛ КОПИРОВАНИЯ "КИРПИЧКА" В БЛОК
          ; АДРЕС ОЧЕРЕДНОГО БАЙТА "КИРПИЧКА" В РЕГИСТРЕ
          ; АДРЕСА КГД
          MOV KGBD,(R2)+
          ADD @R2,R3
          DEC R1
          BPL COPY
          ; *** "КИРПИЧКИ" СКОПИРОВАНЫ В БЛОК
          ; В R2-СЧЕТЧИК СТОЛБЦОВ В БЛОКЕ
          MOV @R1,R2 ; *** ЦИКЛ ПЕРЕКОДИРОВКИ И ПЕЧАТИ БЛОКА
          ; В R1-СЧЕТЧИК БАЙТОВ В БЛОКЕ
          ; *** ЦИКЛ СДВИГА ОДНОГО СТОЛБЦА
          PRINT: MOV @R1,R1
          CNF: RORB BLOCK(R1)
          RORB PRINTB
          DEC R1
          BPL CNF
          ; *** СТОЛБЕЦ СДВИНУТ В PRINT
          ; *** БЛОК ПЕРЕКОДИРОВАН И РАСПЕЧАТАН
          ; ПЕРЕХОД К СЛЕДУЮЩЕМУ "КИРПИЧКУ"
          L4: TSTB LPS
          BPL L4
          MOVB PRINTB,LPB
          DEC R2
          BPL PRINT
          INC ADR0 ; *** ГРАФИЧЕСКАЯ СТРОКА ПУ ВЫВЕДЕНА.
          DEC R4 ; ПЕРЕХОД К СЛЕДУЮЩЕЙ СТРОКЕ
          BPL STRNG
          ADD @R3B,ADR0
          DEC R0
          BPL PAGE
          ; *** ГРАФИЧЕСКАЯ СТРАНИЦА ВЫВЕДЕНА
          MOV @INITLP,R1
          JSR PC,BENDLP ; ИНИЦИАЛИЗАЦИЯ ПУ
          RTS PC
          GRFCOD: .ASCIZ <12><15><33>^/ <5><220><1> ; ПС, ВК И ПЕРЕВОД ПУ В ГРАФИЧ РЕЖИМ
          INTERV: .ASCIZ <33>/^/ <10> ; УСТАНОВКА ИНТЕРВАЛА В/72"
          PRINTB: .BLKB 1 ; БАЙТ, ВЫВОДИМЫЙ НА ПЕЧАТЬ
          INITLP: .ASCIZ <33>^/ ; ИНИЦИАЛИЗАЦИЯ ПУ
          ADR0: .WORD 0 ; АДРЕС ПЕРВОГО БАЙТА "КИРПИЧКА" ИЗОБРАЖЕНИЯ
          BLOCK: .BLKB 10 ; БЛОК С КОПИЕЙ "КИРПИЧКА" ИЗОБРАЖЕНИЯ
          .EVEN
    
```

### Приложение 3

#### Порядок вызова и параметры подпрограмм

##### HDCOPY

Назначение: Получение твердой копии изображения с экрана на бумаге.

Вызов из Фортрана: CALL HDCOPY,

Параметры: нет.

##### RPICST WPICST

Назначение: Запись, чтение изображения с диска на экран. Вызов из Фортрана.

CALL WPICST (FILNAM, IRTCOD),  
CALL RPICST (FILNAM, IRTCOD).

Параметры:

FILNAM — REAL \* 8 — спецификация файла в коде RAD50.

IRTCOD — INTEGER \* 2 — код возврата.

IRTCOD=0 — нормальное завершение,

IRTCOD=1 — ошибка при открытии файла,

IRTCOD=2 — ошибка ввода (вывода).

##### FILCOD

Назначение: кодировка спецификации файла из символической строки в формате операционной системы в код RAD 50 в соответствии с макрокомандой .DBLK.

Вызов из Фортрана: CALL, FILCOD (STR, FILNAM, IRTCOD).

Параметры:

STR — массив длиной 15 байт. Содержит спецификацию файла в символическом виде, завершающуюся кодом «BK» (015).

FILNAM — REAL \* 8. До вызова содержит спецификацию файла по умолчанию в коде RAD 50. После выхода содержит

новую спецификацию, полученную из старой заменой частей, указанных явно в символической строке STR.

IRTCOD — INTEGER \* 2 — код возврата.

IRTCOD=0 — нормальное завершение,

IRTCOD=1 — недопустимая спецификация файла в STR.

##### NEG

Назначение: Обращение изображения на экране.

Вызов из Фортрана: CALL NEG.

Параметры: нет.

# ПЕРЕСТРАИВАЕМАЯ МИКРОПРОЦЕССОРНАЯ СИСТЕМА УПРАВЛЕНИЯ САМОХОДНЫМ РЕНТГЕНОВСКИМ ДЕФЕКТОСКОПОМ «СИРЕНА-1»

(К ст. Данильченко Н. Т., Лозового Л. Н., которая будет опубликована в очередном номере журнала)



Гибкая автоматизированная система контроля качества кольцевых сварных швов трубопроводов построена на базе самоходного импульсного рентгеновского аппарата «СИРЕНА-1» и МПК БИС КР580. Использование автоматизированных средств доставки рентгеновского излучателя в зону контролируемого шва по внутренней поверхности трубопровода полностью исключает влияние внешних условий. При облучении кольцевого сварного шва изнутри трубопровода с его продольной оси можно получить пол-

ный панорамный снимок этого шва за одну экспозицию.

В автоматическом режиме выполняются следующие команды: перемещение дефектоскопа по внутренней поверхности нитки трубопровода на номинальной или пониженной скорости, остановка дефектоскопа в плоскости контролируемого шва с высокой точностью, отсчет заданного времени экспозиции рентгеновского излучателя, блокировка системы управления и автоматический выезд дефектоскопа из трубопровода при снижении запаса энергии

в источнике питания до заранее заданного уровня, остановка при отработке заданной установки времени таймером дальности хода.

Комплекс «СИРЕНА-1» демонстрировался на ряде международных выставок, а также на ВДНХ СССР, где получил 11 медалей. В настоящее время он производится серийно, одна из его модификаций экспортируется в Англию.

197198, Ленинград, ул. Блохина, д. 4, кв. 52, Лозовой Л. Н.

# ИНТЕЛЛЕКТУАЛЬНЫЙ КОНТРОЛЛЕР ЛОКАЛЬНОЙ СЕТИ

(К ст. Хмелевского А. М., Камкова А. А., Ковалева А. Г., которая будет опубликована в одном из очередных номеров журнала)

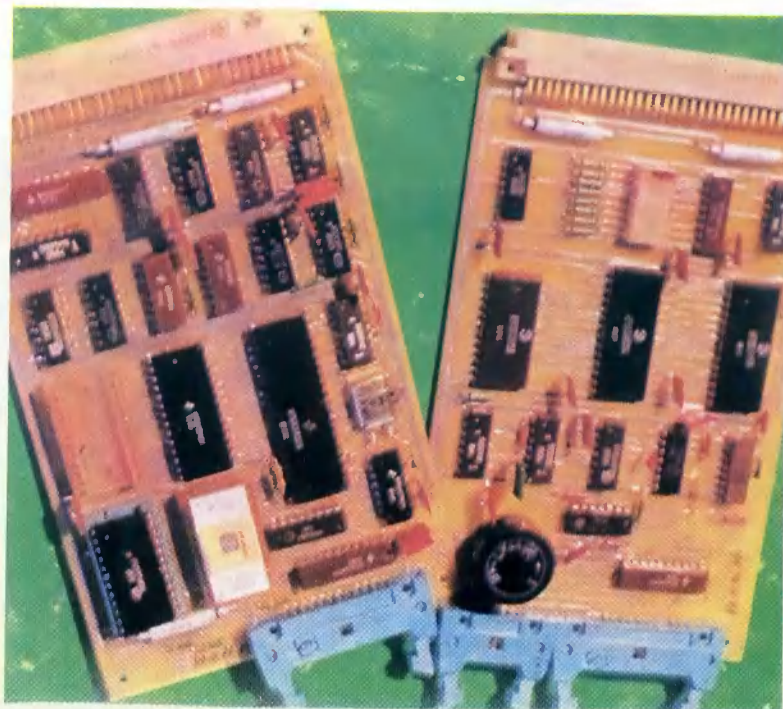
Для специалистов, разрабатывающих гибкие производственные системы или автоматизированные измерительные комплексы на базе микропроцессорной техники, созданы аппаратно-программные средства, реализующие уровень канала передачи данных среднескоростной локальной производственной сети.

Доступ к линии связи локальной сети со стороны абонентов осуществляется через интеллектуальный контроллер локальной сети (ИКЛС). Функционально и конструктивно ИКЛС состоит из двух частей: интеллектуального периферийного контроллера (ИП) и контроллера сети (КС). Все узлы ИП объединены внутренней шиной. Взаимодействие КС с ИП осуществляется через дополнительную шину, ориентированную на подключение БИС серии КР580. Программируемый приемопередатчик КС выполнен на БИС КР580ВВ51А, а временные формирователи — на БИС КР580ВИ53.

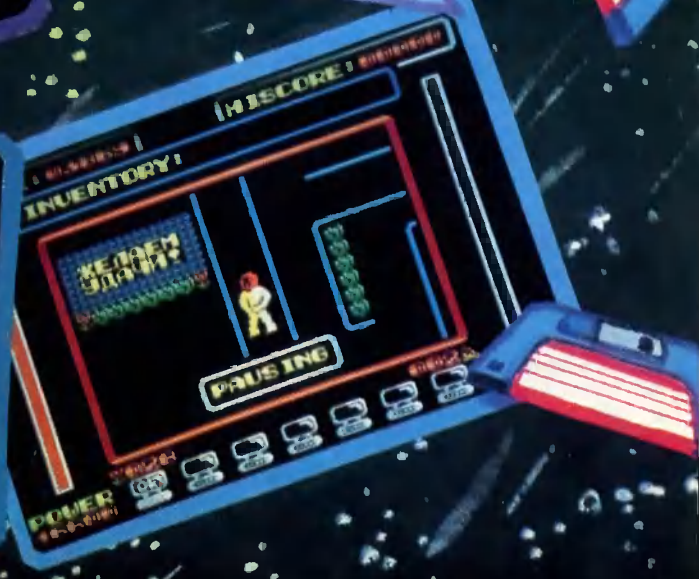
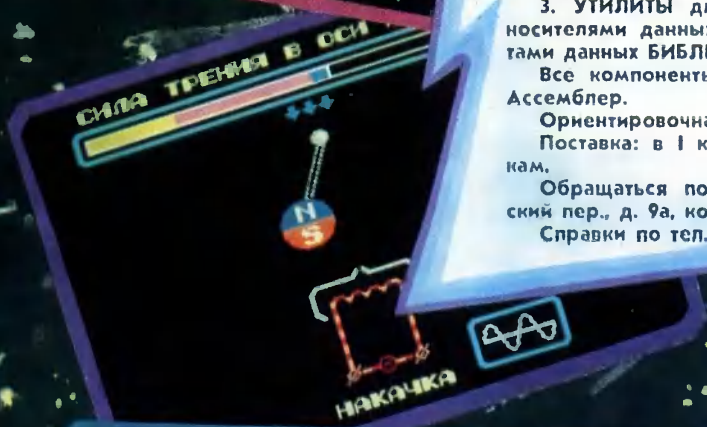
ИКЛС разработан для систем управления технологическим оборудованием, выполненных на базе МПК БИС серии КР580. На базе ИКЛС создана сеть для связи между системой управления участком и гибкими производственными модулями механообработки.

## Основные характеристики локальной сети

Скорость обмена, Кбод	50
Топология сети	Моноканал
Среда передачи	Экранированная витая пара
Число абонентов, подключаемых к сегменту сети	До 16



109052, Москва, «РАТЕКС»



### «МАЭСТРО»

— программное средство для специалистов, работающих в области рекламы, разработки графического интерфейса, системных и прикладных (в том числе учебных и игровых) программ на компьютерах MSX и MSX2.

Если Вы приобретете «Маэстро», Вам станут доступны такие современные средства программирования как параллельные процессы, окна, работа с манипулятором «мышь», специализированные методы доступа и обработки графических данных, а также многое другое.

Программное средство «Маэстро» включает:

1. **БИБЛИОТЕКУ** для языка С (более 200 функций), которая обеспечивает доступ ко всей аппаратуре и BIOS/SUBROM MSX2 и MSX, а также предоставляет ряд новых возможностей для программирования сложных и ресурсоемких задач.

2. **РЕДАКТОР** знакомоестной графики, использующий манипулятор «мышь» и оконные меню для разработки полиэкранов, движущихся объектов, их сохранения и модификации, а также управления ими из программ, использующих БИБЛИОТЕКУ.

3. **УТИЛИТЫ** для работы с различной информацией и носителями данных. Форматы данных совместимы с форматами данных БИБЛИОТЕКИ и РЕДАКТОРА.

Все компоненты ПС «Маэстро» написаны на языках С и Ассемблер.

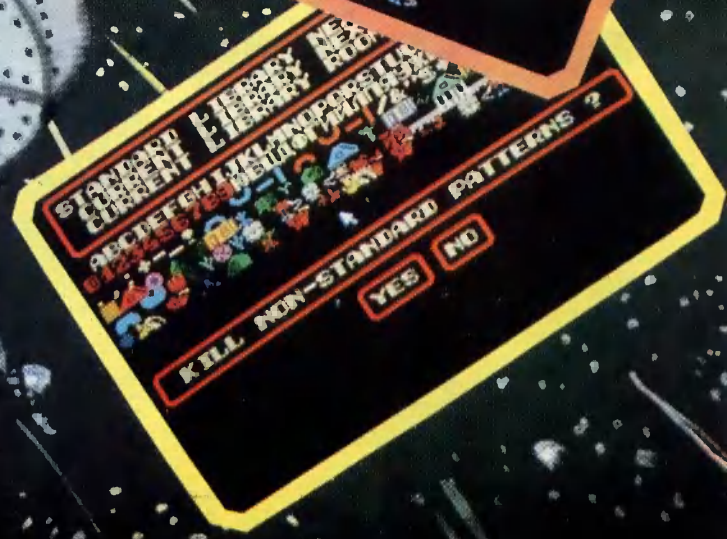
Ориентировочная стоимость ПС «Маэстро» — 9740 руб.

Поставка: в I квартале 1989 г. по предварительным заявкам.

Обращаться по адресу: 103051, г. Москва, М. Сухаревский пер., д. 9а, кооператив «Элекс».

Справки по тел. 237-57-87.

# МАСТРО





На выставке можно было ознакомиться с лабораторным образцом нового поколения роботов с многозвенной рукой — роботом ЦИМ-SKR 30. Робот имеет шесть осей поворота, оснащенных сервоприводами постоянного тока, пневматическую систему смены инструмента WWS 30, гибкий захват для инструмента с интегрированной системой управления (IG 20 NC), сило-моментный датчик.

Робот SKR 30 показан как основное звено сборочного модуля. Сам модуль представляет собой участок автоматической сборочной линии мини-компрессоров.



## Микроэлектроника обеспечивает выигрыш во времени

Выставка стала самым широким и важным смотром достижений народного хозяйства, который ГДР когда-либо проводила за рубежом. Восемь тематических экспозиций демонстрировали ключевые технологии и новейшие результаты совместных исследований и научно-технической кооперации.

Одна из основных и представительных экспозиций выставки — микроэлектроника, включающая микроэлектронные элементы и модули, базовые технологии для высокоинтегральных схем, современные решения технической подготовки и реализации производства (САПР, АСУТП), гибкие автоматизированные производственные системы в металлообрабатывающей промышленности и т. д.

## Гибкие автоматические системы и робототехника

Продемонстрированы высокопроизводительные машинные системы, производственные участки и отдельные машины гибкой автоматизации и робототехники.



1. Программное обеспечение ДВК. Фортран. Графический пакет // Руководство программиста.
2. Печатающее устройство Robotron SM6329.02-м. Инструкция по эксплуатации. Системное описание.— 1986.
3. Плата КГД. Техническое описание. ПБА 4 135.998.ТО.— 1985.
4. Программное обеспечение СМ ЭВМ. Операционная система с разделением функций РАФОС. Фортран // Руководство системного программиста.— 1980.— Т. 5. кн. 2, ч. 3.  
Телефон 57-34-25, Гомель

Статья поступила 01.02.88

УДК 681.3

А. Л. Адушкин, М. В. Богданов, А. И. Земцов,  
Ю. М. Хамидулин

## ИНТЕГРИРОВАННАЯ ГРАФИЧЕСКАЯ СИСТЕМА ДЕКЛАРАТИВНОГО ПРОГРАММИРОВАНИЯ

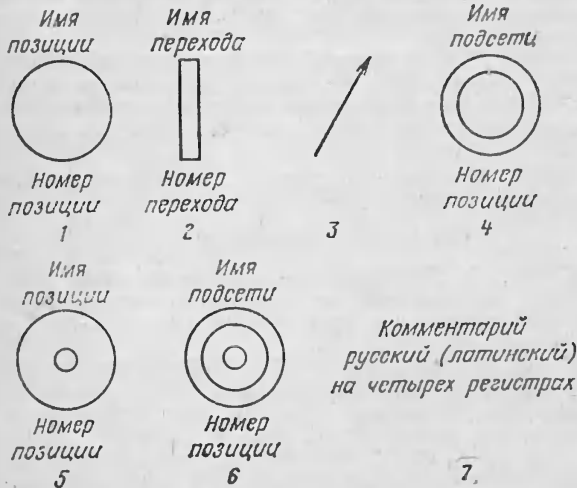
Интегрированная графическая система декларативного программирования (ИГСДП) позволяет:

описывать параллельные асинхронные процессы, используя формальный аппарат сетей Петри, в терминах ациклической сети-процесса с возможностью стратификации подсетей [1]. Каждая подсеть может быть представлена в другой подсети в качестве специально отмеченной позиции. Аппарат стратификации аналогичен вложенности подпрограмм в традиционных языках программирования. Такое представление дает возможность наиболее наглядно отразить все причинно-следственные связи, возникающие в параллельных асинхронных процессах. В настоящее время разработан достаточно мощный математический аппарат анализа процессов [2];

проводить предварительную верификацию описанного процесса с целью обнаружения тупиков, циклов и решения задачи достижимости;

управлять параллельными процессами в реальном времени;

моделировать реальные процессы;



Набор графических примитивов:

1 — позиция в сети; 2 — переход в сети; 3 — ориентированная дуга; 4 — стратифицированная позиция в сети; 5 — маркированная позиция; 6 — маркированная стратифицированная позиция; 7 — текст комментария

осуществлять трассировку выполнения процессов в реальном времени или пошаговом отладочном режиме.

Система ориентирована на машины, имеющие в своем составе растровый дисплей, матричное печатающее устройство или графопостроитель. Исходной информацией для ИГСДП служит графическое описание процессов в виде сетей Петри и графических примитивов, представленных на рисунке.

На выходе ИГСДП формируется загрузочный модуль, включающий описание причинно-следственных связей в исполняемом процессе в форме матриц инцидентности; процедуру исполнения процесса в соответствии с описанной логикой; библиотеку элементарных процедур, осуществляющих действия, описанные в терминах сетей Петри.

В состав ИГСДП входят следующие компоненты:

**Многоуровневый графический редактор**, позволяющий описывать процессы в виде сетей Петри с различной степенью детализации и при необходимости оперативно их корректировать, вставлять и удалять примитивы, находить примитив по имени (номеру), удалять часть сети, переносить фрагмент сети.

**Внутреннее графическое описание сети Петри** представляет собой набор связанных линейных списков, что дает возможность при редактировании свести время реакции системы к минимуму. **Траеклятор**, формирующий машинно-ориентированное описание процесса в виде шести таблиц, содержащих перекрестные ссылки:

**ТАВГ** — с адресами процедур, связанных с позициями в сети;

**F** — ссылки на ТАВГ;

**T** — с адресами процедур, связанных с переходами в сети;

**AT** — ссылки на T;

**FA, AF** — ссылки на F.

Эти таблицы используются процедурой поддержки исполнения процесса.

**Система верификации**, позволяющая тестировать описанный процесс с использованием матричного представления сетей Петри и проверять ограниченность, достижимость и активность сети.

**Система обслуживания библиотек элементарных процедур**, поддерживающая библиотеки в актуальном состоянии и позволяющая организовать просмотр и вывод на печать каталога библиотеки, вставку-удаление функций из библиотеки.

**Система поддержки выполнения процесса** на вычислительном комплексе, обеспечивающая функционирование процесса в реальном времени и отладочном интерактивном режиме. В состав системы может быть включена процедура трассировки выполнения процесса.

**Системный монитор**, поддерживающий функционирование всех компонентов интегрированной системы, связь с операционной системой, сохранение описания процессов и загрузочных модулей на внешних носителях и создание твердой копии на матричном принтере или графопостроителе.

Система ИГСДП использует команды СМ4 для ДВК3 и ОС РАФОС. Большинство компонентов ИГСДП реализовано на языке PL-11, а некоторые компоненты — на языке Паскаль. Суммарный объем программного обеспечения составляет 80 Кбайт двоичных кодов. Система может функционировать в вычислительной системе минимальной конфигурации, включающей графический дисплей и ИГМД. Необходимый объем оперативной памяти не менее 56 Кбайт.

В настоящее время ведутся работы по адаптации ИГСДП для ПЭВМ, программно совместимых с IBM PC.

Телефон 279-14-82, Москва

### ЛИТЕРАТУРА

1. Котов В. Е. Сети Петри.— М.: Наука, 1984.
2. Питерсон Дж. Теория сетей Петри и моделирование систем.— М.: Мир, 1984.

Статья поступила 28.04.88

## СИСТЕМА ВИЗУАЛЬНОГО ВЫВОДА ПОЛУТОНОВЫХ ИЗОБРАЖЕНИЙ НА БАЗЕ ПЭВМ «ИСКРА 226»

Стандартными базисными элементами (примитивами вывода) в блоке отображения символично-графической информации (БОСГИ) ПЭВМ «Искра 226» служат точки и отрезки прямых линий [1]. Для представления сложного изображения необходимо использовать большое число указанных элементов, что затруднительно выполнить из-за ограниченной внешней и оперативной памяти ПЭВМ и отсутствия в БОСГИ модулятора яркости.

Предлагаются программные средства для отображения на экране стандартного БОСГИ большого объема графической информации. При этом сложное полутоновое изображение кодируется в ПЭВМ и формируется на экране как совокупность псевдотоновых кругов и выпуклых многоугольников, имеющих различное местоположение на экране, размеры и квантованную яркость свечения [2, 3].

Уровни яркости свечения элементов изображения моделируются путем задания на микроэрастрах  $2 \times 2$  или  $3 \times 3$  различного числа точек. При выборе микроэрастра предпочтение отдавалось более регулярным структурам, которые проще реализуются и быстрее воспроизводятся на экране дисплея. Формирование изображения с использованием микроэрастра  $2 \times 2$  обеспечивает лучшую

Применение для кодирования и воспроизведения изображений только многоугольников общего вида не рационально с точки зрения затрат памяти ПЭВМ и времени формирования. Поэтому наряду с ними в качестве базисных элементов используются прямоугольники, параллелограммы и трапеции. Для заполнения внутренней области многоугольников предложена модификация быстрого алгоритма Брезенхэма, ранее применяемого только для построения отрезков прямых [4]. Этот алгоритм позволил повысить точность воспроизведения элементов и уменьшить затраты машинного времени при их формировании.

Структура разработанной системы визуального вывода изображений (ТОН) представлена на рисунке. Система может функционировать в следующих режимах.

1. Формирование полутонового изображения из базисных элементов на микроэрастрах  $3 \times 3$  или  $2 \times 2$  с одновременным отображением на устройстве графического вывода (дисплее или графопостроителе).

2. Запись синтезированного изображения в архив на магнитном диске для длительного хранения.

3. Преобразование изображения с отображением на экране и соответствующей коррекцией данных на магнитном диске.

4. Вывод изображения из архива с отображением на устройстве графического вывода.

Работой системы ТОН управляет головная программа, осуществляющая вызов любой из указанных процедур. В свою очередь, каждая процедура завершается выходом на головную программу. При синтезе нового изображения программа ввода обращается к программе формирования изображения, которая воспроизводит элементы изображения (прямоугольники, параллелограммы, произвольные многоугольники, круги). В процессе создания изображения можно проводить преобразования отдельных элементов. В системе предусмотрены следующие преобразования: стирание, перемещение параллельно осям X и Y, растяжение-сжатие, поворот на угол, кратный  $90^\circ$ . После преобразований (либо минуя их) изображение может быть записано на магнитный диск с помощью программы ведения архива изображений. Причем записываются не все точки изображения, а компактное описание составляющих элементов. Последнее обстоятельство является весьма существенным в условиях ограниченной емкости магнитных носителей, используемых в ПЭВМ. Записанное на диске изображение можно ввести в ЭВМ для коррекции с отображением на устройстве графического вывода всех проводимых изменений и последующей перезаписью на диск.

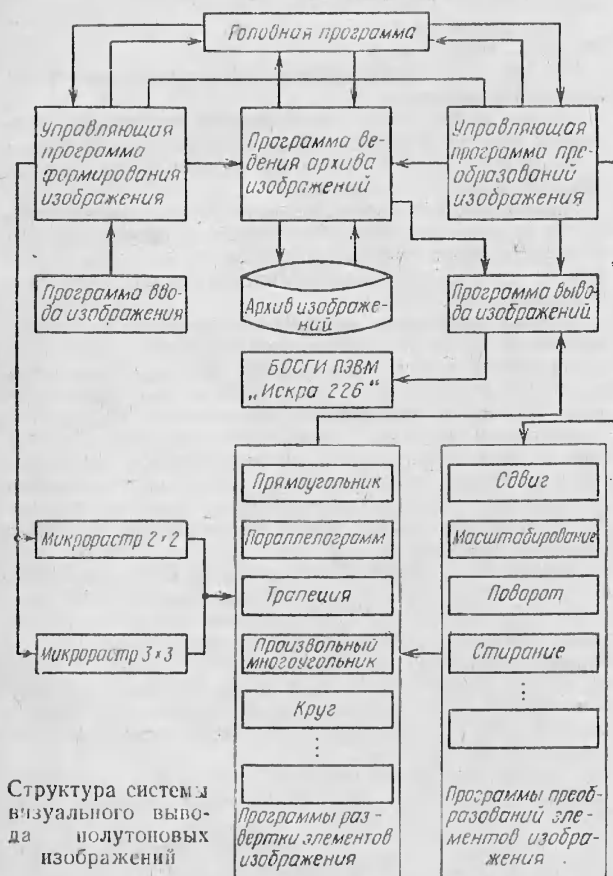
Система ТОН занимает 42 Кбайт оперативной памяти ПЭВМ «Искра 226». Программы написаны на языке БЕЙСИК-2. Для нормального функционирования системы достаточно иметь минимальную конфигурацию машины.

460352, Оренбург, пр. Победы, Политехнический институт, тел. 7-98-56.

### ЛИТЕРАТУРА

1. Сасов А. Ю. Микротомография и цифровая обработка изображений на микроЭВМ «Искра 226» // Микропроцессорные средства и системы.—1986.— № 1.— С. 53—58.
2. Feng H.—Y.F., Pavlidis T. Decompositions of polygons into simpler components: feature generation for syntactic pattern recognition // IEEE Transaction on computers.—1975.—С. 24.—Р. 636—650.
3. Агронян О. С. Сегментация изображений с использованием замощения плоскости многоугольниками Вороного // Автоматика и телемеханика.—1986.— № 10.— С. 95—100.
4. Фоли Дж., Вэндэм А. Основы интерактивной машинной графики.—М.: Мир, 1985.—Т. 2.—362 с.

Статья поступила 13.04.88



Структура системы визуального вывода полутоновых изображений

геометрическую точность воспроизведения, а на микроэрастре  $3 \times 3$  достигается более высокое качество отображения полутонов.



## ИНТЕРПРЕТАТОР ЯЗЫКА БЕЙСИК С ГРАФИЧЕСКИМИ ВОЗМОЖНОСТЯМИ

Предлагается расширенная версия языка БЕЙСИК на основе BASIC/RAFOS. Дополнительно в него введены подпрограммы для работы с контроллером графического дисплея\*.

Интерпретатор реализован на микроЭВМ ДВК и работает под управлением операционной системы RT-11 версий 4 и 5. Для аппаратуры EIS (без оверлейности и расширенных сообщений об ошибках) интерпретатор занимает 58 блоков.

Графическая часть интерпретатора позволяет проводить:

независимое включение и выключение графического и алфавитно-цифрового режимов работы;

очистку, заполнение и инвертирование экрана;

абсолютное и относительное позиционирование в точку;

прорисовку точки и вектора в абсолютных и относительных координатах, окружности, прямоугольника (в режимах заполнения, стирания и инвертирования);

заполнение замкнутой области;

получение твердой копии на устройстве УВВПЧ-30-004 с режимами столбцовой и растровой печати.

Данная версия интерпретатора разработана на кафедре «Высшая математика» АН АТН.

В дальнейшем предполагается расширить интерпретатор подпрограммами прорисовки символов (в графическом режиме), а также подпрограммами для работы с целыми переменными на уровне битов (инверсия, сдвиги, логические операции).

Телефон 2-19-17, Андропов

Сообщение поступило 6.01.88

## ЦВЕТНАЯ ГРАФИКА РЕАЛЬНОГО ВРЕМЕНИ ДЛЯ МИКРОЭВМ ТИПА «ЭЛЕКТРОНИКА 60»

Пакет программ RTG (Real Time Graphic) выполняет строго ограниченную функцию: вывод графиков и диаграмм (закрашенных графиков) в темпе реального времени в заданные прямоугольные окна экрана. Сам факт поступления отображаемой информации позволяет отказаться от ее промежуточного накопления в памяти микроЭВМ, реализованного, например, в [1]. Пакет RTG ориентирован на контроллер «Электроника МС 4702» [2] с растром  $256 \times 256$  и количеством цветов — 8. Число окон неограничено. В каждое окно выводится до семи графиков или трех диаграмм с использованием «логического сложения» цветов. Тип отображаемых данных (целевой или действительный) для каждого окна фиксирован. Масштабирование каждой функции по оси ординат (линейное или логарифмическое) выполняется независимо от всех прочих, выводимых в данное окно. Ось абсцисс отображается тем же цветом, что и соответствующая функция. Управление графикой реального времени из программы пользовате-

Безобразов В. С., Димов В. А., Мякотин А. В. в др. Контроллер графического дисплея // Микропроцессорные средства и системы. — 1986. — № 3.

ля обеспечивается примитивами RTG: инициализировать RTG, захватить окно, вывести временную срезку функций в окно, освободить окно.

Временная срезка функций — массив текущих значений всех функций, отображаемых в указанном окне. Каждое окно заполняется слева направо, причем темп заполнения разных окон независим. При достижении крайней правой позиции окна заполнение автоматически продолжается с его крайней левой позиции. Текущая позиция отличается маркером окна.

Пакет RTG легко адаптируется к любому типу графического или алфавитно-цифрового дисплея, допускающего прямую адресацию курсора.

Область применения пакета RTG — информационно-измерительные системы, системы автоматизации научных исследований, АСУ ТП.

Пакет RTG реализован на Макроассемблере ОС РАФОС. Минимальный объем памяти, занимаемой в загрузочном модуле, — около 2 Кбайт плюс 70 байт на одно окно. Имеется версия, реализованная на Фортране ОС РАФОС.

443099, Куйбышев, ул. Льва Толстого, 23, КЭИС, каф. «Вычислительная техника»; тел. 39-11-10

## ЛИТЕРАТУРА

1. Ковалев Б. М., Николаенко Н. Н., Чернис Б. З. Система отображения информации в многоканальном измерительно-вычислительном комплексе. — М.: ВЗМИ, 1987. — С. 78—82. — (В сб. «Электропривод и автоматизация в машиностроении».)
2. Бабкин П. А., Солоненко О. П., Тарасов Б. В., Федорин В. Г. Средства цветной машинной графики для микроЭВМ «Электроника 60» // Микропроцессорные средства и системы. — 1987. — № 4. — С. 60.

Статья поступила 26.02.88

## ДРАЙВЕР ГРАФИЧЕСКОГО ДИСПЛЕЯ ДЛЯ ДВКЗ

При получении результатов работы программы на языке БЕЙСИК в виде графиков используют оператор PRINT TAB, который выводит изображение на экран дисплея или АЦПУ с помощью комбинации пробелов и символов текста.

Наличие в составе микроЭВМ ДВКЗ графического дисплея с разрешением  $400 \times 286$  точек делает желательным построение более качественных графиков. Предлагаемый драйвер графического дисплея для ДВКЗ позволяет строить графики из программ на языке БЕЙСИК, но его можно также использовать с программами на других языках.

В драйвере графического дисплея реализованы следующие возможности:

- очистка экрана (команда CL);
- инверсия экрана (команда IN);
- задание режима построения отрезков прямых линий путем засветки, гашения или инверсии изображения (команды LS, LC, LI соответственно);
- построение координатных осей (команды AX и AY);
- построение заполненного прямоугольника (команда SQ);
- разрешение/запрещение индикации (команды EN и DS соответственно).

Команды драйвера должны отделяться от параметров и других команд символами «пробел» или «возврат каретки». В качестве параметров команд используются

```

*****
;*
;* ДРАЙВЕР ГРАФИЧЕСКОГО ДИСПЛЕЯ
;* ДЛЯ ДВК-3
;*
*****
.TITLE GD
.IDENT /01.03/
.MCALL .DRBEG, .DREND, .DRFIN, .PUSH, .POP
TIMMIT=0
MMGHT=0
ERLHG=0
HDERR=1
GDRCS= 176640
GDBYT= 176642
GDADR= 176644
WCNT= 12-4
BUFF= 10-4
FUNC= 6-4
;

```

```

.DRBEG GD,0,0,0
MOV GDCGE,R5
MOV WCNT(5),R3 ;СЧЕТЧИК СЛОВ
MOV BUFF(5),R4 ;АДРЕС БУФЕРА
ASL R3
BEQ DONE
BCC ERR ;ЧТЕНИЕ ЗАПРЕЩЕНО
NEG R3
WRITE: MOV (4)+,R0
CALL GDE
SOB R3,WRITE

DONE:
BDINT: CLR WCNT(5)
EXIT: .DRFIN
ERR: BIS #HDERR,0-(5) ;ОШИБКА ДРАЙВЕРА
BR EXIT
GDEERR: TST (SP)+
.IRP Z,<CNT,NUM,PNT,FTR,AUX,MIN>
.CLR GDE'Z
.ENDR

BR ERR
.ENABL LSB
55x: INC GDEMIN
54x: MOV #10000,GDEPNT
GDE: BR 13x
BIC #C177,R0 ;РАЗБОР ПЕРЕДАНЫХ
CMP R0,#40 ;БАЙТОВ
BLE 40x
CMP R0,#'-
BEQ 55x
CMP R0,#'
BEQ 56x
CMP R0,#'
BEQ 60x
CMP R0,#'0
BLT 54x
CMP R0,#'9
BGT 67x
TST GDEPNT ;БЫЛА ТОЧКА?
BMI 1x
BEQ 11x
CMP R0,#'5 ;ОКРУГЛЕНИЕ
BLT 12x
INC GDENUM
12x: ASL GDEPNT
BR 13x
11x: MOV GDENUM,R1 ;ПРЕОБРАЗОВАНИЕ К
BIC #C17,R0 ;ЦЕЛОМУ ДЕСЯТИЧНОМУ
MUL #10,R1
ADD R0,R1
MOV R1,GDENUM
13x: MOV #1,GDEFTR
1x: RETURN
56x: BIS #4000,GDEPNT
RETURN
41x: TST (PC)+ ;ПРИЗНАК ЧИСЛА
GDEFTR: 0
BEQ 42x
CALL LLIMITS ;ОГРАНИЧЕНИЕ И (0,0)
ADD PC,R0 ;ЗАПОЛНЕНИЕ МАСКВА
ADD #XYARR-,R0 ;ПАРАМЕТРОВ
MOV R1,R0
CLR GDEFTR
CLR (PC)+
GDENUM: 0 ;АККУМУЛЯТОР ЧИСЛА
ADD #2,GDECNT
CMP GDECNT,#6
BLE 42x
CALL AUXLIN ;ОСИ И КВАДРАТ

```

```

60x: CALL LINE00
MOV K1,X2
MOV Y1,Y2
AUXEXT: CLR GDECNT
42x: RETURN
40x: TSTB GDEPNT
BEQ 43x
MOV #399,GDENUM
43x: CLR (PC)+ ;ПРИЗНАК ТОЧКИ
GDEPNT: 0
TST GDEMIN
BEQ 44x
CLR GDENUM
CLR (PC)+ ;ПРИЗНАК МИНУСА
GDEMIN: 0
44x: TST GDEMIN
BEQ 41x
CLR R0
MOV PC,R1
ADD #MNEMS-,R1
2x: TST QR1 ;МНЕМОНИКА
BEQ GDEERR ;НЕ НАЙДЕНА
CNP GDEMIN,(1)+ ;ПОИСК МНЕМОНИКИ
BEQ 3x
TST (0)+
BR 2x
67x: TST GDEPNT
BNE 70x
MOVB GDEMIN+1,GDEMIN ;ЗАПОЛНЕНИЕ
MOVB R0,GDEMIN+1 ;МНЕМОНИКИ
70x: RETURN
3x: CLR (PC)+ ;МНЕМОНИКА КОМАНДЫ
GDEMIN: 0
CLR (PC)+ ;СЧЕТЧИК ПАРАМЕТРОВ
GDECNT: 0
.IRP Z,<CL,IN,LS,LI,LC,EN,AX,AY,SQ>
BR Z
.ENDR
DS: BIC #10000,GDRCS ;ЗАПРЕТИТЬ ИНДИКАЦИЮ
RETURN
EN: BIS #10000,GDRCS ;РАЗРЕШИТЬ ИНДИКАЦИЮ
RETURN
.ENABL LSB
CL: MOV #105037,1x ;CLRБ 0#
CLR X2
MOV #285.,Y2
BR 2x
IN: MOV #105137,1x ;СОВБ 0#
2x: MOV #50.*286.,R0
CLR #GDADR
1x: CLRБ #GDBYT ;ОЧИСТКА/ИНВЕРСИЯ
INC #GDADR ;ЭКРАНА
SOB R0,1x
RETURN
LS: MOV #50037,R0 ;BIS R0,0#
BR 3x
LI: MOV #74037,R0 ;XOR R0,0#
BR 3x
LC: MOV #40037,R0 ;BIC R0,0#
3x: MOV R0,GRFC
RETURN
AX: MOV #2,GDEAUX ;ОСЬ АБСЦИСС
RETURN
AY: MOV #4,GDEAUX ;ОСЬ ОРДИНАТ
RETURN
SQ: MOV #6,GDEAUX ;КВАДРАТ
RETURN
.AUXGRF
.ENABL LSB
MNEMS: "CL","IN","LS","LI","LC","EN","AX","AY","SQ","DS",0
LINE00: .PUSH (R5,R4,R3)
JSR R5,LIN
XYARR=:
X1: 0 ;МАССИВ КООРДИНАТ
Y1: 0
X2: DY: 0
Y2: DX: 285.
.POP (R3,R4,R5)
RETURN
.ENABL LSB
MASK: .BYTE 1,2,4,10,20,40,100,200 ;БИТОВАЯ МАСКА
POS: 1 ;АДРЕС БАЙТА
ADR: 0
XYP: .PUSH R0
MOV POS,R0
ADD R2,PC ;+++
.IRP Z,<20,1,2,3,4,5,6,7,10,11,12,13,14,15,16>
BR 'Z'x
.ENDR ;+++

```

```

17к: SUB    $50.,ADR    ;ОТРАВОТКА
3к:13к: CALL ROLPOS   ;ДВИЖЕНИЯ ТОЧКИ
2к: BR     GRF         ;ПО ОСЯМ
5к: ADD    $50.,ADR    ;И ДИАГОНАЛЯМ
1к:11к: CALL RDRPOS   ;
20к: BR     GRF
7к: CALL  ROLPOS
4к:6к: ADD    $50.,ADR
10к: BR     GRF
15к: CALL  RDRPOS
14к:16к: SUB    $50.,ADR
12к: ;BR     GRF
GRF: MOV    ADR,0$GDADR
GRFC: XOR   R0,0$GDBYT
      MOV   R0,POS
      .POP  R0
RETURN
RDRPOS: ROLB  R0        ;СДВИГ ТОЧКИ
        BCC  40к       ;ВПРАВО
        ROLB  R0
        INC  ADR
40к: RETURN
ROLPOS: RORB  R0        ;СДВИГ ТОЧКИ
        BCC  41к       ;ВЛЕВО
        RORB  R0
        DEC  ADR
41к: RETURN
.ENDBL LSB
2,10,2
LIN:: MOV    PC,R4
      CMP   -(4),-(4)
      MOV   (5)+,R0
      MOV   (5)+,R1
      MUL   $50.,R1
      MOV   R1,R2
      MOV   R0,R1
      BIC   $^C7,R1
      ADD   PC,R1
      MOVB  MASK-(1),POS
      ASH   $-3,R0
      ADD   R0,R2
      MOV   R2,ADR
57к: CMP   -(5),-(5)
      CLR   R2
      MOV   #-1,R3
      MOV   4(5),R0
      SUB   (5)+,R0
      BGE   1к
      DIS   $^B100,R2
1к: NEG   R0
      MOV   4(5),R1
      SUB   (5)+,R1
      BGE   2к
      BIS   $^B10000,R2
2к: NEG   R1
      CMP   R0,R1
      BCC   3к
      .PUSH R1
      MOV   R0,R1
      .POP  R0
      TST  -(4)
3к: INC   R1
      MOV   R0,-(SP)
      ADD   R1,SP
      BIC   $^B1010,R2
      CALL  XYP
      DEC   QSP
      BLE   10к
4к: BIC   $^B1010,R2
      BIS   QR4,R2
      DEC   QSP
      ADD   R1,R3
      CMP   R3,R0
      BLD   5к
      SUB   R0,R3
      BIS   -2(4),R2
      DEC   QSP
5к: CALL  XYP
      TST  QSP
      BGT  4к
10к: TST  (SP)+
      CMP  (5)+,(5)+
      RTS  R5
      .REND GD
.END

```

Рис. 1

```

10 REM ПРОГРАММА ПОСТРОЕНИЯ РОЗЫ RO=A*SIN(M/L*F)
20 REM В ПОЛЯРНЫХ КООРДИНАТАХ
30 OPEN 'GD:' FOR OUTPUT AS FILE $1
40 INPUT M,L,D,A
50 M=M/L \ L=L*PI
60 D=PI/D/M \ F=-D
70 PRINT $1,'EN LS CL',200,160
80 F=F+D
90 B=A*SIN(M*F)
100 PRINT $1,200+B*COS(F),160+B*SIN(F),',',
110 IF F<L THEN 80
120 PRINT $1,'AX 80 160 330 20'
130 PRINT $1,'AY 200 40 20 285'
140 CLOSE $1

```

RUN

5  
1  
17  
110

Рис. 2

READY

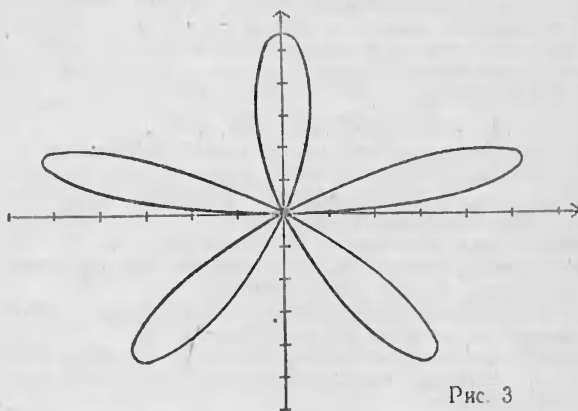


Рис. 3

вещественные с фиксированной запятой или целые десятичные числа в символическом коде. Числа задают абсолютные координаты точек в диапазоне 0...399 для абсцисс и 0...285 для ординат.

Графики вычерчиваются в виде отрезков прямых между соседними точками. Если задаются четыре координаты X1, Y1, X2, Y2, то проводится отрезок от первой точки до второй; если задаются две координаты X1, Y1, за которыми следует запятая, то проводится отрезок от предыдущей точки к заданной.

Координатные оси строятся с помощью команд AX, Xп, Y, Xк, Д и AY, X, Yп, Д, Yк, где Xп, Yп, Xк, Yк — соответственно координаты начал и конца осей; X, Y — смещение осей относительно точки (0, 0); Д — шаг разметки. Ось строится от точки (п) к точке (к), на конце оси вычерчивается стрелка.

Заполненный прямоугольник строится после подачи команды SQ X1, Y1 X2 Y2, где X1, Y1, X2 Y2 — координаты точек, расположенных на диагонали прямоугольника.

Для всех перечисленных команд действует режим построения, заданный командами LS, LC, LI.

Текст драйвера приведен на рис. 1. Для генерации рабочего драйвера необходимо выполнить команды  
MACRO GD  
LINK/EXE:GD.SYS GD  
DEL/NOQ GD.OBJ  
INS GD:

Текст программы на языке БЕЙСИК иллюстрирует взаимодействие с драйвером графического дисплея (рис. 2).

Содержимое экрана дисплея после завершения программы см. на рис. 3.

Телефон 290-79-13, Киев

Сообщение поступило 08.07.88

**ДРАЙВЕРЫ ГРАФИЧЕСКОГО ТЕРМИНАЛА И ГРАФИЧЕСКОГО ПРИНТЕРА**

Драйверы представляют собой удобное средство для получения твердых копий графических изображений, формируемых на экране графического дисплея (ГД) комплекса ДВКЗ, на матричном печатающем устройстве (ПУ), имеющемся в составе этого комплекса.

Драйверы, предназначенные для использования в операционной системе ОС РАФОС, имеют наборы SET-параметров, позволяющие настраиваться на конкретный тип ПУ (пока реализованы УВВПЧ-30 — 0004, D-100 и ROBOTRON CM 6329.02-M), а также осуществлять очистку экрана ГД, наложение изображений, инвертирование и т. д.

Сама операция получения твердой копии заключается в поблочном копировании устройства T: на устройство P: с нулевого блока и так, пока не кончится файл. Таким образом, программный модуль HRDCOP, осуществляющий получение твердой копии, должен выполнить три процедуры: загрузить файлы GT.SYS и GP.SYS, осуществить операцию копирования, выгрузить драйверы. В простейшем случае можно воспользоваться командой монитора

**COPY GT: GP:**

Следует заметить, что предлагаемый подход к операции получения твердой копии, несмотря на некоторую громоздкость, является более предпочтительным по сравнению с общепринятым методом, когда модуль HRDCOP осуществляет непосредственное, т. е. без использования драйверов, копирование графического изображения, так как здесь обеспечивается:

независимость разрабатываемого программного обеспечения (ПО) от типа применяемого ПУ;

легкость перевода готового ПО на другие типы ПУ, так как отпадает необходимость перетрансляции исходных программ;

преимущество ПО по отношению к новым типам ПУ и ГД — необходимо лишь изменить соответствующие драйверы с добавлением новых SET-параметров;

возможность записи изображения с экрана ГД на внешнее запоминающее устройство в виде файла, хранения и обратного копирования на ГД или ПУ.

Вышперечисленное особенно важно для внедрения операции получения твердой копии в графически-ориентированные версии интерпретирующих языков высокого уровня (Фокал, БЕЙСИК) с введением соответствующих операторов.

*Телефон 93-429, Псков, Берников Ю. Л.*

*Сообщение поступило 21.04.87*

УДК 681.3

**В. Г. Адрианов, Д. А. Сухов**

**ДРАЙВЕР ДЛЯ ГРАФОПОСТРОИТЕЛЯ ТИПА Н-306, Н-307 С АНАЛОГОВЫМ ВХОДОМ**

Графопостроитель типа Н-306 широко применяется для вывода графической информации в измерительно-вычислительных комплексах на основе микроЭВМ «Электроника 60» и ДВК с ОС RT-11 (РАФОС и ФО-ДОС). Включение в систему соответствующего драйвера существенно облегчает создание программного обеспечения для вывода графической информации. Предлагаемый драйвер позволяет значительно расширить возможности использования «интеллектуальных» графопостроителей. Вывод графиков на аналоговые графопостроители осложнен необходимостью отслеживать положение пера. Поэтому необходимо устанавливать соответствующие задержки. Имея драйвер в системе, пользователь освобождается от такой рутинной рабо-

ты, как настройка с помощью задержек своих программ на соответствующий тип графопостроителя. Настройка драйвера на конкретный тип графопостроителя осуществляется SET-параметрами SPEED, DELAY. Толщина линий определяется скоростью вывода точек.

Драйвер управляет работой графопостроителя через стандарт КАМАК со следующим набором модулей: два цифро-аналоговых преобразователя (ЦАП-10), выходной регистр (польский 350), установочный счетчик (польский 420-А), генератор импульсов (польский 730-А).

Данные из ЭВМ на графопостроитель передаются по прерываниям от счетчика. Генератор и счетчик импульсов моделируют интервальный таймер. Счетчик дает прерывание на переполнение, следовательно, чтобы получить необходимый интервал времени, нужно записать в него логическое дополнение необходимого интервала времени. Драйвер управляет разными задержками с помощью записи соответствующих чисел в счетчик. Вместо указанного набора модулей можно использовать специальный модуль управления графопостроителем.

Разрядность ЦАП влияет на качество линий. В драйвер поступает поток данных в виде набора координат в формате целых чисел. Вывод осуществляется отрезками от текущей точки в новую точку. Знаковый бит в новой точке управляет видом отрезка, который выводится по точкам растра. Вывод графической информации заканчивается после двойного нуля в потоке точек или по исчерпанию элемента очереди.

Драйвер транслируется и компоуется стандартными средствами ОС RT-11. Прилагается специальный файл определен PWDEF.MAC, в котором, изменяя соответствующие параметры, можно адаптировать драйвер на разные схемы управления, включая специальные модули управления для такого типа устройств:

**EXEC/EXE: SY:PW.SYS/OBJ:PW PWDEF+PW INST PW**

После установки драйвера его можно использовать обычным способом, разрешенным в RT-11 для устройств последовательного вывода.

Вывод осуществляется из любых доступных языков программирования.

*Телефон 186-88-61, Ленинград*

*Статья поступила 10.06.88*

**КРАТКОЕ СООБЩЕНИЕ**

При разработке программного обеспечения наибольшую часть времени программист работает с редактором текста. При этом скорость работы определяется квалификацией программиста и удобством работы с программой-редактором.

Кадровые (экранные) редакторы более удобны в работе, чем строчные, но, к сожалению, не для всех ЭВМ они предоставляются. Для любых типов микроЭВМ на основе МП КР580ИК80 на кафедре АЭИ Киевского политехнического института разработан аппаратно-независимый кадровый редактор, удобный в работе, обладающий достаточным набором сервисных функций. Аппаратная и системная независимость обеспечивается сменными подпрограммами для взаимодействия с операционной системой и применяемым дисплеем.

С его помощью редактируются файлы практически неограниченной длины (с длиной строки, большей строки экрана дисплея). Среди сервисных функций — переход на строку с указанным номером, переход вверх-вниз по тексту на указанное число строки, его ликвидация, поиск строки по образцу, удаление строк до обнаружения образца, вставка в текст другого текста, запись на файл указанных участков текста.

Программа прошла «обкатку» в операционных системах CP/M и ISIS—II.

*252056, Киев-56, пр. Победы, 37, КПИ, кафедра АЭИ; тел. 441-97-67, Федиву В. В.*

*Сообщение поступило 23.04.88*

А. В. Лигская

## ПРОБЛЕМЫ ЗАЩИТЫ АВТОРСКИХ ПРАВ НА ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ В США

Защита прав интеллектуальной собственности — один из наиболее острых вопросов в развитии наукоемких отраслей. В США в его основе лежит конституционное положение, обязывающее конгресс «способствовать прогрессу науки и полезных ремесел путем сохранения в течение ограниченного времени исключительных прав для автора или новатора на его, соответственно, сочинение или открытие» [1]. В 1954 г. в материалах Верховного суда США по делу «Mazer vs. Stein» это положение комментировалось следующим образом: «Экономическая философия, содержащаяся в статье, обязывающей конгресс выдавать патенты и авторские права, состоит в том, что поощрение индивидуальных усилий путем личного вознаграждения является лучшим способом повышения общественного благосостояния через стимулирование творческой деятельности авторов и изобретателей» [2].

Более 20 лет назад, в 1964 г. компьютерные программы стали приниматься на регистрацию Управлением по авторским правам США, и уже в 1965 г. это Управление в одном из своих документов поднимало ряд вопросов, касающихся методов и средств защиты прав собственности на программное обеспечение, многие из которых остались неразрешенными и поныне.

Наиболее сильный всплеск интереса к этой проблеме произошел во второй половине 70-х годов и был вызван принятием нового закона об авторских правах в США.

В 1974 г. в период подготовки новой редакции закона конгресс США утвердил создание Комиссии по использованию авторских прав в областях новейших технологий (CONTU — National Commission on New Technological Uses of Copyrighted Work) для изучения вопроса об изменениях в Законе об авторских правах в связи с развитием новой техники и технологии. В рамках CONTU функционировал специальный подкомитет по программному обеспечению. Его цель — составление рекомендаций о включении ПО в сферу действия Закона об авторских правах и средствах защиты авторских прав на программы. В 1976 г. CONTU закончила свои исследования, но еще до того был принят новый закон об авторских правах, в котором ПО упомянуто не было.

Объекты, относящиеся к интеллектуальной собственности (ИС), делятся на три типа: work of art (произведения искусства), work of fact (энциклопедии, справочники, базы данных) и work of function (изобретения, нововведения, ноу-хау). Соответственно существует и несколько механизмов защиты прав ИС — авторские права, патенты, коммерческие тайны и общее законодательство о несправедливой конкуренции.

Из трех типов ИС программы могут быть наиболее естественно классифицированы как «work of function», однако они часто не удовлетворяют требованиям, оговоренным в патентном праве (иметь физическую форму и содержать в себе изобретение или открытие). Действительно, как говорилось в докладе, подготовленном Национальным бюро стандартов для CONTU, «...очень незначительное число программ содержит новые концепции, такие уникальные, как например, симплекс-метод для решения задач линейного программирования или быстрое преобразование Фурье» [3]. Более того, патент защищает только функциональные аспекты программы и считается, что он может быть выдан только на программу, которая управляет выполнением физических операций, т. е. патентуются лишь те программы, которые непосредственно связаны с функционированием промышленного оборудования (управлением технологи-

ческими процессами, станками с ЧПУ, электронным полиграфическим оборудованием). Таким образом, очень незначительная часть программ может быть запатентована.

Еще в 1965 г. Управление по авторским правам поднимало вопросы о том, является ли программа авторским произведением и можно ли физическую запись программы на магнитном носителе рассматривать как письменную форму (form of writing). Поводом для постановки таких вопросов послужило распространение среди юристов и государственных чиновников мнение, что программа — это часть ЭВМ, так как без ПО компьютер функционировать не может и, как любое техническое средство, она просто замещает часть живого труда. В то же время ПО является результатом коллективного труда, основанного на фиксированных научных и технических достижениях. Следовательно, оно не может рассматриваться ни как произведение искусства, ни даже как авторская работа.

Американские эксперты в области вычислительной техники, профессиональные программисты, бизнесмены и юристы, специализирующиеся в обслуживании отрасли ПО, склоняются к тому, что на современном уровне развития программирование является если не искусством, то техническим ремеслом специфического рода, которое значительно ближе к искусству, чем к науке. Можно привести ряд характерных высказываний на эту тему. Например, в докладе, подготовленном ADAPSO — торговой ассоциацией фирм по программному обеспечению и вычислительным услугам, для CONTU программы сравниваются с сочинениями в высокотехнологических сферах деятельности. Вице-президент фирмы Phillips по научным исследованиям описывает процесс программирования следующими словами: «У меня до сих пор нет метода доказательства правильности программ. Я не имею даже простейших средств для измерения эффективности конкретной программы. Поэтому я работаю как свободный художник. Другой такой же художник оценивает мою работу в соответствующих нашим методам терминах: «Хороший стиль! Эти места особенно элегантны» или — «Вот здесь несколько рыхловато» [4].

Если принять точку зрения, что программа является авторским произведением, то любые виды программ независимо от того, управляют ли они работой компьютера, технологическим процессом, или предназначены для решения прикладных задач, т. е. независимо от функциональных особенностей, могут входить в категорию «произведения искусства» и подлежать защите через механизм авторских прав. Но именно в этом и коренятся проблемы, ограничивающие эффективность защиты программ в форме авторских прав, так как авторские права защищают только форму выражения идеи. В данном случае символьный код программы, и не распространяются на функции, логическую структуру и методологию решения задачи. В 70-х годах была широко распространена защита программ с помощью коммерческих тайн, предназначенная в основном для обеспечения сохранения в секрете технологических процессов, таких, например, как рецепт изготовления «Кока-колы». Эта форма защиты программ также обладает рядом недостатков. Она требует больших затрат на обеспечение неразглашения оговоренных в контракте секретов, пригодна лишь для программ, разрабатываемых компаниями для собственных нужд или по индивидуальным заказам. Широкое применение этого вида защиты объяснялось, главным образом, преобладанием на рынке ПО до 80-х годов контрактного программирования и незначительными объемами продаж готовых программ.

По данным ADAPSO 35 % членов этой организации использовали для защиты программ коммерческие тайны, 33 % — авторские права, 6 % — патенты, причем обычно использовалась не одна форма защиты, а комбинация из двух или трех форм, и мало кто отдавал

предпочтение одной из них, считая ее наиболее эффективной [3]. Это подтверждают и данные, приводимые в докладе подкомитета CONTU по программному обеспечению. Несмотря на то, что примерно в трети случаев для защиты программ использовались авторские права, с 1964 по 1976 г. в Управлении по авторским правам было зарегистрировано только 1300 программ, из них 971 принадлежала двум компаниям — IBM и Burroughs. В то же время число программ, на которые ставились пометки об авторских правах без регистрации их в Управлении, несравнимо больше, что говорит о неуверенности компаний в этой форме защиты.

Исследования, проведенные CONTU, ADAPSO, Национальным бюро стандартов и другими организациями в период подготовки закона об авторских правах, показали, что нецелесообразно выделять одно из средств защиты как наиболее эффективное и адекватно отражающее особенности ПО как специфического типа интеллектуальной собственности. В итоговом докладе, подготовленном CONTU для конгресса, особо подчеркивалось, что закон об авторских правах не должен ограничивать использование любых других средств защиты прав интеллектуальной собственности на программы.

В то же время очевидным было и то, что необходимо разработать специальный механизм для защиты прав собственности на программное обеспечение, так как по существующей классификации программа является чем-то средним между work of art и work of function, лишь частично охватываемым каждым из этих двух типов ИС. По образному выражению Д. Сандина — президента фирмы Application Software Inc: «Мы пытаемся объединить яблоки и апельсины. Это абсурд. Патенты предназначены для технических средств. Авторские права — для книг и средств массовой информации». Поэтому, заявляет он, необходимо разработать новый закон «в соответствующей терминологии» [3].

О том, как и кто должен разрабатывать такой закон, существуют различные мнения. Одни считают, что разработку нового закона лучше всего поручить Национальному бюро стандартов или другому федеральному ведомству, непосредственно связанному с техникой и новыми технологиями. Другие предлагают в очередной раз модифицировать закон об авторских правах, разделив в нем обычные объекты, относящиеся к традиционному пониманию категории «произведения искусства», и объекты, порожденные развитием новых технологий, в частности программы, и сформулировать для каждого типа объектов специальные правила защиты. В этом случае разработкой нового закона должно заняться Управление по авторским правам. Но все едино во мнении, что в работе над новым законом обязательно должны принимать участие эксперты по программированию и вычислительной технике.

Споры вокруг этой проблемы не утихают уже более десяти лет, несмотря на то, что в 80-е годы стало ясно, что совершенствование защиты прав собственности на программы идет в рамках законодательства об авторских правах. В 1980 г. в статью 117 Закона об авторских правах была внесена поправка, включающая программы в сферу действия этого закона, но не поясняющая методов и границ их защиты. Причинами этого послужили, во-первых, четко проявившаяся в этот период тенденция к возрастанию доли использования готовых программ по сравнению с контрактным программированием и, во-вторых, быстрый рост рынка программ для персональных компьютеров (ПК). Именно на рынке программ для ПК проблемы, связанные с защитой программ, наиболее ярко высечены в настоящее время, так как здесь наиболее велика доля программ, «покупасмых с полки».

Проблемы защиты ИС пока не так сильно затронули компании, разрабатывающие программы для базовых и мини-ЭВМ. Нарушение авторских прав не получило широкого распространения на этих сегментах рынка ПО. Программы здесь значительно более сложные, чем

программы для ПК, включают миллионы строк кода. В сопровождение программных систем для больших машин входит широкий спектр работ, требующий специалистов высокого класса, как правило, из фирм-разработчиков. Большую роль играют и такие моменты, как высокая стоимость аппаратуры и значительная доля разработок по индивидуальным заказам или внесения в универсальные программы изменений, вызванных специфическими запросами пользователей.

Однако ситуацию, сложившуюся сейчас на рынке программного обеспечения для ПК, можно рассматривать как модель будущих проблем всей отрасли ПО.

В последние два-три года отмечается очередной всплеск внимания к проблеме защиты авторских прав на программы. Она обсуждается на трех уровнях. Вопросы первого уровня связаны с незаконным копированием программ и проблемой получения авторских гонораров. Вопросы второго уровня возникают в связи с распространением плагията в программировании. Вопросы третьего уровня — международные, порожденные ростом торговли программами на мировом рынке и несогласованностью законодательства об охране авторских прав в различных странах.

### Копирование программ

Незаконное копирование программ пользователями широко распространено на рынке программ для персональных компьютеров. Занимаются им как отдельные пользователи, так и целые корпорации. Копирование программ в первую очередь бьет по программистам, лишая их части, нередко весьма существенной, заработанных гонораров. По оценкам экспертов на одну законно снятую копию приходится от одной до трех нелегальных. Специалисты из ADAPSO оценивают потери, которые несет рынок программ для ПК из-за распространения незаконного копирования, приблизительно в 50 % от общего объема продаж на нем [5]. Определенную часть доходов из-за незаконной деятельности пользователей теряют компании-разработчики и издатели ПО, но больше всего страдают так называемые «свободные программисты». Частной практикой по разработке программ в свободное от основной работы время занимаются более половины из 250 000 профессиональных программистов [6].

Обычно в документации к ПО указывается, что купленная программа может использоваться только на одном компьютере. Но такая оговорка равнозначна контракту, подписанному только одной стороной, и юридически не обязывает пользователей следовать ей. Получить компенсацию через суд можно, если тиражирование программы приобрело массовый характер. Известен ряд случаев, когда корпорации, купившие одну копию программ и многократно размножившие ее для использования в различных офисах и на множестве машин, привлекались к судебной ответственности и по решению суда были принуждены выплачивать крупные суммы компенсации компании-разработчику. Однако трудно приостановить размножение программ, приобретенных для индивидуальных целей и распространяемых на основе личных связей. В частности, многие ученые, инженеры и другие специалисты, использующие ПК в домашних условиях, призывают своих коллег обмениваться программами, не считая такой обмен копиями незаконным.

Технические средства защиты от копирования, как правило, неэффективны, пользователь может разрушить систему защиты программ без особых усилий.

Сами возможности защиты программ с помощью технических средств не так уж широки. Иногда программы снабжаются специальным кодом, который гарантирует, что программа будет работать только на одной машине. Практикуется этот метод компаниями, выпускающими одновременно и ПК и программы для них, и реализуется частично аппаратно. Недавно в продаже появились гибкие диски Techline компании Vault, специально пред-

назначенные для компаний, разрабатывающих ПО, копирование программы с которых возможно только после набора серии паролей пользователем и техническим специалистом из фирмы-разработчика. Такие диски предназначены для вполне определенных марок ПК [7]. Но возможность использования аппаратных средств для защиты от копирования ограничена вследствие большого разнообразия различных моделей и марок персональных компьютеров на рынке США.

В некоторых случаях используются и так называемые «бомбы замедленного действия». Основное назначение таких «бомб» — гарантировать оплату программы. Перед тем как купить программу, пользователь обычно хочет удостовериться, что это именно то, что ему нужно, и научиться работать с ней. Раньше компании практиковали обучение пользователей и демонстрацию пакетов программ в специализированных учебно-демонстрационных лабораториях или представляли пользователю неполный, упрощенный вариант программы для испытания. Но эти методы не удовлетворяли потребителей. «Бомбы замедленного действия» предоставляют полную программу для опытного использования обычно на срок от одного до трех месяцев, по истечении которого пользователь должен решить, купит ли он программу или нет. Существуют два типа «бомб замедленного действия». Первый — это встроены часы, которые включаются при первом обращении пользователя к программе и по истечении определенного времени, установленного разработчиком, блокируют работу программы. Второй тип «бомб замедленного действия» использует так называемый «метод случайных последовательностей». Работа программы блокируется в том случае, если определенные команды (команды копирования или выполнения некоторых операций) повторяются с заданной частотой или используются больше допустимого числа раз. При копировании программы в копию переносится значение счетчика или таймера из исходной программы, и, следовательно, каждая следующая копия оказывается работоспособной в течение все меньшего времени.

Однако большинство программ не снабжаются средствами защиты от копирования вообще. Более того, первая операция, которую учится выполнять пользователь, — снятие запасной копии с программы. Иметь такую копию необходимо, так как неосторожные действия пользователя могут случайно разрушить код программы или уничтожить данные. Все меньше число компаний включает какие-либо средства защиты от копирования в свои программы: невозможность иметь запасную копию на случай нарушения кода программы снижает ее ценность.

В настоящее время рынок программ для ПК еще не достиг насыщения, спрос на программы превышает их предложение, рыночные цены значительно выше реальной стоимости. Специфика этого рынка состоит также и в том, что стоимость разработки программы очень высока, а тиражирование просто и дешево. Среди независимых программистов большое распространение приобретают методы продажи программ, получившие название freeware или user supported software (свободно распространяемые программы или программное обеспечение, поддерживаемое пользователями). Программист поощряет копирование, и пользователю предлагается распространять копии программы среди своих знакомых и коллег в неограниченных количествах. Каждый, кому программа понравится и он решит ее использовать, должен прислать весьма умеренную плату за нее по почте. Например, автор популярной программы PC-Talk III продает свою программу за 35 долл. (в 10 раз ниже средней цены на подобные программы на рынке США). В ответ на вылату денег пользователю гарантируется высылка новой улучшенной версии программы. Внесение платы не контролируется поставщиком и целиком лежит на совести пользователя.

Программисты, практикующие такой метод, оценива-

ют количество пользователей программ, заплативших за них составляет примерно 10 %, но при циркулировании нескольких тысяч копий они получают доходы, с избытком покрывающие их затраты. В эпоху «информационного общества» необходимы и новые подходы к экономике, так как «стоимость размножения информации мала по сравнению со стоимостью ее производства» [5, 6].

### Развитие юридических форм защиты программ

В последние два-три года по судам прокатилась волна дел о программном плагиате.

По статье 102,6 Закона об авторских правах, действующего с 1 января 1978 г., авторскими правами может быть защищена только форма выражения идеи «Ни в одном случае, — записано в этой статье, — авторские права не распространяются на какую-либо идею, процедуру, процесс, систему, метод или операцию, концепцию, принцип или открытие, независимо от формы, в которой они описаны, объяснены, проиллюстрированы или воплощены в конкретной работе» [2].

Точных критериев классификации программного плагиата нет, решение каждого случая полностью находится на усмотрении суда. Верховный суд США не планирует вмешиваться в урегулирование этого вопроса путем введения определенных правил или стандартов.

По поправке, внесенной в 1980 г. в статью 117 Закона об авторских правах, защите подлежат исходный (символьный) код программы, т. е. запрещен лишь простой перенос программы с одной ЭВМ на другую и перевод с одного алгоритмического языка на другой, аналогично запрету на несанкционированные переводы литературных произведений. Это положение было расширено судами и включило в себя объектный код и микродод прикладных программ, затронуло коды операционных систем, несмотря на то, что авторские права не защищают «процесс, систему или метод действия». В протоколе по делу фирмы «Apple Computer Inc.» против «Franklin Computer Corp.» было записано, что в отношении операционных систем возможна защита только формы написания команд управления компьютером, а не метод, которым осуществляется выполнение операционных функций управления системой.

Следующим шагом стало включение в сферу действия авторских прав изображения на экране и команд управления программой с экрана независимо от того, был ли скопирован код программы или нет. Подобные решения сохранились в деле «Unison World Inc.» против «Broderbund Software Inc.» о копировании программы Print Shop. Решение по делу «Unison» вызвало множество возражений. Требование обеспечения уникальности команд управления с экрана неизбежно приведет к тому, что сами команды станут громоздкими и неудобными и затруднится переход от использования одной программы к другой. Разработчики стремятся унифицировать систему команд, используемых в пакетах различного назначения. По словам генерального юрисконсульта компании «Unison», требование сделать команды управления с экрана непохожими друг на друга в различных пакетах аналогично требованию выпускать пишущие машинки с разными клавиатурами [8]. Однако данное судебное решение неоднократно использовалось как прецедент при вынесении заключения по аналогичным искам.

Таким образом, практически все элементы программы могут быть в настоящее время защищены авторскими правами. Несмотря на это сторонники усиления защиты прав на программы выражают недовольство тем фактом, что через авторские права можно защитить лишь форму написания программы, а не ее функции и логическую структуру. Одни и те же функции могут быть закодированы различным образом, поэтому разные программные коды могут дать функциональный эквивалент программы, т. е. скопировать программу можно и не копируя кода.

В 1986 г. были приняты два принципиальных судебных решения по делам о программном плагате, расширяющих границы защиты программ за пределы кода до способа выполнения программы, ее «структуры, последовательности и организации». Программист Элайн Виллан возбудил дело против «Jaslow Dental Laboratory», по заказу которой он разработал программу финансовых расчетов для мини ЭВМ IBM серии 1. В 1983 г. в фирме написана программа финансовых расчетов на БЕЙСИКе для ПК. Апелляционный суд третьего округа установил, что программа фирмы не была простой транслитерацией программы Виллана, но нарушала его авторские права, так как обе программы были «существенно схожи» в структуре, организации и последовательности действий. Аналогичное решение вынес тот же суд по делу «Lotus Development Corp.» против «Mosaic Software».

Программист считается нарушителем авторских прав, если его программа просто похожа на другую программу с зарегистрированными авторскими правами. Для оценки используется интегрированный тест на идентичность двух программ. Его проводят непрофессионалы и эксперты, если предварительно установлено, что в программах заложена общая идея, однако результат часто зависит от субъективной оценки непрофессионалов (юристов). Об этом говорят расплывчатые формулировки, встречающиеся в следственных документах: «...обычный человек вряд ли мог бы отрицать, что две программы выглядят похожими»; «общее ощущение от этих программ практически одинаковое»; «одна программа выглядит похожей на другую» [2, 9].

Расширение сферы защиты авторских прав на программы ограничивает деятельность так называемых «клонеров», копирующих популярные программы или вносящих лишь тривиальные изменения. С другой стороны, похожими могут оказаться и программы, разработанные совершенно самостоятельно, независимо друг от друга. Для решения этой проблемы предлагается ввести в практику разделение авторских прав и патентов между несколькими компаниями. Пострадавшими могут оказаться не только программисты, но и пользователи. Фирма, признанная нарушителем авторских прав, обязана изъять свои программы из продажи и лишается возможности развивать их. По решению суда уже проданные программы могут быть конфискованы у пользователей, или фирма, понесшая большие финансовые потери в результате судебных издержек и оказавшаяся на грани банкротства, может отказаться по предписанию суда от лицензионных соглашений и объявить залицензированное ПО как имущество несостоятельного должника.

Более сильная защита авторских прав на ПО может оказать отрицательное влияние на конкуренцию, распространение передовой технологии и поток нововведений в отрасли. Так, например, профессор Йельского университета Р. Левин считает, что «хотя имитация не приносит реальной общественной выгоды, иногда распространение опыта из одной компании в другую может быть полезно» [10]. Уменьшение конкурирующих товаров на рынке способствует ослаблению стимулов к снижению цен и улучшению программ. Увеличится время и затраты на разработку новых программ, так как потребуются глубокое изучение уже имеющихся на рынке пакетов. Рост расходов на исследования и разработки отрицательно отражается на ценах, возникает опасность неоправданной монополизации рынка одним поставщиком. Повышение степени монополизации также снижает эффективность конкуренции. В целом, то что хорошо для одной компании, может быть плохо для всей отрасли. Опыт западно-европейских стран в производстве вычислительной техники убедительно показал, насколько неэффективны структуры, основанные на принципе «единого лидера».

Быстрое развитие информационной технологии постоянно выдвигает новые проблемы. С появлением на рын-

ке в 1987 г. микропроцессорных схем RISC («Reduce Instruction Set Computer» chip), в которых микрокод частично заменен схемными решениями, вновь возник вопрос о применении «правила Ингрэма», защищающего микрокод авторскими правами, а не патентом. Отмена этого правила может произойти в 1988 г. По причине нарушения судебной этики должен состояться пересмотр дела «Intel Corp.» против «Nippon Electric Company», в результате которого было установлено «правило Ингрэма» [11].

Все больше сомнений вызывает возможность защиты структуры и организации программы вследствие расширения использования в ПО четвертого поколения средств, с помощью которых пользователь может менять конфигурацию программы. В частности, такими средствами снабдила свою новую программу компания Lotus — один из наиболее активных борцов за введение более строгих законов.

### Международные аспекты

Огромное количество копий и подделок самых разнообразных товаров производится в развивающихся странах и продается под торговыми марками крупнейших фирм. На рынок США уже проникли персональные компьютеры IBM и Apple, собранные в Гонконге и Тайване. За ними устремился поток копий популярных пакетов программ для ПК. В магазинах Сингапура ассортимент квазиамериканских программ насчитывает 142 пакета и включает практически все лучшие программы и даже транслятор с языка ADA, предназначенный для выполнения сложных расчетов в военных областях [12]. Цены на копии популярных программ на местных рынках баснословно низкие. Так, пакет «Frameword II» компании Ashiton—Tate стоит 270 гонконгских долларов, что равно примерно 35 долл. США, по сравнению с ценой подлинника на американском рынке в 700 долл.

По официальным оценкам, промышленность США ежегодно теряет от 8 до 20 млрд. долл. из-за того, что новые индустриальные, развивающиеся и некоторые развитые страны не обеспечивают достаточную защиту американских патентов и авторских прав, из них около 2 млрд. долл. приходится на потери, вызванные нарушением авторских прав. Около 800 млн. долл. в год теряет американская отрасль ПО, что больше потерь фирм индустрии звукозаписи и книгоиздательства [13, 14].

Федеральное правительство США в последнее время уделяет большое внимание защите прав ИС, включив ее в круг важнейших вопросов внешнеэкономической политики США. Еще в 1983 г. президент Рейган заявил, что одной из главных задач своей администрации он считает обеспечение технологического лидерства США не только на период своего правления, но и в XXI веке. Активная деятельность различных организаций, таких как Международный союз интеллектуальной собственности, Корпорация владельцев интеллектуальной собственности, имеющих сильное лобби в конгрессе, привела к тому, что администрация и члены конгресса стали рассматривать защиту прав ИС как критический фактор будущего положения США на международной экономической арене.

В январе 1987 г. в речи, посвященной вопросам внешней торговли, президент Рейган сказал: «Когда правительство допускает нарушение авторских прав или копирование американских товаров — это обкрадывает наше будущее» [13].

В настоящее время регулированием международных вопросов защиты прав ИС занимаются несколько международных организаций: Международная организация интеллектуальной собственности ООН (МОИС), Конференция ООН по торговле и развитию (ЮНКТАД), Генеральное соглашение по торговле и тарифам (ГАТТ) и Международная торговая комиссия США. Существует ряд международных соглашений об охране прав



ИС: Бернская конвенция о защите произведений литературы и искусства, Всеобщая конвенция об авторских правах, Парижская конвенция. Круг участников этих соглашений очень широк. Парижскую конвенцию подписали 95 стран. Однако такой широкий состав форумов, посвященных защите прав интеллектуальной собственности, снижает их эффективность. В промышленно развитых странах законы об авторских правах обеспечивают примерно одинаковый уровень защиты и не вызывают разногласий у правительств этих стран. Развивающиеся страны не только не стремятся усилить очень слабую защиту ИС в своих странах, но и противостоят давлению развитых стран в этой области. Особенно активно в этом направлении действует Группа 77, выступающая за установление нового экономического порядка. Такая политика развивающихся и новых индустриальных стран имеет глубокие основания. Ее цель — облегчение доступа к новой технологии национальным фирмам. При несбалансированных потоках высокотехнологичных товаров частная собственность на знания ограничивает рост экономики стран третьего мира, создает дополнительные препятствия для их развития. Представители этих стран ссылаются на то, что уровень защиты ИС должен соответствовать уровню развития экономики. Преследуют они также и более близкие цели: снижение розничных цен и уменьшение бюджетных расходов на государственные закупки.

Трудности, связанные с урегулированием вопроса о несбалансированности национальных систем защиты прав интеллектуальной собственности, вынуждают США обращаться к ГАТТ и двусторонним соглашениям. Здесь США имеют возможность действовать традиционными для них методами силового давления. Так, по принятому в 1984 г. конгрессом США Закону о торговле и тарифах решение об обновлении соглашений по Генеральной системе льготных таможенных пошлин, особенно о беспощинном импорте из стран Карибского бассейна, будет зависеть от уровня обеспечения торговыми партнерами США защиты американских патентов и авторских прав [14]. Действия администрации США по укреплению защиты патентов и авторских прав на американские товары рассматриваются как еще одна попытка усиления протекционизма новыми методами.

В рамках двусторонних переговоров США уже удалось добиться некоторых успехов. Более сильную защиту прав интеллектуальной собственности ввели правительства Южной Кореи и Тайваня. В конце 1987 г. в Китае начал действовать первый в этой стране закон об авторских правах на кино- и видеоленты, звукозаписи. Правительство Китая сообщило о намерении включить ПО в закон об авторских правах в 1988 г.

Однако достижение успеха в переговорах еще не означает успеха на деле. Так, в 1985 г. министерство внешней торговли и промышленности Японии сообщило об отмене закона о принудительном лицензировании ПО и введении общепринятой защиты программ через авторские права. Это решение было принято под давлением США и вызвало сильную отрицательную реакцию японских фирм [16]. Однако до сих пор в Японии не предприняли никаких реальных мер для выполнения своего обещания. Японское правительство создает оптимальный режим для национальных фирм, облегчая им доступ к американским программам, разрешая перекодирование программ и пренебрегая проникновением иностранных фирм на внутренний рынок Японии. Сейчас, по мнению экспертов из Управления оценки технологий США, японская отрасль ПО отстает от американской примерно на 20 лет [14]. Однако, используя проверенную стратегию ограждения национальных компаний от

иностранных конкурентов на внутреннем рынке и поощряя использование их ноу-хау, в перспективе отрасль ПО в Японии может превратиться в такого же опасного конкурента для компаний по программному обеспечению США, каким сейчас является японская отрасль по производству ЭВМ.

Ситуация, сложившаяся во второй половине 80-х годов, показывает необходимость привлечения экспертов и специалистов-техников к разработке законов, регулирующих защиту ИС в областях науки и производства.

Будущая конкурентоспособность отрасли ПО США во многом будет зависеть от эффективности защиты прав интеллектуальной собственности на программы.

Телефон 202-45-18, Москва

#### ЛИТЕРАТУРА

1. Constitution of the United States. National and States.—Dobbs Ferry, New York: Oceania Publications, 1974.—P. 3.
2. Schachter E. R. Software Protection in the Tropics of Legal Morass/Datamation, June.—1987.—Vol. 33.—N 11.—P. 49—58.
3. Myers E. D. Should Software be Copyrighted//Datamation, March.—1978.—Vol. 24.—N 3.—P. 125—137.
4. Громов Г. Р. Национальные информационные ресурсы: проблемы промышленной эксплуатации.—М.: Наука, 1985.—С. 83.
5. Lewenstein B. V. The Ethics of Software Piracy//PC Magazine, April.—1985.—Vol. 4.—N 9.—P. 181—187.
6. Boody Traps that Catch Deadbeats//Business Week, May, 31.—1982.—P. 64.
7. Preventing Software Pirates from Plundering Advice//Business Week, September, 14.—1987.—P. 62.
8. Bulkeley W. M. Courts Expand the Copyright Protection of Software, but Many Questions Remain//Wall Street Journal, November.—1986.—Vol. CCVIII.—N 99.—P. 35.
9. Tell L. I. Software Copyright: keep out the pirates—but let innovators in//Business Week, August, 31.—1987.—P. 21.
10. Gulert F. V. Patents: Potent Weapon for High-Tech Companies//Dun's Business Month, July.—1986.—Vol. 128.—N 1.—P. 32—34.
11. Brandt R. For Chipmakers, a landmark ruling gets wiped off the map//Business Week, January, 11.—1988.—P. 27—28.
12. Hebditch D. Pirates Paradise//Datamation, September.—1986.—Vol. 32.—N 17.—P. 71—72.
13. Stoke B. Intellectual Piracy Captures the Attention of the President and Congress//National Journal, February.—1986.—Vol. 18.—N 8.—P. 442—444.
14. International Competition in Services.—Wash.: U.S. Government Printing Office, 1987.—P. 318—321, 165—175.
15. China may build a wall against copyright cheapening//Business Week, September, 14.—1987.—P. 62.
16. Yoder S. K. Japan Scuttles Patent Proposal for Software//Wall Street Journal, March.—1985.—Vol. CCY.—N 54.—P. 33.

Статья поступила 26.01.88

## ОБМЕН ОПЫТОМ РАЗРАБОТКИ АППАРАТНО-ПРОГРАММНЫХ СРЕДСТВ МП-ТЕХНИКИ

УДК 681.3.86

Ю. И. Жуков, П. Н. Кармалин, А. В. Боровиков

### ПРОГРАММИРОВАНИЕ ПЗУ В МОДУЛЕ 15УЗПП-16К×16 В КОНСТРУКТИВЕ «ЭЛЕКТРОНИКА 60» ПОД ОС РАФОС

После разработки, отладки и тестирования программного обеспечения (ПО) встраиваемых микропроцессорных устройств на базе микроЭВМ «Электроника 60» необходимо ПО разместить в соответствующий модуль ПЗУ (серии К573). Применяемые для этой цели универсальные программаторы [1, 2] не всегда доступны разработчику по ряду объективных причин, поэтому авторы использовали аппаратный комплекс, целиком состоящий из стандартных оборудования и ПО. Исключение составила лишь собственно программа для записи в ПЗУ.

Программирование микросхем производится на стандартном модуле 15УЗПП-16К×16-1, устанавливаемом в конструктиве микроЭВМ перед ее включением. Наиболее удачное размещение в поле памяти — когда модуль занимает банки 1...4. При этом банки 0, 5, 6 после загрузки будут заняты операционной системой и программой записи (банк 0).

Такой вариант компоновки комплекса предназначен в основном для программирования ПЗУ, так как для многих системных программ свободная область ОЗУ (банк 0) недостаточна. Размещение целевого ПО в банках 1...4 удобно и тем, что позволяет производить отладку и тестирование в тех же адресах ОЗУ под управлением ОС. После завершения отладки пользователю остается заменить модуль ОЗУ (например, модуль ПЗ), занимающий банки 1...4, на модуль ПЗУ, перезагрузить систему и переписать в ПЗУ окончательный вариант ПО, находящийся на каком-либо носителе с файловой структурой в формате отображения памяти (например, DX1:PROG.SAU).

Создавая резидентное ПО ЭВМ с системой команд LSI-11, следует учитывать при работе с ПЗУ, что ряд команд выполняется в цикле считывание — модификация — запись. Если ПЗУ не позволяет выполнять такой цикл обращения, команды LSI-11 использовать нельзя.

При невозможности оперативной смены блоков ОЗУ на модуль ПЗУ и перезагрузки системы можно работать с одним банком памяти модуля ПЗУ. В этом случае его удобно разместить над РМОМ и драйвером SY: (например, в банк 6). Чтобы обеспечить загрузку ОС ниже банка 6, необходимо запретить выработку сигнала СИП модуля ПЗУ на момент загрузки ОС. Это можно сделать, разорвав цепь сигнала СИП модуля. Целевое ПО при этом загружается блоками по 4К. Для загрузки очередного блока запрограммированные микросхемы заменяются на «чистые».

В заключение приведем текст программы, поддерживающий режим программирования первого варианта комплекса (банки 1...4). Программа позволяет записывать в ПЗУ программу пользователя с носителя, сравнивать ее с эталонной программой, размещаемой на носителе. Записывать и сравнивать банки 1...4 можно в любом порядке. При сравнении записанной программы и программы на носителе вычисляется контрольная сумма, которая распечатывается по окончании сравне-

```

PROGRAM ROM;
(*****
PASCAL/RAFOS V.01.00.01
Программа записи в модуль ппзз типа 15УЗПП-16К×16
с проверкой и вычислением контрольных сумм блоков
с носителя с файловой структурой программы в формате
отображения памяти
*****
LABEL 1,2,3,4;
CONST DNMEM=200000;
TYPE UNSIGN=1..65535;
VAR ARR=ARRAY[0..16383] OF UNSIGN;
VAR INF:TEXT;
HLIM,ADRMEM:UNSIGN;
AR ORIGIN DNMEM;ARR;
FILENAME:ARRAY[1..20] OF CHAR;
CH:CHAR;
BUF:ARRAY[0..1] OF CHAR;
BUFMEM,C:J,NUMMEM,I,K:INTEGER;
NUMBER:ARRAY[1..4] OF BOOLEAN;
CHSUM:ARRAY[1..4] OF CHAR;
(*Проверка блока 0*)
PROCEDURE PULL;
BEGIN
  RESET(INF,FILENAME); ADRMEM:=0B;
  REPEAT
    BUF[0]:=INF< GET(INF); BUF[1]:=INF< GET(INF);
    ADRMEM:=ADRMEM+2;
  UNTIL ADRMEM=DNMEM;
END;
(*Загрузка 50M*)
PROCEDURE TIM50MS;
VAR C:INTEGER;
BEGIN
  FOR C:=1 TO 2000 DO;
  END;
(*Переключение терминала в русский регистр*)
PROCEDURE RUS;
BEGIN
  WRITE(CHR(142));
END;
(*Переключение терминала в латинский регистр*)
PROCEDURE ENGL;
BEGIN
  WRITE(CHR(143));
END;
(*Начало основной программы*)
BEGIN
  ADRMEM:=0;
RUS; WRITELN('Запись/проверка ПЗУ в банки 1..4');
WRITE(' Имя файла: '); ENGL; READLN(FILENAME);
RESET(INF,FILENAME);
(*Считывание верхнего адреса программы*)
WHILE ADRMEM<X28 DO
  BEGIN
    BUF[0]:=INF< GET(INF); BUF[1]:=INF< GET(INF);
    HLIM:=ORD(BUF[0])AND 377B+ORD(BUF[1])AND 377B+256;
    ADRMEM:=ADRMEM+2;
  END;
RUS; WRITE('Верхний адрес ='); ENGL; Writeln(HLIM:8);
RUS; WRITE('Вы проверяете?'); ENGL; Writeln('Y/N '); CH:='';
REPEAT READ(CH) UNTIL (CH='Y')OR(CH='N');
IF CH='Y' THEN
  BEGIN
    FOR K:=1 TO 4 DO
      BEGIN
        RUS; WRITE('Проверка банка '); ENGL; WRITE(K:3,' V/N ');
        CH:='';
        REPEAT READ(CH) UNTIL (CH='Y') OR (CH='N');
        IF CH='Y' THEN NUMBER[K]:=TRUE ELSE NUMBER[K]:=FALSE;
      END;
    GOTC 3;
  END;
(*Считывание номера банка для записи*)
REPEAT
  FOR K:=1 TO 4 DO
    BEGIN
      RUS; WRITE('Запись в банк'); ENGL; WRITE(K:3,' V/N ');

```

## УСТРОЙСТВА ПРИОРИТЕТА В БАЗИСЕ ПЛМ — КОМПАКТНОСТЬ И БЫСТРОДЕЙСТВИЕ

Устройства приоритета (УП) применяются в микропроцессорной (МП) технике для разрешения конфликтов, возникающих при одновременном использовании несколькими устройствами общего ресурса. Они необходимы для организации доступа периферийных устройств к процессору через систему прерываний, к общей памяти по каналу прямого доступа, при управлении многопортовой памятью и т. д. Этим устройствам посвящен большой список публикаций (например, [1, 2]), включающий в себя несколько сотен изобретенных классов МКИ<sup>4</sup> GO6F 9/46 и 7/06. Однако известные УП (на базе ИС малой и средней интеграции) требуют больших затрат оборудования, не обеспечивая при этом высокого быстродействия, — в МП системах они часто становятся «узким местом».

Авторам известны БИС приоритетных прерываний, в том числе БИС К589ИК14 и КР580ВН59. Последняя позволяет организовать обслуживание прерываний по линейной или циклической дисциплине обслуживания восьми источников (с возможностью расширения до 64), однако она работает под управлением микропроцессора КР580ИК80А [3]. Эта ИС требует программной организации специального диалога со стороны процессора. Ей необходима фаза инициализации (команды КИ1...КИ3) и фаза управления (команды КУ1...КУ3). Выходная информация имеет специальное представление: в виде команды CALL с 16-разрядным адресом подпрограммы обслуживания прерывания или же требуется программный опрос.

Данная микросхема — специализированная, т. е. не может функционировать автономно.

Основное преимущество предложенных авторами УП — их автономность. Они не требуют программной поддержки (настройки и опроса) и могут работать самостоятельно. В этом смысле они являются «открытыми» и могут быть использованы как элементы МП систем любой организации и с любым МП. Их также можно использовать в объектах программно недоступных процессору или без процессорного элемента, например в системах арбитража канала доступа или многопортовых блока памяти. Отметим, что использование микросхемы КР580ВН59 для решения задач, поставленных в статье, влечет значительные потери по быстродействию (программный обмен) при больших затратах по оборудованию.

Высокий уровень разработки УП может быть достигнут, если в качестве базового элемента устройства использовать программируемые логические матрицы (ПЛМ) [4, 5]. В статье предлагаются три типа УП, реализующих линейную, циклическую и динамическую дисциплины обслуживания (ДО), выполненные на ПЛМ К556РТ1. Эти микросхемы имеют выходы с открытым коллектором и рассчитаны на организацию монтажной логики И. Нагрузочные резисторы на рисунках не показаны. Таблицы программирования выполнены в соответствии с [4].

Входная информация для УП — это сигналы запросов от источников, требующих доступа к общему ресурсу. УП дает разрешение на пользование общим ресурсом одному из источников и отказывает остальным. Выбор осуществляется на основании характерной для данного устройства дисциплины обслуживания.

При линейной дисциплине обслуживания разрешение дается источнику запроса с наименьшим номером. Пример устройства, обслуживающего 45 источников, приведен на рис. 1, 2. Запросы Р разбиваются на группы

```

CH:=' ':
REPEAT READ(CH) UNTIL (CH='Y') OR (CH='N');
IF CH='Y' THEN NUMBANKI:=TRUE ELSE NUMBANKI:=FALSE;
END;
RUS: WRITE('Запись в банки '); ENCL;
FOR K:=1 TO 4 DO
  BEGIN
    IF NUMBANKI THEN WRITE(K:3); CH:=' ';
  END;
RUS: WRITE('*9Кл. 268жж*'); ENCL; WRITE('Y/N ');
REPEAT READ(CH) UNTIL (CH='Y') OR (CH='N');
UNTIL CH='Y';
PULL:
(*Запись в ПЗУ*)
FOR K:=1 TO 4 DO
  BEGIN
    RUS: IF NUMBANKI THEN WRITELN('Запись в банк ',K:3);
    REPEAT
      IF ADRMEM=HLIM THEN GOTO 1;
      BUF[0]:=INF^; GET(INF); BUF[1]:=INF^; GET(INF);
      IF NUMBANKI THEN
        BEGIN
          C:=ORD(BUF[0])AND 377B+ORD(BUF[1])AND 377B*256;
          A:=C(ADRMEM-DNMEM)DIV 21:=C;
          TMSMS;
        END;
      ADRMEM:=ADRMEM+2;
      UNTIL ADRMEM=DNMEM*K-2+DNMEM;
    END;
  1: WRITELN('Запись окончена. выкл. 26В');
  (*Проверка программы в ПЗУ*)
  3: RUS: WRITE('Вы проверяете банки ');
  FOR K:=1 TO 4 DO
    BEGIN
      CHSUM[K]:=CHR(0); IF NUMBANKI THEN WRITE(K:3); ENCL;
      WRITELN: ENCL;
      WRITE('Y/N '); CH:=' ';
    END;
  REPEAT READ(CH) UNTIL (CH='Y') OR (CH='N');
  IF CH='Y' THEN
    BEGIN
      PULL:
      FOR K:=1 TO 4 DO
        BEGIN
          RUS:
          IF NUMBANKI THEN
            BEGIN
              WRITELN('Проверка банка ',K:3);
              REPEAT
                BUF[0]:=INF^; GET(INF); BUF[1]:=INF^; GET(INF);
                IF NUMBANKI THEN
                  BEGIN
                    (*Проверка банка*)
                    C:=ORD(BUF[0])AND 377B+ORD(BUF[1])AND 377B*256;
                    IF ADRMEM=HLIM THEN
                      BEGIN
                        WRITE('fail '); ENCL; WRITE(FILENAME);
                        RUS: WRITELN('= файл в ПЗУ='); GOTO 4;
                      END;
                    IF A:=C(ADRMEM-DNMEM)DIV 21<X THEN
                      BEGIN
                        WRITELN('Ошибка сравнения в банке ',K:3);
                        WRITE('Адрес '); ENCL; WRITE(ADRMEM-8);
                        RUS: WRITE(' ПЗУ='); ENCL;
                        WRITE(A:=C(ADRMEM-DNMEM)DIV 21:-8); RUS:
                        WRITE(' файл='); ENCL; WRITELN(C:-8); GOTO 2;
                      END;
                      CHSUM[K]:=CHSUM[K]+CHR(C AND 377B);
                    END;
                    ADRMEM:=ADRMEM+2;
                    UNTIL ADRMEM=DNMEM*K-2+DNMEM;
                  END;
                END;
              4: ENCL; FOR K:=1 TO 4 DO
                IF NUMBANKI THEN
                  WRITE('CHSUM',K:2, ': '=ORD(CHSUM[K]) AND 377B:-5, ' ');
                ENCL; WRITELN; (*Конец сравнения*)
              END.

```

ния. Если запись не верна, печатается адрес ячейки, ее содержимое и содержимое эталона. Программа работает в интерактивном режиме и сопровождается элементами самодокументирования, поэтому проста в использовании.

Телефон 183-50-97, Ленинград

### ЛИТЕРАТУРА

1. Дианов А. П., Щелкунов Н. Н. — Методика программирования микросхем ПЗУ // Микропроцессорные средства и системы. — 1985. — № 3. — С. 80—83.
2. Лукьянов Д. А. — Схемотехника универсальных программаторов ПЗУ // Микропроцессорные средства и системы. — 1985. — № 3. — С. 84—88.

Статья поступила 02.04.87



Конъюнкция		Уровень активности 0 0 0 0 1 1 1 1	
Входная переменная		Выходная функция	
A15		B07	
A14		B06	
A13		B05	
A12		B04	
A11		B03	
A10		B02	
A09		B01	
A08		B00	
A07			
A06			
A05			
A04			
A03			
A02			
A01			
A00			

Рис. 4. Таблица программирования ПЛМ DD1 для устройства приоритета с циклической дисциплиной обслуживания

Конъюнкция		Уровень активности 0 0 0 0 1 1 1 1	
Входная переменная		Выходная функция	
A15		B07	
A14		B06	
A13		B05	
A12		B04	
A11		B03	
A10		B02	
A09		B01	
A08		B00	
A07			
A06			
A05			
A04			
A03			
A02			
A01			
A00			

Рис. 5. Таблица программирования ПЛМ DD2 для устройства приоритета с циклической дисциплиной обслуживания

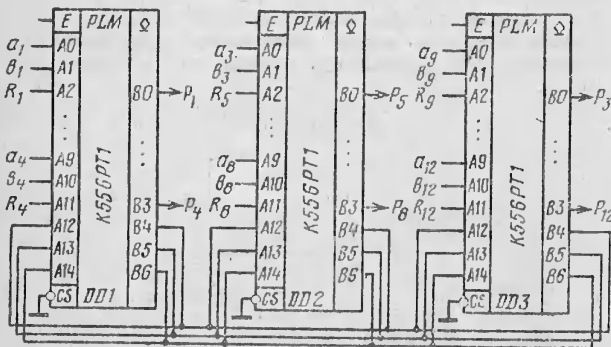


Рис. 6. Функциональная схема устройства приоритета с динамической дисциплиной обслуживания

са из поступивших на ее входы. Если значение этого приоритета равно 11 (10, 01, 00), то на выходах B4, B5, B6 вырабатывается код 000 (100, 110, 111). Поразрядная конъюнкция нескольких таких кодов равна максимальному из них. Так как соответствующие выходы B4, B5, B6 всех ПЛМ соединены между собой, то согласно принципу монтажной логики И на входы A12,

Конъюнкция		Уровень активности 1 0 0 0 1 1 1 1	
Входная переменная		Выходная функция	
A15		B07	
A14		B06	
A13		B05	
A12		B04	
A11		B03	
A10		B02	
A09		B01	
A08		B00	
A07			
A06			
A05			
A04			
A03			
A02			
A01			
A00			

Рис. 7. Таблица программирования ПЛМ для устройства приоритета с динамической дисциплиной обслуживания

1. Гойхман П. А., Добровинский М. Е. и др. Устройство приоритетного прерывания для микроЭВМ «Электроника 60» // Микропроцессорные средства и системы. — 1986. — № 1. — С. 73—75.
2. Bentley O. W. FPLA Arbiter Concept Adapts to Application Needs // Computer Design. — 1981. — V. 20. — N 6. — P. 149—155.
3. Алексенко А. Г., Галицын А. А., Иванников А. Д. Проектирование радиоэлектронной аппаратуры на микропроцессорах. — М.: Радио и связь, 1984.
4. ОСТ 11 340.915—82.
5. Щелкунов Н. Н., Дианов А. П. Процедуры программирования логических матриц // Микропроцессорные средства и системы. — 1986. — № 2. — С. 71—76.

Телефон 34-97-20, Минск

Статья поступила 06.01.88

A13, A14 поступает код, характеризующий максимальный приоритет источника, имеющего запрос. Сравнивая его значение с приоритетами своих запросов, каждая ПЛМ вырабатывает разрешения на выходах В0...В3, которые соответствуют запросам с максимальным приоритетом.

При наличии источников равного приоритета на выходах УП с динамической дисциплиной обслуживания может возникнуть несколько сигналов разрешения. Для однозначного выбора выходы Р<sub>1</sub>...Р<sub>12</sub> следует подключить к УП с линейной или циклической дисциплиной обслуживания.

Устройство можно легко расширить для кодов приоритетов любой разрядности. Досаточно соединить последовательно нужное число УП (рис. 6) так, чтобы выходы разрешений Р<sub>1</sub>...Р<sub>12</sub> предыдущего являлись входами запросов Р<sub>1</sub>...Р<sub>12</sub> последующего. Каждое входное УП увеличивает разрядность кодов приоритетов на два разряда.

По сравнению с известными устройствами УП в базе ПЛМ имеют в 5...7 раз меньший объем оборудования, превышая их по быстродействию в 2...5 раз.

## УСТРОЙСТВА СВЯЗИ МИКРОЭВМ С ОБЪЕКТАМИ

УДК 681.324

А. Е. Михуткин

### КОНТРОЛЛЕР КР580ВН59 В УСТРОЙСТВАХ ПРИОРИТЕТНОГО ПРЕРЫВАНИЯ МИКРОЭВМ «ЭЛЕКТРОНИКА 60»

Разработка устройств приоритетного прерывания (УПП) для микроЭВМ «Электроника 60» на элементах малой степени интеграции [1] с использованием блоков приоритетного прерывания К589ИК14 [2] сложна и имеет определенные ограничения. Более перспективным для построения УПП является программируемый контроллер БИС КР580ВН59 [3] с широкими функциональными возможностями, способный реализовать сложные стратегии прерываний. Контроллер позволяет построить 8-уровневую векторную систему прерываний с возможностью маскирования и динамического изменения дисциплины обслуживания. Каскадированием БИС КР580ВН59 число уровней прерывания можно увеличить до 64. Контроллер используется для организации обмена информацией в режимах прерываний и программно-управляемого обмена. В первом случае БИС на приоритетной основе формирует запрос на прерывание для процессора, код команды CALL микропроцессора КР580ИК80 и двухбайтовый адрес подпрограммы обслуживания, во втором — процессор, считывая информацию с внутренних регистров контроллера, определяет устройство с наивысшим приоритетом, готовое к обмену, или состояние входов запросов прерывания (ЗП).

Контроллер позволяет организовать простой приоритетный режим и режим циклического приоритета обслуживания прерываний. В простом приоритетном режиме всем восьми входам запросов прерывания (ЗП0...ЗП7) присваиваются фиксированные приоритеты: высший приоритет входу ЗП0, низший — ЗП7. В режиме циклического приоритета после окончания обслуживания любого устройства приоритеты входов циклически изменяются таким образом, что входу, обслуженному последним, присваивается низший приоритет, а следующему за ним по порядку — высший. Кроме того, при работе в режиме циклического приоритета низший приоритет может быть программно присвоен любому входу запроса.

Запросы прерывания от внешних устройств (ВУ) подаются на входы ЗП0...ЗП7 контроллера (рис. 1) и

запоминаются в регистре запросов. Регистр состояния БИС, каждый разряд которого соответствует одному из входов, содержит все запросы прерывания, обслуживаемые в текущий момент. Регистр маски содержит единицы в разрядах, соответствующих маскируемым в настоящий момент входам запросов. Каждый последующий запрос на прерывание воспринимается контроллером только после выполнения подпрограммы обслуживания текущего запроса по данному входу и сброса соответствующего разряда регистра состояния, что осуществляется программным способом. Установка в единицу того или иного разряда регистра маски блокирует передачу запроса на прерывание. Однако, если через некоторое время после подачи запроса на прерывание маска снимается программой, запрос будет обслужен. Содержимое регистра маски меняться в любой момент времени.

В контроллере КР580ВН59 можно организовать режим специального маскирования обслуживаемого прерывания для того, чтобы разрешить обслуживание входов запросов с более низким приоритетом, чем маскируемый вход, во время выполнения подпрограммы обслуживания запроса прерывания по маскируемому входу.

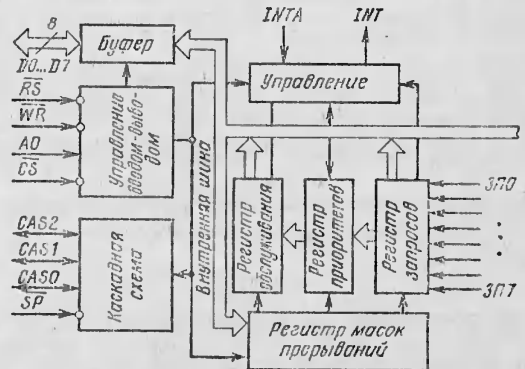


Рис. 1. Функциональная схема контроллера КР580ВН59

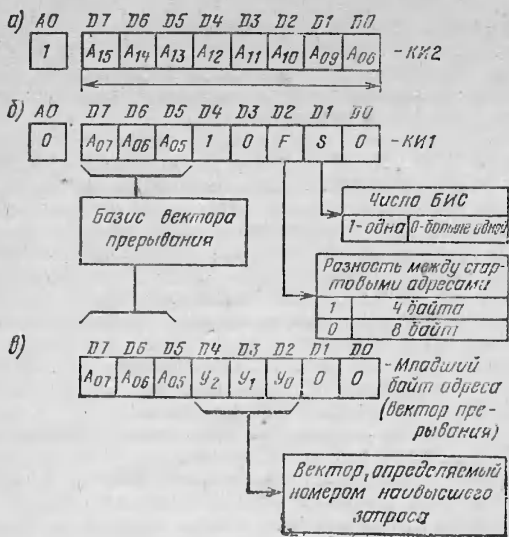


Рис. 2. Структура команд инициализации КИ2 (а), КИ1 (б) и младшего байта адреса (в)

Программирование контроллера осуществляется командами инициализации (КИ1 и КИ2) и управления режимом (КУ1, КУ2, КУ3) [3]. Команды инициализации задают стартовые адреса подпрограмм обслуживания прерываний (рис. 2, а, б), расстояния между соседними стартовыми адресами и указывают на наличие других контроллеров в системе. Команды управления режимом служат для оперативного изменения режимов

обслуживания прерываний и могут подаваться в любое время в процессе работы контроллера.

Последовательность обработки сигналов прерывания в БИС выглядит следующим образом. В режиме обслуживания прерываний при одновременно поступивших запросах от ВУ контроллер выявляет запрос с высшим приоритетом (маски учитываются) и выдает сигнал с выхода INT центральному процессору. При поступлении от процессора сигнала INTA БИС выдает на магистраль данных D0...D7 код команды CALL. Сигнал INT не снимается. В ответ на удерживаемый после первого сигнала INTA уровень запроса прерывания INT процессор вырабатывает еще два сигнала INTA, позволяющие контроллеру послать последовательно в магистраль данных D0...D7 адрес подпрограммы обслуживания прерывания. Адрес подпрограммы состоит из заранее запрограммированного базиса и вектора, двоично-десятичный код которого соответствует порядковому номеру (0...7) обслуживаемого запроса. По второму сигналу выдается младший байт адреса (рис. 2, в), по третьему — старший байт адреса, соответствующий КИ2.

Режим прерываний для микроЭВМ «Электроника 60» заключается в формировании ВУ (при его готовности) сигнала «требование прерывания» (К ТПР) и 8-разрядного вектора прерывания, закрепленного за ВУ, в ответ на один сигнал процессора «подтверждение прерывания» (К ППР). Вектор прерывания представляет собой младший байт адреса ячейки памяти, в которой записан стартовый адрес подпрограммы обслуживания прерывания. Старший байт адреса в процессоре автоматически устанавливается в нуль. Следующая ячейка памяти предназначена для хранения слова состояния процессора при выполнении подпрограммы обслуживания [4].

Младший байт адреса, формируемый БИС, так же как и вектор прерывания микроЭВМ, представляет со-

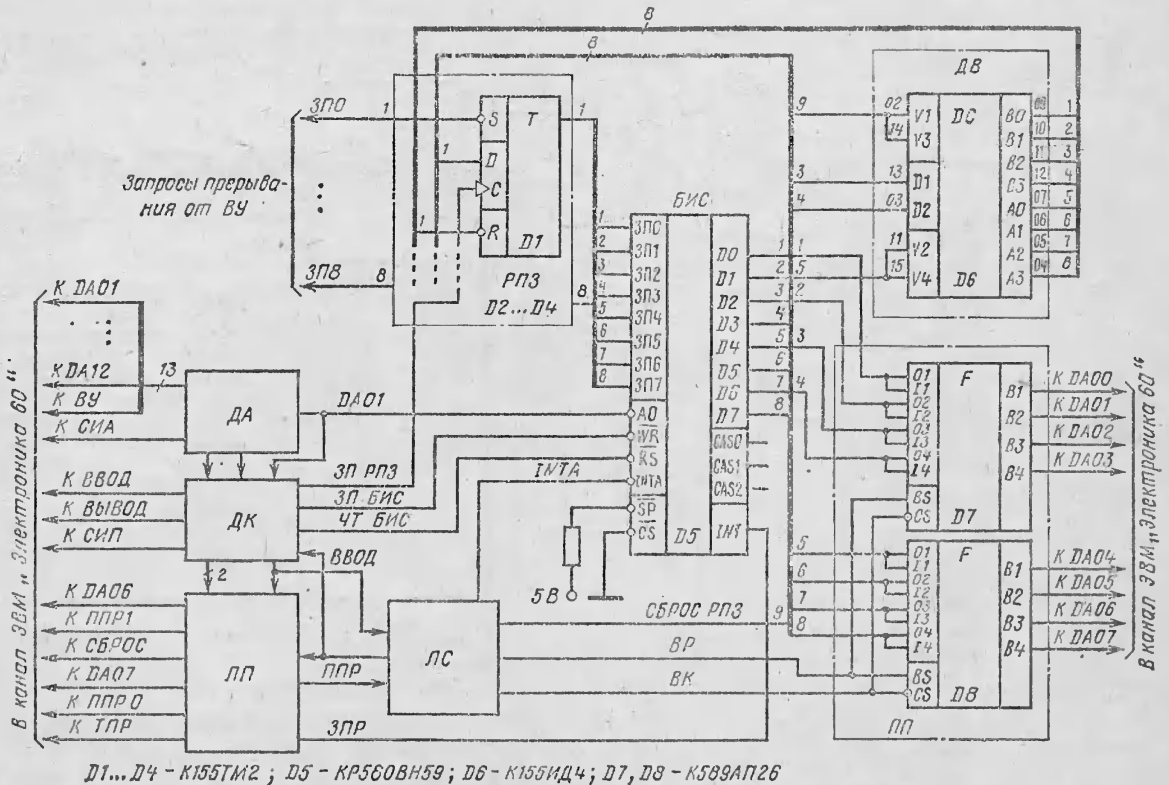


Рис. 3. Функциональная схема УПП-8

бой 8-разрядный код, содержимое которого определяется базисом (разряды D5, D6, D7) и номером входа ЗП (разряды D2, D3, D4). Смежные адреса при инициализации могут быть заданы кратными четырем, что необходимо и для векторов прерывания микроЭВМ «Электроника 60». Это соответствие и дополнительное схемное обрамление БИС позволяют обеспечить согласованные протоколы обмена контроллера КР580ВН59 и микроЭВМ «Электроника 60» и реализовать на его основе УПП.

Устройство приоритетного прерывания на 8 уровней (УПП-8) содержит дешифратор адреса (ДА), дешифратор команд (ДК), логику прерывания (ЛП) процессора, логику синхронизации (ЛС), БИС КР580ВН59, приемопередатчики (ПП), дешифратор вектора (ДВ), регистр памяти запросов (РПЗ (рис. 3).

По отношению к процессору УПП-8 является внешним устройством, имеющим четыре программно доступных регистра: регистр состояния (РС), регистры данных РД0, РД1, РПЗ, обмен информацией с которыми осуществляется по стандартным циклам ВВОД и ВЫВОД [4]. Дешифратор адреса (рис. 4) определяет адреса регистров в адресном пространстве микроЭВМ и разрешает выполнение внутренних операций при обращении к одному из них. Одновременно дешифратор сигналом ДА01 выбирает один из внутренних регистров контроллера КР580ВН59 (РД0 — А0=0; РД1 — А0=1). Дешифратор команд в зависимости от типа цикла обращения вырабатывает сигналы ЗП БИС и ЧТ БИС для контроллера, а также управляет записью информации в РС и РПЗ. Данные с общих шин К ДА00... К ДА07 передаются в контроллер и РПЗ через шинные формователи ПП (К589АП26).

Работа с УПП-8 начинается с программной инициализации, осуществляемой путем записи КИ1 и КИ2 в регистры РД0 и РД1 соответственно. В команде КИ1 задается базис (D5, D6, D7) для вектора прерывания,

указание о единственности контроллера (S=1) и разность между стартовыми адресами 4 (F=1). Содержимое команды КИ2 для УПП-8 безразлично. После проведения инициализации устройство готово к работе в простом приоритетном режиме.

При появлении запроса прерывания по одному из входов ЗП0...ЗП7 (см. рис. 3) соответствующий разряд РПЗ устанавливается в единичное состояние, тем самым вызывая появление запроса на соответствующем входе БИС. При разрешенном прерывании по данному входу (маска не установлена) контроллер выставляет сигнал «запрос установления» (ЗПР), поступающий в логику прерывания (ЛП) процессора (рис. 5), реализованную согласно рекомендациям предприятия-разработчика [4], и отвечает за разрешение или запрет формирования сигнала К ТПР. Организация РС в ЛП идентична организации РС, принятой для ЭВМ «Электроника 60». Управление РС программно.

При наличии разрешения прерывания процессора от УПП-8 (шестой разряд РС установлен в 1) запрещается сквозное прохождение сигнала К ППР, а появление сигнала ЗПР вызывает формирование сигнала К ТПР. Последний прерывает выполнение текущей программы, и процессор переходит к циклу реакции на внешнее прерывание, т. е. генерирует сигналы К ВВОД и К ППР, требуя от ВУ, вызвавшего прерывание, выставить на шинах К ДА00... К ДА07 своей вектор.

Сигнал К ППР1 через ЛП поступает в ЛС (рис. 5), которая генерирует последовательно три импульса INTA (рис. 6). По второму импульсу INTA формируются два сигнала ВР и ВК, переключающие ПП на передачу данных с шин D0...D07 на общие шины К ДА00... К ДА07, а ДК формирует сигнал К СИП, информирующий процессор о наличии вектора прерывания на шинах данных. В остальные периоды времени ПП блокирует связь между шинами процессора и шинами данных БИС. При обмене информацией с ре-

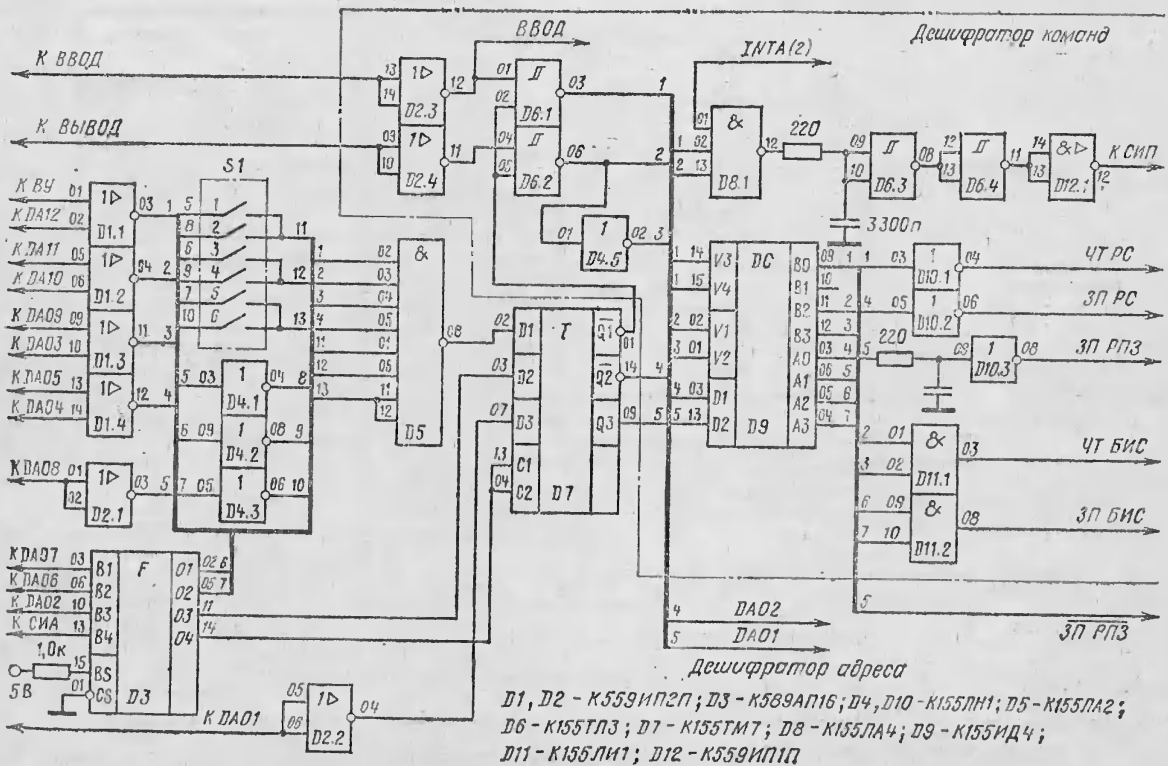
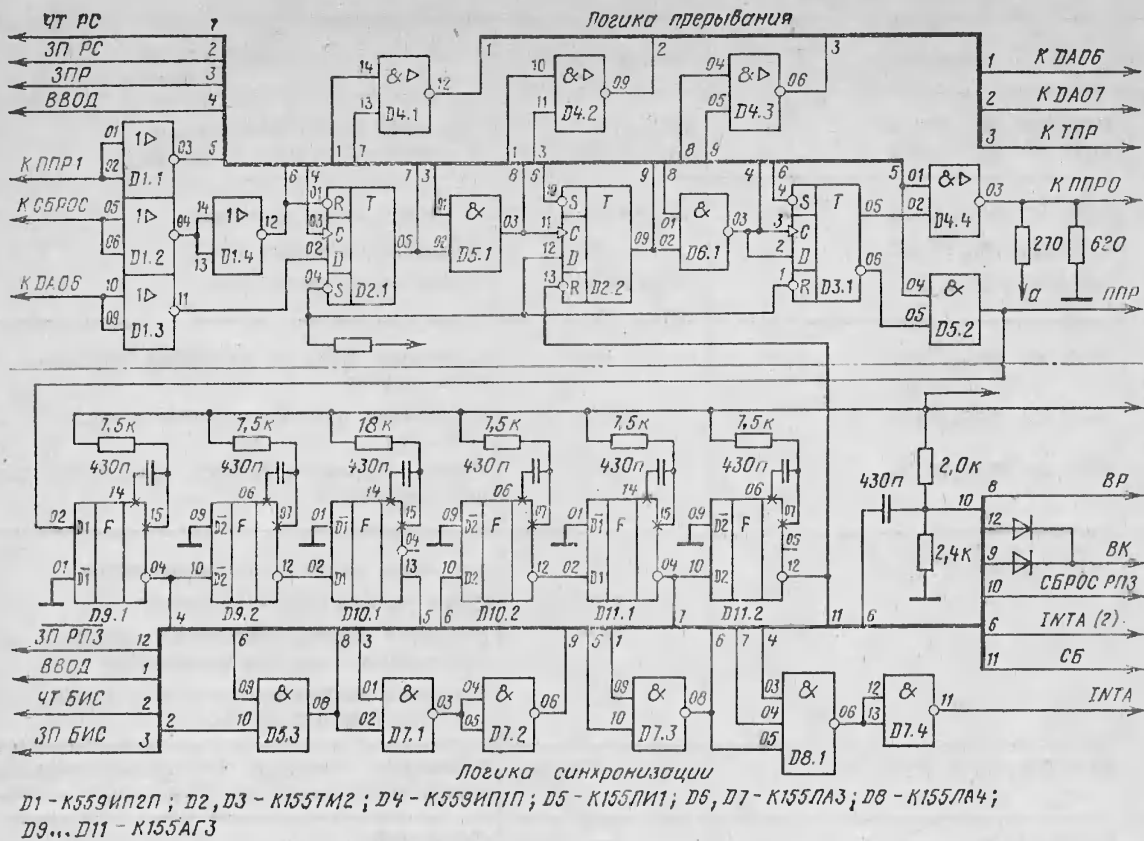


Рис. 4. Схема дешифратора адреса (ДА) и дешифратора команд (ДК) УПП-8





Д1 - К1559ИП2П; Д2, Д3 - К155ТМ2; Д4 - К1559ИП1П; Д5 - К155ЛН1; Д6, Д7 - К155ЛАЗ; Д8 - К155ЛАЗ; Д9... Д11 - К155АГ3

Рис. 5. Схема логики прерывания (ЛП) и логики синхронизации (ЛС) УПП-8

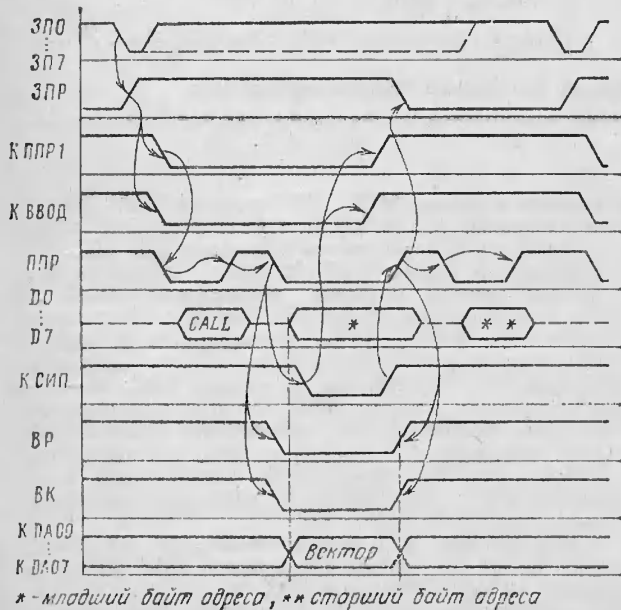


Рис. 6. Временные диаграммы режима прерывания

гистрами УПП-8 ЛС обеспечивает соответствующее переключение ПП.

Логика работы БИС КР580ВН59 требует снятия сигнала ЗПР после его обработки, иначе блокируется обработка других сигналов ЗПР независимо от их приоритета. Снятие сигнала ЗПР обеспечивается дешифратором ДВ и формированием на его выходах сигналов, приводящих к сбросу по концу второго импульса INTA соответствующего триггера РПЗ. Завершает цикл обработки прерывания по запросу программный сброс соответствующего разряда внутреннего регистра состояния контроллера посредством записи команды КУ2 в регистр РД0. При этом возможно изменение статуса приоритетов входов [3]. Установка и чтение маски прерывания осуществляются путем записи и чтения информации с регистра РД1.

Программно-управляемый обмен в УПП-8 реализуется через РД0 при использовании вариантов команды КУ3 [3].

В УПП-8 предусмотрена возможность программного управления содержимым РПЗ, доступного только для записи. Прочитать его содержимое (при необходимости) можно через регистр РД0 командами КУ3. Такая возможность обеспечивает программное изменение (сброс или установку) содержимого РПЗ в процессе работы, проверку работоспособности УПП и отладку программного обеспечения путем имитационного моделирования поступающих на входы запросов прерывания.

Программирование обмена информацией и управление УПП-8 осуществляются стандартными командами микроЭВМ «Электроника 60» (см. таблицу). Использование в системе нескольких УПП-8 с различными ба-

Команда	Мнемоник	Содержание 0 и 1	Комментарий
КИ1 КИ2	MOV R0, @#177474 MOV R0, @#177476	AAA10110, XXXXXXXXX,	Установка базиса AAA вектора X — значение разряда безразлично
КУ1	MOV R0, @#177476 BIS#101, @#177476 MOV@#177476, R1	AAAAAAA, 40 UUUUUUU,	Установка маски прерывания Маскирование входов 1 и 7 Чтение маски прерывания
КУ2	MOV R0, @#177474  MOV R0, @#177474  MOV R0, @#177474	40 <sub>8</sub>  30N <sub>8</sub>  250 <sub>8</sub>	Сообщение БИС об окончании обработки текущего запроса Присвоение низшего приоритета входу N (N = 0...7) Сброс наивысшего запроса, циклическое изменение приоритетов
КУ3	MOV R0, @#177474	150 <sub>8</sub> 110 <sub>8</sub> 14 <sub>0</sub> * 12 <sub>8</sub> *	Установка режима спецмаскирования Сброс режима спецмаскирования Установка режима считывания следующей командой запроса с высшим приоритетом Установка режима считывания следующей командой состояния входов ЗП
	MOV@#177474, R1	UUUUUUUU,*	Считывание запросов (передается информация, определяемая одной из предыдущих команд)
РПЗ	CLR@#177472 MOV R0, @#177472 MOV R0, @#177474 MOV@#177474, R1	— AAAAAAA, 12 <sub>8</sub> UUUUUUUU,	Сброс РПЗ Запись информации в РПЗ Последовательность двух команд, обеспечивающих чтение информации с РПЗ
РС	TSTB@#177470 BISB#100, @#177470  CLR@#177470	— — —	Проверка готовности УПП Установка разрешения прерывания процессора сигналом с УПП Запрет прерывания процессора сигналом с УПП

Примечание. \* — основные команды, обеспечивающие организацию программно-управляемого обмена.

зисами, заданными при инициализации, увеличивает число входов запросов. С помощью переключателей S1 регистры УПП-8 смещаются в адресном пространстве микроЭВМ. При указанных на схеме (см. рис. 4) положениях переключателей установлены следующие адреса для регистров: РС — 177470; РД0 — 177474; РД1 — 177476; РПЗ — 177472.

Пример программы, включающей необходимые команды для организации работы с УПП-8, приведен на рис. 7.

Устройство приоритетного прерывания на 50 входов (УПП-50). Увеличить число уровней приоритета до 50 можно путем простого каскадного наращивания БИС КР580ВН59 (рис. 8). При этом одна из микросхем (БИС0) назначается ведущей (S=1), а остальные (БИС1... БИС6) — ведомыми (S=0). По отношению к схеме УПП-8 в УПП-50 внесены изменения в дешифратор адреса и введен дешифратор ведомой БИС (ДВ БИС). Регистры РПЗ и дешифраторы ДВ для каждой из ведомых БИС, ДК, ЛП, ЛС идентичны рассмотренным выше. Выходы INT ведомых БИС под-

ключаются к входам ЗП2... ЗП7 ведущей БИС0. Шины каскадирования (CAS0, CAS1, CAS2) всех БИС запаралилизуются и соединяются с входами ДВ БИС.

Дешифратор адреса УПП-50 (ДА1) вырабатывает отдельные сигналы «выборка микросхемы» (ВМ0... ВМ7) и распределяет сигнал ЗП РПЗ на соответствующие регистры РПЗ1... РПЗ6 в зависимости от адреса команды процессора. Закрепленная область адресов в ДА1 (рис. 9) — 1767XY, где X — номер БИС; Y — номер внутреннего регистра БИС и соответствующего ей РПЗ. Такое построение ДА1 обеспечивает организацию обмена информацией с каждой из БИС в отдельности, что необходимо для инициализации и управления. Инициализация БИС КР580ВН59 при каскадном включении подробно описана в работе [3].

В дешифраторе ДА1 использованы ПЗУ D8, D9 с приведенными на рис. 10 прошивками. ПЗУ D8 обеспечивает выборку отдельной БИС только при обращении к внутренним регистрам. Прошивка D9 по сигналу ЗП РПЗ формирует 1 на соответствующем выходе ЗП РПЗ1... ЗП РПЗ6 в зависимости от номера БИС

ОСНОВНАЯ ПРОГРАММА

ОСНОВНЫЕ ПАРАМЕТРЫ ДЛЯ УПП

(V0) - АДРЕС НАЧАЛЬНОГО ВЕКТОРА ПЕРЕРВАНИЯ для УПП
(CR) - ВЕКТОР (В СЛОВАХ), СОДЕРЖАЩИЙ АДРЕСА ПОДПРОГРАММ ОБРАБОТКИ ПЕРЕРВАНИЯ

V0: .WORD 200 ; МОЖНО ЗАДАВАТЬ 100, 140, 240, 300, 340

PR: .WORD PR0 ; PR0 ... PR7 СТАРТОВЫЕ АДРЕСА ПОДПРОГРАММ
.WORD PR1 ; АДРЕСА ПОДПРОГРАММ
.WORD PR2 ; ОБРАБОТКИ СИГНАЛОВ
.WORD PR3 ; ПО ВХОДАМ "ВХО"... "ВК7"
.WORD PR4 ;
.WORD PR5 ;
.WORD PR6 ;
.WORD PR7 ;

РАБОЧИЙ СТЕК
.STEK .WORD 0,0,0,0,0,0,0,0,0,0,0
.STEK .WORD 0 ; ВЕРШИНА СТЕКА

ФОРМИРОВАНИЕ УПРАВЛЯЮЩЕГО СЛОВА КИ1

MOV V0,R0
ADD #25,R0

ПРОЦЕДУРА ИНИЦИАЛИЗАЦИИ УПП

MOV R0,#177474 ; КОМАНДА ( КИ1 )
MOV R0,#177476 ; КОМАНДА ( КИ2 )
MOV #377, #177476 ; УСТАНОВКА МАСКИ ПЕРЕРВАНИЯ, ЗАПРЕТ ПО ВСЕМ ВХОДАМ УПП.
MOV #307, #177474 ; НАЗНАЧЕНИЕ НИЗВЕГО ПРИОРИТЕТА ВХОДУ #17.

УСТАНОВКА ПАРАМЕТРОВ ВЕКТОРОВ ПЕРЕРВАНИЯ

MOV STEK,R6 ; УСТАНОВКА СТЕКА
MOV V0,R0 ; ЗАПИСЬ В R0 АДРЕСА НАЧАЛЬНОГО ВЕКТОРА
MOV PR,R1 ; ЗАПИСЬ АДРЕСА ВЕКТОРА PR
CLR R2

S1: MOV (R1)+,(R0)+ ; УСТАНОВКА СТАРТОВЫХ АДРЕСОВ ПОДПРОГРАММ В ВЕКТОРЫ ПЕРЕРВАНИЯ
MOV #200,(R0)+ ; УСТАНОВКА ССП для команд из ПОДПРОГРАММ ПЕРЕРВАНИЯ ПРОГРАММ ЗАПРЕЩАЕТСЯ

INC R2
SFB #3,R2
BLE S1 ; ОРГАНИЗАЦИЯ ЦИКЛА

CLR #177472 ; СБРОС РПЗ УПП.
MOV #376,#177476 ; УСТАНОВКА МАСКИ ПЕРЕРВАНИЯ ИЛИ РАЗРЕШЕНИЕ ПЕРЕРВАНИЯ ТОЛЬКО ПО ВХОДУ "ВХО"
MOV #200,#177470 ; УСТАНОВКА ШЕСТОГО РАЗРЯДА РС УПП СРАЗУЩЕНИЕ ПРЕТРАЖЕНИЯ ПРОЦЕССОРА ОТ УПП

CLR R0 ; УСТАНОВКА ССП И РАЗРЕШЕНИЕ ПЕРЕРВАНИЯ ПРОЦЕССОРА
MTPS R0 ; СОРА ВНЕШНИЙ СИГНАЛИ

ТЕЛО ОСНОВНОЙ ПРОГРАММЫ

S2: WAIT ; ЗАЦИКЛИВАНИЕ
BR S2 ; КОМАНДА "WAIT"

КОНЕЦ ОСНОВНОЙ ПРОГРАММЫ

СТРУКТУРА ПОСТРОЕНИЯ ПОДПРОГРАММ ОБРАБОТКИ ПЕРЕРВАНИЯ

ПОДПРОГРАММА "PR0"

PR0: NOP
MOV #40,#177474 ; СБРОС РЕГИСТРА СОСТОЯНИЯ КОНТРОЛЕРА КР СБРОСОМ В УПП ( \* )
MOV #0,#177472 ; СБРОС СОДЕРЖИМОГО РПЗ
MOV #0,#177476 ; ИЗМЕНЕНИЕ МАСКИ ПЕРЕРВАНИЯ ИЛИ РАЗРЕШЕНИЕ ПЕРЕРВАНИЯ ПО ВСЕМ ВХОДАМ

ТЕЛО ПОДПРОГРАММЫ

RTI ; ВЫХОД ИЗ ПОДПРОГРАММЫ

ПОДПРОГРАММА "PR1"

PR1: NOP
MOV #40,#177474 ; ( \* )
MOV #340,#177474 ; ЦИКЛИЧЕСКОЕ ИЗМЕНЕНИЕ ПРИОРИТЕТНОСТИ ВХОДОВ ИЛИ ДЛИМ НАЗНАЧАЕТСЯ ВХОД "ВК5"

ТЕЛО ПОДПРОГРАММЫ

RTI ; ВЫХОД ИЗ ПОДПРОГРАММЫ

ПРИМЕЧАНИЕ

ОПЕРАТОР ВНЕШЕННИЙ ( \* ) ЯВЛЯЕТСЯ ОБЯЗАТЕЛЬНЫМ В ПОДПРОГРАММАХ. ОСТАЛЬНЫЕ ОПЕРАТОРЫ УПРАВЛЕНИЯ УПП (СМ. ТАБЛИЦУ) ВВОДЯТСЯ ПРИ НЕОБХОДИМОСТИ.

Рис. 7. Пример программы, включающей необходимые команды для организации работы с УПП-8

в адресной части команды. Информация в РПЗ с кодом Х=0 (БИС0) во все регистры записывается одновременно (используется при начальной установке УПП-50).

При возникновении требования прерывания по одному из входов соответствующая БИС выставляет сигнал (INT=1), который поступает на один из входов ЗП2...ЗП7 ведущей БИС0. При разрешенном прерывании по данному входу БИС0 формирует сигнал ЗПР, вызывающий прерывание выполнения текущей программы в процессоре.

В ответ на сигнал К ППР1 логика синхронизации формирует три сигнала INTA, поступающие на все БИС одновременно. При этом все БИС выбраны сигналами ВМ0...ВМ7 с ДА1. По первому импульсу INTA ведущая БИС выставляет код команды CALL, а на шинах CAS0...CAS2 — код ведомой БИС, с которой поступил запрос. Информация с шин каскадирования поступает в ДВ БИС, подготавливая его к работе.

По второму импульсу INTA ведомая БИС, выбранная информацией на шинах каскадирования, выставляет вектор прерывания, определяемый своим номером входа запроса и базисом, записанным при инициализации. По

окончании второго импульса INTA в ЛС формируется сигнал СБРОС РПЗ, поступающий через дешифратор ДВ БИС на один из дешифраторов вектора (ДВ1...ДВ6) в зависимости от номера выбранной ведомой БИС. Этот сигнал через ДВ сбрасывает соответствующий разряд РПЗ.

В УПП-50 применено шесть ведомых БИС, обеспечивающих организацию 48 уровней прерывания. Использование большего числа БИС (ведомых) для реализации, например 64 уровней прерывания, целесообразно, так как область векторов 0008...0768 в микрЭВМ «Электроника 60» закреплена за системными прерываниями и прерываниями от обычно используемых стандартных ВУ, а свободной остается область 1008...3768. Последняя полностью перекрывается УПП-50 при задании базисов для ведомых БИС 0102...1112. Прерывания по входам ЗП0 и ЗП1 размещаются в области 0408...0468 (базис для БИС0 — 0012).

При реализации простого приоритетного режима в УПП-50 приоритетность входов уменьшается с увеличением номера входа. В режиме циклического приори-

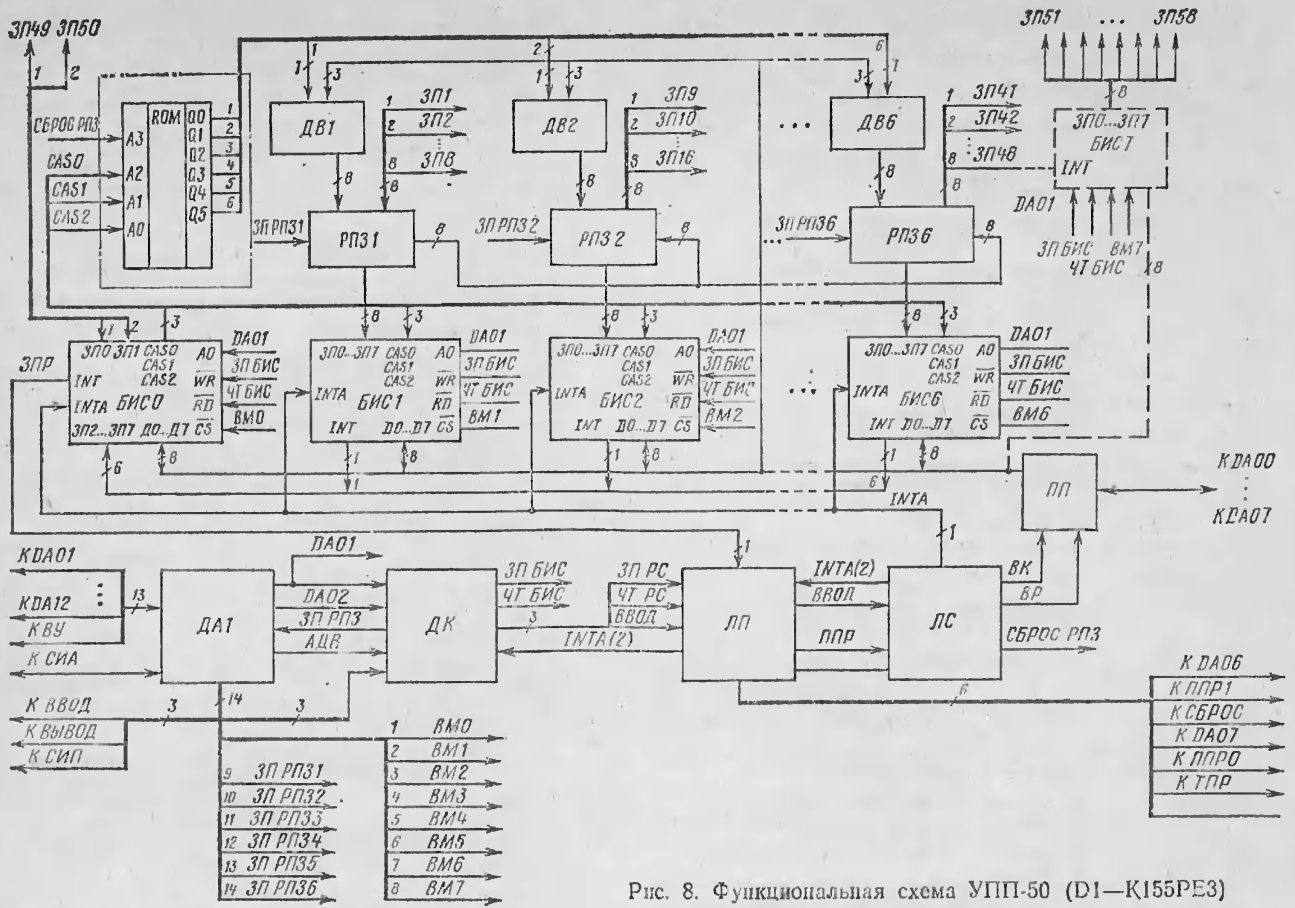


Рис. 8. Функциональная схема УПП-50 (D1—K155PE3)

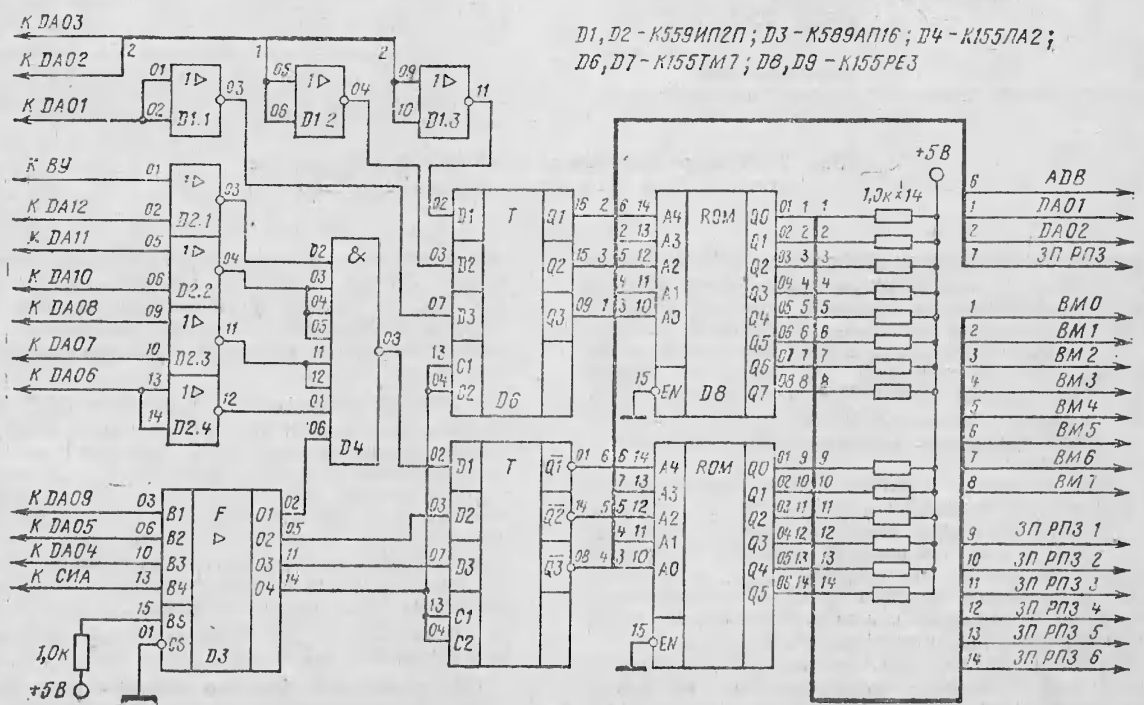


Рис. 9. Схема дешифратора адреса УПП-50

ADR. X=	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
- DATA < HEX > -																
D8	00X	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
	01X	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
- DATA < HEX > -																
D9	00X	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
	01X	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Рис. 10. Карты прошивки ПЗУ D8 и D9 дешифратора ДА1

тега приоритетность каждой БИС может задаваться отдельно и независимо друг от друга.

УПП с числом уровней более 50. Использование программно-управляемого обмена позволяет обеспечить дальнейшее наращивание уровней прерывания. Схематически это обеспечивается подключением дополнительных БИС ко входам ЗПО...ЗП50 УПП-50. Пример подключения одной дополнительной БИС показан на рис. 8 (пунктирными линиями). При дальнейшем наращивании числа дополнительных БИС необходимо внести изменения только в дешифратор адреса.

Запрос прерывания от дополнительных БИС обнаруживается по вектору прерывания, а идентификация запроса прерывания по входам запросов осуществляется программным опросом внутренних регистров БИС. Для последних, так же как и для основных БИС, возможны все стратегии маскирования и режимы.

УДК 681.3.06

С. Ю. Банников, К. С. Марфенко, Б. И. Подлепецкий, С. В. Турбаров

## ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС КПС-1813 ДЛЯ ПРОЕКТИРОВАНИЯ БИМЕДИЦИНСКИХ УСТРОЙСТВ С ЦИФРОВЫМ ПРОЦЕССОРОМ СИГНАЛОВ КМ1813ВЕ1

Первый отечественный цифровой процессор сигналов (ЦПС) КМ1813ВЕ1 [1, 2] включает в себя кроме процессора и памяти элементы аналогового интерфейса (АЦП, ЦАП, схемы выборки-хранения, мультиплексоры и др.). ЦПС могут быть использованы в аналоговой аппаратуре без подключения дополнительных схем сопряжения.

Основное назначение этого ЦПС — построение устройств обработки аналоговых сигналов в полосе частот 0...10 кГц. С помощью КМ1813ВЕ1 реализуются следующие функциональные узлы: цифровые фильтры, пороговые детекторы, ограничители, спектроанализаторы, модуляторы и демодуляторы, формирователи нелинейных функций, многочастотные генераторы и т. п. [1]. Возможности процессора КМ1813ВЕ1 позволяют эффективно решать задачи обработки сигналов при создании биомедицинских измерительных приборов и систем: улучшить их точностные параметры, повысить стабильность и воспроизводимость метрологических характеристик, использовать более

сложные методы обработки сигналов в реальном масштабе времени [3]. Особенности архитектуры ЦПС КМ1813ВЕ1 и невозможность доступа к его внутренним регистрам,

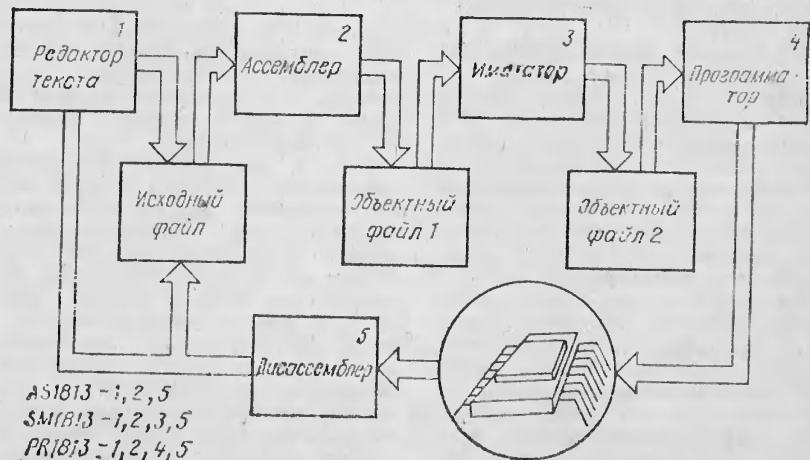


Рис. 1. Структура КПС-1813

Рассмотренные УПП выполнены в виде стандартных плат микроЭВМ «Электроника 60»: УПП-8 — одинарная; УПП-50 — двойная. Для повышения помехоустойчивости при работе в реальных условиях УПП-8 реализовано в виде двух модификаций: УПП с оптронной развязкой по входам запросов прерывания УПП-8/ОП; УПП с низкочастотными фильтрами УПП-8/Ф. УПП-8/Ф внедрено в автоматизированной системе неразрушающего контроля с обработкой информации на микроЭВМ «Электроника 60».

320625, ГСП-10, Днепропетровск, пр. Гагарина, 72,  
Днепропетровский Госуниверситет,  
кафедра радиозлектронной автоматики

### ЛИТЕРАТУРА

1. Гойхман П. А., Добровинский М. Е., Удовченко А. Б., Фланчик Б. С. Устройство приоритетного прерывания для микроЭВМ «Электроника 60» // Микропроцессорные средства и системы. — 1986. — № 1. — С. 73—75.
2. Лукьянов Д. А. КР 580 — автоматизация без проблем // Микропроцессорные средства и системы. — 1985. — № 1. С. 85—90.
3. Алексенко А. Г., Галицин А. А., Иванов А. Д. Проектирование радиоэлектронной аппаратуры на микропроцессорах. — М.: Радио и связь, 1984.
4. Центральный процессор М2. — Техническое описание и инструкция по эксплуатации 3.852.ТО — 1979.

Статья поступила 12.03.87

щинам и ячейкам памяти делают неприемлемыми традиционные методы при разработке аппаратуры на ЦПС. В связи с этим программное обеспечение (ПО) для КМ1813ВЕ1 проектируется с помощью соответствующих кросс-средств. Расширяется и круг вопросов, решаемых кросс-средствами. Необходимы также анализ характеристик устройств на ЦПС и оптимизация алгоритмов цифровой обработки сигналов.

Комплекс кросс-средств КПС-1813, разработанный на кафедре микро-

электроники МИФИ, состоит из ассемблера-дисассемблера AS1813, программной модели (имитатора) ЦПС SM1813, драйвера программатора PR1813 и специального программатора [4]. Состав и назначение программ комплекса (рис. 1) — традиционны [5].

К особенностям относится, во-первых, расширенный набор функциональных и эксплуатационных возможностей программ комплекса, ускоряющий проектирование. Например, имитатор SM1813 имеет встроенный редактор и транслятор (рис. 1). Это позволяет при отладке оперативно изменять программу. Во-вторых, комплекс построен так, что можно исследовать методы цифровой обработки сигналов (например, изучать процессы накопления погрешностей в различных структурах многозвенных цифровых фильтров, предельные циклы фильтров и т. д.). В-третьих, особенности архитектуры процессора определяют специфику структуры программной модели.

Все программы комплекса написаны на языках Паскаль и ассемблер Макро II и предназначены для работы на ЭВМ типа ДВК-2; ДВК-3, «Электроника 60», «Электроника 100/25» под управлением систем РАФОС, ОС ДВК, RT-11.

Ассемблер-дисассемблер AS1813 — это многофункциональная программа-транслятор для преобразования прикладных программ процессора KM1813 из текста в объектные коды и обратно. Транслятор AS1813 позволяет осуществлять ассемблирование прикладной программы из текстового файла с созданием объектного и листингового файлов, протокола трансляции на принтере; дисассемблирование прикладной программы из объектного файла; просмотр текстовых и объектных файлов (с дисассемблированием); редактирование текстовых файлов.

Полученный после работы с ассемблером AS1813 файл объектных кодов — исходный для программной модели SM1813.

**Программная модель SM1813.** Ядро комплекса, самая сложная его программа — это имитатор SM1813, работа с которым занимает 80...90 % времени проектирования [3]. Имитатор реализует свои функции по 56 командам, вводимым пользователем при работе в одном из следующих режимов: настройка имитатора, моделирование работы ЦПС, анализ результатов моделирования.

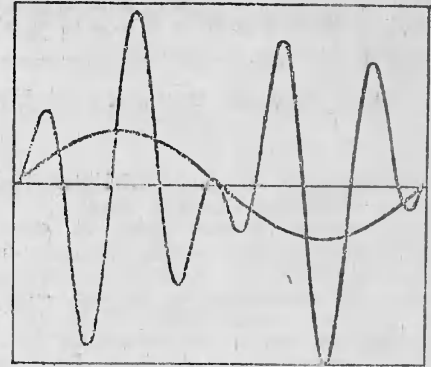
Настройка имитатора. На этапе подготовки имитатора к процессу моделирования необходимо: загрузить в память программ имитатора моделируемую программу, установить состояния флагов процессора, задать входные сигналы, установить точки остановов и трассировки, определить режим отладочной печати, если необходимо, загрузить

```
*IN1=0.5*SIN(T*PI*400)
*IN2=(SIN(T*PI*1700))*IN1/0.3
*BREAKPOINT COUNT=25 AND OUT3<=0.18
*TRACE IN1,IN2
*GRAPHICS SCREEN
*SIMULATE FROM 0
TRACE = IN1,IN2
```

Время начала имитации  
0.0000000

Количество кризисов: 2

Текущее время  
0.0153789



```
*SIMULATE/STEP FROM 15
TRACE = IN1,IN2
15 LDA ALFA,BETA,R00,N0P
SOURCE 0.0000000=0.0000000 00000000 00000000 B
DIBT 0.0000000=0.0000000 00000000 00000000 B
RESULT 0.0000000=0.0000000 00000000 00000000 B
DAR 0.0039062=0.0000000 11111111 11111111 B
OVF= 0 NOF= 1 CY= 0 OVE= 1 EOP= 1
T= 0.0002600 INP= 0.0000000
IN1,IN2
*QUALIFIER PG=0 AND T-TC)1E-5
*SRAPHICS SYMBOL
*SIMULATE FROM 0
TRACE = IN2
0.0898206!
0.2513453!
0.3876405!
0.3940390!
0.1599173!
-0.2757992!
-0.7095585!
-0.9008112!
-0.7109770!
-0.2930877!
0.2302253!
0.6909723!
0.9418168!
*DAR
DAR 0.0039062= 0.0000000 11111111 11111111 B
```

Рис. 2. Протокол работы с SM1813

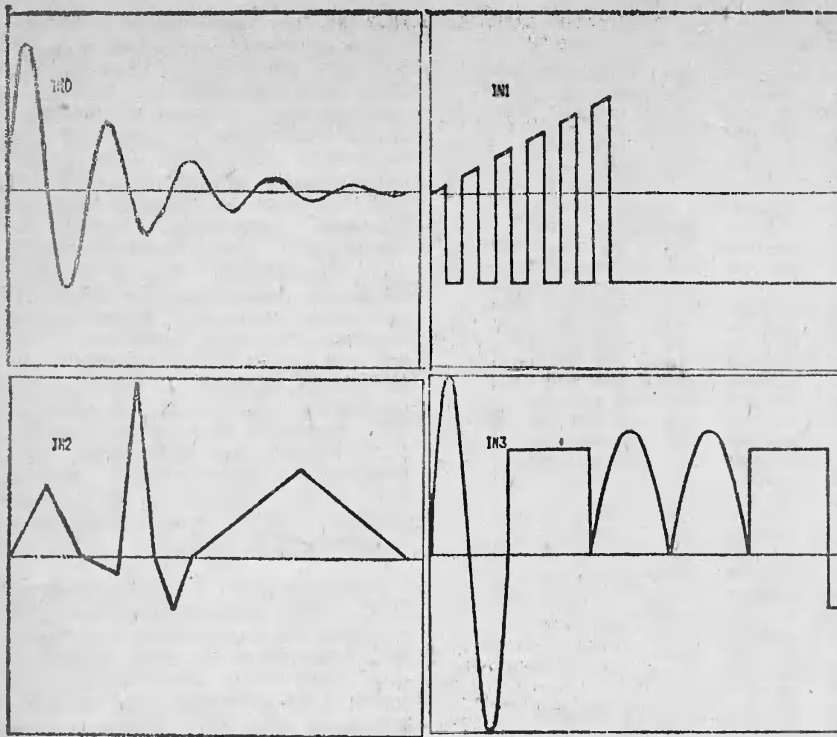
или очистить ячейки ОЗУ. В имитаторе предусмотрено также задание временных режимов работы процессора, т. е. изменение тактовой частоты и, соответственно, времени выполнения программ.

Если в моделируемой программе информация вводится в цифровом или аналоговом виде, то необходимо задать сигналы по соответствующим входам. В имитаторе предусмотрены четыре типа внешних воздействий для каждого из четырех входов процессора. Для их описания используются: функциональная зависимость от времени и других параметров, функциональная зависимость с проверкой логического условия, кусочно-функциональное представление и кусочно-линейная аппроксимация функции от времени. Таким образом, входные сигналы можно зада-

вать в виде сложных выражений (рис. 2, 3), включающих в себя наборы элементарных функций (SIN, COS, ABS, EXP), констант ( $\pi$ ,  $\pi/2$ ,  $2\pi$ ), логических конструкций (IF... THEN...ELSE...).

Процессом имитации управляют с помощью точек останова, элементов и условий трассировки. Условия трассировки определяют моменты вывода значений элементов трассировки.

Имитация работы ЦПС. В программе SM1813 реализованы непрерывный и пошаговый режимы имитации. В обоих режимах программы запускаются с любого адреса. Полученные во время имитации значения элементов трассировки записываются во внутренний буфер имитатора и одновременно выводятся на экран дисплея в виде таблицы или графика (см. рис. 2).



```

IN0=EXP(-T/0.00025)*SIN(T*PI*5E3)
IN1=IF T<0.005 AND P<10 THEN T*200 ELSE -0.5
IN2=TABLE 0.75,0.3 0.15,0 0.3,-0.2 0.35,1
        0.4,0 0.45,-0.4 0.5,0 0.8,0.4 1.1,0
IN3=CASE T 0,0.001:SIN(T*PI*1E3) 0.001,0.002:0.6
        0.002,0.004:0.7*ABS(SIN((T-0.002)*PI*0.5E3))
        0.004,.005:0.6 ELSE -0.5

```

Рис. 3. Моделирование входных сигналов

Для машин, имеющих графические дисплеи, реализован режим точечной графики (250×300 точек).

Анализ результатов имитации. После завершения или прерывания моделирования пользователь SM1813 может проанализировать и зафиксировать результаты. С этой целью в программе SM1813 предусмотрена возможность просмотра любого фрагмента или всего содержимого буфера трассировки, а также интересующих ячеек ОЗУ, состояния флагов и выводов, значений входных сигналов. Кроме того, можно рассчитывать необходимые выражения с помощью встроенного калькулятора программы SM1813. Вся информация, выводимая на терминал на этапах настройки имитатора, имитации и анализа результатов, может быть одновременно распечатана на построочно-печатающем устройстве или выведена в файл на гибком диске. Для сокращения времени подготовки имитатора к работе можно воспользоваться режимом загрузки команд из командного файла. При этом командный файл создается либо автоматически при вводе команд SM1813 с пульта, либо заранее с

использованием любого имеющегося в ОС редактора текста.

Часто при отладке необходима оперативная коррекция текста программы. Во всех известных программных комплексах для этого следует заново подготовить программу, оттранслировать ее и вновь загрузить в имитатор. В SM1813 имеется встроенный редактор, работающий с текстом моделируемой программы, а также транслятор для выявления ошибок и получения объектных кодов скорректированной программы. Это в 2,5...3 раза сокращает время отладки программы. Экранный редактор имитатора позволяет не только изменять тексты имеющихся программ, но и создавать новые. По своим функциональным возможностям он аналогичен таким редакторам, как SCREEN, EDIK, K52, USED. С его помощью можно изменять строки программы, вставлять новые фрагменты и стирать ненужные, перемещать и повторять отдельные блоки текста, находить необходимые символы или слова. После завершения работы с редактором команды ассемблера автоматически транслируются в объектные коды.

По завершении работы с имитатором SM1813 на магнитном диске создается файл, содержащий объектные коды отлаженной программы.

Заключительный этап проектирования устройства на ЦПС KM1813BE1 — это запись программ из полученного объектного файла во внутреннее ППЗУ процессора с помощью аппаратной приставки программатора и ее драйвера PR1813.

Драйвер программатора PR1813 реализует: запись кодов программ во внутреннее ППЗУ процессора KM1813BE1 из объектных файлов; считывание содержимого внутреннего ППЗУ процессора и сохранение кодов в виде объектных или текстовых файлов с дизассемблированными программами, записанными в памяти процессора;

просмотр на экране терминала содержимого ППЗУ в виде объектных кодов и текстов на ассемблере;

изменение режима записи информации в ППЗУ (начальный и конечный адреса записи, число циклов программирования);

проверку возможности записи программ обработки в память процессора полностью или частично;

редактирование и коррекцию текстов целевых программ перед их записью в ППЗУ;

трансляцию отредактированных текстов программ и сохранение их объектных кодов в виде файлов.

Все это значительно упрощает программирование ППЗУ, ускоряет коррекцию программ при использовании частично годных процессоров KM1813BE1, увеличивает срок службы кристаллов за счет увеличения циклов их перепрограммирования.

**Программатор для KM1813BE1.** Разработаны и созданы два варианта программатора для записи кодов программ во внутреннее ППЗУ процессора. Первый — универсальное устройство, работающее и автономно и в качестве приставки к микроЭВМ. В автономном режиме возможны ввод и редактирование программ в виде шестнадцатеричных кодов. Данные вводятся с помощью клавиатуры, а отображаются на цифровом табло, состоящем из 10 семисегментных индикаторов. Режимы считывания, программирования, контроля задаются клавишами. При работе программатора совместно с микроЭВМ все операции выполняются по командам от машины под управлением драйвера PR1813. Данный программатор реализован в виде микроконтроллера на базе МПК ГИС серии КР580. Объем ППЗУ управляющей программы и буферного ОЗУ по 2 Кбайт. К программатору можно подключить адаптеры для программирования ППЗУ других типов (КР573РФ1/2, КР556РТ4/5). Второй программатор имеет более простую схему, включающую 4 ИС (2×КР589ИР12, КР589АП16,

```

0 00C6EF LDA DAR KPO R00 INO 47 74C2FF LDA NP KPO R00 CNDS
1 08B23D ADD N KP4 R13 INO 48 4844EF LDA DAR Q R00
2 0200FF LDA N875 N1 R00 INO 49 70B2EF LDA N KPO R00 CNDS
3 02005A SUB N875 N1 R03 INO 50 4CDA6F LDA W KP5 R12
4 0600FF LDA N125 N1 R00 INO 51 4EDACC ADD W KP7 R07
5 06005C ADD N125 N1 R03 INO 52 4EDA6C ADD W KP7 R04
6 6818EF LDA Q Q0 R00 CVTS 53 44DA0C ADD W KP1 R01
7 4418EF LDA F3 N875 R00 54 40C6EF LDA DAR KPO R00
8 4410EB SUB F3 N R00 55 4C70EB SUB W SUM R00
9 7510FF LDA F4 N R00 CVT7 56 FD700A SUB W SUM R01 CND7
10 4618FB SUB F4 N125 R00 57 ED702A SUB W SUM R02 CND6
11 4032FF LDA Q0 DAR R00 58 DD704A SUB W SUM R03 CND5
12 4818EB SUB Q Q0 R00 59 CD706A SUB W SUM R04 CND4
13 4292ED ADD Q KP2 R00 60 BD70BA SUB W SUM R05 CND3
14 4840EF LDA M Q R00 61 AD70AA SUB W SUM R06 CND2
15 4A40E3 AND M F3 R00 62 9D70CA SUB W SUM R07 CND1
16 4A48E3 AND M F4 R00 63 8D70FA SUB W SUM R08 CND0
17 4EBAE1 XOR F3 KM1 R00 64 40CEEB SUB DAR KP1 R00
18 4C10E3 AND F3 Q R00 65 49C68A SUB DAR KP4 R05
19 4EBAF1 XOR F4 KM1 R00 66 40660C ADD DAR DAR R01
20 4C10F3 AND F4 Q R00 67 40668C ADD DAR DAR R05
21 4064EF LDA DAR M R00 68 4000EF NOP
22 70CA1C ADD NW KP1 R01 CNDS 69 4000EF NOP
23 7400EF LDA N1 N R00 CNDS 70 4000EF NOP
24 74C2EF LDA ERR KPO R00 CNDS 71 4000EF NOP
25 4A44EF LDA DAR F3 R00 72 8000EF OUTO
26 74CA1C ADD NP KP1 R01 CNDS 73 8000EF OUTO
27 7ECAEF LDA ERR KP7 R00 CNDS 74 8000EF OUTO
28 4A4CEEF LDA DAR F4 R00 75 8000EF OUTO
29 74CA1C ADD NP KP1 R01 CNDS 76 8000EF OUTO
30 7ECAEF LDA ERR KP7 R00 CNDS 77 8000EF OUTO
31 48C6EF LDA DAR KP4 R00 78 8000EF OUTO
32 406CEB SUB DAR NW R00 79 8000EF OUTO
33 7878EF LDA SUM SUMT R00 CNDS 80 4264EF LDA DAR ERR R00
34 70505E LDA SUMT N R03 CNDS 81 4000EF NOP
35 70CA1E LDA NW KP1 R01 CNDS 82 4000EF NOP
36 4066CB SUB DAR DAR L01 83 4000EF NOP
37 4064E3 AND DAR M R00 84 4000EF NOP
38 70505C ADD SUMT N R03 CNDS 85 9000EF OUT1
39 42C6EF LDA DAR KP2 R00 86 9000EF OUT1
40 426CEB SUB DAR NP R00 87 9000EF OUT1
41 70CA1E LDA NW KP1 R01 CNDS 88 9000EF OUT1
42 70505E LDA SUMT N R03 CNDS 89 9000EF OUT1
43 7400EF LDA N1 N R00 CNDS 90 9000EF OUT1
44 74C2FF LDA NP KPO R00 CNDS 91 9000EF OUT1
45 74C2EF LDA ERR KPO R00 CNDS 92 9000EF OUT1
46 4064EF LDA DAR M R00
    
```

правило, по длительности сердечного цикла между соседними R-зубцами. Для обнаружения R-зубца или всего QRS-комплекса электрокардиографического сигнала применяются схемы детекторов на серийных операционных усилителях либо с использованием микроконтроллеров на МП общего назначения. Анализ функций подобного устройства (фильтрация помех фоновых сигналов, артефактов; выделение QRS-комплекса; идентификация характерных точек комплекса; формирование выходного сигнала) показал, что все они могут быть выполнены на ЦПС KM1813BE1. С этой целью составлена программа фильтрации ЭКГ-сигнала с полосой пропускания 9...15 Гц, а также выделения QRS-комплекса и формирования сигнала пульса [6]. Отладка и запись программы в ППЗУ велась с помощью комплекса КПС-1813.

Устройство обработки ЧСС. При мониторинговании ЭКГ в условиях повышенной двигательной активности человека увеличивается вероятность ложного срабатывания QRS-детектора. Это происходит из-за повышения уровня многографического шума (например, при выполнении физических упражнений). Кроме того, учащаются сбои детектора вследствие обрабатывания артефактов, связанных с движением соединительных проводов и смещением электродов относительно кожи. В связи с этим необходимо устройство для анализа и обработки информации о ЧСС, достаточно просто реализуемое на базе ЦПС KM1813BE1. Процессор преобразует последовательности импульсов с выхода детектора в среднее значение ЧСС. На выходе ЦПС формируется уровень напряжения, пропорциональный ЧСС, выраженной в ударах сердца в минуту. Это значение подается в устройство отображения информации (цифровой индикатор) либо фиксируется на магнитной или бумажной ленте.

Для реализации алгоритма обработки ЧСС была создана программа (рис. 4), выполняющая: прием сигналов от детектора, «отбраковку» заведомо ошибочных R—R интервалов по длительности в сравнении с предыдущими циклами, усреднение нескольких последовательных интервалов, преобразование длительности в значение ЧСС (ул./мин), формирование уровня напряжения, пропорционального средней ЧСС, и сигнала ошибочного срабатывания детектора. Для отладки программы использован имитатор процессора SM1813. После записи программы в ППЗУ были проведены испытания устройства в масштабе реального времени, причем в качестве QRS-детектора применялось устройство [6] на базе ЦПС.

Рис. 4. Программа для устройства обработки ЧСС

ИМЯ	ЯЧЕЙКА	ОЗУ	ИМЯ	ЯЧЕЙКА	ОЗУ	ИМЯ	ЯЧЕЙКА	ОЗУ
N		00	ERR		10			
N875	1	F3	NP		11			
N1	2	F4	SUM		12			
N125	3	M	SUMT		13			
Q	4	NW	W		14			
ОШИБОК	:	0						
ПРЕДУПРЕЖДЕНИЙ	:	0						
ЯЧЕЕК ОЗУ	:	15						
ЯЧЕЕК ПЗУ	:	93						

K155ЛАЗ), и работает только в качестве приставки к микроЭВМ. Управляет ею специальный драйвер PR1813, входящий в состав КПС-1813. При программировании используется внешний источник на +25 В. Подключение к машине производится через интерфейс ИРПР или интерфейсную плату И2.

Таким образом, КПС-1813 и программатор — это законченная технологическая цепочка для проектирования устройств на ЦПС KM1813BE1.

Биомедицинские устройства на основе процессора KM1813BE1. Ис-

пользуя ЦПС KM1813BE1, можно реализовать разнообразные устройства обработки результатов измерений физиологических параметров. Ниже рассматриваются примеры внедрения процессора в медицинскую аппаратуру для обработки и анализа ЭКГ-сигналов [7].

Преобразователь ЭКГ—ЧСС. Для наблюдения за стационарными больными и профилактических обследований распространены системы оценки состояния человека по параметрам ЭКГ-сигнала, в частности частоте сердечных сокращений (ЧСС). Этот параметр определяется, как



**Характеристики устройства обработки ЧСС**

Диапазон измерения ЧСС, уд./мин	40...200
Погрешность измерений, уд./мин	±2
Периодичность цикла информации о ЧСС в отсутствие ложных срабатываний	1 раз на 8 сердечных циклов
Потребляемая мощность, Вт	~1

Таким образом, использование БИС КМ1813ВЕ1 при реализации описанных выше устройств обработки сигналов ЭКГ значительно упростило их структуру — число корпусов ИС по сравнению с аналогичными устройствами на операционных усилителях и МП КР580ИК80 сократилось в 9...10 раз. Кроме того, повысилась технологичность разработок, поскольку не требуется подгонка номиналов аналоговых компонентов при настройке фильтров. Применение комплекта КПС-1813 значительно сокра-

тило сроки разработки аппаратуры на ЦПС КМ1813ВЕ1.

Телефон 324-31-34, Москва

**ЛИТЕРАТУРА**

1. Цифровой процессор обработки сигналов КМ1813ВЕ1 и его применение/Под ред. А. А. Ланнэ, Г. Ф. Страутманиса.— М.: ЭКОС, 1987.
2. Марфенко К. С., Подлепецкий Б. И. БИС специализированных цифровых процессоров для обработки сигналов//Зарубежная электронная техника.—1985.— № 3.— С. 3—42.
3. Банников С. Ю., Марфенко К. С., Подлепецкий Б. И. Проектирование систем обработки электрофизических сигналов на основе цифрового процессора сигналов//Микроэлектронные методы и средства цифрового преобразования и обработки сигналов: Тез. докл.— Рига: Изд-е ин-та электроники и вычислительной техники АН ЛАССР, 1986. Т. 1.
4. Банников С. Ю., Марфенко К. С. Имитационное моделирова-

ние как средство разработки и отладки прикладных программ для цифровых процессоров сигналов// Программное обеспечение и применение микропроцессорных систем в устройстве: Материалы семинара.— М.: МДНТП, 1986. С. 88—90.

5. Кросс-системы—эффективные средства автоматизации программирования микропроцессорных устройств//Микропроцессорные средства и системы.—1986.— № 3.— С. 23.
6. Банников С. Ю., Марфенко К. С., Подлепецкий Б. И. Проектирование систем обработки биосигналов на базе цифрового процессора сигналов//Проблемы техники в медицине: Тез. докл.— Тбилиси: Изд-е ТПИ, 1986. С. 60—61.
7. Марфенко К. С. Система цифровой обработки биосигналов на базе однокристального цифрового процессора//Техника средств связи. Сер. ОТ.—1985.— Вып. 3.— С. 37—44.

Статья поступила 07.04.87.

УДК 681.327.8+621.317 : 621.3.541.138

А. В. Васильев, Е. А. Новиков, В. Н. Ширшов, С. П. Распонин, И. Ф. Ничков

**ДВУНАПРАВЛЕННЫЙ ИНТЕРФЕЙС СВЯЗИ С МИКРОЭВМ «ЭЛЕКТРОНИКА ДЗ-28»**

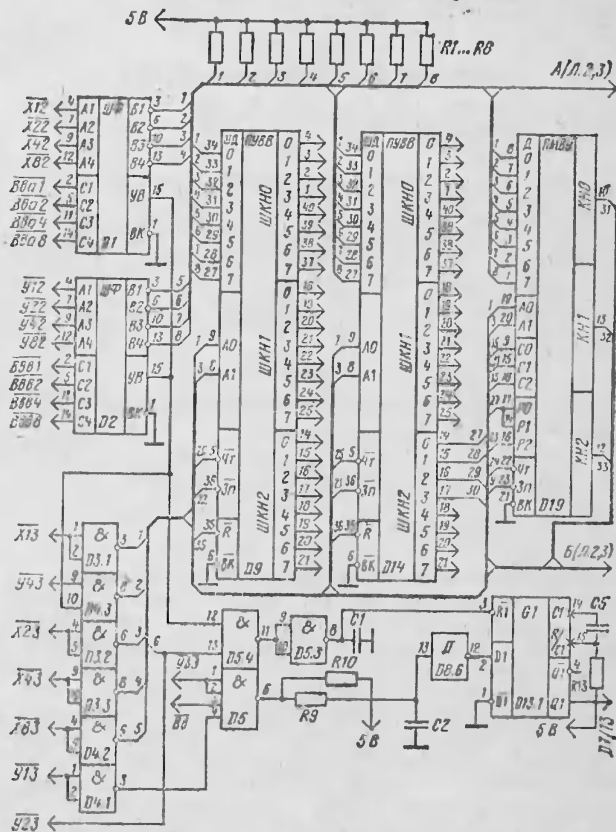
Разработанное устройство связи микроЭВМ «Электроника ДЗ-28» (ДЗ-28) с различными экспериментальными установками выполнено на БИС серий К580, К589, КР572 в отличие от аналогичных устройств, исполненных на микросхемах средней степени интеграции [1—6]. Интерфейс может быть использован как для одно-, так и двунаправленного обмена данными между ДЗ-28 и периферийными устройствами (ПУ). Он состоит из двунаправленного шинного формирователя КР589АП26, программируемого таймера КР580ИВ53, программируемого параллельного интерфейса КР580ВВ55, формирователя адресов и сигналов управления чтением и записью на микросхемах серии К155 и цифроаналогового преобразователя КР572ПА1А (см. рисунок).

Устройство в режимах ввода и вывода работает при наличии или отсутствии сигнала СИМ на шинах ДЗ-28. По нему идет формирование импульсов чтения или записи информации для D9, D14 и D19.

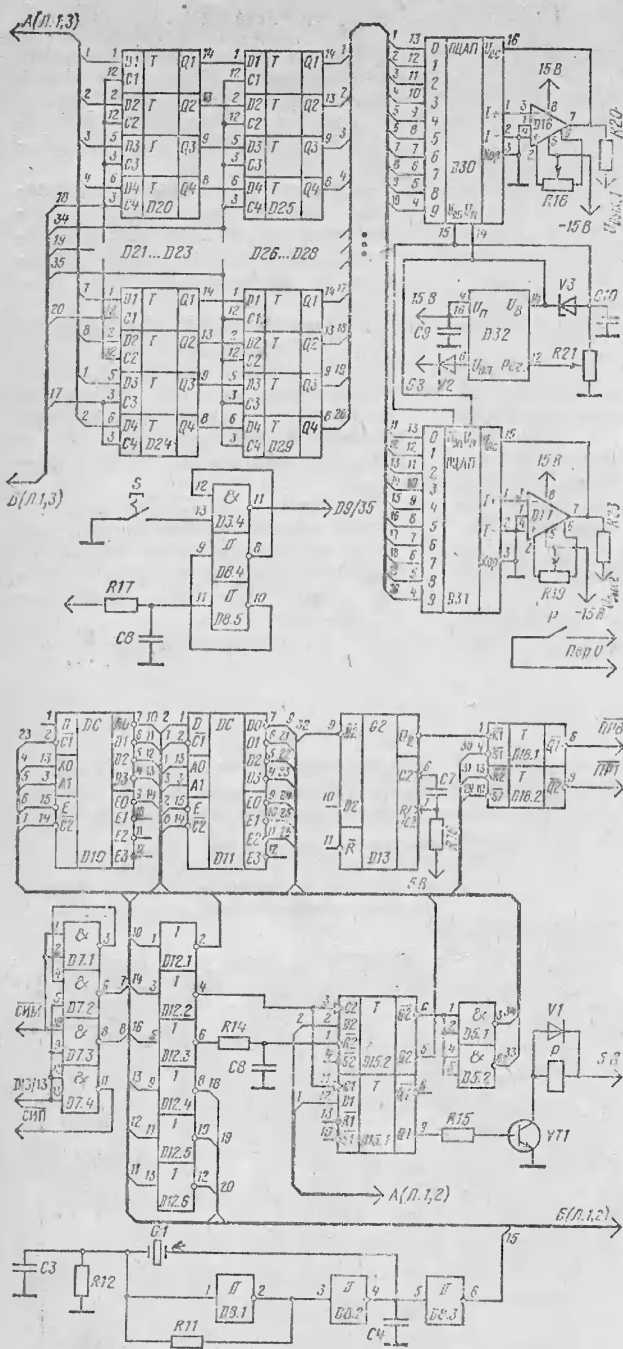
Режим ввода может быть организован по инициативе микроЭВМ или внешних устройств сигналами

ПР1, ПР2, ПР4 и ПР8. Сигналы ПР1 и ПР8 используются интерфейсом. В начале работы в D9, D14 и D19 записываются служебные слова, задающие режимы работы этих БИС [7]. Если D9 находится в режиме 0 (основной режим параллельного интерфейса КР580ВВ55) и на шинах УПР ДЗ-28 выставлен один из кодов: 01 08, 01 09, 01 10 в шестнадцатеричной системе исчисления, то по сигналу Вв от ДЗ-28 на D9 поступит сигнал Чт, сформированный микросхемами D7 и D10, а на ДЗ-28 по срезу Вв—сигнал СИП, формируемый микросхемами D6, D8, D13 и D7. По сигналу Чт на

D9 информация, сформированная на шинах ШКН1—ШКН3, записывается в буферный регистр этих каналов и без стробирования передается на шину ШД согласно



Принципиальная схема интерфейса (начало)



Принципиальная схема интерфейса (оконание):

D2, D1 — КР589АП26, D3... D5, D7 — К155ЛА3  
 D6 — К155ЛА6, D8 — К155ТЛ2, D9, D14 — КР580ВВ55,  
 D10, D11 — К155ИД4, D12 — К155ЛН1, D14 — К155АГ3,  
 D15, D18 — К155ТМ2, D16, D17 — К140УД3А,  
 D19 — КР580ВН153, D20... D29 — К133ТМ5, D30,  
 D31 — КР572ПА1А, D32 — К142ЕН1Б, G1 — кварцевый  
 резонатор на 32768 Гц, VT1 — КТ315Б, V1, D9,  
 V2 — ДЗ1Т, V3 — 2С456А, R1... R10, R15, R20, R23 — 1К,  
 R11... R13 — 2К7, R14 — К82, R16, R17, R22 — 10К,  
 R18, R19, R21 — 47К, C1 — 300пФ, C2 — 1н,  
 C3 — 6800пФ, C4 — 680 пФ, C5 — 1,0 мкФ, C6 — 47 н,  
 C7 — 0,33 мкФ, C8 — 1,0 мкФ, C9 — 4,7 мкФ,  
 C10 — 14,7 мкФ.

выставленному адресу на шинах А0—А1. Далее информация через микросхемы D1 и D2 поступает в D3-28. При этом сигналы УВ D1 и D2 установлены в 1. По снятию сигнала Вв интерфейс снимает сигнал СИП.

Таким образом, за чтение отвечает сигнал У13, а за запись — сигналы У43 и У13 одновременно. Кроме того, интерфейс выдает два аналоговых сигнала для управления графопостроителем Н 306, но при необходимости их можно использовать и для других целей. Роль буферных регистров в ИЦАП КР572ПА1А выполняют триггеры-зашелки на микросхемах К133ТМ5.

Узел управления пером графопостроителя выполнен на микросхеме D15,2, транзисторе VT1 и герконовом реле Р. Функцию таймера выполняет БИС КР580ВН153, которая представляет собой три независимых 16-битовых программируемых счетчика [7]. Счетчики С0 и С1 соединены последовательно так, что на счетный вход С0 поступают тактовые импульсы от генератора С1 через формирователь D8.1—D8.3 с частотой 32768 Гц, а на счетный вход С1 — сигнал КН0, который представляет собой программируемую функцию времени счетчика С0. Для удобства можно использовать режим 2 для всех счетчиков с записью в С0 и С2 числа 32768, что соответствует следованию сигналов КН0 и КН1 с периодом 1 с. В этом случае на вход С1 поступят импульсы с частотой 1 Гц. Это дает возможность записывать в С1 время непосредственно в секундах. По истечении заданного интервала времени С1 вырабатывает сигнал КН1, который через микросхемы D13.2 и D18.2 формирует сигнал ПР8. Сигнал КН2 через микросхему D18.1 формирует ПР1.

Сигналы ПР1 и ПР8 снимаются D3-28 через D14 по окончании обработки прерывания, как и рекомендовано в паспорте на микроЭВМ. БИС D14 устанавливает триггеры D18.1 и D18.2 в состояние Лог. 1. Микросхема D13.2 исключает возможность одновременной подачи сигналов S и R на триггер D18.2 и снятия прерывания раньше 1 с. Синхронизация таймера осуществляется микросхемой D14, которая по инициативе D3-28 подает сигналы разрешения счета на входы P0, P1 и P2 таймера.

Схема реализации адресов ПУ выполнена на дешифраторах К155ИД4 — D10, D11. Вспомогательную роль играют схемы 2И-НЕ и схемы 4И-НЕ, НЕ, микросхемы К155ЛА3, К155ЛА6 и К155ЛН1.

Конструктивно интерфейс собран на одной плате М12, которая помещена в металлический корпус вместе с блоками питания. Подводящие кабели представляют собой экранированные провода и витые пары.

Описанный интерфейс несколько лет работает в установке управления многоцелевым электрохимическим комплексом.

Телефон 44-88-13, Свердловск

#### ЛИТЕРАТУРА

1. Петров Н. С. // ПТЭ. — 1983. — № 4. — С. 94.
2. Афонин Ю. Д., Шалагинов В. Н., Бекетов А. Р. // ПТЭ. — 1984. — № 6. — С. 70.
3. Фрадков А. И., Зимин Г. П., Кацюба О. А. // ПТЭ. — 1982. — № 6. — С. 47.
4. Антонович А. И., Буцкий В. В., Миколенко В. Ф., Натансон И. Л. // ПТЭ. — 1982. — № 5. — С. 74.
5. Титаренко В. И., Луговой В. П., Шаповал В. И., Городыцкий А. В. // Укр. хим. журнал. — 1985. — № 3. — С. 268.
6. Александров А. Л., Базарный Е. М., Гуляев И. Б. // Электрохимия. — 1983. — № 1. — С. 198.
7. Алексенко А. Г., Галицын А. А., Иванов А. Д. Проектирование радиоэлектронной аппаратуры на микропроцессорах. — М.: Радио и связь, 1984.

Статья поступила 01.07.87

### ПРЕОБРАЗОВАТЕЛЬ ЦАП—АЦП ДЛЯ РАБОТЫ С ПРОЦЕССОРОМ КР580ИК80

В микропроцессорных системах сбора данных и устройствах управления широко используются ЦАП—АЦП. При работе с медленными процессами наиболее целесообразно использовать БИС К572ПВ1. Микросхема не является функционально законченным преобразователем, поэтому для сопряжения с системной шиной микроЭВМ на базе микропроцессора КР580ИК80 необходимы дополнительные элементы (рис. 1). Из 12 разрядов микросхемы К572ПВ1 использовано восемь. Предусмотрена возможность изменения разрядности преобразования в режиме АЦП через изменение времени преобразования. Работа ЦАП—АЦП осуществляется в процессе обращения процессора к источнику или приемнику аналоговой информации как порту ввода-вывода. Цикл ввода-вывода процессора удлиняется на время преобразования в режиме АЦП, время загрузки в микросхему К572ПВ1 и время, необходимое для работы приемника информации, снятием сигнала «готовность» процессора.

Все вспомогательные сигналы ЦАП—АЦП формируются программируемым таймером КР580ВИ53. Счетчик 0 таймера работает в режиме 1 (генератор меанд-

ра) и формирует меандр с частотой 250 кГц; счетчики 1 и 2—в режиме 2 (ждущий мультивибратор)\*. Временные диаграммы приведены на рис. 2, где  $\tau$ — время, необходимое приемнику аналоговой информации в режиме ЦАП,  $T=4$  мкс— период сигнала, стро- бирующего ЦАП—АЦП.

Сигнал «Строб» синхронизирует внешнее устройство, работающее на аналоговый вход-выход, и обеспечивает подключение к ЦАП—АЦП нескольких внешних устройств. В режиме АЦП длительность строба определяется временем преобразования, в режиме ЦАП— устанавливается программно в соответствии с циклом работы периферийного оборудования. При подключении к выходу ЦАП схемы выборки-хранения аналоговой информации  $\tau$  выбирается больше, чем время заряда емкости ячейки хранения.

Амплитуда аналогового сигнала  $U_{анал.}$  пропорциональна опорному напряжению  $U_{оп.}$  Коэффициент пропорциональности определяется информацией, снимаемой с шины данных микроЭВМ. Напряжение опорного источника можно изменять в пределах  $\pm 12$  В. В этих же пределах, но с обратным знаком, изменяется напряжение сигнала на выходе схемы ЦАП.

В режиме АЦП к микросхеме К572ПВ1 подключаются операционный усилитель К574УД1 и компаратор К554СА3Б. В режиме ЦАП компаратор отключен. При-

\* Торгов Ю. И. Программируемый таймер КР580ВИ53 и его применение // Микропроцессорные средства и системы. — 1984. — № 1. — С. 77—84.

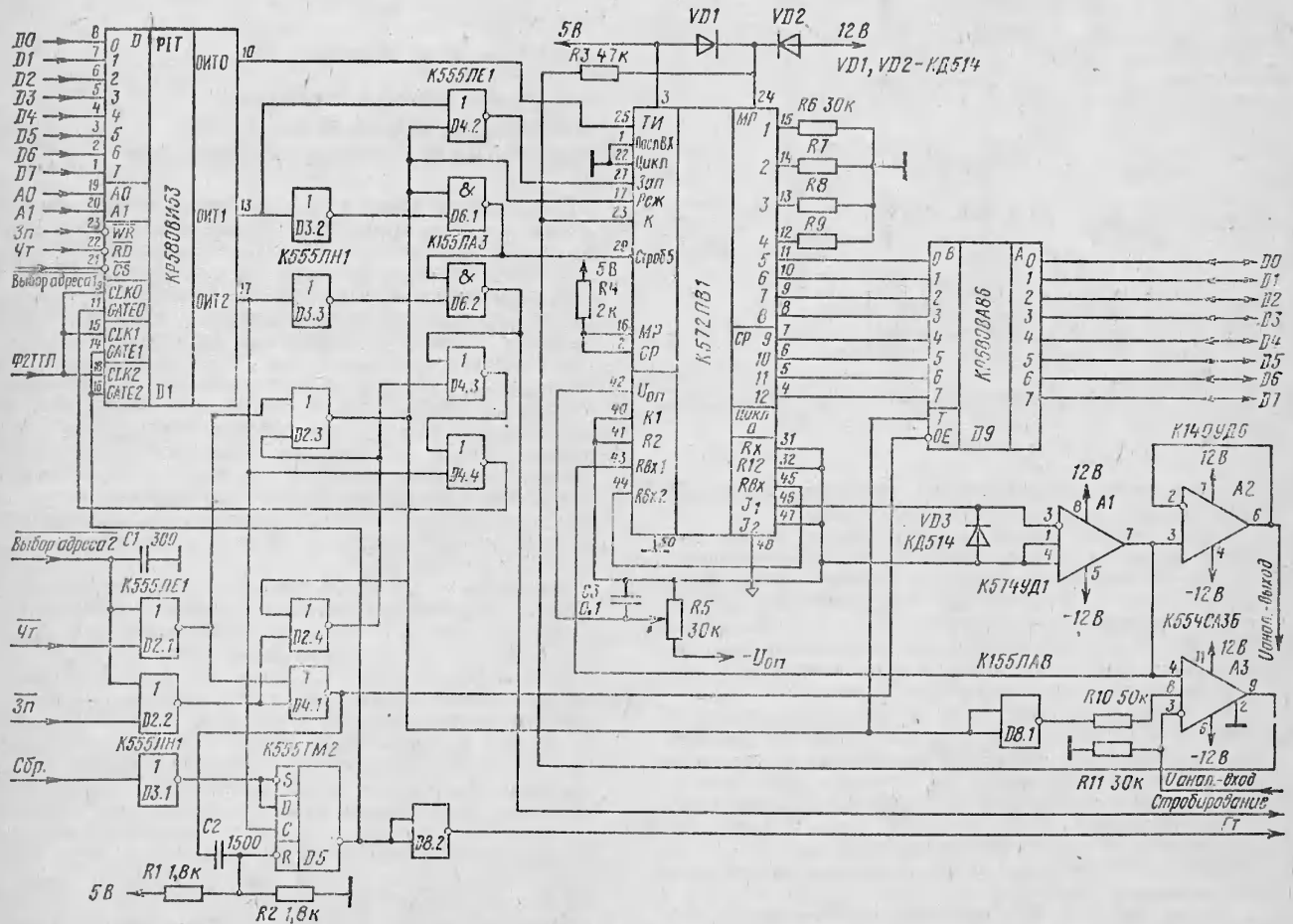


Рис. 1. Принципиальная схема ЦАП—АЦП

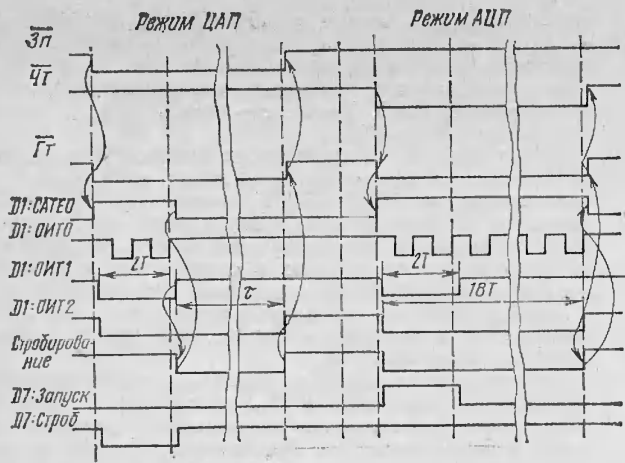


Рис. 2. Временные диаграммы работы ЦАП—АЦП

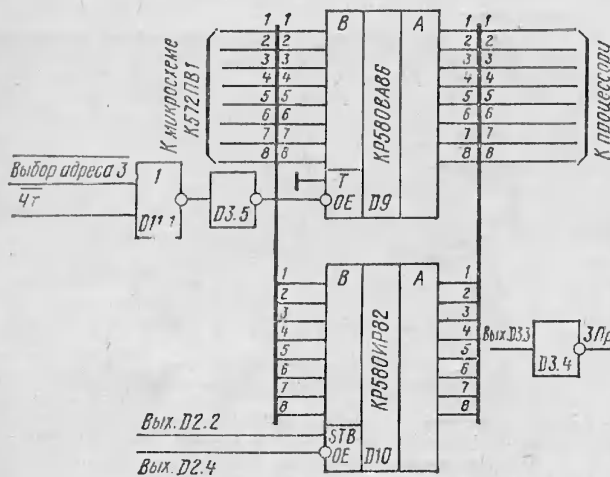


Рис. 3. Доработка принципиальной схемы ЦАП—АЦП

менение такой схемы позволяет обеспечить максимальную стабильность статических параметров.

Использование восьми разрядов дает погрешность квантования не больше 0,8%. Время преобразования АЦП — 72 мкс, время работы в режиме ЦАП — 8 мкс плюс время  $\tau$ . Погрешность преобразования определяется погрешностью квантования и стабильностью  $U_{оп}$ . Используя небольшую доработку схемы (рис. 3), можно значительно сократить потери времени. При этом преобразователь работает в двух режимах:

по аппаратному запросу на прерывание сигнал 0 T2 таймера КР580В153 подается на контроллер прерывания и в программе обработки прерывания производится опрос содержимого порта D9 в режиме АЦП или запись в регистр D10 новой информации в режиме ЦАП; по обращению к D9 или D10 через определенные программно-учтенные промежутки времени.

При доработке схемы необходимо отсоединить шину Гт от выхода D8.2.

Операторы 11 и 13 в программе инициализации таймера (рис. 4) задают время  $\tau$  в режиме ЦАП. Число, которое необходимо внести в эти операторы,  $N = \tau f_{\Phi 2}$ ,

СОДЕРЖАНИЕ; ТРН; настройка порта  
 OUT PУС  
 MVI A, 53H  
 OUT PУС  
 MVI A, 63H  
 OUT PУС  
 MVI A, 0BH; загрузка  
 OUT канал 0  
 MVI A, 16H  
 OUT канал 1  
 MVI A, 44H  
 OUT канал 2  
 MVI A, 01H  
 OUT канал 2  
 RET  
 END

Рис. 4. Программа инициализации таймера КР580В153 для работы в режиме ЦАП при тактовой частоте 2 кГц

где  $f_{\Phi 2}$  — тактовая частота процессора. Если оставить эти операторы без изменения (как в режиме АЦП), то  $\tau = 72$  мкс.

420084, г. Казань, ул. К. Маркса, 10,  
 КАИ Студенческое КБ «Прометей»; тел. 32-55-53,  
 39-71-65

Сообщение поступило 16.02.87

УДК 681.326

С. Н. Азин, Н. И. Муравьев

### МОДУЛЬ ВВОДА-ВЫВОДА ВИДЕОСИГНАЛА В ПАМЯТЬ МИКРОЭВМ «ЭЛЕКТРОНИКА 60»

Предлагается простой модуль ввода-вывода изображений в память микроЭВМ «Электроника 60». Модуль принимает цифровой код с устройств формирования видеосигнала (камеры на видеоконе или матрице ПЗС с цифровым выходом) и заносит его в память ЭВМ, читает содержимое памяти, преобразует код в аналоговый сигнал и формирует видеосигнал для бытового телевизора. Модуль предназначается для использования в гибридных системах обработки изображений и оцифровки изображений определенного класса. Его можно использовать в качестве контроллера полупроводникового дисплея, работающего с произвольной областью памяти (экранной ОЗУ), в том числе и с системной. Последнее представляет интерес для процесса обучения при демонстрации загрузки операционной системы, динамического распределения памяти работающей системой или программой.

В отличие от известных устройств ввода изображения модуль не использует внешнюю быстродействующую буферную память для хранения изображения или ввода изображения «по столбцам» (одна точка за время развертки одной строки) [1]. Здесь оцифрованный видеосигнал записывается прямо в штатную или дополнительную память микроЭВМ в течение времени развертки одного телевизионного кадра. Обмен между памятью и модулем организован в режиме прямого доступа. Для согласования скорости обмена и скорости ввода-вывода видеосигнала размеры изображения выбраны равными  $256 \times 256$  точек, а в одно машинное слово закодированы четыре точки изображения, т. е. на одну точку используются 4 бита. Выбор таких, на первый взгляд низких, информационных параметров изображения существенно упрощает построение схем ввода-вывода изображения и позволяет использовать стандартные блоки памяти микроЭВМ «Электроника 60».

Такой выбор вполне оправдан в ряде случаев обработки изображений. Так, при использовании модуля в гибридной системе 16 уровней яркости и 64К пиксел вполне достаточно для управления процессом обработки, так как собственно обработка проводится оптическим процессором. В тех задачах распознавания изображений, где нет необходимости выделять мелкие детали объекта (тем более, что зачастую оптическая система, строящая изображение, и не разрешает их), модуль можно использовать и для оцифровки изображений при последующей цифровой обработке на более мощных, чем «Электроника 60», ЭВМ.

Модуль может работать в трех программно переключаемых режимах. Режим 1 — пассивный: модуль не обменивается с памятью, канал предоставлен процессору, видеосигнал с камеры транслируется на телевизор. Режим 2 — запись: модуль активен в течение одного кадра телевизионной развертки. Режим 3 — чтение: модуль обращается по прямому доступу к указанной области памяти и формирует видеосигнал, соответствующий содержанию этой области.

В режимах записи и чтения модуль обменивается с памятью блоками по 64 слова, соответствующими одной строке развертки телевизионного изображения. Развертка телевизионного сигнала для телекамеры и телевизора построена. В течение импульсов гашения экрана модуль освобождает канал и процессор может продолжать выполнение программ или проводить регенерацию памяти (при этом скорость выполнения программ уменьшается в три раза). Так как прямой доступ возможен и во время цикла регенерации памяти (регенерация приостанавливается), то при соответствующем выборе частоты регенерации и длительности импульсов гашения регенерацию способен проводить сам процессор без участия модуля.

Технические характеристики модуля приведены ниже.

Размер изображения, строк×точек в строке . . .	256×256
Число уровней квантования яркости при вводе	16
Число уровней яркости при выводе на экран телевизора:	
черно-белый . . . . .	16
цветной . . . . .	4 плюс двухуровневые цветоразностные сигналы
Объем памяти одного кадра изображения, Кбайт	32
Объем используемой памяти, Кбайт	152
Число одновременно хранимых в памяти кадров	4
Время ввода одного кадра изображения, мс . . .	20
Время ввода-вывода строки изображения, мкс	37
Длительность строчного гашающего импульса, мкс	27
Длительность кадрового гашающего импульса, мс	3,5

Экранное ОЗУ может располагаться в любом месте памяти с дискретностью 8 Кбайт как в основной памяти объемом 56 Кбайт, так и в дополнительной. Адрес начала экранного ОЗУ задается программно. Модуль ввода-вывода не проверяет наличие ячеек памяти на общей шине, поэтому в области экранного ОЗУ могут отсутствовать отдельные блоки памяти. Для использования дополнительной памяти модуль формирует на общей шине расширения адреса по разрядам A16, A17, которые также устанавливаются программно.

Полная конфигурация памяти системы, используемой модулем, и схема организации ввода-вывода приведены на рис. 1. Старшие четыре банка системной памяти и дополнительная память реализованы на платах ПЗ с необходимой комбинацией расширенных адресных разрядов. Младшие три банка системной памяти не используют расширенный адрес. Управление модулем осуществляется через стандартную плату параллельного интерфейса И2.

Для управления модулем используются разряды PC00, PC01, PC06 и TPA регистра состояния интерфейсной платы И2 (регистр состояния модуля) и разряды VD00, VD01, VD02, VD12, VD14, VD15 регистра вывода (регистр адреса ввода-вывода) (рис. 2, 3).

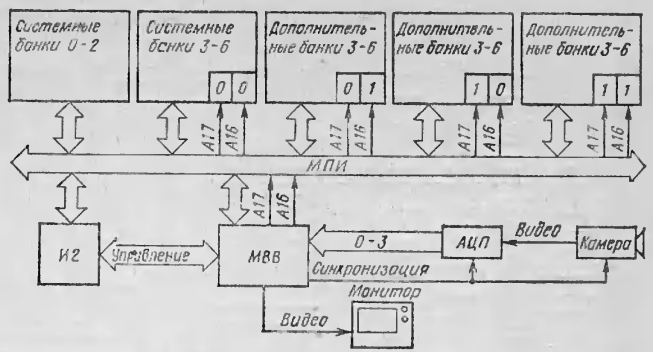


Рис. 1. Функциональная схема организации ввода-вывода видеосигнала в память

Регистр адреса ввода-вывода предназначен для управления расширенными адресными разрядами A16, A17 и задания адреса начального ОЗУ с дискретностью в один банк. Разряды 0, 1, 2 используются для управления доступом к дополнительной памяти. Разряды 0 и 1 соответствуют A16, A17; разряд 2 имеет следующее значение: если он сброшен, то расширенные адресные разряды передаются из регистра адреса в канал только во время прямого доступа (для работы операционной системы во время гашающих импульсов разряды A16, A17 очищаются); при установленном разряде 2 расширенный адрес передается на шину постоянно. Дополнительную память можно использовать и для других целей, например в качестве буфера для электронного диска.

Работа модуля. Принципиальная схема модуля изображена на рис. 4. Через разъем XP1 подключена плата И2, а через разъем XP2 — телевизор и АЦП видеосигнала с телекамеры. На схеме приняты следующие обозначения: TV — полный видеосигнал для телевизора; R — Y и BY — цветоразностные сигналы, сформированные из двух младших разрядов цифрового кода, хранящегося в памяти; IO, П, I2, I3 — входы оцифрованного видеосигнала; КСИ, ССИ — сигналы для синхронизации телевизионной камеры; ТИ — тактовый сигнал частотой 7 МГц для внешнего аналого-цифрового преобразования видеосигнала.

Для расширения возможности вывода с помощью переключателя S1 предусмотрен переход от 16-уровне-

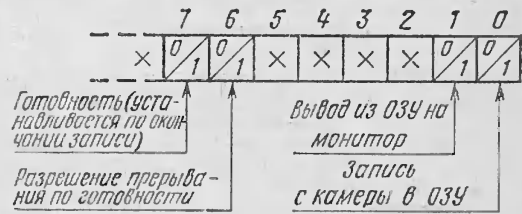


Рис. 2. Формат регистра управления модулем ввода-вывода

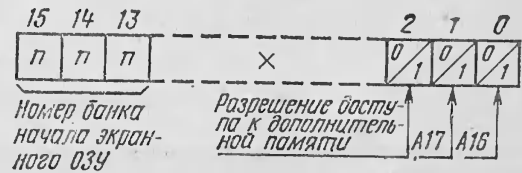


Рис. 3. Формат регистра адреса ввода-вывода

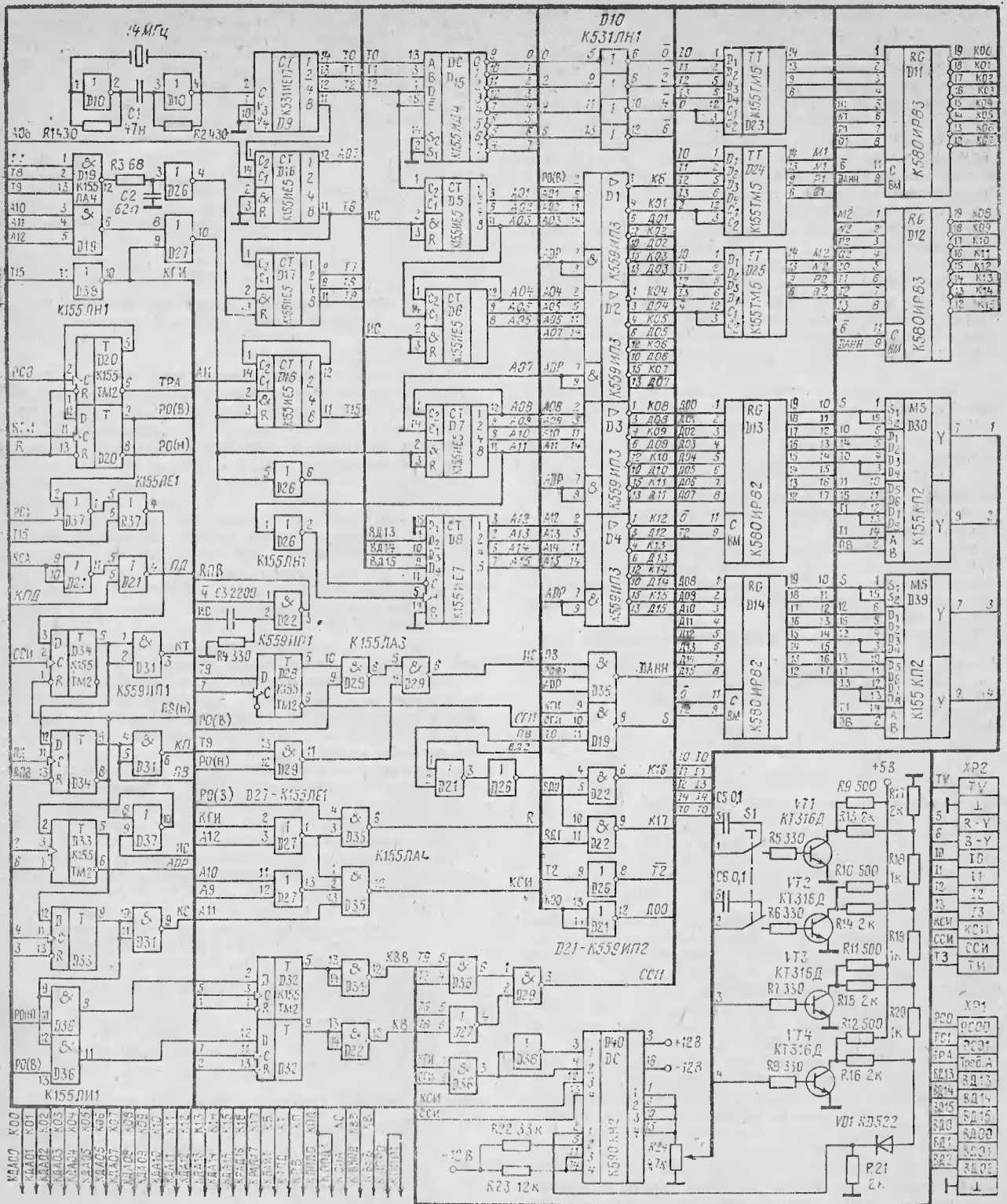


Рис. 4. Принципиальная схема модуля ввода-вывода видеосигнала

вого выходного видеосигнала к 4-уровневому и двум цветоразностным.

Счетчик адреса, формирователь строчных и кадровых синхронимпульсов (ССИ, КСИ) и гасящих импульсов

(СГИ, КГИ) собраны на ИС D5...D9, D16...D18. Так как при записи изображения сначала запакуются четыре точки и полученное слово записывается в память, а при чтении сначала читается слово и потом

распаковывается, то возможен сдвиг записанного изображения по отношению к наблюдаемому в пассивном режиме работы модуля. Для предотвращения сдвига счетчик тактов в строке (D16, D17) задублирован счетчиком адреса (D5, D6). При выводе они работают синхронно, а при записи адрес отстает на четыре такта. Счетчик D8 работает синхронно со счетчиком D18; первый формирует старшие разряды адреса, а второй — кадровый гасящий импульс (КГИ).

Запаковка кодов яркости четырех точек изображения в одно слово осуществляется микросхемами D23...D25 и D11, D12, а распаковка — D13, D14, D30, D39. Данные с регистров D13, D14 поочередно выставляются на коммутаторы D30, D39, которые, в свою очередь, еще два раза коммутируют данные на вход ЦАП. При сброшенных разрядах 0,1 регистра управления входной сигнал с АЦП проходит непосредственно на ЦАП для визуализации изображения перед записью.

Логика обмена данными с каналом ЭВМ реализована на триггерах D20, D32...D34 и синхронизируется с работой схем распаковки и запаковки дешифратором D15. При установленном режиме обмена срезом ССИ активизируется сигнал требования прямого доступа (ТПД). Это происходит примерно за 13 мкс до конца строчного гасящего импульса, так как время между требованием и предоставлением прямого доступа может достигать до 7,2 мкс. При выполнении команд расширенной арифметики оно может быть больше, поэтому, чтобы не было утери информации в начале строк при записи изображения, следует использовать прерывание по окончании записи с «зависанием» процессора по команде WAIT. Получив сигнал подтверждения прямого доступа (ППД) модуль устанавливает подтверждение выбора (ПВ), сбрасывает ТПД и проводит каналные циклы обмена. При тактовой частоте 7 МГц время запаковки и записи одного слова составляет 0,6 мкс. Когда содержимое счетчика тактов увеличится до 256, т. е. будет записана-считана строка видеосигнала в 64 слова памяти, счетчик строк наращивается на единицу, сбрасывается сигнал ПВ, и канал предоставляется процессору на время порядка 14 мкс до следующего ТПД. После обработки всех 256 строк, канал предоставляется процессору до конца КГИ.

Канальным сигналом КСБРОС модуль устанавливается в пассивный режим. Для работы с модулем использовалась бытовая камера «Электроника Л 801БК» и телевизор «Электроника ЛЦ 432». Камера была несколько изменена для синхронизации работы от внешнего генератора и вывода цифрового видеосигнала. Аналого-цифровой преобразователь выполнен на микросхеме К1107ПВ1.

Программная поддержка модуля. В системах, построенных на базе микроЭВМ «Электроника 60» и работающих под управлением дисковой операционной системы, дополнительную память, используемую модулем ввода-вывода, можно представить в виде внешнего устройства произвольного доступа с файловой структурой, например в виде электронного диска [2]. Такое представление позволяет осуществлять доступ к экранному ОЗУ средствами операционной системы. Создав на подобном электронном диске ряд файлов, можно запоминать изображения на других внешних устройствах системы, просматривать хранящиеся на них файлы изображения (или файлы вообще), проводить обработку или генерацию изображений программами, написанными на языках со средствами доступа к устройствам файловой структуры.

Так, в операционной системе типа РАФОС три дополнительные платы памяти ПЗ соответствуют электронному диску емкостью 192 блока. Три неперекрывающихся изображения будут занимать по 64 блока каждое и располагаться начиная с блоков 0, 64 и 128. При использовании справочника диска под первое изображение отводятся блоки 16...63, так как экранное ОЗУ может

```

.TITLE IM
.MCALL .DRDEF
.DRDEF IM,0,FILST#,192.,167752,0
ADSW =330 ;АДРЕС СВЯЗКИ.

;ИСПОЛЬЗОВАНИЕ СЕКЦИИ SET ДЛЯ УПРАВЛЕНИЯ ВВОДОМ-ВЫВОДОМ
.DRSET INP,2,0.INP,0CT ;ВВОД ИЗ ОЗУ
.DRSET OUT,3,0.OUT,0CT ;ЗАПИСЬ ИЗОБРАЖЕНИЯ
.DRSET MEM,2,0.MEM,0CT ;ВЫВОД ИЗ СИСТЕМ. ОЗУ
.DRSET WIZ,400,0.WIZ ;КАМЕРА НА МОНИТОР.
O.INP:
O.OUT: BIC #177774,R0
      BNE OUT1
      MOV #1000001,R0
      BR INIT
OUT1: ADD #60000,R0
INIT: MOV R0,#IM#CSR
      MOV R3,#<IM#CSR-2>
      RTS PC
O.WIZ: CLR R0
      BR INIT
O.MEM: BIC #177774,R0
      ASH #12., R0
      BR INIT

.DRBEG IM ;ВХОД В ДРАЙВЕР.
MOV #ADSW, R4 ;ПЕРЕПИСАТЬ СВЯЗКУ.
MOV %7, R5
ADD #<SNQZ-.>,R5
MOV (%5)+, (%4)+
BNE -2
MOV #<SWL1-SNQZ+ADSW>,R4;АДРЕС МОДИФИКАЦИИ В R4

MOV IMCRQ, R5
MOV (R5)+, R0 ;НОМЕР БЛОКА В R0.
MOV (R5)+, R1 ;КОД ФУНКЦИИ В R1.
TSTB R1 ;УКАЗАНЫ СПЕЦФУНКЦИИ ?
      BNE DONE ;ВЫХОД,ЕСЛИ СПЕЦФУНКЦ.
MOV (R5)+, R2 ;АДРЕС БУФЕРА В R2.
MOV @R5, R1 ;СЧЕТЧИК СЛОВ В R1.
BEQ DONE ;ВЫХОД,ЕСЛИ СЧЕТЧИК =0.

MOV R0, R3 ;УСТАНОВИТЬ В R3
BIC #177477,R3 ;РАЗРЯДЫ 16,17.
ASH #-6., R3
ADD #5, R3
BIC #177700,R0
ASH #9., R0
ADD #60000,R0

TST R1 ;R1=0 - ЧТЕНИЕ.
BPL WRDR1

WRDR1: MOV (%7)+, @R4 ;ИНАЧЕ,ЗАПИСЬ.
      MOV R4, (R0)+ ;ПОДГОТОВИТЬ ЗАПИСЬ.
      NEG R1 ;ИНВЕРТИРОВАТЬ СЧЕТЧИК.
      MOV R1, R5 ;ДОПОЛНЕНИЕ R5 ДО 256.
      BIC #377, R5
L3DR1: MOV (R2)+, R4 ;СЛОВО ИЗ БУФЕРА В R4.
      JSR %7, @#ADSW ;ОБРАТИТЬСЯ К СВЯЗКЕ.
      DEC R1 ;УМЕНЬШИТЬ СЧЕТЧИК.
      BNE L3DR1 ; # 0 ? ,ПРОДОЛЖИТЬ.
      CLR R4

L4DR1: TSTB R5 ;ВОЗВРАТ В СИСТЕМУ,ЕСЛИ
      BEQ DONE ;ЦЕЛОЕ ЧИСЛО БЛОКОВ.
      JSR %7, @#ADSW ;ИНАЧЕ,ОЧИСТИТЬ
      INC R5 ;НЕЗАПОЛНЕННЫЙ БЛОК.
      BR L4DR1

REDR1: JSR %7, @#ADSW ;ПРОЧИТАТЬ БЛОК ИЗ ИМ.
      MOV R4, (R2)+
      DEC R1
      BNE REDR1

IMINT:
DONE: MOV IMCRQ, R5 ;ВОЗВРАТ В СИСТЕМУ.
      CLR 6(R5)

```

Рис. 5. Драйвер электронного диска (начало)

```

SHQZ: ; ТЕКСТ СВЯЗКИ СИСТЕМНОЙ И ДОПОЛНИТЕЛЬНОЙ ПАМЯТИ
MFPS - (SP)
MOV @#IM#CSR, SWL2+2 ; ЗАПОМНИТЬ РЕГИСТР IM.
MTPS #2#0# ; ЗАПРЕТИТЬ ПРЕРЫВАНИЯ.
MOV R3, @#IM#CSR ; УСТАНОВИТЬ P16, 17.
SWL1: MOV (R0)+, R4 ; ПЕРЕСЛАТЬ СЛОВО.
CMP #16#0#0#, R0 ; АДРЕС ЗА ГРАНИЦЕЙ ПЗ ?
BNE SWL2 ; ПЕРЕХОД ЕСЛИ НЕТ.
INC R3 ; НАРАСТИТЬ P16, 17.
MOV #6#0#0#, R0 ; АДРЕС НАЧАЛА ПЗ.
SWL2: MOV #2#0#, @#IM#CSR ; ВОССТАН. РЕГИСТР IM.
CLR @#IM#CSR ; ОЧИСТИТЬ P16, 17.
MTPS (SP)+
RTS #7
.word 0 ; КОНЕЦ ТЕКСТА СВЯЗКИ.

```

```

.DREND IM ; КОНЕЦ ДРАЙВЕРА.
.END

```

Рис. 5. Драйвер электронного диска (окончание)

располагаться в памяти только с начала банка, а блоки 0 ... 15 — под каталог и как системные.

Чтобы реализовать эту возможность для операционной системы РАФОС, разработан драйвер электронного диска. Особенность драйвера — подпрограмма, которую драйвер переписывает при вызове в старшую неиспользуемую часть области векторов. Подпрограмма служит для обмена данными между дополнительной памятью и системой в то время, когда установлены расширенные адресные разряды и старшие четыре банка системной памяти заменяются в адресном пространстве канала экранного ОЗУ на дополнительную память. Банки же 0 ... 2 всегда доступны процессору независимо от разрядов A16, A17 и подпрограмма может обращаться к экранному ОЗУ.

Секция SET драйвера используется для управления модулем ввода-вывода. Запись изображения в ОЗУ производится по команде

SET IM OUT=*n*,

где при *n*=0 изображение (без последних 64 строк) записывается начиная с 16 блока, не накладываясь на каталог; при *n*=1 весь кадр изображения записывается начиная с блока 0; при *n*=2 или 3 — начиная, соответственно, с блоков 64 и 128. По команде SET IM INP=*n* содержимое экранного ОЗУ выводится на телевизор. Числовое значение параметра INP интерпретируется так же, как и в случае записи. Для просмотра содержимого системной памяти используется команда

SET IM MEM=0, 1, 2, 3.

Значение MEM определяет номер банка, начиная с которого отображается содержимое памяти. Командой

SET IM WIZ

камера переключается на телевизор и модуль ввода-вывода прекращает обращение к каналу по прямому доступу.

Во время вывода изображения из ОЗУ на телевизор допускаются операции обмена с электронным диском. Однако при этом следует помнить, что драйвер может установить разряды A16, A17 в то время, когда модуль использует другую комбинацию разрядов и произойдет переключение ОЗУ с драйвером. Текст драйвера приведен на рис. 5.

Телефон 65-60-56, Горький, ГГУ, радиофак

ЛИТЕРАТУРА

1. Сасов А. Ю. Микротомография и цифровая обработка изображений на микроЭВМ «Искра 226» // Микропроцессорные средства и системы. — 1986. — № 1. — С. 53—58.
2. Лукьянов Д. А. «Электроника 256 К» — эмулятор диска для комплексов на основе микроЭВМ «Электроника 60» и ДВК // Микропроцессорные средства и системы. — 1986. — № 2. — С. 62—65.

Статья поступила 09.04.87

М. И. Ройзен, А. Н. Карпушов, А. Ю. Бургасов, А. Г. Белый

СИСТЕМА ОРГАНИЗАЦИИ ПРЯМОГО ЦИФРОВОГО УПРАВЛЕНИЯ НА БАЗЕ МИКРОЭВМ ТИПА «ЭЛЕКТРОНИКА 60»

МикроЭВМ типа «Электроника 60» позволяют обеспечить прямое цифровое управление несколькими контурами регулирования. Таким системам присущи периодический характер работы отдельных контуров (в заданном режиме), синхронизации вычислительного процесса в ЭВМ с работой датчиков и преобразователей информации (чаще многоканальных АЦП). При этом отдельные контуры управления, а также части одного контура могут работать в соответствии со спектром обрабатываемых сигналов на различных частотах дискретизации (различающихся, например, в радиотехнических системах на 1—2 порядка).

Такие МП системы чаще всего встраивают в оборудование. Выполняют их на простых микроЭВМ (микроконтроллерах с минимальным внешним оборудованием и практически без встроенной ОС).

Функции диспетчера в этом случае целесообразно выполнять аппаратно. Это обеспечивает максимальную производительность ЭВМ, многоканальных датчиков и преобразователей, т. е. позволяет максимально использовать возможности аппаратуры в отличие, например, от чисто программной системы реального времени [1].

В данной статье рассматривается типичный пример такой системы на базе управляющей микроЭВМ типа «Электроника 60» и многоканального АЦП.

Структура системы (рис. 1) и принцип работы. Многоканальный АЦП состоит из аналогового мультиплексора, устройства выборки-хранения (УВХ) и непосредственно АЦП.

Процессами преобразования АЦП, ЦАП и вычисления ЭВМ управляет блок микрокомандного управления (БМУ), состоящий из генератора тактовых импульсов, счетчика и ПЗУ микрокоманд (ПЗУМК).

После подачи тактового импульса с периодом *T<sub>t</sub>* счетчик увеличивает адрес ПЗУМК на единицу. Поле П1 микрокоманды определяет номер входного канала, подключаемого аналоговым мультиплексором на вход АЦП. Затем с соответствующей задержкой УВХ переводится в режим хранения и запускается АЦП. После

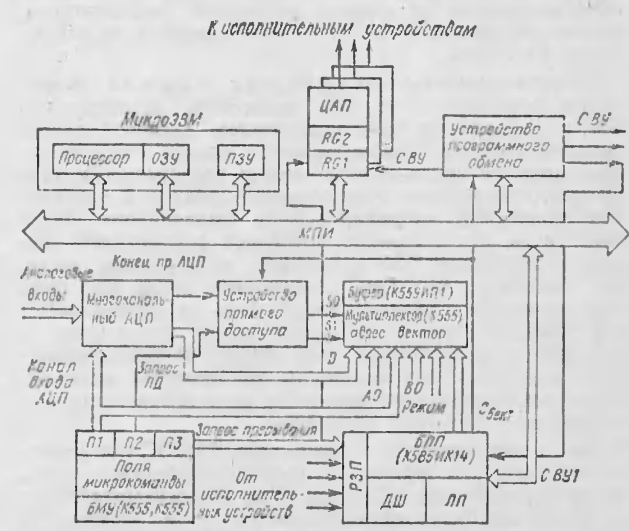


Рис. 1. Структурная схема системы



преобразования АЦП выдает сигнал «Конец преобразования АЦП» (Конец пр. АЦП) на устройство прямого доступа, которое согласно требованиям интерфейса МПИ записывает данные D с выхода АЦП через мультиплексор с буфером в ячейку ОЗУ микроЭВМ. Старшие разряды адреса ОЗУ  $A_0$  задаются при этом перемычками (начальный адрес массива каналов АЦП в ОЗУ), а младшие — тем же полем П1 микрокоманды, определяющим преобразованный канал АЦП.

Так же проводится преобразование и пословный прямой доступ в ОЗУ микроЭВМ следующего входного канала АЦП.

Последовательность преобразуемых каналов АЦП определяется заранее и записывается в ПЗУ МК. Для этого входные каналы разбиваются по периоду дискретизации на группы с номерами  $n=1, 2, \dots, N$  (в примере  $N=4$ ). Последовательность каналов группы  $n < N$  может быть прервана с глубиной  $N-n$  последовательностями каналов других групп, имеющих большую частоту дискретизации. Для осуществления требуемой периодичности, как будет показано ниже, в некоторых микрокомандах битом П2 «Запрет ПД» (см. рис. 1) прямой доступ в ОЗУ должен быть запрещен.

В микрокоманде преобразования и ввода данных АЦП по прямому доступу в ОЗУ микроЭВМ последнего входного канала  $n$ -й группы в разряде  $n$  поля П3 запросов прерываний микрокоманды записывается Лог.1. Это приводит к установке RS-триггера в  $n$ -м разряде регистра запросов прерываний, подающем далее сигнал запроса прерывания на соответствующий вход блока приоритетного прерывания (БПП).

Прерывания групп с большей частотой дискретизации подаются на входы БПП, имеющие больший приоритет. ЭВМ переходит на обработку  $n$ -й группы данных в тот момент, когда приоритет ее запроса станет самым большим относительно других групп, запрашивающих прерывание, а также больше, чем приоритет текущей программы процессора, записанный программно в БПП, и при условии, что прерывания процессора разрешены. Устройство прерываний организует прерывание по вектору. Старшие разряды  $V_0$  вектора задаются перемычками (начальный адрес массива векторов прерываний в ПЗУ), а младшие — кодом на выходе БПП. Этот код указывает группу входов, после которой проведено прерывание. При необходимости на один или несколько разрядов вектора может быть подан код режима, например от пульта управления или исполнительных устройств. В этом случае для каждого режима могут быть свои программы обработки групп входов.

При предоставлении прерываний через дешифратор происходит сброс триггера обслуженного запроса прерывания. После окончания самого большого периода дискретизации группы 1 битом П4 («Сброс СТМК») микрокоманды (см. таблицу) производится сброс в нуль счетчика микрокоманд и начинается новый цикл БМУ.

Таким образом, вся программа ЭВМ — это система вложенных периодически повторяющихся прерываний глубиной  $N-1$ .

Время непосредственного преобразования многоканального АЦП на микросхемах K590АН6, K1100СК2 и K572ПВ1, мкс . . . . . 140  
 Максимальное время предоставления прямого доступа для микроЭВМ «Электроника 6Б», мкс . . . . . 2  
 Цикл вывода данных по прямому доступу в ОЗУ микроЭВМ, мкс . . . . . 1  
 Максимальное время входа и выхода в программу прерываний (без времени ожидания предоставления прерываний) [2], мкс . . . . . 31

При выборе частот дискретизации примерно 10, 40, 80 и 550 Гц общее число прерываний в секунду равно 680. При этом за секунду максимальное число преобразований АЦП в системе может быть  $1/(140+9+1) \cdot 10^{-6} = 6711$ . Это составляет 94% от максимально возможного для данной схемы АЦП числа преобразо-

Адрес МК	П1		П2	П3				«Сброс СТМК» («Лог. 1»)
	Поле канала входа АЦП		«Запрет ПД» («Лог. 1»)	Поле управления прерываниями				
	группа	порядковый номер в группе		группы входов 1 2 3 4				
0	4	1	0	0	0	0	0	0
1	4	2	0	0	0	0	0	0
3	4	3	0	0	0	0	1	0
.	.	.	.	.	.	.	.	.
125	—	—	1	0	0	0	0	0
126	2	1	0	0	0	0	0	0
127	4	1	0	0	0	0	0	0
128	4	2	0	0	0	0	0	0
129	4	3	0	0	0	0	1	0
130	3	1	0	0	0	0	0	0
131	3	2	0	0	0	0	0	0
132	3	3	0	0	0	0	0	0
133	3	4	0	0	0	0	0	0
134	3	5	0	0	0	0	0	0
135	2	2	0	0	0	1	0	0
136	4	1	0	0	0	0	0	0
137	4	2	0	0	0	0	0	0
138	4	3	0	0	0	0	1	0
139	2	3	0	0	0	0	0	0
140	2	4	0	0	0	0	0	0
.	.	.	.	.	.	.	.	.
503	—	—	0	0	0	0	0	0
504	—	—	0	0	0	0	0	1
.	.	.	.	.	.	.	.	1

ваний за секунду ( $1/140 \cdot 10^{-6} = 7143$ ). Введением после АЦП конвейерного регистра эти временные потери вообще можно свести к нулю.

Передача информации по прямому доступу занимает только  $1 \cdot 100/(140+1) = 0,7\%$ , а организация прерываний —  $100 \cdot (31 \cdot 680) \cdot 10^{-6} = 2,1\%$  машинного времени микроЭВМ. Видно, что временные характеристики рассматриваемой системы близки к потенциально возможным для используемой аппаратуры.

**Пример микропрограммы.** Выше были описаны форматы микропрограмм и принципы выбора приоритетов в последовательности прямого доступа и прерываний. Кроме того, целесообразно выбрать периоды групп кратными тактовой частоте  $T_T$  счетчика микрокоманд и другим (меньшим) периодам. В этом случае обеспечивается строгая периодичность преобразования и обработки данных всех групп и минимальная емкость ПЗУ микрокоманд, в котором каналы групп с максимальным периодом записываются один раз, а каналы остальных групп — несколько раз в соответствии с их периодом.

В рассматриваемом примере выбираем периоды групп  $T_1 = 2^3 \cdot 3^2 \cdot 7 T_T = 504 T_T$ ;  $T_2 = 2 \cdot 3^2 \cdot 7 T_T = 126 T_T$ ;  $T_3 = 3^2 \times 7 = 63 \cdot T_T$ ;  $T_4 = 3^2 T_T$ . Это при  $T_T = 0,2$  мс соответствует требуемым частотам 9,9; 39,7; 79,4; 555,5 Гц.

В таблице показан фрагмент микропрограммы для числа каналов в группах соответственно 38, 20, 5 и 3. В поле канала АЦП вместо конкретного кода аналогового мультиплексора для наглядности приведены номер группы и порядковый номер канала в группе.

Для большей универсальности системы ПЗУ МК может быть заменено на ОЗУ микрокоманд, загружаемое программно от микроЭВМ. Сама микропрограмма может быть также вычислена на этой же ЭВМ по специальной программе.

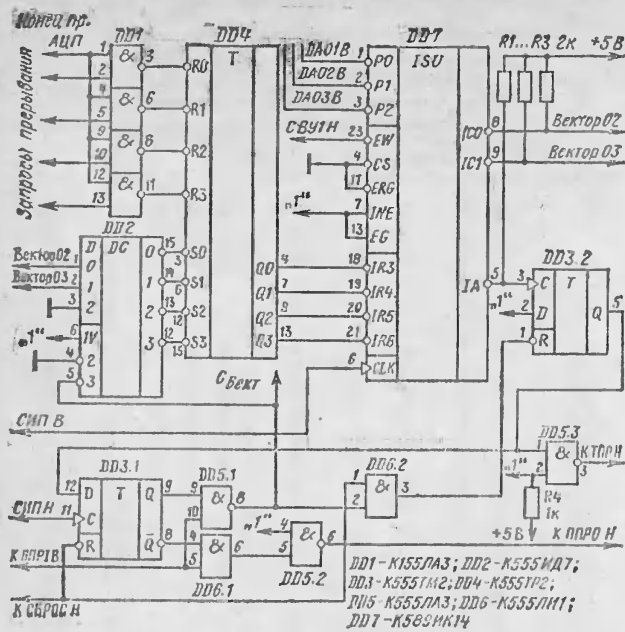


Рис. 2. Схема устройства прерываний

Электрическая схема. В структурной схеме (рис. 1) целесообразно рассмотреть только устройство прерываний и прямой доступа, поскольку остальные устройства собраны по типовым схемам.

На рис. 2 показана электрическая схема устройства прерываний. Текущий приоритет процессора записывается в БПП по стробу С ВУ1 с устройства программного обмена. Временная диаграмма устройства соответствует требованиям МПИ. При выборке импульса С<sub>вект</sub> уст-

ройство прямого доступа выставляет сигналы управления  $SOS1=10$  и мультиплексор (рис. 1) выдает в разряды (2...15) канала вектор прерываний, а устройство программного обмена (см. рис. 1) [2] — сигнал К СИПН.

Временная диаграмма устройства прямого доступа (рис. 3) соответствует требованиям МПИ. Мультиплексор (см. рис. 1) при управляющих сигналах  $SOS1=01$  выдает при прямом доступе в канал МПИ адрес  $A1...A15$  ячеек ОЗУ, а при  $SOS1=11$  — данные  $D0...D15$ , записываемые в эти ячейки. Обозначения канальных сигналов на схемах соответствуют [2]. Описываемые схемы могут быть построены и на интегральных БИС, например серии К588.

Структура программного обеспечения показала на рис. 4 для четырех групп входов в одного режима.

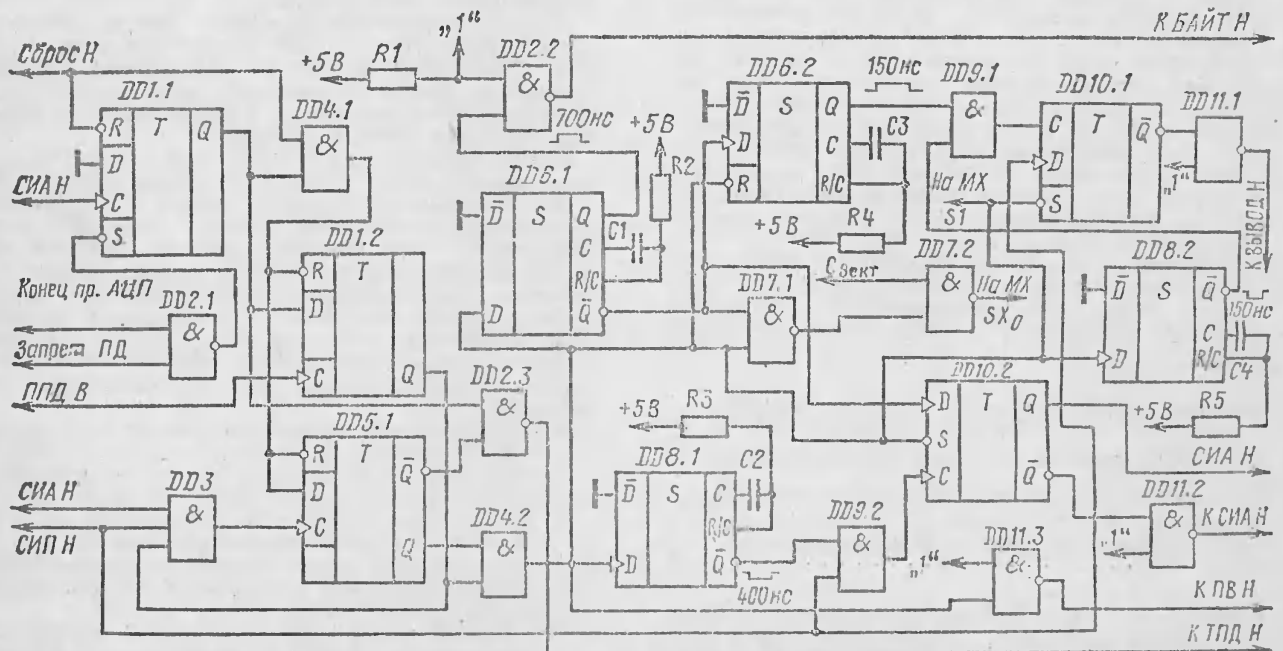
При включении ЭВМ по вектору 24 начинает выполняться подпрограмма START, которая может запускать тесты, вводить начальные условия в рекуррентные формулы цифровых фильтров. После этого подпрограмма разрешает все прерывания и переходит на непрерывное выполнение системы четырех встроенных с глубиной 3 прерываний (INT0...INT3). Высший приоритет имеет подпрограмма INT3. При разработке программы учитывалось следующее:

регистр текущего приоритета в БПП КР589ИК14 доступен только для записи;

после прерываний БПП самоблокируется до записи в него нового текущего приоритета;

запись в БПП текущего приоритета должна проходить при запрете прерываний процессора для исключения возможного сдвигания прерываний.

С увеличением числа групп и режимов соответственно растет число подпрограмм прерываний при сохранении общей структуры программного обеспечения. Тело каждой подпрограммы прерываний представляет собой программу обработки группы данных и выработки управляющих сигналов для отдельных контуров или их



DD1, DD5, DD10 — К155ТМ2; DD2, DD11 — К555ИП1; DD3, DD4, DD9 — К155ЛИ1; DD6, DD8 — К155АГ3; DD7 — К155ЛА3

Рис. 3. Схема прямого доступа

----- ЗАГОЛОВОК -----

АСЕМБЛЕР MACRO-11

```

;
;
; *****
;  DMAINT  -  СТРУКТУРА ПРОГРАМНОГО ОБЕСПЕЧЕНИЯ ОБРАБОТКИ ГРУПП ДАННЫХ,
;
;  ПОСТУПАЮЩИХ В ОЗУ ЭВМ ПО ПРЯМОМУ ДОСТУПУ, В РЕЖИМЕ 4-Х ВЛОЖЕННЫХ ПРЕ-
;
;  РЫВАНИЙ И ВЫРАБОТКИ УПРАВЛЯЮЩИХ СИГНАЛОВ ПРЯМОГО ЦИФРОВОГО УПРАВЛЕНИЯ
;
; *****

```

----- ОПРЕДЕЛЕНИЯ -----

```

PRIORT=160000 ;АДРЕС РЕГИСТРА ТЕКУЩЕГО ПРИОРИТЕТА В БПП K589IK14
;ТЕКУЩИЕ ПРИОРИТЕТЫ БПП
INTPR3=4 ;ВЫСШИЙ УРОВЕНЬ ПРИОРИТЕТА ПРЕРЫВАНИЯ
INTPR2=6
INTPR1=10
INTPR0=12 ;НИЗШИЙ УРОВЕНЬ ПРИОРИТЕТА ПРЕРЫВАНИЯ

PRIOR2: .BLKW 1 ;ЯЧЕЙКА ДЛЯ ДУБЛИРОВАНИЯ ТЕКУЩЕГО ПРИОРИТЕТА БПП

.ASECT ;АБСОЛЮТНАЯ СЕКЦИЯ, СОДЕРЖАЩАЯ ВЕКТОРЫ ПРЕРЫВАНИЙ
.=24 ;ВЕКТОР ПРЕРЫВАНИЯ ПО
VECST: .WORD START,200 ; ВКЛЮЧЕНИЮ ПИТАНИЯ
.=540
VEC0: .WORD INT0,200 ;VEC0...VEC3 - ВЕКТОРЫ ПРЕРЫВАНИЙ
VEC3: .WORD INT3,200 ;INT0...INT3 - СТАРТОВЫЕ АДРЕСА П/П ПРЕРЫВАНИЙ
VEC2: .WORD INT2,200 ;PS=200 - ЗАПРЕТ ПРЕРЫВАНИЙ ПРОЦЕССОРУ
VEC1: .WORD INT1,200
.PSECT

```

----- ОСНОВНАЯ ПРОГРАММА -----

```

* ПОДПРОГРАММА ОБРАБОТКИ ПРЕРЫВАНИЯ ПО ВКЛЮЧЕНИЮ ПИТАНИЯ

LC=
.=LC+1000 ;СМЕЩЕНИЕ РАБОЧЕЙ ПРОГРАММЫ ДЛЯ РЕЗЕРВИРОВАНИЯ
; ОБЛАСТИ ПЗУ ПОД ВЕКТОРЫ ПРЕРЫВАНИЙ
START: MOV #11000,SP ;НАЧАЛЬНАЯ ЗАГРУЗКА УКАЗАТЕЛЯ СТЕКА
;ПРЕРЫВАНИЯ ЗАПРЕЩЕНЫ ПО ВЕКТОРУ ПУСКА VECST
; (УСТАНОВКА НАЧАЛЬНЫХ УСЛОВИЙ)
;
; ТЕЛО ПОДПРОГРАММЫ
MOV #INTPR0,PRIOR2 ;ЗАПОМНИТЬ ПРИОРИТЕТ
MOV #INTPR0,@PRIORT ;РАЗРЕШЕНИЕ БПП ПРЕРЫВАНИЙ ВСЕХ УРОВНЕЙ
1X: MTPS #0 ;РАЗРЕШЕНИЕ ПРЕРЫВАНИЙ ПРОЦЕССОРУ И
BR 1X ; ОЖИДАНИЕ ПРЕРЫВАНИЙ С ГЕНЕРАЦИЕЙ
; Б КАНАЛЕ ЦИКЛА "ВВОД" ДЛЯ синхронизации БПП

```

Рис. 4,а. Структура программного обеспечения

частей, в частности реализацию цифровых фильтров с одинаковой частотой дискретизации.

В рассмотренном примере непосредственно для организации прерываний требуется 80 слов ПЗУ, для хранения массива данных ЦАП — 70 слов ОЗУ.

В подпрограммах обслуживания прерываний на основе поступивших по прямому доступу данных вычисляются управляющие воздействия на исполнительные устройства, подаваемые на ЦАП типа код — напряжение, код — временной интервал и др. Для обеспечения максимальной производительности ЭВМ и строгой периодичности выдачи управляющих воздействий на исполнительные устройства информация выводится в ЦАП по программному обмену в первую ступень буферного

регистра ЦАП и переписывается в его вторую ступень соответствующими сигналами запросов прерываний. В этом случае от исполнительных устройств возможны только однократные прерывания, например по ошибке устройства (см. рис. 1). Для сохранения периодичности управления приоритет однократных прерываний целесообразно установить ниже приоритета периодических прерываний.

Описанная структура была реализована на базе процессора M2 с использованием резидентного ОЗУ. Отметим, что организация прямого доступа по одному слову не нарушает процесс регенерации динамического ОЗУ [2]. Однако время регенерации необходимо учитывать при расчете времени выполнения программ.

```

; ПОДПРОГРАММА ОБРАБОТКИ ПЕРЕРЫВАНИЙ НИЗШЕГО УРОВНЯ
INT0:  MOV #INTPR0,PRIOR2 ;ЗАПОМИНАНИЕ ПРИОРИТЕТА БП
      MOV #INTPR0,@#PRIORT ;ЗАПИСЬ ПРИОРИТЕТА 0-ГО УРОВНЯ В БП
      MTPS #0 ;РАЗРЕШЕНИЕ ПЕРЕРЫВАНИЙ ПРОЦЕССОРУ
;ТЕЛО ПОДПРОГРАММЫ
      RTI

; ПОДПРОГРАММА ОБРАБОТКИ ПЕРЕРЫВАНИЯ 1-ГО УРОВНЯ
INT1:  MOV PRIOR2,-(SP) ;СОХРАНЕНИЕ ПЕРЕРВАННОГО ПРИОРИТЕТА БП
      MOV #INTPR1,PRIOR2 ;ЗАПОМИНАНИЕ НОВОГО ПРИОРИТЕТА
      MOV #INTPR1,@#PRIORT ;ЗАПИСЬ НОВОГО ПРИОРИТЕТА
      MTPS #0 ;РАЗРЕШЕНИЕ ПЕРЕРЫВАНИЙ ПРОЦЕССОРУ
; СОХРАНЕНИЕ В СТЕКЕ РЕГИСТРОВ ОБЩЕГО НАЗНАЧЕНИЯ,ИСПОЛЬЗУЕМЫХ В П/П
; ТЕЛО ПОДПРОГРАММЫ
; ВОССТАНОВЛЕНИЕ ЗНАЧЕНИЙ РОН
      MTPS #200 ;ЗАПРЕТ ПЕРЕРЫВАНИЙ ПРОЦЕССОРУ
      MOV (SP)+,PRIOR2 ;ВОССТАНОВЛЕНИЕ ПЕРЕРВАННОГО УРОВНЯ ПРИОРИТЕТА
      MOV (SP)+,@#PRIORT ; И ЗАПИСЬ ЕГО В БП
      MTPS #0 ;РАЗРЕШЕНИЕ ПЕРЕРЫВАНИЙ ПРОЦЕССОРУ
      RTI

; ПОДПРОГРАММА ОБРАБОТКИ ПЕРЕРЫВАНИЙ 2-ГО УРОВНЯ
INT2:  MOV PRIOR2,-(SP) ;СОХРАНЕНИЕ ПЕРЕРВАННОГО ПРИОРИТЕТА БП
      MOV #INTPR2,PRIOR2 ;ЗАПОМИНАНИЕ НОВОГО ПРИОРИТЕТА
      MOV #INTPR2,@#PRIORT ;ЗАПИСЬ НОВОГО ПРИОРИТЕТА
      MTPS #0 ;РАЗРЕШЕНИЕ ПЕРЕРЫВАНИЙ ПРОЦЕССОРУ
; СОХРАНЕНИЕ В СТЕКЕ РЕГИСТРОВ ОБЩЕГО НАЗНАЧЕНИЯ,ИСПОЛЬЗУЕМЫХ В П/П
; ТЕЛО ПОДПРОГРАММЫ
; ВОССТАНОВЛЕНИЕ ЗНАЧЕНИЙ РОН
      MTPS #200 ;ЗАПРЕТ ПЕРЕРЫВАНИЙ ПРОЦЕССОРУ
      MOV (SP)+,PRIOR2 ;ВОССТАНОВЛЕНИЕ ПЕРЕРВАННОГО УРОВНЯ ПРИОРИТЕТА
      MOV (SP)+,@#PRIORT ; И ЗАПИСЬ ЕГО В БП
      MTPS #0 ;РАЗРЕШЕНИЕ ПЕРЕРЫВАНИЙ ПРОЦЕССОРУ
      RTI

; ПОДПРОГРАММА ОБРАБОТКИ ПЕРЕРЫВАНИЯ ВЫСШЕГО УРОВНЯ
INT3:  СОХРАНЕНИЕ В СТЕКЕ РОН,ИСПОЛЬЗУЕМЫХ В ПОДПРОГРАММЕ
; ТЕЛО ПОДПРОГРАММЫ
; ВОССТАНОВЛЕНИЕ РОН
      MOV PRIOR2,@#PRIORT ;ДУБЛИРОВАНИЕ ПЕРЕРВАННОГО ПРИОРИТЕТА ДЛЯ
      MTPS #0 ; РАЗБЛОКИРОВКИ БП И РАЗРЕШЕНИЕ ПЕРЕРЫВАНИЙ
      RTI ; ПРОЦЕССОРУ
      .END

```

Рис. 4.6. Структура программного обеспечения

Наиболее эффективные программы обработки прерываний могут быть написаны на ассемблере. Из языков высокого уровня, входящих в программное обеспечение микроЭВМ «Электроника 60», только «Модуль-2» [3] позволяет непосредственно программировать прерывание.

Можно использовать для работы с прерываниями более распространенные языки (Фортран, Паскаль), если включить в программу фрагменты на ассемблере, аналогичные представленным на рис. 4.

Пример такой программы для версии Паскаль / ФО-ДОС приведен в работе [4].

Телефон 75-07-69, Тула

Статья поступила 12.05.87

#### ЛИТЕРАТУРА

1. Щелкунов Н. Н., Днанов А. А. Программирование микросистем реального времени // Микропроцессорные средства и системы. — 1985. — № 4. — С. 31—35.
2. Центральный процессор М2. Техническое описание и инструкция по эксплуатации 3.858.382 ТО.
3. Вирт Н. Модуль-2. — В кн.: Языки программирования. — М.: Наука, 1985.
4. Бутомо И. Д., Самочадип А. В., Усанова Д. В. Программирование на алгоритмическом языке Паскаль для микроЭВМ. — Л.: Изд-во ЛГУ, 1985.

# УСТРОЙСТВО ВВОДА-ВЫВОДА АНАЛОГОВЫХ СИГНАЛОВ МИКРОЭВМ

Разработка УСО для локальных программируемых контроллеров — одна из актуальных задач в области проектирования современных микропроцессорных систем управления. Устройство является частью контроллера, но может быть выделено в отдельный блок.

меньше времени установления процессов в объекте управления, поэтому задержка переключения не влияет на процесс регулирования.

Аналоговые сигналы через замкнутые контакты 1 и 2 реле K2, K4, K6, K8 поступают на входы устройства выборки и хранения (УВХ) DA1...

опроса и преобразования входных сигналов); УВХ стробируются сигналом Лог. 1 по входам 8. Строб формируется в результате инвертирования (схемой DD1.1) сигнала низкого уровня с выхода 1 дешифратора DD2. Этот сигнал возникает при комбинации 0H на выходе портов P24...P27. Аналоговые сигналы, зафиксированные в УВХ, поступают на вход АЦП при замыкании контактов 2 и 3 реле K1, K3, K5, K7.

Последовательность подключения определяется программой контроллера. Сигналы для управления реле записываются в регистр DD10 по стробу с выхода 1 конъюнктора DD3. Строб возникает при совпадении сигнала записи WR и сигнала с выхода 9 DD2 (при комбинации 8H портов P24...P27). При записи Лог. 0 в соответствующий разряд регистра DD10 реле срабатывает: для K1 — в первый разряд, K2 — во второй, K5 — в тре-

### Технические характеристики

Число аналоговых входов	4
Диапазон изменения входного сигнала, В	0...5,12
Дискретность измерения входного сигнала, мВ	1,25
Погрешность измерения, мВ	0,625
Время измерения входного сигнала, мкс, не более	130
Число аналоговых выходов	3
Диапазон регулирования, В	0...5,12
Дискретность, мВ	20
Точность установки выходного сигнала, мВ	±10
Время установки выходного сигнала, мкс	±5

Для повышения помехозащищенности по входным цепям применен метод программного формирования среднего значения. Усредняется значение по 256 измерениям. В этом случае время измерения составляет 33,33 мс

Неискаженный прием столь малых сигналов возможен при использовании герконовых реле. Время переключения реле РЭС-55 существует

...DA4 (рис. 1). Устройства одновременно фиксируют уровни входных сигналов и хранят их более 10 мс (этого времени достаточно для

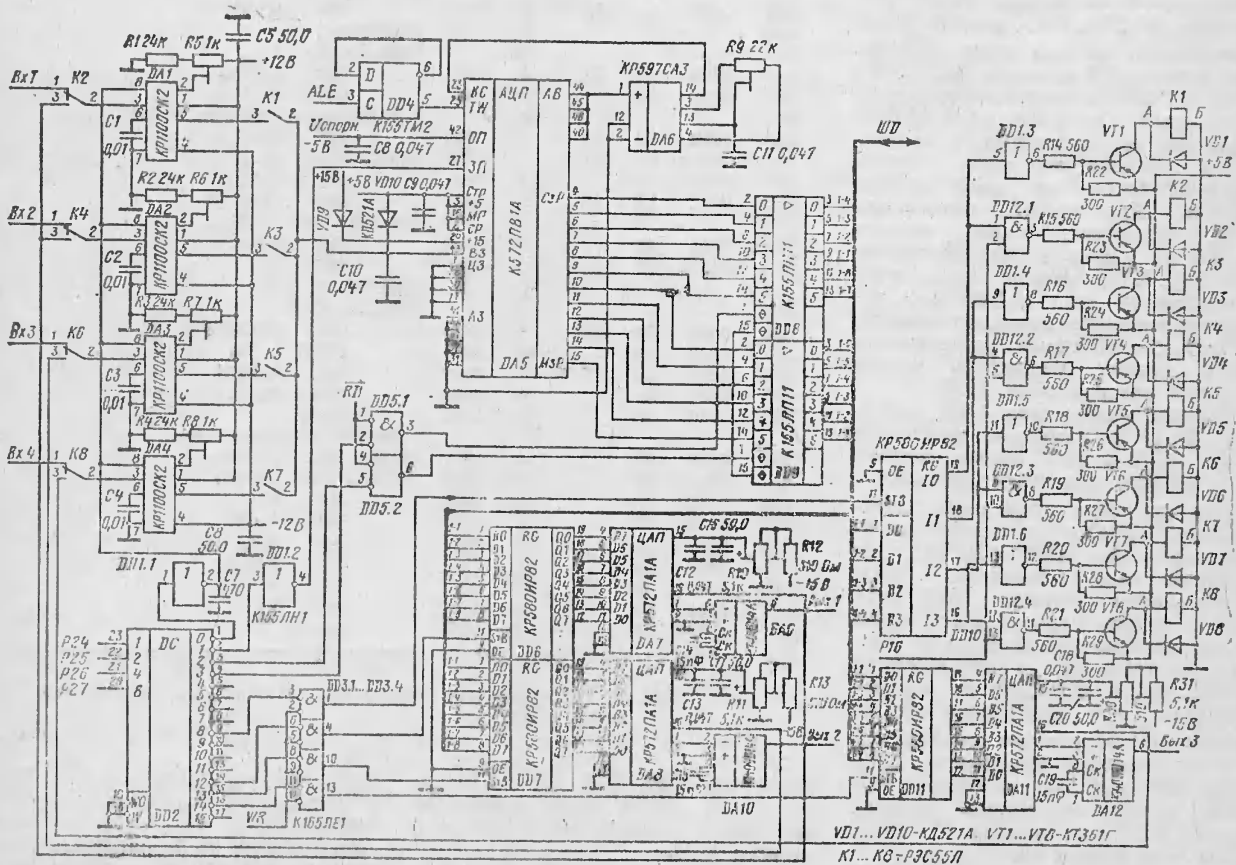


Рис. 1. Принципиальная схема УСО

тий, K7 — в четвертый. На выходе порта P16 программно устанавливается Лог. 0, запрещающий работу конъюнкторов DD12. Реле K2, K4, K6, K8 используются в режиме внутренней диагностики аналоговых каналов, но об этом ниже.

Запуск АЦП производится импульсом положительной полярности, полученным после инвертирования схемой DD12 сигнала низкого уровня с выхода 2 DD2 (при комбинации ИН P24...P27) через время, превышающее «дребезг» сработавшего реле (более 3 мс). Длительность импульса запуска 10...12 мкс. ОЭВМ переходит в режим ожидания. Тактовые импульсы формируются триггером DD4.1 из сигнала ALE ОЭВМ. При использовании в ОЭВМ кварцевого резонатора на 6 МГц период тактовых импульсов АЦП равен 5 мкс, время преобразования АЦП составит 120 мкс, затем ОЭВМ считывает результат преобразования, подключая шины данных АЦП к шине данных ОЭВМ по сигналу чтения RD. Оно осуществляется в два приема: вначале восемь младших битов, затем четыре старших. Роль коммутатора данных выполняют схемы DD8, DD9. Для первого считывания ОЭВМ формирует сигнал низкого уровня на выходе 3 DD2 (состоящие 2H P24...P27), разрешающий прохождение сигнала чтения RD через схему DD5.2 на входы 1 и 15 DD9 и вход 15 DD8. По этому сигналу все выходы схемы DD9 и выходы 11, 13 DD8 переходят из высокоомного состояния в активное, передавая восемь младших битов на шину данных ОЭВМ. Для второго считывания ОЭВМ формирует сигнал низкого уровня на выходе 4 DD2 (состоящие 3H P24...P27), сигнал коммутации поступает на вход 1 DD8 и в активное состояние переходят выходы 3, 5, 7, 9 этой схемы, передавая на шину данных микроЭВМ четыре старших разряда кода АЦП.

Опорное напряжение АЦП выбрано — 5, 12 В. Разрешающая способность АЦП в этом случае составит 1,25 мВ (вес младшего разряда).

Для генерации аналоговых управляющих сигналов применены интегральные схемы ЦАП К527ПА1, для хранения кодов управления ЦАП используются регистры DD6, DD7, DD11. Запись управляющих кодов в эти регистры производится по стробам, поступающим с конъюнкторов DD3.2...DD3.4. Каждый из стробов возникает при совпадении сигнала записи WR и сигналов с соответствующих выходов дешифратора DD2: строб DD6 при наличии сигнала низкого уровня на выходе 14 DD2; строб DD7 — на выходе 15; строб DD11 — на выходе 16.

Опорное напряжение ЦАП (—5, 12 В) устанавливается резисторами грубой (R10, R12, R30) и точной

914	0493	BF	00	4	0083	RED: MOV R7,#0 ;считываю усреднения
915				0		*запрет тестирования; коммутации каналов
916	0495	FE		4	0085	FOTO: MOV A,R6
917	0486	8F	B3	4	0086	ORL P1,#10111000B
918	0488	9F	BF	4	0088	ANL P1,#10111111B
919	048A	94	4D	4	008A	CALL 04KC
920				0		*считываем значение находится во внутр.
921				0		*ОЗУ; ячейки-26,27H
922				0		*запрет обращения к другим устройствам.
923	048C	9A	0F	4	008C	ANL P2,#FH
924				0		*РАЗРЕШЕНИЕ ОБРАЩЕНИЯ К ВНЕШНЕМУ ОЗУ.
925	048E	9F	7F	4	008E	ANL P1,#01111111B
926				0		*R00:= адрес яч.10
927				0		*R01:= адрес яч.27H
928	0490	B3	00	4	0090	MOV R0,#0
929	0492	B9	27	4	0092	MOV R1,#27
930	0494	97		4	0094	CLR C
931	0495	94	E9	4	0095	CALL CFEFH
932	0497	EA	00	4	0097	MOV R2,#0
933	0499	80		4	0099	MOVX A,#R0
934	049A	7A		4	009A	ADDC A,R2
935	049B	90		4	009B	MOVX @R0,A
936	049C	EF	85	4	009C	DJNZ R7,FOTO
937	049E	93		4	009E	RETR
						*****
850	044D	8A	8F	4	004D	ФКК: ORL P2,#2FH
851	044F	9A	8F	4	004F	ANL P2,#8FH
852	0451	90		4	0051	MOVX @R0,A
853				0		*защита от дребезга реле 2х1,5 мс
854	0452	E8	52	4	0052	f1: DJNZ R0,f1
855	0454	E8	54	4	0054	f2: DJNZ R0,f2
856				0		*DD2-10:=1
857				0		*DD2-01:=0
858	0456	9A	0F	4	0056	ANL P2,#0FH
859				0		*DD2-01:=1
860				0		*DD2-02:=0
861	0458	8A	1F	4	0058	ORL P2,#1FH
862				0		*DD2-02:=1
863				0		*DD2-03:=0
864	045A	23	2F	4	005A	MOV A,#2FH
865	045C	3A		4	005C	CJTL P2,A
866				0		* задержка считывания 130мкс.
867	045D	B3	1A	4	005D	MOV R0,#26
868	045F	E8	5F	4	005F	test1:DJNZ R0,test11
869	0461	80		4	0061	MOVX A,#R0
870	0462	B9	27	4	0062	MOV R1,#27H
871				0		*младший байт в @R1.
872	0464	A1		4	0064	MOV @R1,A
873	0465	C9		4	0065	DEC R1
874				0		*DD2-03:=1
875				0		*DD2-04:=0
876	0466	8A	3F	4	0066	ORL P2,#3FH
877	0468	80		4	0068	MOVX A,#R0
878	0469	53	0F	4	0069	ANL A,#0FH
879	046B	A1		4	006B	MOV @R1,A
880	046C	93		4	006C	RETR
						*****
994	04E9	94	E9	4	00E9	СРЕДН: CALL CFEFH
995	04EB	80		4	00EB	СРЕДН1: MOVX A,#R0
996	04EC	AA		4	00EC	MOV R2,A
997	04ED	F1		4	00ED	MOV A,@R1
998	04EE	7A		4	00EE	ADDC A,R2
999	04EF	90		4	00EF	MOVX @R0,A
1000	04F0	18		4	00F0	INC R0
1001	04F1	C9		4	00F1	DEC R1
1002	04F2	83		4	00F2	RET

Рис. 2. Подпрограмма обработки аналогового сигнала

установки (R12, R13, R31). Выходные сигналы ЦАП используются для проведения диагностики аналоговых каналов УСО.

В режиме диагностики на вход исследуемого канала подключается выход одного из ЦАП. Для этого

порт P16 устанавливается в Лог. 1 и тем самым снимается запрет с конъюнкторов DD12. После записи Лог. 0 в один из разрядов DD10, наряду со срабатыванием реле с нечетным номером, срабатывает реле с четным номером, т. е. при диагности-

638				0	*при наличии кода "8a" вывести уставки.
639	0332	23	8A	3	0032 AN6: MOV A,#8AH
640	0334	DE		3	0034 XRL A,R6
641	0335	96	39	3	0035 JNZ AN7
642	0337	84	8B	3	0037 JMP MYW
949				0	Выход уставки в канал.
950	04AB	99	EB	4	00AB MYW: ANL P1,#10111011B
951	04AA	89	80	4	00AA ORL P1,#80H
952				0	*отличить все напряжения управления.
953	04AC	27		4	00AC CLR A
954	04AD	8A	FF	4	00AD ORL P2,#FFH
955	04AF	90		4	00AF MOVX @R0,A
956	04B0	9A	EF	4	00B0 ANL P2,#EFH
957	04E2	90		4	00E2 MOVX @R0,A
958				0	*загрузить период воздействия.
959	04B3	B9	32	4	00B3 MOV R1,#32H
960	04B5	B4	02	4	00B5 CALL zaust
961	04B7	94	F3	4	00B7 CALL inter
962				0	*загрузить величину воздействия.
963	04B9	8A	DF	4	00B9 ORL P2,#DFH
964	04BB	9A	DF	4	00BB ANL P2,#DFH
965	04BD	B9	30	4	00BD MOV R1,#30H
966	04BF	94	9F	4	00BF CALL ustaw
967	04C1	A5		4	00C1 CLR F1
968	04C2	B5		4	00C2 CPL F1
969	04C3	89	BC	4	00C3 idet: ORL P1,#10111100B
970				0	*считать содержимое таймера
971	04C5	9A	4F	4	00C5 ANL P2,#4FH
972	04C7	B3	03	4	00C7 MOV R0,#3
973	04C9	B9	39	4	00C9 MOV R1,#39H
974	04CB	23	80	4	00CB MOV A,#10000000B
975	04CD	90		4	00CD MOVX @R0,A
976	04CE	C8		4	00CE DEC R0
977	04CF	80		4	00CF MOVX A,@R0
978	04D0	A1		4	00D0 MOV @R1,A
979	04D1	C9		4	00D1 DEC R1
980	04D2	80		4	00D2 MOVX A,@R0
981	04D3	A1		4	00D3 MOV @R1,A
982	04D4	BD	DF	4	00D4 MOV RS,#04
983	04D6	23	38	4	00D6 MOV A,#38H
984	04D8	BE	80	4	00D8 MOV R6,#80H
985	04DA	BF	03	4	00DA MOV R7,#3
986	04DC	74	67	4	00DC CALL ANA
987	04DE	05		4	00DE EN I
988	04DF	76	C3	4	00DF JF1 idet
989	04E1	99	B8	4	00E1 ANL P1,#10111000B
990	04E3	9A	DF	4	00E3 ANL P2,#DFH
991	04E5	27		4	00E5 CLR A
992	04E6	90		4	00E6 MOVX @R0,A
993	04E7	64	BB	4	00E7 JMP prokon ;процесс окончен
938				0	*сдвиг слова на 1 разряд вправо
939				0	Выход 8-и младших разрядов уставки.
940	049F	F1		4	009F ustaw: MOV A,@R1
941	04A0	97		4	00A0 CLR C
942	04A1	67		4	00A1 RRC A
943	04A2	2A		4	00A2 XCH A,R2
944	04A3	19		4	00A3 INC R1
945	04A4	F1		4	00A4 MOV A,@R1
946	04A5	67		4	00A5 RRC A
947	04A6	90		4	00A6 MOVX @R0,A
948	04A7	93		4	00A7 RETR

Рис. 3. Подпрограмма формирования управляющих сигналов

ровании первого канала включаются K1 и K2, второго — K3 и K4, третьего — K5 и K6, четвертого — K7 и K8. При такой коммутации на входы первого и второго каналов будет подано напряжение ЦАП DA7, третьего канала — DA8, четвертого — DA11. После преобразования ском-

мутированного сигнала производится сравнение восьми старших разрядов кода с АЦП и кода, записанного в соответствующий ЦАП. Если разница превышает предварительно заданную величину, результат рассматривается как нарушение работы аналогового канала, и конт-

УДК 681.327

А. В. Токмачев

## ЭМУЛЯТОР ТЕРМИНАЛА ДЛЯ ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА ROBOTRON 1715

На ПК ROBOTRON 1715 поддерживается одна из версий операционной системы CP/M. Хорошая аппаратная реализация (микропроцессорный комплект U880 с гибкими дисками емкостью до 800 Кбайт и программно-управляемым печатающим устройством с широким набором шрифтов), большой пакет программного обеспечения позволяют решать сложные математические, экономические, бухгалтерские и другие задачи.

Предложено использовать незадействованную часть периферийного последовательного порта U856, выходы которого через согласующие схемы подключены к разъему 24 на задней стороне ПК для сопряжения ROBOTRON 1715 с другими ЭВМ (например, СМ ЭВМ) в качестве терминала.

Программа «Эмулятор терминала» написана на ассемблере Z80, занимает в ОЗУ менее 4 Кбайт и позволяет заменять следующие типы терминалов: VDT52100, MERA7253, BTA20000.15.

Характеристики программы:

работает с внешней линией на прием информации по прерываниям;

обслуживает основные ESC-последовательности (непосредственную адресацию маркера, режим ALTERNATE KEYPAD MODE, режим HOLD SCREEN, режим AUTO COPY и др.);

позволяет с клавиатуры управлять режимом получения жесткой копии экрана;

обеспечивает настройку физических параметров обмена по линии (скорость, число битов данных, паритет, число стоповых битов);

позволяет с помощью клавиатуры выйти из режима эмуляции терминала в автономный режим.

Применение «Эмулятора терминала» позволило с помощью рабочих мест на базе ПК ROBOTRON 1715 работать в операционных системах ОС PB, РАФОС и др. (СМ ЭВМ) и CP/M (ROBOTRON 1715).

Ведутся работы по включению в программу возможностей обмена двоичными и текстовыми файлами между CP/M (ROBOTRON 1715) и ОС PB (CM1420).

Сообщение поступило 18.01.88

Рис. 4 Подпрограмма диагностики

664					0	*при наличии кода "89" тестировать как
665	035A	23	89	3	005A	AN10: MOV A, #89H
666	035C	DE		3	005C	XRL A, R6
667	035D	96	65	3	005D	JNZ AN11
668	035F	BF	F7	3	005F	MOV R7, #1111011B
669	0361	BE	DF	3	0061	MOV R6, #1FH ;adres cap
670	0363	84	6D	3	0063	JMP test2
671	0365	04	97	3	0065	AN11: JMP wzp32
672				0		*****
881				0		*****
882				0		* тест2 настройка канала *
883				0		*****
884	046D	B9	27	4	006D	test2: MOV R1, #27H
885				0		*адрес ячейки хранения
886				0		*считанной информации (26, 27)
887	046F	23	F8	4	006F	MOV A, #11111000B
888	0471	39		4	0071	OUTL P1, A
889				0		*P2:= адрес ЦАП
890	0472	FE		4	0072	MOV A, R6
891	0473	3A		4	0073	OUTL P2, A
892	0474	23	AA	4	0074	MOV A, #AAH
893	0476	90		4	0076	MOVX @R0, A
894				0		*A:= ключ коммутатора.
895	0477	FF		4	0077	MOV A, R7
896	0478	94	4D	4	0078	CALL ФМКС
897	047A	04	8F	4	007A	JMP test2

Телефон: 468-72-12, Москва

Статья поступила 19.08.87

УДК 681.327

В. П. Омельченко, С. И. Семенов, С. И. Трегубов

## УСТРОЙСТВО ВВОДА МНОГОКАНАЛЬНОЙ ИНФОРМАЦИИ В МИКРОЭВМ «ЭЛЕКТРОНИКА 60»

При обработке электрофизиологической информации, снимаемой непосредственно с человека, на микроЭВМ «Электроника 60», кроме выполнения аналого-цифрового преобразования многоканального сигнала и ввода его в ЭВМ, необходима электробезопасность пациента. Простое гальваническое соединение выходных цепей медицинского прибора с узлами ЭВМ не допустимо. Решить эти задачи позволяет использование устройства сбора аналоговых данных «Электроника МС 8201» (УСАД), которое обеспечивает многоканальное преобразование аналоговых сигналов в цифровой код, передачу их по оптоволоконному тракту и выдачу в виде 16-разрядного параллельного цифрового кода в TTL-уровнях. Под номер канала отводятся 4 разряда, остальные 12 — информационные (последовательно опрашиваются 16 аналоговых каналов, время преобразования по одному каналу — 40 мкс).

Вести большой массив данных в микроЭВМ «Электроника 60» с такой скоростью невозможно, так как каждые 2 мс необходимо регенерировать память, которая длится 150 мкс. Однако во многих случаях такая скорость получения данных не нужна. Например, при анализе электрокардиограмм и электроэнцефалограмм используются частоты выборки сигнала 256 и 128 Гц. В предложенном устройстве сопряжения блок памяти запоминает данные из всех 16 каналов за один цикл опроса, затем несколько циклов пропускает (см. рисунок).

Информационный код из УСАД поступает в блок памяти (D12...D14) и записывается в ячейку. Запись разрешается положительным выходным импульсом формирователя цикла записи (выход 6 D22.1). Инвертированный импульс цикла записи (выход 5 D22.1) включает формирователь коротких отрицательных импульсов записи (D21.1...D3). Они соответствуют пе-

репадам напряжения в младшем разряде кода адреса, поступающего из УСАД.

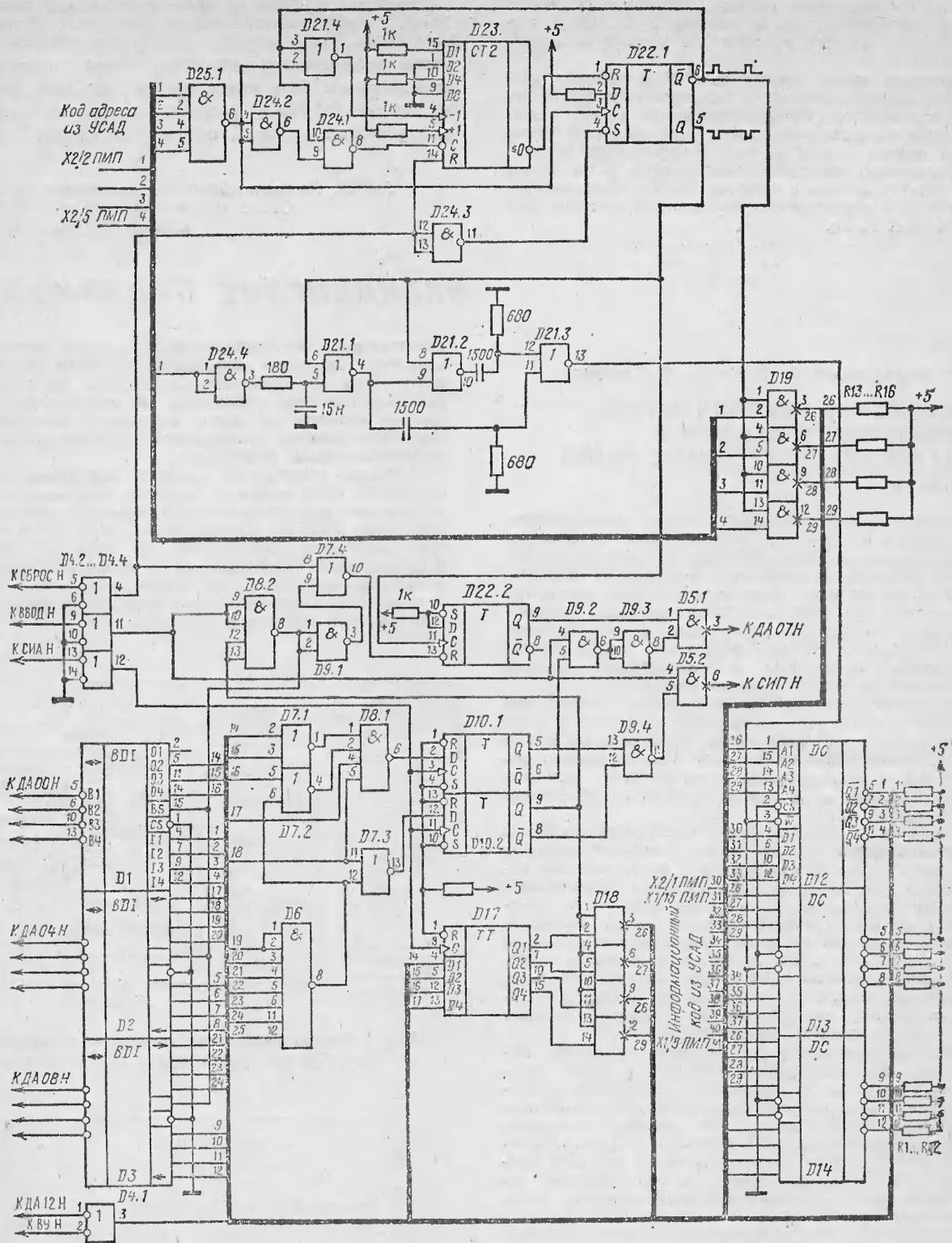
Отрицательные синхронимпульсы с выхода дешифратора адреса D25.1 через D21.4 поступают на счетчик D23. На его информационных входах с помощью переключек установлено число циклов преобразования (N), которое необходимо пропустить. Например, если N=5, то в ЭВМ будет поступать каждый шестой цикл и, следовательно, частота выборки исследуемого сигнала окажется около 260 Гц.

Фронт последнего синхронимпульса обнуляет счетчик и на выходе появляется нулевой потенциал, сбрасывающий триггер D22.1. Высокий уровень триггера на выходе 6 разрешает подачу кода адреса из УСАД на микросхемы памяти и открывает ключ D24.1: Следующий синхронимпульс, поступая через этот ключ на вход С счетчика D23, опять фиксирует число циклов преобразования, заданное переключками, на выходе счетчика появляется высокий потенциал и по фронту синхронимпульса передается «1» в триггер D22.1. Таким образом, на выходе 5 этого триггера за один цикл опроса каналов УСАД формируется отрицательный импульс, разрешающий работу формирователя импульсов записи. Фронт этого импульса записывает «1» (наличие данных в памяти устройства и готовность ввода в ЭВМ) в 07 разряд регистра состояний устройства сопряжения (микросхема D22.2).

Обмен данными с микроЭВМ осуществляется через двунаправленные шинные формирователи D1...D3. Режим вывода адреса из процессора обеспечивает высокий уровень на входах BS формирователей.

Селектор адресов собран на элементах D6, D7.1...D3, D8.1. Физический адрес регистра состояний 177760. При обращении процессора к этому регистру в триггер D10.1 по сигналу К СИА Н записывается «0». Адреса от 177700 до 177736 присвоены 16 ячейкам памяти. При обращении к ним в триггер D10.2 заносится «1», а в регистр D17 — 4 разряда адреса, образующие номер ячейки памяти. Высокий уровень сигнала на выходе триггера D10.2 открывает адресные ключи D18 и разрешает прохождение сигнала ВВОД через ключ D8.2 к шинным формирователям D1...D3, а также сбрасывает триггер готовности D22.2. Таким образом, триггер готовности устанавливается один раз после цикла преобразования, а сбрасывается при вводе пер-





Принципальная схема платы сопряжения микроЭВМ «Электроника 60» с устройством сбора аналоговых данных «Электроника МС 8201»:

- D1...D3 — К589АП26; D4 — К559ИП2; D5, D18, D19 — К599ИП1; D6 — К155ЛА2; D7, D21 — К155ЛЕ1; D8, D25 — К155ЛА6; D9, D24 — К155ЛА3; D10, D22 — К155ТМ2; D12...D14 — К155РУ2; D17 — К155ТМ8; D23 — К155НЕ7; R1...R16 — 2,7 КОМ

вого из опрашиваемых каналов. Канальный сигнал К СИП Н формируется из сигнала К ВВОД Н при обращении к одному из регистров устройства сопряжения.

Программа ввода данных из УСАД работает следующим образом: процессор постоянно опрашивает регистр состояния устройства сопряжения 177760 и при появлении «1» в его разряде 07 (после окончания цикла записи данных в память) вводит информацию в ОЗУ. Быстродействие микроЭВМ «Электроника 60М» позволяет вводить данные с частотой 780 Гц. Нижний предел скорости определяется разрядностью счетчика D23 и составляет 90 Гц.

Устройство собрано на плате стандартного размера 135×240 мм и входит в каркас микроЭВМ «Электроника 60М». Кроме использования при медико-биологических исследованиях устройство сбора аналоговых данных может быть применено для передачи данных на большие (до 300 м) расстояния в условиях сильных электромагнитных помех, например в заводском цехе.

344718, Ростов-на-Дону, Нахичеванский пер., 29,  
Отдел медэлектроники; тел. 65-31-02  
Статья поступила 19.05.87

УДК 616.1.072

Ф. Ф. Водоватов, В. В. Панов, А. В. Сорокин

## ПРИМЕНЕНИЕ ПЕРСОНАЛЬНОГО КОМПЬЮТЕРА В МАССОВОЙ ДИАГНОСТИКЕ И ТРЕНИНГЕ РИТМА СЕРДЦА

С развитием вычислительной техники, автоматизации производства и введением компьютерного обучения резко увеличился поток информации в сфере производства и обучения. Восприятие и переработка информационного потока могут быть оптимальными лишь в случае нормального функционального состояния организма человека. Трудности, возникающие при преодолении информационного потока, формируют эмоциональное напряжение, приводящее к вегетативным сдвигам в организме, не контролируемым сознанием.

Возрастание степени эмоционального напряжения приводит к срыву высшей нервной деятельности и соответственно к расогласованию механизма регуляции произвольных (автоматических) физиологических реакций [1], а в дальнейшем к необратимым патологическим явлениям и, в первую очередь, в сердечно-сосудистой системе.

Регистрация и преобразование электробиоинформации о произвольных физиологических реакциях, вывод ее на экран с помощью компьютера и предоставление видеoinформации человеку предполагают подключение сознания на более ранних стадиях напряжения, диагностики и саморегуляции функционального состояния сердечно-сосудистой системы внешней обратной связью через зрительный канал. За методом обучения (самоуправления) произвольными физиологическими реакциями с помощью электронных систем закрепился термин «адаптивная регуляция», «биологическая обратная связь» (БОС) или БОС-тренинг [2].

Ритм сердца — наиболее изученный показатель деятельности сердечно-сосудистой системы и общей реакции организма на эмоциональное напряжение, возникающее при восприятии и переработке информационного потока. Можно считать, что любому заданному уровню функционирования организма соответствует эквивалентный уровень функционирования аппарата кровообращения и, следовательно, частоты сердечных сокращений (ЧСС). Поэтому конечный результат самоуправления ритмом сердца с помощью БОС (помимо предупреждения «информационного невроза») — обеспечение заданного уровня функционального состояния организма человека.

Для исследования вопросов, связанных с саморегуляцией ритма сердца, была создана экспериментальная установка (рис. 1).

Электрокардиосигнал (ЭКГ-сигнал) усиливается кардиографом, преобразуется и формируется в прямоуголь-

ные импульсы, поступающие на вход порта микроЭВМ с частотой сердечных сокращений. На экран выводится информация об изменении длительности кардиоинтервала. Программное обеспечение и экспериментальные данные хранятся на ленте кассетного магнитофона. Распечатка данных производится графопроектором и цифронечающим устройством.

Измерение зависимости разности потенциалов электрического поля сердца от времени производится отечественным портативным одноканальным электрокардиографом ЭКГ-04. Сигнал на его вход поступает от трех электродов, наложенных на руки обучаемого и правую ногу. Усилитель состоит из предварительного усилителя, собранного по дифференциальной схеме, и усилителя постоянного тока (электродвигатель лентопротяжного механизма отключен).

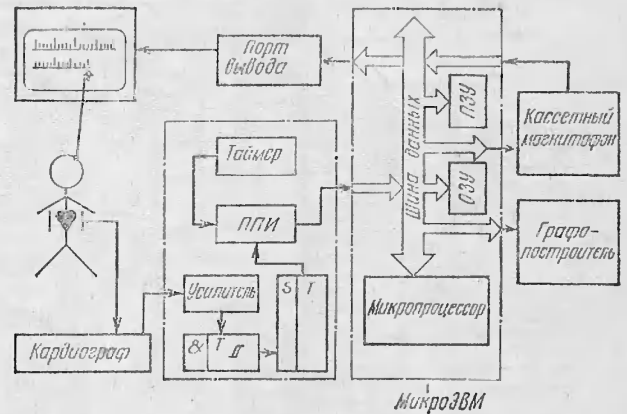


Рис. 1. Структурная схема установки диагностики и саморегуляции ритма сердца БОС-тренингом

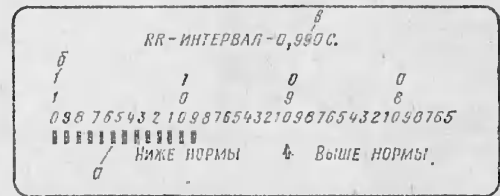


Рис. 2. Изображение изменения длительности кардиоинтервала на экране дисплея:

а — линия изменения кардиоинтервала; б — шкала длительности сердечного цикла; в — текущее значение кардиоинтервала

На вход интерфейса с кардиографа поступает ЭКГ-сигнал, усиленный, а затем формирующий триггером Шмитта (Т) в прямоугольные импульсы с частотой сокращения миокарда. С триггера Т сигнал поступает на stroбирующий вход счетного ST-триггера. Его инверсный выход связан с информационным входом, а на прямом выходе формируется прямоугольный сигнал, длительность которого соответствует периоду сокращения миокарда.

Для ввода сигналов с выхода счетного ST-триггера в микроЭВМ служит БИС программируемого параллельного интерфейса (ППИ) [3].

Измерение временных интервалов ЭКГ-сигналов обеспечивается подсчетом импульсов тактовой частоты, поступающих на вход таймера. Линию изменения кардиоинтервала запускает срез прямоугольного импульса, синхронизированный появлением ЭКГ-сигнала (рис. 2). Одновременно на экране регистрируется текущее значение кардиоинтервала (RR) в цифрах.

Обработка данных изменения кардиоритма во время обучения производится серийно выпускаемой бытовой микроЭВМ «Микроша» [3] (ОЗУ — 32 Кбайта, ПЗУ — 2 Кбайта).

Программное обеспечение установки: программа измерения временного интервала ЭКГ-сигнала миокарда написана на ассемблере (рис. 3);

программа обработки набранных статистических данных и выдачи итоговых графиков (интервалограммы и гистограммы) — на языке БЕЙСИК (рис. 4). Эта программа позволяет также записать статистические данные на магнитную ленту и считать их.

Первая программа состоит из нескольких подпрограмм: формирования динамически меняющегося изображения шкал измеряемых величин; измерения интервала ЭКГ-сигнала миокарда; выдачи числовых значений на экран.

#### Методика проведения эксперимента

Обучающийся помещается в звукоизолированную комнату и располагается в удобном кресле перед дисплеем установки. Обратная связь замыкается зрительным восприятием изменения длительности кардиоинтервала в реальном времени в виде изменяющейся по длине горизонтальной линии (рис. 2). Обучаемый инструктируется о необходимости удерживать последнюю вертикальную полосу горизонтальной меняющейся по длине линии на уровне стрелки-мишени, уве-

```

0 A0=28726
10 K=500
20 DIM(50),ICH(K)
30 N=K/50
40 P1=28928:A1=26624:A2=28672
50 PRINTCHR(31)
100 PRINT"**** РЕЖИМ РАБОТЫ КОМПЛЕКСА ****"
110 PRINT" 1 - СБОР СТАТИСТИКИ"
120 PRINT" 2 - ОБРАБОТКА СТАТИСТИКИ"
130 PRINT" 3 - ЗАПИСЬ СТАТИСТИКИ НА ЛЕН"
140 PRINT" 4 - ЧТЕНИЕ СТАТИСТИКИ С ЛЕН"
150 PRINT"ЗАДАЙТЕ РЕЖИМ":INPUT I
160 ON I GOTO 1000,5000,2000,2500
170 PRINT
180 PRINT"РЕЖИМ ЗАДАН НЕ ПРАВИЛЬНО!"
190 PRINT
200 GOTO100
1000 PRINT"ФАМИЛИЯ, ИМЯ, ОТЧЕСТВО ОБУЧАЕМОГО":INPUT PR
1001 GOSUB1050
1002 AA=A2:L=LEN(PR):IF L>45 THEN L=45:PR=LEFT$(PR,45)
1004 FORI=1TOL:POKE AA,ASC(MID$(PR,I,1)):AA=AA+1:NEXT
1006 POKE AA,0
1007 PRINTCHR(31)
1010 PRINT"ОБУЧЕНИЕ :",PR
1015 C=USR(P1)
1017 PRINTCHR(31);"ОБУЧЕНИЕ ";PR;" ЗАКОНЧЕНО"
1020 GOSUB1100
1040 GOTO90
1050 PRINT"НОРМА RR-ИНТЕРВАЛА ":INPUTM1
1060 IFN1<0.5 OR N1>1.1THEN PRINT"НЕПРАВИЛЬНАЯ НОРМА !":GOTO1050
1065 N1=INT(N1*100)
1070 N2=INT(N1/256):N1=N1-N2*256
1080 POKE A3,N1:POKE A3+1,N2
1090 RETURN
1100 AA=A1
1200 FORI=1TO K
1300 ICH(I)=(PEEK(AA)+PEEK(AA+1)*256)/1000
1400 AA=AA+2
1420 NEXT
1440 RETURN

```

Рис. 4. Фрагмент программы на БЕЙСИКе

дить полосу в заданном направлении от стрелки-мишени или фиксировать ее при достижении заданного значения кардиоинтервала. Требуемая степень саморегуляции достигается приемами аутотренинга, контролем дыхания, активированием воображения и фаза-тазии.

Отклонение величины текущего значения кардиоинтервала от значения, фиксируемого стрелкой-мишенью, информирует обучающегося о приближении к требуемой цели. Для обучения саморегуляции могут быть использованы звук, свет, электрическое раздражение, меняющиеся в соответствии с изменением кардиоинтервала.

#### Обработка результатов эксперимента

Задача математического анализа сердечного ритма — оценить состояние системы саморегуляции, механизма адаптации и в целом функциональное состояние организма. К настоящему времени наиболее разработан математический анализ ритма сердца с позиций теории вероятности, где последовательность временных кардиоинтервалов рассматривается как эргодический случайный процесс.

Результаты измерений после обучения записываются в виде кардиоинтервалограмм (рис. 5), где по оси ординат откладываются значения длительности сердечного цикла, в секундах, а по оси абсцисс — порядковые номера цикла. Кардиоинтервалограммы позволяют анализировать переходные процессы, амплитуды и длительности фаз. Пунктирная кривая (см. рис. 5, а) определяет усредненное значение изменения кардиоинтервала за время обучения (500 циклов); оценивается способность обучающегося к саморегуляции.

В кардиоинтервалограмме можно «спросовать» информацию, суммируя интервалы. Это позволяет устранить влияние на длительность кардиоинтервала дыхательной ритмии (в макете предусмотрено усреднение 10 интервалов).

Плотность распределения кардиоинтервалов описывается гистограммой. Для сравнения различных гистограмм необходимо при расчетах учитывать ширину разряда (шага) либо все разряды должны быть одинаковыми.

1420:	7230	CB6372	WRS:	CALL	NN
1430:	7233	112E70		LXI	D,ADREF+1
1440:	7236	21FF67		LXI	H,BUF-1
1450:	7239	0EE6		MOV	C,0E6H
1460:	723B	CB0CFB		CALL	OFBOCH
1470:	723E	23	WR:	INX	H
1480:	723F	4E		MOV	C,M
1490:	7240	CB0CFB		CALL	OFBOCH
1500:	7243	CBF272		CALL	COMPR
1510:	7246	C23E72		JNZ	WR
1520:	7249	CB6372		CALL	NN
1530:	724C	C9		RET	
1540:	724D	112E70	RDS:	LXI	D,ADREF+1
1550:	7250	210068		LXI	H,BUF
1560:	7253	3EFF		MOV	A,OFFH
1570:	7255	CB06F8	RD:	CALL	OF806H
1580:	7258	77		MOV	M,A
1590:	7259	CBF272		CALL	COMPR
1600:	725C	C8		RZ	
1610:	725D	23		INX	H
1620:	725E	3E08		MVI	A,B
1630:	7260	C3B572		JMP	RD
1640:	7263	1600	NN:	MVI	D,0
1650:	7265	0E00	NN1:	MVI	C,0
1660:	7267	CB0CFB		CALL	OFBOCH
1670:	726A	14		INR	D
1680:	726B	C8		RZ	
1690:	726C	C3B572		JMP	NN1
1700:	726F		ERR:	SET	ERR
1710:	726F	1A	MS:	LDA	D
1720:	7270	B7		ORA	A
1730:	7271	C8		RZ	
1740:	7272	77		MOV	M,A
1750:	7273	23		INX	H
1760:	7274	13		INX	D

Рис. 3. Фрагмент программы на ассемблере

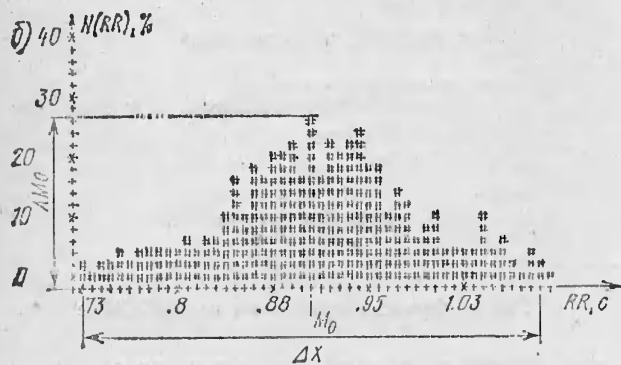
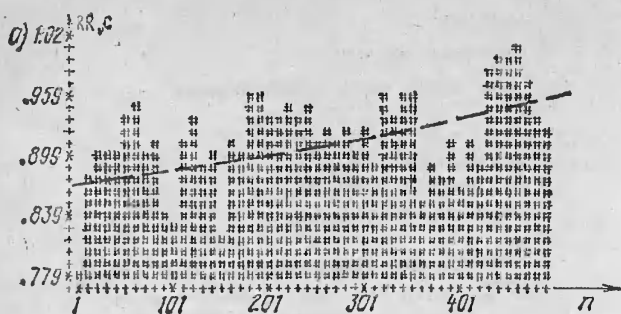


Рис. 5. Результаты БОС-тренинга обучаемого:  
а — интервалограмма; б — гистограмма

Разброс значений продолжительности сердечного цикла определяется изменением регуляции тонуса блуждающего нерва, симпатической, парасимпатической и гуморальной систем. Вариационный размах  $\Delta X$  интерпретируется как активность вагусной регуляции ритма сердца [4].

Наиболее часто встречающееся значение RR-интервала в гистограмме (см. рис. 5, б) носит название «Мода» ( $M_0$ ) и соответствует наиболее вероятному для периода обучения уровню функционирования систем регуляции и адаптации. Его физиологическая интерпретация — активность гуморального канала регуляции ритма сердца [5].

Для физиологического анализа гистограммы используется также вегетативный показатель ритма ВПР [6]  
$$ВПР = 1/M_0 \Delta X$$

и индекс напряжения  
$$ИН = AM_0 / 2 \Delta X M_0,$$

выражающий степень напряжения регуляторных механизмов сердца, где  $AM_0$  — наиболее вероятное значение «Моды» из пульсограммы в процентах.

Применение метода компьютерной биологической обратной связи поможет предупреждать «информационные неврозы», подключая сознание на более ранних стадиях степени напряжения, возникающей при эмоциональных нагрузках и увеличенных информационных потоках, характерных для интеллектуальной деятельности.

Телефон 291-43-97, Москва (после 20 ч.),  
Водоватов Федор Федорович

#### ЛИТЕРАТУРА

1. Превентивная кардиология / Под ред. Г. И. Кошечко. — М.: Медицина, 1977.
2. Черниговская Н. В., Мовсесянц С. А., Тимофеева Л. Н. Клиническое значение адаптивного биоуправления. — Л.: Медицина, 1983.

3. Горшков Д. И., Зеленко Г. В., Озеров Ю. В., Пайов В. В. Персональный радиолюбительский компьютер «Радио-86РК» // Радио, 1986.
4. Ритм сердца у спортсменов / Под ред. Р. М. Басевского, Р. Е. Мотылянской. — М.: Физкультура и спорт, 1986.
5. Бавеский Р. М. Синусовая аритмия с точки зрения кибернетики // Математические методы анализа сердечного ритма. — М.: Наука, 1968.
6. Сидоренко Г. И. Ранняя инструментальная диагностика гипертонической болезни и атеросклероза. — Минск: Белорусь, 1973.

Статья поступила 15.05.87

УДК 681.3.06

А. Ю. Барановский, А. Ю. Неверова, В. Н. Филинов,  
Т. Л. Цветкова

### ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ПРОГНОЗА ТЕЧЕНИЯ И ИСХОДОВ ЯЗВЕННОЙ БОЛЕЗНИ ЖЕЛУДКА

Автоматизированное рабочее место (АРМ) врача-гастроэнтеролога на базе ПЭВМ «Искра 226» реализует ввод и накопление данных на внешнем ЗУ (накопителе на гибком магнитном диске — ГМД); поиск, коррекцию и печать данных; вычисление прогностических коэффициентов по методу неоднородной последовательной процедуры (НПП) и коэффициентов дискриминантной функции; прогноз течения и исходов язвенной болезни желудка (ЯБЖ) на основе НПП и дискриминантного анализа. Функционирование программы начинается с выбора режима:

«Аннотация» содержит краткие сведения о программе, структуре файлов, дополнительных сервисных возможностях;

«Подготовка данных» — данные карт истории болезни с клавиатуры, в режиме «вопрос-ответ» либо в виде вектора строки градаций признаков заносятся на ГМД (с прогнозом при работе медицинского персонала), размерность записи — 84 байт;

«Поиск» осуществляется по 13 признакам, связанным конъюнктивно или дизъюнктивно;

«Коррекция» — исправленные данные вносятся в базу данных на ГМД (с прогнозом при работе медицинского персонала и изменениями в карты изучения истории болезни);

«Печать» реализует печать базы данных, данных по условиям поиска, данных карты изучения истории болезни интересующего больного;

«Вычисление прогностических коэффициентов» (коэффициентов дискриминантной функции) работает только в инженерном режиме. Коэффициенты вычисляются для каждой пары течений и исходов ЯБЖ и записываются в соответствующие зоны ГМД;

«Прогноз» — в зависимости от выбранного пользователем метода прогноза с ГМД считываются либо прогностические коэффициенты, либо коэффициенты дискриминантной функции (по мере возрастания базы данных коэффициенты могут быть уточнены в соответствующем режиме).

Программное обеспечение универсально, является типовым для решения других задач прогнозирования, изменяется только структура записи и назначение составляющих ее элементов.

194175, Ленинград, Лебедева, 6, ВМедА им. С. М. Кирова, кафедра ГУВ-2; телефон 251-68-21

Сообщение поступило 21.09.87

И. В. Кочин, С. И. Берстижевский, Т. В. Колосова

## КОМПЬЮТЕРНАЯ ТЕХНОЛОГИЯ В СОЦИАЛЬНОЙ ПРОФИЛАКТИКЕ ЗАБОЛЕВАНИЙ

Заключение цехового врача о пригодности для работы в конкретной профессии (с определенными санитарно-гигиеническими и психо-физиологическими условиями труда) должно быть обосновано данными медицинского осмотра, инструментальных и лабораторных исследований, а также оценкой образа жизни. Эти данные сопоставляются с характеристиками рабочего места и в комплексе с социально-гигиеническими условиями быта поступающего на работу определяют степень риска возникновения заболевания.

Компетентное заключение о возможности работы конкретного лица в данных условиях труда без ущерба для его здоровья с одновременным учетом образа жизни, поведения и социальных привычек цеховый врач не в состоянии сделать в реальном масштабе времени по причине своих ограниченных физических и информационных возможностей как человека. Выход из такого положения — использование компьютерной технологии на базе ПЭВМ для помощи цеховому врачу.

Предлагаемая компьютерная технология создана на основе анализа предметной области. Автоматически рассчитываются отклонения функции риска возникновения заболевания (соответствующей некоторой однородной по образу жизни группе работающих, к которой отнесен вновь поступающий на работу) от каждой из функций риска возникновения заболеваемости (соответствующей однородным по структуре заболеваемости группам производств).

Для формирования однородных по образу жизни групп работающих исследованы организация и социально-гигиенические условия труда; материально-жилищные условия; условия питания, отдыха, занятий физкультурой и спортом; взаимоотношения в семье, коллективе.

Для каждого работающего формируется интегральный показатель образа жизни в виде суммы логарифмов отношения вероятностей социально-гигиенических факторов по каждому работающему. Совокупность значений интегрального показателя образа жизни разбивается на однородные группы при помощи алгоритма классификации дихотомического типа.

Исследование социально-гигиенических условий труда и быта для получения однородных по образу жизни

групп было проведено на Запорожском коксохимическом заводе (ЗКХЗ). Совокупность значений интегрального показателя образа жизни составила 2185 единиц и была разделена на 18 групп. Анализ этой классификации показал статистическую значимость 15 групп.

Однородные по структуре заболеваемости группы производств промышленного предприятия определены при помощи индекса особенностей структур заболеваемости, который вычисляется так:

данные о заболеваемости структурируются по классам согласно международной классификации болезней (МКБ-9);

вычисляются интенсивные показатели (случаев выявленных заболеваний на 100 работающих по каждому классу болезней) и общий уровень заболеваемости; процентные отношения (удельные веса) числа случаев заболеваний на 100 работающих по каждому классу болезней к общему уровню заболеваемости; индекс особенностей структур заболеваемости через накопленные, взвешенные суммы удельных весов:  $I = [50N - (50 + d_1(N-1) + d_2(N-2) + \dots + d_{N-1})] / 50(N-1)$ , где  $i$  — индекс особенностей структур заболеваемости;  $d_i$  — удельный вес  $i$ -го класса заболеваний ( $i=1, 2, \dots, N$ );  $N$  — количество классов болезней по МКБ-9.

Совокупность значений индекса разбивается на группы при помощи алгоритма классификации дихотомического типа.

Функции риска возникновения заболевания и заболеваемости рассчитаны по данным ретроспективных исследований методами многомерного статистического анализа. Ретроспективные исследования социально-гигиенической системы факторов для работающих на ЗКХЗ проведены в 1978—1986 гг. И. В. Кочиным.

Цеховой врач проводит регламентированный медицинский осмотр вновь поступающего на работу и анкетный опрос о его образе жизни, затем, используя компьютерную технологию, получает рекомендации о целесообразности принятия на работу вновь поступающего лица в конкретное производство с указанием степени риска возникновения заболевания. Право окончательного заключения остается за цеховым врачом.

Компьютерная технология в виде программного продукта реализуется авторами на ПЭВМ, программно совместимой с IBM PC.

330096, Запорожье, бул. Винтера, 20,  
Запорожский ин-т усовершенствования  
врачей им. Горького, Кафедра  
социальной гигиены и организации  
здравоохранения; тел. 57-06-24

Статья поступила 15.11.87

Окончание. Начало см. на с. 18

Мир персональных компьютеров становится 32-разрядным (только на МП 80386 в США к сентябрю 1987 г. было анонсировано около 40 новых моделей ПЭВМ); далее он может развиваться только в сторону ПЭВМ с элементами пятого поколения ЭВМ.

129001, Москва, ул. Касаткина, д. 3,  
ВНИИИМТ; тел. 283-97-64

### ЛИТЕРАТУРА

- Компания IBM стремится изменить неблагоприятный ход событий // Электроника. — 1987. — № 3. — С. 12—13.
- Технико-экономический обзор // Электроника. — 1987. — С. 19—54.
- Bender E. IBM Tells a PS/2 Success Story // PC WORLD.— USA.— 1988. Vol. 6.— N 2.
- PS/2 Model 50, 60 Technical Reference.— USA: IBM Corp. 80x 0902.— 1987.
- INTEL Catalog. INTEL Corp.— USA.— 1987.
- DOS 3.3 System Reference. MICROSOFT.— USA.— 1987.
- Duncan R. The OS/2 Application Family // BYTE.— 1987.— N 10.— P. 109—118.
- Duncan R. A Programmers Introduction to OS/2.— 1987.— N 9.
- Reed S. B. Do you need a new operating system? // Personal Computing.— 1988.— N 2.— P. 99—101.
- Petzold C. OS/2—Ready to take on DOS—Has a Familiar Fell // PS Magazine.— 1988.— Vol. 7.— N 2.— P. 33—36.
- Горностаев Ю. М., Дрожжинов В. И. Сергеева интеграция автоматизированного машиностроительного производства на базе протоколов связи MAP/TOP.— М.: МЦНТИ, 1987.
- Пройдаков Э. М., Бабанов И. И. Средства обеспечения горизонтальной мобильности прикладных и инструментальных систем // Машинно-независимые операционные системы.— М.: МЦНТИ, 1987.— С. 63—67.
- Franklin C., Grehan R. The IBM PS/2 Model 80 // BYTE.— 1987.— N 13.— P. 143—148.
- 3.86 Operating Systems/Personal Computing.— March 1987.— P. 72.

Статья поступила 11.03.1988

К. А. Глушченко, Н. В. Кирианаки, О. Б. Котыло,  
С. Ю. Юриш

**ИЗМЕРИТЕЛИ ЧАСТОТНО-ВРЕМЕННЫХ ПАРАМЕТРОВ СИГНАЛОВ НА БАЗЕ ОЭВМ СЕРИИ K1816**

Для измерения частотно-временных параметров измерителями на базе ОЭВМ серии K1816 используется метод зависимого счета [1], эффективно реализуемый в измерителях частотно-временных параметров (ИЧВП) на основе микропроцессорных комплектов [2].

На рис. 1 представлена функциональная схема ИЧВП, использующего этот метод измерения, на рис. 2 — временные диаграммы работы. Аппаратурные затраты ограничены входным формирователем (ВФ), нормализующим входной сигнал  $f_x$ , и блоком цифровой индикации (БЦИ), подсоединенным к ОЭВМ KM1816BE48 через квазидвухнаправленные порты ввода-вывода P1, P2 и двухнаправленный порт BUS.

При нажатии кнопки «Сброс-Пуск» (сигнал CLR) происходит общий сброс ОЭВМ, выбирается нулевой блок памяти и нулевой банк регистров, таймер-счетчик событий (ТС) остановлен, признак переполнения таймера FT сброшен, все прерывания в системе запрещены. Нормализованные импульсы входной частоты подаются на вход запроса прерывания от внешнего источника — INT ОЭВМ. Программируемый ТС настраивается на функционирование в качестве таймера для формирования первого образцового интервала времени  $T_0$ , в течение которого происходит подсчет числа  $N_1$  импульсов периодической последовательности измеряемой частоты  $f_x$ . Это число фиксируется в регистрах R2 и R4 ОЭВМ. Параллельно с этим процессом происходит программное формирование второго временного интервала  $T'_0$ , строго синхронизированного с импульсами измеряемой частоты. Фронт  $T'_0$  соответствует импульсу последовательности  $f_x$ , появившемуся в начале временного интервала  $T'_0$ , а срез — импульсу, возникшему после окончания этого интервала. Длительность второго интервала времени определяется как  $T'_0 = N_1 T_x$ .

Погрешность округления исключается из-за строгой синхронизации  $T_0$  с импульсами  $f_x$ .

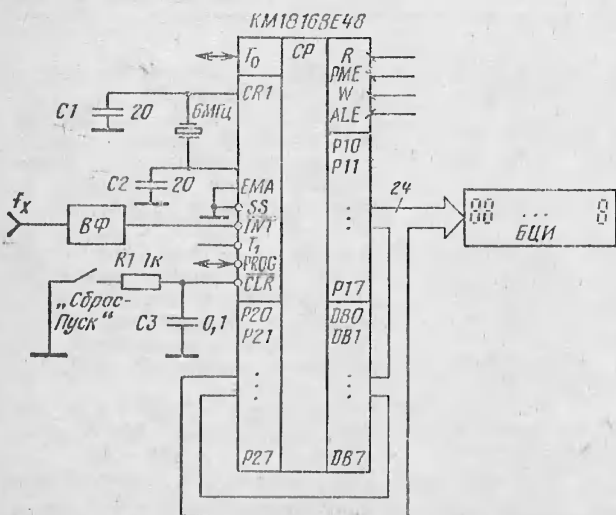


Рис. 1. Функциональная схема ИЧВП

Второй временной интервал  $T'_0$  заполняется импульсами высокой образцовой частоты  $f_c$ . Число импульсов  $N_2$  определяется счетчиками, образованными регистрами R3 и R5. Значение измеряемой частоты  $f_x = N_1 \cdot f_c / N_2$ .

На рис. 3 дана блок-схема алгоритма измерения частоты. С помощью подпрограммы обработки результатов измерения MATH вычисляется частота  $f_x$  или период  $T_x = N_2 / N_1 f_c$ ; с помощью подпрограммы INDEX результат измерения частоты или периода индицируется на БЦИ, после чего ИЧВП вновь готов к измерению.

Программа измерения частотно-временных параметров сигнала приведена на рис. 4.

С помощью команды очистки аккумулятора и команд пересылки обнуляются счетчики-регистры R2...R5. Число 50 загружается в R6 и определяет число перезагрузок ТС. Максимальная временная задержка, реализуемая ТС, составляет 20 мс, поэтому для формирования образцового интервала  $T_0 = 1$  с понадобится 50 перезагрузок ТС. Подсчет переполнений контролируется командой DJNZ R6, LOOP, передающей управление после перезагрузки ТС в точку программы с адресом LOOP. Подсчет чисел  $N_1$ ,  $N_2$  осуществляется с помощью команд инкремента содержимого регистров R2...R5 с учетом переноса  $C=1$ . Такая организация счета позволяет программе работать с 16-разрядными числами  $N_1$ ,  $N_2$ . Команда NOP (нет операции) используется для выравнивания времени выполнения ветвей цикла, ее длительность 2,5 мкс. Равенство времен различных участков программы (независимо от выполнения условия) позволяет программу моделировать равномерное квантование интервала  $T'_0$  импульсами  $f_c$  и равномерный подсчет импульсов  $f_x$  на входе INT. Период образцовой частоты  $f_c$  и максимальное значение измеряемой частоты  $f_x$  зависят от времени выполнения команд участков программы между двумя командами инкремента регистров, подсчитывающими числа  $N_1$  и  $N_2$ . При этом максимальная измеряемая частота  $f_x = 22,2$  кГц, а  $f_c = 28,5$  кГц.

Относительная погрешность квантования определяется как

$$\delta f_{кв} = \pm f_c / (T_0 + 50 T_{II}),$$

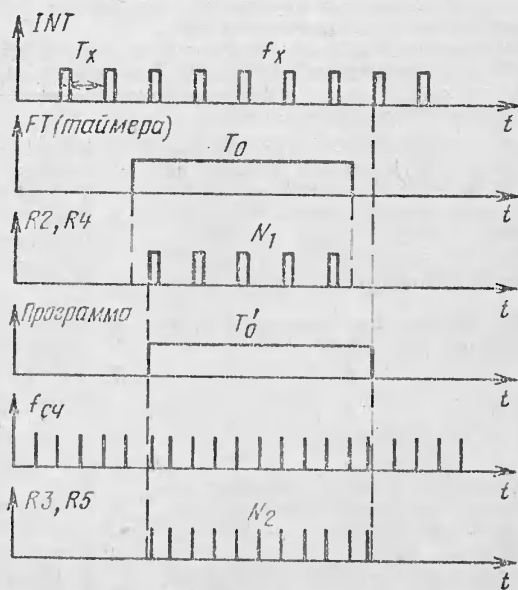


Рис. 2. Временные диаграммы

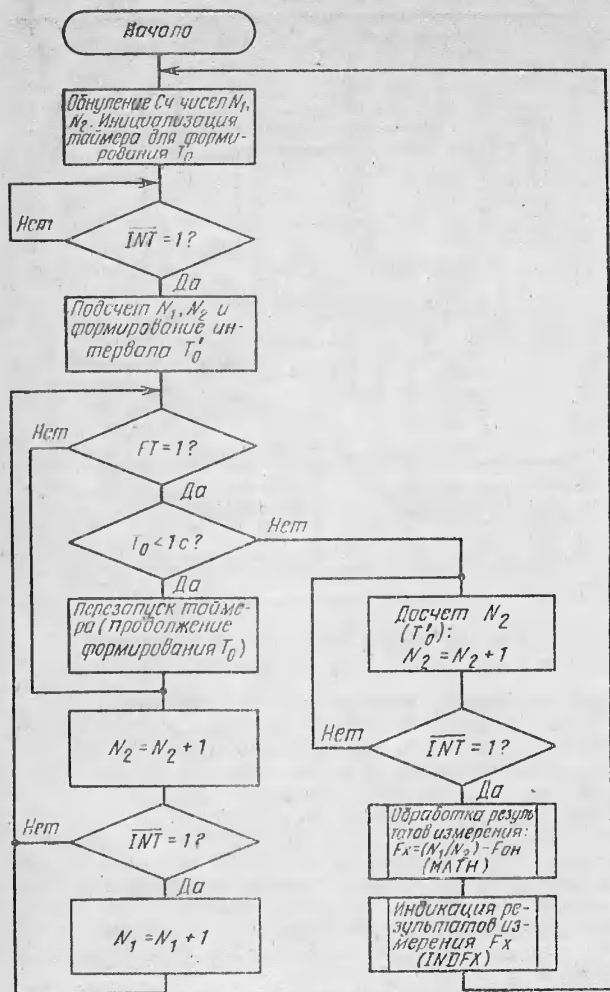


Рис. 3. Блок-схема алгоритма измерения частоты

где  $T_{\Pi}$  — время выполнения команд перезагрузки  $T_C$  после каждого его переполнения. В данном случае время выполнения трех команд перезагрузки составляет 10 мкс при частоте стандартного кварцевого резонатора осциллятора  $f_0 = 5,994$  МГц. Тогда  $T_{\Pi} = 5 \cdot 10^{-4}$  с. Численное значение погрешности квантования  $\delta i_{кв}$  составляет  $\pm 3,5 \cdot 10^{-3}$  %.

Важная особенность используемого метода измерения частотно-временных параметров сигналов — независимость погрешности квантования  $\delta i_{кв}$  от измеряемых величин во всем диапазоне измерения  $f_x$  и  $T_x$ .

При программной реализации метода измерения частоты устраняется случайная погрешность запуска (преобразования), обусловленная шумовой помехой, проявляющейся при формировании стробирующего импульса  $T_0$  аппаратными средствами (триггерными схемами).

Программное обеспечение ИЧВП написано на языке ассемблера для ОЭВМ КМ1816ВЕ48. Его объем без учета стандартных подпрограмм MATH и INDEX 65 байт. Следовательно, для хранения ПО достаточно 1 Кбайт резидентного УФРПЗУ самой ОЭВМ.

Использование ОЭВМ КМ1816ВЕ39 увеличит диапазон измеряемой частоты до 43 кГц, так как ее максимальная тактовая частота составляет 11 МГц. Резидентное ПЗУ отсутствует в данной БИС, поэтому необхо-

\*\*\*\*\*  
 ПРОГРАММА ИЗМЕРЕНИЯ ЧАСТОТНО-ВРЕМЕННЫХ ПАРАМЕТРОВ  
 ЭЛЕКТРИЧЕСКИХ СИГНАЛОВ  
 \*\*\*\*\*

```

START: CLR      A          ;ОРГАНИЗАЦИЯ СЧЕТЧИКОВ ЧИСЕЛ N1, N2
        MOV      R2, A    ;МЛАДШИЙ БАЙТ N2
        MOV      R3, A    ;МЛАДШИЙ БАЙТ N1
        MOV      R4, A    ;СТАРШИЙ БАЙТ N1
        MOV      R5, A    ;СТАРШИЙ БАЙТ N2
        MOV      R6, #56  ;ИНИЦИАЛИЗАЦИЯ ТАЙМЕРА И ФОРМИРОВАНИЕ ИНТЕРВАЛА T0=1C
        MOV      T, A
        STRT
DEL:    JN1      WAIT     ;ПЕРВЫЙ ИМПУЛЬС fx ПРИШЕЛ ?
        JMP      COUNT
WAIT:   JMP      DEL      ;НЕТ. ОЖИДАНИЕ ПЕРВОГО ИМПУЛЬСА fx
COUNT: INC      R2      ;ДА. СЧЕТ N1, N2, ФОРМИРОВАНИЕ T0
        INC      R3
        REPIT:  JTF      REDEL  ;T0 ЗАКОНЧИЛ ?
        NOP
        NOP
        NOP
        LOOP:  INC      R3      ;НЕТ. ПРОДОЛЖИТЬ СЧЕТ N1, N2 С УЧЕТОМ ПЕРЕПОЛНЕНИЯ РЕГИСТРОВ R3, R2, ЕСЛИ ЮНО ПРИЗОШЛО
        JNC      MLN2
        INC      R5
        JMP      STN2
MLN2:   NOP
        NOP
        STN2:  JN1      REP1
        INC      R2
        JNC      MLN1
        INC      R4
        JMP      MLN1
MLN1:   NOP
        NOP
        STN1:  JMP      REPIT
REDEL:  MOV      T, A      ;ПЕРЕЗАГРУЗКА ТАЙМЕРА, ЕСЛИ T0 < 1C И ПРОДОЛЖЕНИЕ ФОРМИРОВАНИЯ T0
        DJNZ    R6, LOOP
CYCL:   INC      R3
        JNC      RC1
        INC      R5
        JMP      RC2
RC1:    NOP
        NOP
        RC2:   JN1      PROC    ;ПОСЛЕДНИЙ ИМПУЛЬС fx ПРИШЕЛ ?
        JMP      CYCL
PROC:   STOP
        CALL   MATH      ;ОБРАБОТКА РЕЗУЛЬТАТОВ ИЗМЕРЕНИЯ
        CALL   INDEX     ;ИНДИКАЦИЯ РЕЗУЛЬТАТОВ ИЗМЕРЕНИЯ
        JMP      START    ;ПОВТОРИТЬ ПРОЦЕСС ИЗМЕРЕНИЯ
  
```

Рис. 4. Программа измерения частотно-временных параметров

димы дополнительные аппаратные затраты для организации внешней программной памяти.

ОЭВМ КМ1816ВЕ49 имеет резидентное ПЗУ 2 Кбайт и тактовую частоту 11 МГц. Такой объем памяти программ позволяет расширить функциональные возможности ИЧВП введением дополнительных режимов измерения (подсчет числа импульсов, измерение временного интервала, длительности) практически без увеличения аппаратных затрат. Однако дальнейшая модернизация ИЧВП без дополнительной внешней памяти невозможна, так как резидентное ПЗУ данной ОЭВМ не позволяет перезаписывать информацию.

Усовершенствование ИЧВП на основе ОЭВМ КМ1816ВЕ48 производится без затруднений, если изменения в программе не требуют увеличения внутреннего репрограммируемого УФРПЗУ [3].

Построение multifunctional ИЧВП на базе ОЭВМ серии К1816 возможно благодаря сопряжению БИС микропроцессорного комплекса КР580 с ОЭВМ (например, введене в состав ИЧВП БИС программируемого интервального таймера КР580ВИ53, параллельного интерфейса ввода-вывода КР580ВВ55 и т. п.).

Наиболее эффективные ИЧВП с минимальными аппаратными затратами, широкими функциональными возможностями и высокими метрологическими характеристиками — измерители на основе ОЭВМ К1816ВЕ51 [4, 5].

Характерные особенности данной БИС для построения ИЧВП:

наличие двух 16-разрядных многорежимных таймеров-счетчиков, работающих в одном из четырех возмож-

ных режимов, четырех портов ввода-вывода, программной резидентной памяти объемом 4 Кбайт, многоуровневой системы прерываний, высокой тактовой частоты — 12 МГц;

развитая система команд обеспечивает возможность обработки 16-разрядных данных;

команды арифметических операций умножения и деления, разнообразие способы адресации упрощают стандартную подпрограмму обработки результатов измерения и повышают скорость ее выполнения.

Использование ОЭВМ серии К1816 для построения ИЧВП и описанный метод измерения частотно-временных параметров электрических сигналов позволяют создавать приборы с минимальными аппаратными затратами, высокой надежностью, сравнительно низкой стоимостью, удовлетворяющие производственные и технические измерения метрологических характеристик, достаточных для широкого использования ИЧВП при автоматизации технологических процессов, в системах телесмотра, тахометрических комплексах.

290646, Львов-13, ул. Мира, 12, ЛПИ, кафедра «Автоматика и телемеханика»; тел. 39-81-97

## ЛИТЕРАТУРА

1. А. с. 788018 СССР. Способ измерения частоты и периода гармонического сигнала и устройство для его осуществления. / Н. В. Кирианаки, Б. М. Березюк. 1980, Бюл. № 46.
2. Кирианаки Н. В., Витер А. С., Котыло О. Б., Юриш С. Ю. Универсальный микропроцессорный частотомер // Приборы и системы управления. — 1987. — № 8. — С. 26—27.
3. Кобылянский А. В., Липовецкий Г. П. Однокристалльные микроЭВМ серии К1816 // Микропроцессорные средства и системы. — 1986. — № 1. — С. 10—19.
4. Весноватов М. Г., Карацуба Г. В., Павлов В. В., Старшова В. А. Перспективные однокристалльные ЭВМ // Микропроцессорные средства и системы. — 1987. — № 2. — С. 7—8.
5. MCS-51 Intel Family of Single Chip Microcomputers User's Manual. Intel. — 1981. — P. 1. — 1986.

Статья поступила 28.12.87

УДК 621.326

Л. А. Белов, Д. Г. Бухаров, В. М. Дежидов, М. Ю. Фомичев

## ШИРОКОДИАПАЗОННЫЙ МИКРОПРОЦЕССОРНЫЙ ЧАСТОТОМЕР

При создании автоматизированных измерительных систем и проведении экспериментальных исследований возникает задача автоматического измерения частоты сигнала неизвестного заранее диапазона. Частоту повторения неизвестного сигнала можно определить [1] подсчетом числа  $k_n$  импульсов измеряемого сигнала за определенное время  $T_n = k_n \tau_n$ , где  $k_n$  — число счетных импульсов;  $\tau_n$  — эталонный период их следования. Тогда  $f = k_n / T_n$ . Если частота  $f$  низкая, а время  $T_n$  ограничено, целесообразно считать эталонные импульсы за отрезок времени  $T_1 = k_n / f$ , определенный известным числом периодов измеряемого сигнала, а затем вычислять частоту  $f = k_n / T_1 = k_n / k_n \tau_n$ .

На основе микропроцессорного комплекта К580 с помощью данного метода был построен автоматиче-

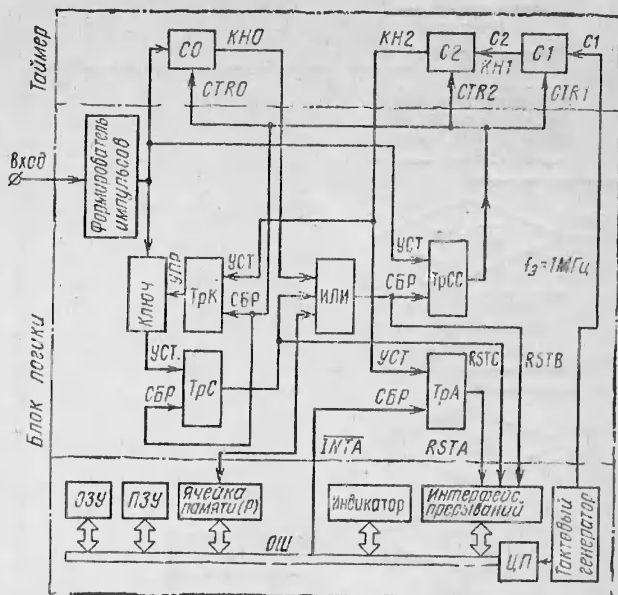


Рис. 1. Структурная схема частотомера

ский частотомер, работающий в диапазоне  $10^{-3} \dots 10^6$  Гц с индикацией четырех значащих десятичных разрядов и десятичного порядка частоты (рис. 1). Погрешность формирования моментов времени, соответствующих определенной фазе входного периодического сигнала, мала; при  $f \leq 999,9$  Гц измеряется период входных сигналов с индикацией расчетного значения частоты. При  $f > 999,9$  кГц и  $f < 1,000$  мГц индицируется выход за допустимые пределы измерений. Относительная погрешность измерения частоты определяется нестабильностью частоты кварцевого генератора и составляет  $\pm 1 \times 10^{-4}$  при относительном уходе эталонной частоты за 10 мин  $-1 \times 10^{-5}$ .

Шестнадцатиразрядные счетчики (рис. 2) программно переводятся на двоично-десятичное вычитание в режиме программируемой задержки (режим 0 для С0 и С1) и управляемого делителя частоты (режим 2 для С2). На счетный вход С0 поступают импульсы измеряемой частоты  $f$ , на С1 — эталонной частоты  $f_0 = 1,000$  МГц от задающего кварцевого генератора микропроцессорной системы. В исходном состоянии основная программа записывает в ячейку памяти Log.1, поэтому старт-стопный триггер (ТрСС) и триггер ключа (ТрК) находятся в состоянии сброса, счет в таймере запрещен по входам СТРО, СТР1, СТР2, ключ (К) разомкнут. В счетчики С0 и С1 записывается число 9999, в С2 — 0099. После программного снятия запрета счета в ячейке памяти блок логики формирует временные ворота для импульсов с выхода ФИ. По прерываниям программно определяется порядок частоты (номер поддиапазона), находят значения цифры частоты, непосредственно по содержимому счетчиков или с помощью вычислений выводится информация на индикатор и осуществляется возврат к исполнению основной программы.

Весь диапазон частот разбивается на 11 поддиапазонов (см. таблицу), D0 и D10 соответствуют выходу частоты входных сигналов за допустимые пределы. Старт-стопный триггер устанавливает первый после разрешения счета входной импульс — начинается счет в таймере.

При  $f \geq 10,00$  кГц первым обнуляется счетчик С0; выходной сигнал KHO через схему ИЛИ сбрасывает ТрСС, счет останавливается и на интерфейс прерываний (ИП) поступает запрос RSTB. Содержимое С2 анали-



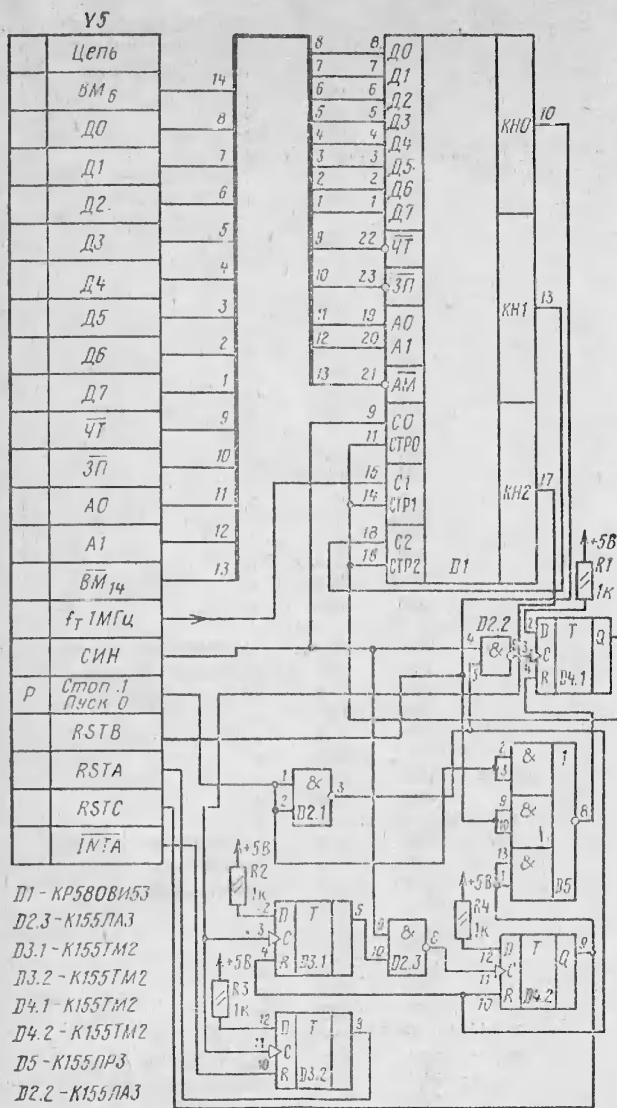


Рис. 2. Принципиальная схема таймера и блока логики

зируется и принимается решение о работе в поддиапазонах Д0, Д1 или Д2 (рис. 3). Значение цифры частоты в поддиапазонах Д1 и Д2 определяются с помощью преобразования и сдвига содержимого С1 и С2 (рис. 4).

При  $f \leq 9,999$  кГц первым обнуляется счетчик С2: сигнал КН2 через ТрК замыкает ключ, а через триггер прерывания А (ТрА) и ИП вызывает прерывание RSTA. Подпрограмма обработки RSTA записывает 9899 в С2 и учитывает ранее подсчитанные 99 импульсов по его входу С2. Очередной импульс с выхода ФИ устанавливает триггер прерывания (ТрС), что приводит к появлению запроса на прерывание RSTC, а через схему ИЛИ к сбросу ТрСС и останову счета. В счетчике С0 содержится число периодов входной частоты  $k_n$  в С1 и С2 — интервал времени  $T_1$ , занимаемый этими импульсами. С помощью подпрограмм обработки прерывания ЦП находят номер поддиапазона Д3, Д4, Д5, Д6 или Д7 вычисляет частоту  $f = k_n/T_1$ .

При  $f \leq 99,99$  МГц через каждые 10 с до прихода второго импульса входной частоты происходит обнуление С2 и возникает прерывание RSTA. Для расширения

Данные для расчета частоты

Интервал частот входного сигнала	Запрос на прерывание при окончании счета	Содержимое счетчиков			
		C2	C1	C0	CA
Д0 $\geq 1,000$ МГц	RSTB	0099	****	9899	99
Д1 $100,0 \dots 999,9$ кГц	RSTB	009X	AAA*	9899	99
Д2 $10,00 \dots 99,99$ кГц	RSTB	00XA	AA**	9899	99
Д3 $1,000 \dots 9,999$ кГц	RSTC	9899	AAAA	XAAA	00
Д4 $100,0 \dots 999,9$ Гц	RSTC	989A	AAA*	9XAA	00
Д5 $10,00 \dots 99,99$ Гц	RSTC	98AA	AA**	99XA	00
Д6 $1,000 \dots 9,999$ Гц	RSTC	9XAA	A***	99YX	00
Д7 $100,0 \dots 999,9$ мГц	RSTC	AAAA	****	9998	00
Д8 $10,00 \dots 99,99$ мГц	RSTC	AAA*	****	9998	00
Д9 $1,000 \dots 9,999$ мГц	RSTC	AA**	****	9998	0B
Д10 $< 1,000$ мГц	RSTA	AA***	****	9999	BA < 09

Примечание. A=0.1...9; X=0.1, 2...8; Y=0.1, 2...7; B=1. 2, 3...; 9; \* — значение игнорируется.

диапазона измеряемых частот до 1 мГц организован программный двоично-десятичный счетчик СА, в котором подсчитывается число этих прерываний. Содержимое этого счетчика, дополненное в поддиапазонах Д8 и Д9 старшими разрядами С2, определяет интервал времени  $T_1$  между запускающим и следующим импульсами входной частоты  $f = 1/T_1$ .

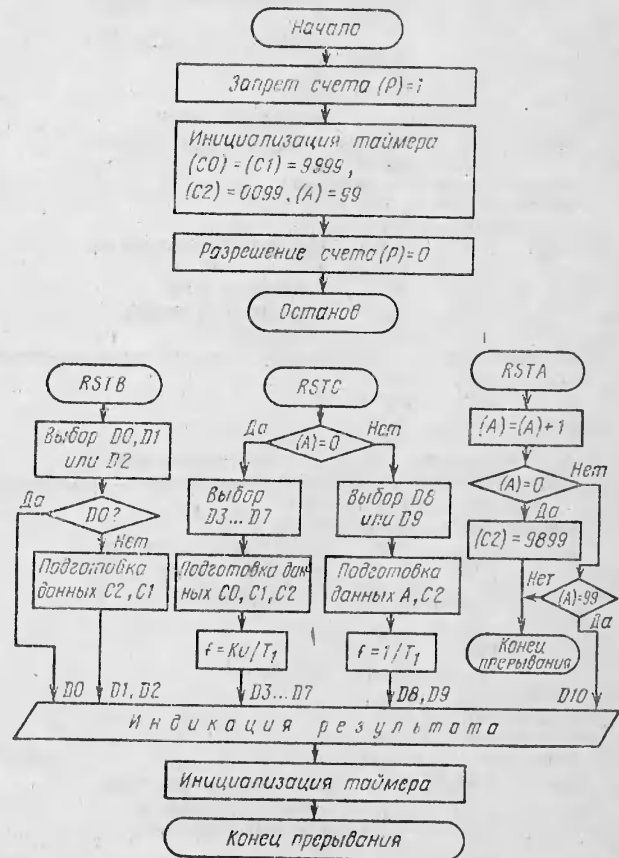


Рис. 3. Алгоритм программного обеспечения

```

; *****
; ШИРОКОДИАПАЗОННЫЙ
; АВТОМАТИЧЕСКИЙ ЧАСТОТОМЕР
; *****
M16: LXI H,UC ;ЗАПИСЬ АДРЕСА РУС В H,L
      MVI M,61 ;ИНИЦИАЛИЗАЦИЯ ТАЙМЕРА
      MVI A,231 ;
      MOV A,B ;ИНИЦИАЛИЗАЦИЯ СЧЕТЧИКА A
      STA C0 ;ЗАПИСЬ В C0 ЧИСЛА 9999
      STA C0 ;
      MVI M,165 ;ИНИЦИАЛИЗАЦИЯ ТАЙМЕРА
      STA C1 ;ЗАПИСЬ В C1 ЧИСЛА 9999
      STA C1 ;
      MVI M,261 ;ИНИЦИАЛИЗАЦИЯ ТАЙМЕРА
      STA C2 ;ЗАПИСЬ В C2 ЧИСЛА 0099
      XRA A ;
      STA C2 ;
      INR A ;
      STA PUSK ;ЗАПУСК СЧЕТА, РАЗМЫКАНИЕ КЛЮЧА
      HLT ;ОЖИДАНИЕ ПРЕРЫВАНИЙ ОТ СЧЕТЧИКОВ
; *****
; ОБРАБОТКА ПРЕРЫВАНИЙ RSTA
; *****
RSTA: MOV A,B ;ОБРАБОТКА ПРЕРЫВАНИЙ RSTA
      CPI 231 ;ОПРЕДЕЛЕНИЕ НОМЕРА
      JZ M60 ;ПОДДИАПАЗОНА
      CPI 230 ;
      JZ M62 ;
      INR A ;ПОДСЧЕТ ПРЕРЫВАНИЙ RSTA
      DAA ;ЗАПОЛНЕНИЕ СЧЕТЧИКА A
      MOV B,A ;
      CPI 231 ;
      JNZ TUB+3 ;
;
; ИНДИКАЦИЯ F<10E-3 ГЦ
;
M60: MVI M,261 ;ИНИЦИАЛИЗАЦИЯ ТАЙМЕРА
      MVI A,231 ;
      STA C2 ;ЗАПИСЬ В C2 ЧИСЛА 9899
      MVI A,230 ;
      DCR B ;
TUB: STA C2 ;
      HLT ;ОЖИДАНИЕ RSTC
M62: MVI M,241 ;ИНИЦИАЛИЗАЦИЯ ТАЙМЕРА
      JMP TUB-1 ;
; *****
; ОБРАБОТКА ПРЕРЫВАНИЙ RSTB
; *****
RSTB: MVI M,221 ;ИНИЦИАЛИЗАЦИЯ ТАЙМЕРА
      CPI 231 ;
      JZ M0 ;ОБРАБОТКА RSTB
      CPI 220 ;ОПРЕДЕЛЕНИЕ НОМЕРА
      JC M1 ;ПОДДИАПАЗОНА
;
M2: CALL M0C ;
      JMP M20 ;
M1: CALL M0A ;
      JMP M20 ;
;M0: ВЫВОД НА ИНДИКАТОР СООБЩЕНИЯ F>1 MHZ
; *****
; ОБРАБОТКА ПРЕРЫВАНИЙ RSTC
; *****
RSTC: CALL DOP ;ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ
      PUSH D ;ЧАСТОТЫ СИГНАЛА
      MVI M,165 ;В ПОДДИАПАЗОНЕ D3
      LDA C1 ;
      MOV E,A ;
      LDA C1 ;
      MOV D,A ;
      CALL MCB ;
M31: POP D ;
      JMP M201 ;
;////////M4////////
M4: CALL DOP ;ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ
      PUSH D ;ЧАСТОТЫ СИГНАЛА В
      MVI M,221 ;ПОДДИАПАЗОНЕ D4
      LDA C2 ;
      CALL M0A ;
      JMP M31 ;
;////////M5////////

```

```

M5: CALL DOP ;ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ
      PUSH D ;ЧАСТОТЫ СИГНАЛА В
      MVI M,221 ;ПОДДИАПАЗОНЕ D5
      LDA C2 ;
      CALL M0C ;
      JMP M31 ;
;////////M6////////
M6: CALL DOP ;ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ
      PUSH D ;ЧАСТОТЫ СИГНАЛА В
      MVI M,261 ;ПОДДИАПАЗОНЕ D6
      LDA C2 ;
      MOV C,A ;
      CALL MH2 ;
      MOV D,A ;
      MOV A,C ;
      CALL MH1 ;
      MOV E,A ;
      LDA C2 ;
      CALL MH1 ;
      ADD D ;
      MOV D,A ;
      MVI M,145 ;
      LDA C1 ;
      CALL MH2 ;
      ADD E ;
      MOV E,A ;
      CALL MCB ;
      JMP M31 ;
;////////M7////////
M7: MVI M,261 ;ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ
      LDA C2 ;ЧАСТОТЫ СИГНАЛА В
      MOV E,A ;ПОДДИАПАЗОНЕ D7
      LDA C2 ;
      MOV D,A ;
      CALL MCB ;
M71: LXI D,1 ;
      JMP M201 ;
;////////M8////////
M8: MOV A,B ;ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ
      CALL MH1 ;ЧАСТОТЫ СИГНАЛА В
      MOV D,A ;ПОДДИАПАЗОНЕ D8
      MVI M,261 ;
      LDA C2 ;
      CALL MH2 ;
      MOV E,A ;
      CALL M0A1 ;
      JMP M71 ;
;////////M9////////
M9: MOV D,B ;ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЯ
      MVI M,241 ;ЧАСТОТЫ СИГНАЛА В
      LDA C2 ;ПОДДИАПАЗОНЕ D9
      MOV E,A ;
      CALL MCB ;
      JMP M71 ;
;////////M10////////
DOP: MOV A,E ;ПОДПРОГРАММА ДЛЯ
      CMA ;ПРЕОБРАЗОВАНИЯ
      INR A ;КОДА, УЧИТЫВАЮЩАЯ,
      ADI 231 ;ЧТО СЧЕТЧИКИ РАБОТАЮТ
      MOV E,A ;В РЕЖИМЕ ВЫЧИТАНИЯ I
      MOV A,D ;
      CMA ;
      INR A ;
      ADI 231 ;
      MOV D,A ;
      RET ;
M0A: CALL MH1 ;ПОДПРОГРАММА ОБРАБОТКИ
      MVI M,165 ;ПОКАЗАНИЙ СЧЕТЧИКОВ
      MOV D,A ;ДЛЯ ФОРМИРОВАНИЯ
      LDA C1 ;ДЕЛИМОГО И ДЕЛИТЕЛЯ
      CALL MH2 ;
      MOV E,A ;
      LDA C1 ;
      MOV C,A ;
      CALL MH1 ;
      ADD E ;
      MOV E,A ;
      MOV A,C ;
      CALL MH2 ;

```

```

ADD D ;
MOV D,A ;
MCB: CALL DBP ;
MOV A,E ;
ANI 17 ;
MOV L,A ;
MVI H,0 ;
MOV A,E ;
CALL MH2 ;
CALL MU3 ;
MOV A,D ;
ANI 17 ;
CALL MU2 ;
MOV A,D ;
ANI 360 ;
CALL MU1 ;
MOV E,H ;
MOV C,L ;
RET ;
;***** HELP *****
MH1: ANI 17 ; ПОДПРОГРАММА
      RLC ; ФОРМИРОВАНИЯ
      RLC ; ПОЛУБАЙТОВ
      RLC ; ДЛЯ ПОДСЧЕТА
      RLC ; ЧАСТОТЫ
      RET ;
MH2: ANI 360 ;
      RRC ;
      RRC ;
      RRC ;
      RRC ;
      RET ;
MH: PUSH H ; ПОДПРОГРАММА
    LXI H,0 ; УМНОЖЕНИЯ И
    MVI B,0 ; ДЕЛЕНИЯ
    MVI D,10 ;
MHA: DAD H ;
      RLC ;
      JNC MUI ;
      DAD B ;
MUI: DCR D ;
      JNZ MHA ;
      POP D ;
      DAD D ;
      RET ;
MU1: MVI C,175 ;
MU4: PUSH D ;
      CALL MU ;
      POP D ;
      RET ;
MU2: MVI C,144 ;
      JMP MU4 ;
MU3: MVI C,12 ;
      JMP MU4 ;
;
DIV: LXI H,0 ;
      XRA A ;
LOOP: EAD H ;
      PUSH PSW ;
      XCHG ;
      DAD H ;
      XCHG ;
      JNC G1 ;
      INX H ;
G1: MOV A,L ;
      SUB C ;
      MOV L,A ;
      MOV A,H ;
      SBB B ;
      MOV H,A ;
      JC G2 ;
      POP PSW ;
G3: INX D ;
      JMP G4 ;
G2: POP PSW ;
      JC G3 ;
      DAD B ;
G4: ADI 20 ;
      JNC LOOP ;
      RET ;

```

```

;
M20: LXI D,114631; ВЫЧИСЛЕНИЕ ЧАСТОТЫ
M201: CALL DIV ; ДЛЯ ПОДАТАЗОНОВ
      MVI A,4 ; D1,D2
      PUSH PSW ;
MSW: POP PSW ;
      DCR A ;
      PUSH PSW ;
      JZ MS4 ; ИНДИКАЦИЯ РЕЗУЛЬТАТА
      CPI 2 ;
      MOV A,E ;
      CMA ;
      JZ MS2 ; ИНДИКАЦИЯ РЕЗУЛЬТАТА
      JM MS3 ; ИНДИКАЦИЯ НОМЕРА ПОДАТАЗОНА
      RET ;

```

Рис. 4. Программа частотомера

Программное обеспечение частотомера занимает 399 ячеек ПЗУ без команд вывода на индикатор, 20 ячеек ОЗУ в виде стека, 74 ячейки ПЗУ для обеспечения работы узла «Индикация». Вспомогательная подпрограмма деления четырехбайтового двоичного числа на двухбайтовое приведена в [3] под именем DIV. Для ее использования двоично-десятичное содержимое C0, C1 и C2 надо преобразовать в двоичное. Для подсчета частоты достаточно иметь результат в виде четырех десятичных разрядов, поэтому используется простая процедура выделения 16 требуемых значащих разрядов из 32 двоичных.

Разработанный автоматический частотомер можно использовать для более высоких входных частот, применив дополнительный внешний СВЧ делитель частоты. Частотомер может быть выполнен в виде автономного измерителя.

105835, Москва, Красноказарменная, 14, МЭИ, Кафедра радиопередающих устройств, тел. 362-77-95.

#### ЛИТЕРАТУРА

1. Мирский Г. Я. Электронные измерения.— М.: Радио и связь, 1986.
2. Алексенко А. Г., Галицын А. А., Иванников А. Д. Проектирование радиоэлектронной аппаратуры на микропроцессорах.— М.: Радио и связь, 1984.
3. Григорьев В. Л. Программное обеспечение микропроцессорных систем.— М.: Энергоатомиздат, 1983.

Статья поступила 1.12.87

## СПРАВОЧНАЯ ИНФОРМАЦИЯ

УДК 681.3.01 : 51

### ИНТЕРФЕЙСНЫЕ БИС МИКРОПРОЦЕССОРНОГО КОМПЛЕКТА K1801: МИКРОСХЕМА K1801ВП1-034

(Продолжение цикла. Начало в № 4, 5, 1988)

Микросхема K1801ВП1-034 представляет собой многофункциональное устройство, работающее в следующих режимах:

передачи информации;  
выдачи вектора прерывания и как компаратор адреса;  
буферного регистра данных.

Используется БИС K1801ВП1-034 как самостоятельное устройство и совместно с микросхемой K1801ВП1-033\* для организации интерфейсов байтового параллельного ввода и 16-разрядного программируемого параллельного ввода-вывода.

\* Интерфейсные БИС микропроцессорного комплекта K1801: микросхема K1801ВП1-033 // Микропроцессорные средства в системах.— 1988.— № 5.— С. 87.

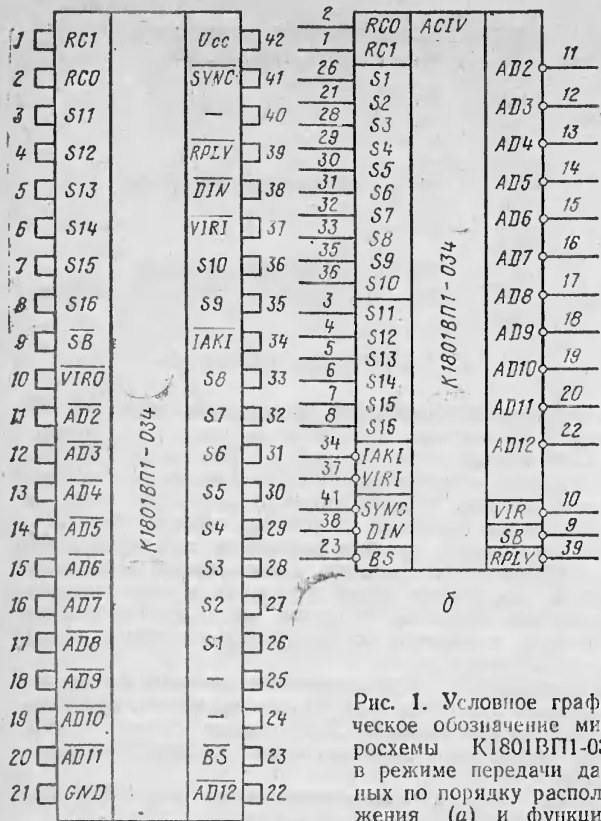


Рис. 1. Условное графическое обозначение микросхемы К1801BP1-034 в режиме передачи данных по порядку расположения (а) и функциональному назначению (б) выводов

Таблица 1

Наименование выводов устройства передачи информации (RC0=1, RC1=0)

Вывод	Обозначение	Тип вывода	Наименование вывода
1	RC1	Вход	Выбор режима 1
2	RC0	»	Выбор режима 0
3...8	B0 ... B5	»	Данные B0 ... B5
9...16	C0 ... C7	Вход-выход	Данные каналов AD0 ... AD7
17...20	D0 ... D3	Выход	Данные D0 ... D3
21	GND	—	Общий
22...25	D4 ... D7	Выход	Данные D4 ... D7
26...33	A0 ... A7	Вход	Данные A0 ... A7
34	R	»	Сброс данных
35	B6	»	Данные B6
36	B7	»	Данные B7
37	CD	»	Разрешение выдачи данных D
38	CB	»	Разрешение выдачи данных B
39	COM	»	Инвертирование данных
40	C	»	Запись данных
41	CA	»	Разрешение выдачи данных A
42	Ucc	—	Напряжение источника питания

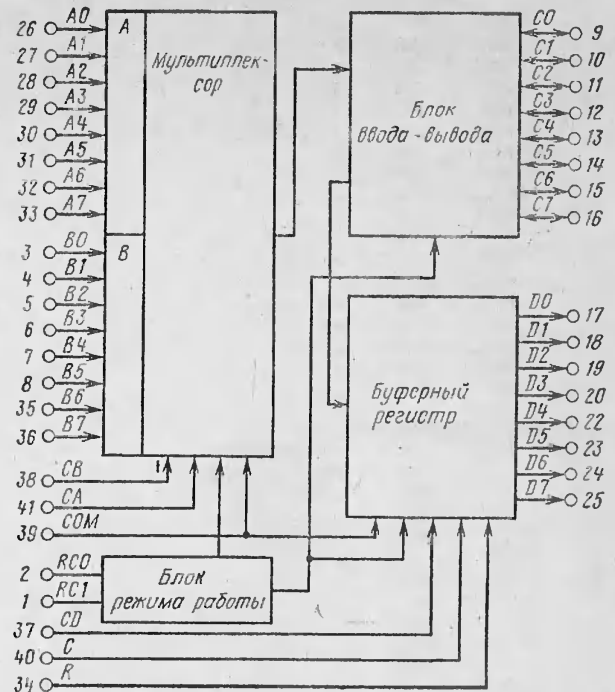


Рис. 2. Электрическая структурная схема устройства передачи информации

Таблица 2

Наименование выводов устройства выдачи вектора прерывания и компаратора адреса (RC0=1, RC1=1)

Вывод	Обозначение	Тип вывода	Наименование вывода
1	RC1	Вход	Выбор режима 1
2	RC0	»	Выбор режима 0
3...8	S11 ... S16	»	Зашивка вектора прерывания 2 ... 7
9	SB	Выход	Устройство выбрано
10	VIR0	»	Запрос на прерывание
11...20	AD2 ... AD11	Вход-выход	Разряды адреса-данных Общий
21	GND	—	Общий
22	AD12	Вход-выход	12-й разряд адреса-данных
23	BS	Вход	Внешнее устройство
24, 25	—	—	Не используются
26...33	S1 ... S8	Вход	Зашивка адреса 3 ... 10
34	IAKI	»	Разрешение прерывания
35	S9	»	Зашивка адреса 11
36	S10	»	Зашивка адреса 12
37	VIRI	»	Запрос прерывания
38	DIN	»	Чтение данных
39	RPLY	Выход	Ответ
40	—	—	Не используется
41	SYNC	Вход	Обмен
42	Ucc	—	Напряжение источника питания

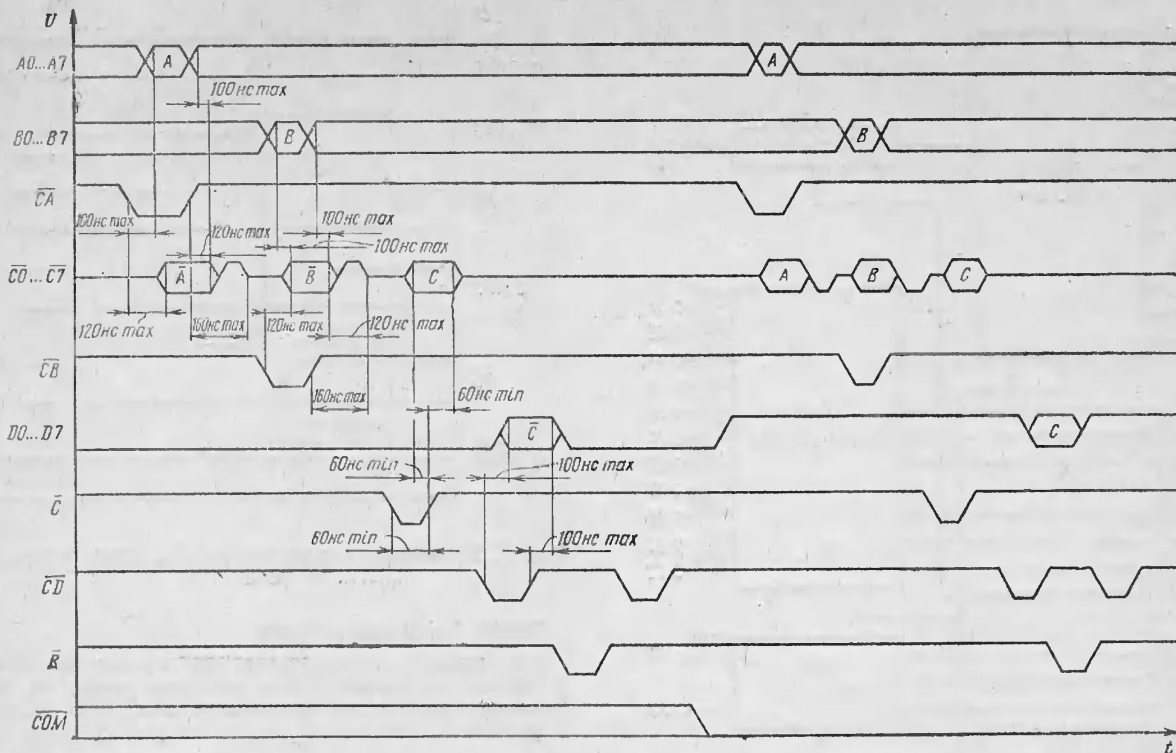


Рис. 3. Временная диаграмма работы устройства передачи информации

**Режим передачи информации**

Условное графическое обозначение микросхемы К1801ВП1-034 в этом режиме приведено на рис. 1, назначение выводов показано в табл. 1, электрическая структурная схема представлена на рис. 2.

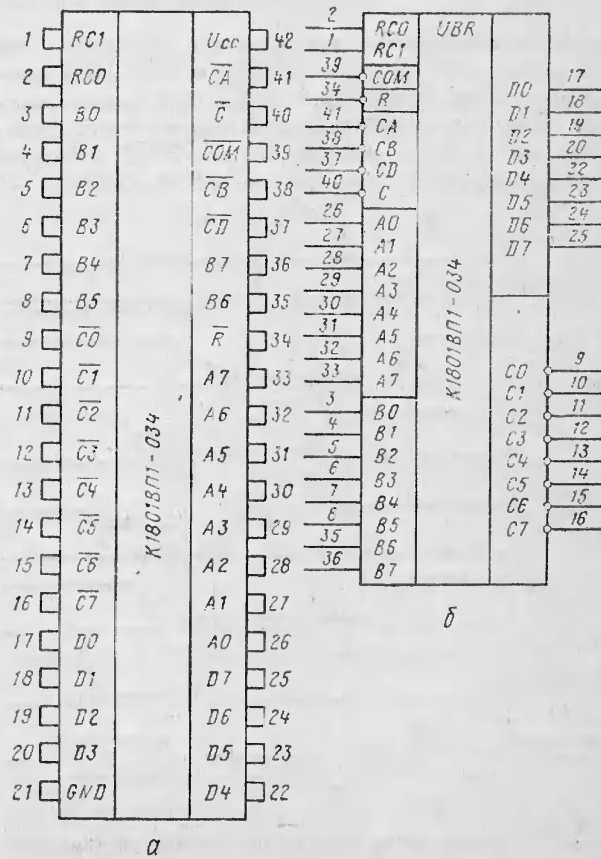
В режим устройства передачи информации (УПИ) микросхема устанавливается подачей на вывод RCO высокого, а на вывод RC1 — низкого уровней напряжения. С помощью управляющих сигналов CA и CB входная информация A0..A7 и B0..B7 соответственно передается на двунаправленные выходы C0..C7 в зависимости от сигнала COM в прямом или инверсном виде (рис. 3). Сигналом C информация с выходов C0..C7 записывается в буферный регистр и с помощью сигнала CD выдается на выходы D0..D7 в зависимости от сигнала COM в прямом или инверсном виде. Сигнал R обнуляет буферный регистр.

Режим выдачи вектора прерывания и компаратора адреса

Условное графическое обозначение микросхемы в этом режиме дано на рис. 4, назначение выводов показано в табл. 2, электрическая структурная схема приведена на рис. 5.

Микросхема устанавливается в режим выдачи вектора прерывания с момента подачи на выходы RCO и RC1 напряжения высокого уровня. Старшие шесть разрядов требуемого адреса вектора прерывания формируются на выводах S11..S16. Временная диаграмма выполнения стандартной процедуры прерывания представлена на рис. 6.

Рис. 4. Условное графическое обозначение микросхемы К1801ВП1-034 в режиме выдачи вектора прерывания и компаратора адреса по порядку расположения (а) и функциональному назначению (б) выводов



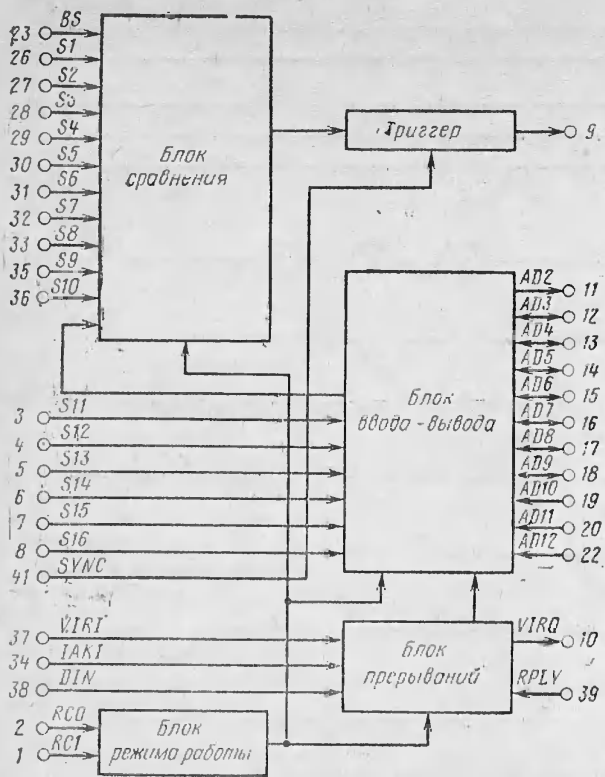


Рис. 5. Электрическая структурная схема устройства выдачи вектора прерывания и компаратора адреса

Адрес, необходимый для сравнения, снимается с выводов S1...S10. Состояния S1...S10 и AD3...AD12 сравниваются при наличии сигнала BS. В результате сравнения вырабатывается сигнал SB, запоминаемый в триггере на все время действия сигнала SYNC. Временная диаграмма работы компаратора адреса приведена на рис. 7.

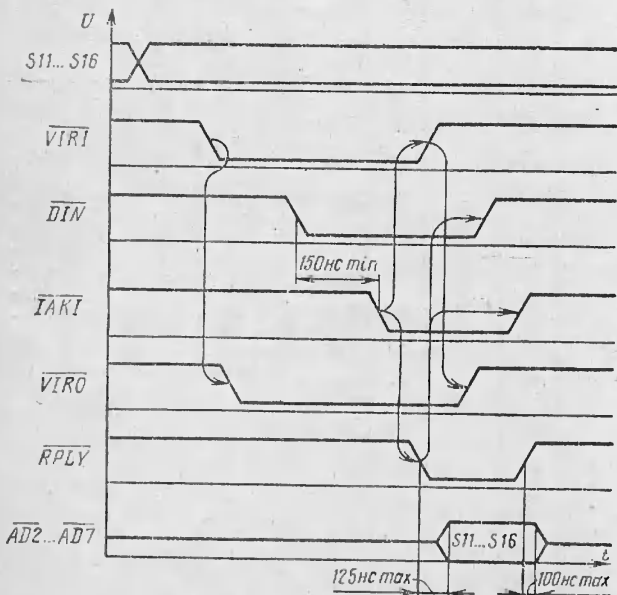


Рис. 6. Временная диаграмма работы устройства выдачи вектора прерывания

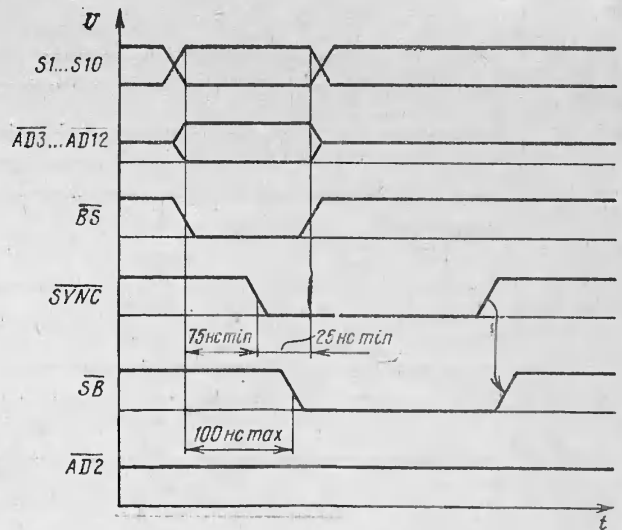


Рис. 7. Временная диаграмма работы компаратора адреса

### Режим буферного регистра

Этот режим устанавливается при напряжении низкого уровня на выводе RCO и высокого уровня на RCI. Условное графическое обозначение БИС К1801ВП1-034 в указанном режиме приведено на рис. 8, назначение

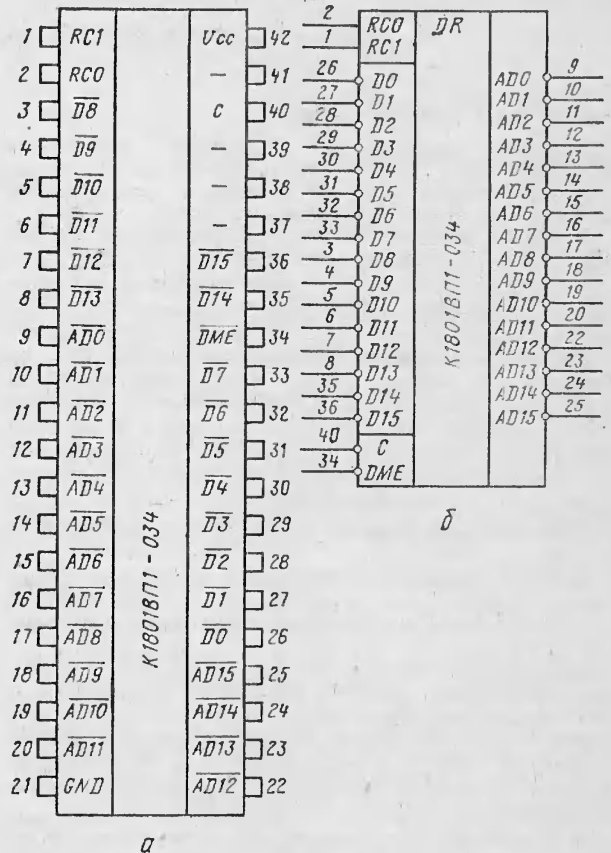


Рис. 8. Условное графическое обозначение микросхемы К1801ВП1-034 в буферном режиме по порядку расположения (а) и функциональному назначению (б) выводов

Таблица 3

Наименование выводов буферного регистра данных  
(RC0=0, RC1=1)

Вывод	Обозначение	Тип вывода	Наименование выводов
1	RC1	Вход	Выбор режима 1
2	RC0	»	Выбор режима 0
3...8	D8...D13	»	Данные D8...D13
9...20	AD0...AD11	Выход	Данные канала AD0...AD11
21	GND	—	Общий
22...25	AD12...AD15	Выход	Данные канала AD12...AD15
26...33	D0...D7	Вход	Данные D0...D7
34	DME	»	Разрешение выдачи данных
35	D14	»	Данные D14
36	D15	»	Данные D15
37...39,41	—	—	Не используется
40	C	Вход	Запись данных
42	U <sub>cc</sub>	—	Напряжение источника питания

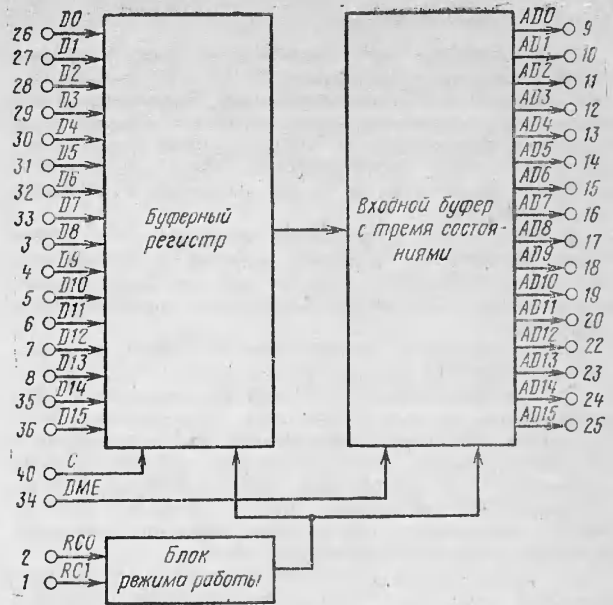
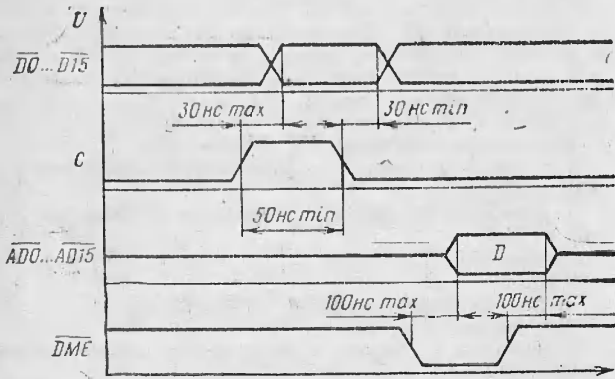


Рис. 9. Электрическая структурная схема буферного регистра данных

Рис. 10. Временная диаграмма режима буферного регистра данных

выводов показано в табл. 3, электрическая структурная схема дана на рис. 9.

Входная информация с выводов D0...D15 сигналом C записывается в 16-разрядный буферный регистр. Сигнал DME разрешает выдачу информации с буферного регистра на выходы AD0...AD15, которые при высоком уровне этого сигнала находятся в отключенном состоянии (рис. 10).

Г. Г. Глушкова, телефон: 208-73-23, Москва

## ПОЛОЖЕНИЕ

о Всесоюзном конкурсе на лучшие разработки профессиональных, учебных, игровых и шахматных программных средств, выполненные самодельными авторами для персональных ЭВМ (ПС ПЭВМ)

1. Всесоюзный конкурс на лучшие разработки профессиональных, учебных, игровых и шахматных программных средств, выполненные авторами для персональных ЭВМ, проводится Государственным комитетом СССТ по вычислительной технике и информатике, ЦК ВЛКСМ, Центральным Советом Всесоюзного общества изобретателей и рационализаторов и Центральным комитетом профсоюза рабочих радиозлектронной промышленности с 1 августа 1988 г. по 31 августа 1989 г.

2. Цель проведения конкурса: всемерное удовлетворение потребностей народного хозяйства и граждан в ПС ПЭВМ; широкое привлечение молодежи, самодельных авторов, членов клубов самодельного технического творчества (КСТТ) и других общественных организаций к совершенствованию существующих и разработке новых ПС ПЭВМ; выявление лучших разработок, пропаганда, обобщение и распространение передового опыта.

### 3. Участники конкурса:

3.1. В конкурсе могут принимать участие члены КСТТ, первичных организаций ВОИР, научно-инженерных обществ (НИО) промышленных предприятий, научно-исследовательских, конструкторских и других организаций, независимо от ведомственной подчиненности, студенческих конструкторских бюро, кружков технического творчества, а также молодежь и самодеятельные авторы.

4.1. На конкурс принимаются разработки ПС ПЭВМ (профессиональные, учебные, игровые и шахматные), при этом обязательно наличие работающего образца или макета, а в случае необходимости, текстового примера.

4.2. На конкурс не принимаются ПС ПЭВМ, разработанные в порядке служебного задания.

4.3. При разработке ПС ПЭВМ рекомендуется ориентироваться на следующие типы персональных ЭВМ: ДВК (всех моделей), «Электроника 85», «Электроника МС 0201», «Искра 1030», «Нейрон», ЕС1840, ЕС1841, «Роботрон 1715», «Роботрон 1834», «Роботрон 1910», «Искра», КУВТ «Корвет», КУВТ «Ямаха», БК0010, БК0011, «Микроша», «Львов ПК», «Криста», «Алогей», «Вектор», «Сура», «ВЭФ-микро», «Агат».

4.4. Рекомендуется написание программ на алгоритмических языках высокого уровня (БЕЙСИК, Фортран, Паскаль, ПЛ/1, Си и др.).

4.5. Рекомендуется следующая примерная тематика ПС ПЭВМ:

- экспертные системы и системы искусственного интеллекта;
- системы обеспечения принятия решений;
- автоматизированные рабочие места специалистов;
- системы автоматизированного проектирования и машинной графики;
- системы автоматизации научных исследований;
- обучающие программы;
- учебные программы;
- шахматные программы.

5. Требования к оформлению материалов на конкурс.

5.1. ПС ПЭВМ должны представляться на машинных носителях.

На конверте для гибкого магнитного диска (магнитной ленты) должны быть надписи следующего содержания:

- название программы;
- тип программы (профессиональная, учебная, игровая, шахматная);
- где и кем сделана программа.

К гибкому магнитному диску (магнитной ленте) необходимо приложить документ, включающий в себя следующее:

- тип машины; язык программирования; операционную систему; потребность в памяти; потребность в периферийных устройствах; время прогона; инструкцию пользования; краткие методические рекомендации по исполнению программы; структурную схему.

5.2. К разработкам, представленным на конкурс, прикладываются сведения об авторе или каждом из соавторов (фамилия, имя, отчество, год рождения, образование, ученая степень, место жительства).

5.3. Разработки, описание которых не соответствует представленным требованиям, не рассматриваются.

### 6. Сроки и порядок проведения конкурса.

6.1. Разработки, материалы и сведения об авторах (соавторах) направляются до 31 июля 1989 г. (дата отправления определяется по штемпелю почтового отделения) в адрес СНПО «Алгоритм» — головную организацию ГКВТИ СССР в развитии самостоятельного творчества, ответственную за организацию проведения

конкурса, по адресу: 109058, г. Москва, Велозаводская ул., д. 4 с пометкой «На конкурс ПС ПЭВМ».

6.2. Все материалы находятся на хранении в СНПО «Алгоритм» и после проведения конкурса авторам не возвращаются (за исключением магнитных носителей информации).

6.3. Для организации работы по проведению конкурса создается организационный комитет.

6.4. Для проведения предварительного изучения и отбора лучших разработок создаются экспертные группы по типам программ (профессиональные, учебные, игровые и шахматные), состав которых утверждается председателем оргкомитета конкурса.

6.5. Для оценки представленных на конкурс разработок экспертные группы привлекают специалистов объединений, организаций и центров информатики ГКВТИ СССР.

6.6. Экспертные группы до 20 августа 1989 г. предварительно рассматривают поступившие материалы и о результатах сообщают оргкомитету конкурса.

6.7. При отборе лучших разработок ПС ПЭВМ как на предварительном этапе, так и при окончательном определении победителей конкурса учитываются следующие факторы:

актуальность, новизна, сфера применения, предполагаемый экономический эффект, социальный эффект, степень завершения ПС ПЭВМ.

6.8. Оргкомитет рассматривает разработки, представленные экспертной группой, и выносит предложение о награждении авторов лучших работ. Итоги конкурса утверждаются совместным решением ГКВТИ СССР, Секретариата ЦК ВЛКСМ, Президиума ЦС ВОИР и Президиума ЦК профсоюза до 30 сентября 1989 г.

6.9. Для поощрения авторов и авторских коллективов — победителей конкурса учреждены 54 диплома и 30 поощрительных премий ГКВТИ СССР, ЦК ВЛКСМ, ЦС ВОИР и ЦК профсоюза, в том числе:

Для профессиональных ПС ПЭВМ:

- 3 диплома I степени с денежными премиями по 1000 рублей;
- 5 дипломов II степени с денежными премиями по 600 рублей;
- 10 дипломов III степени с денежными премиями по 400 рублей;
- 10 поощрительных премий по 50 рублей.

Для учебных ПС ПЭВМ:

- 3 диплома I степени с денежными премиями по 1000 рублей;
- 5 дипломов II степени с денежными премиями по 500 рублей;
- 10 дипломов III степени с денежными премиями по 200 рублей;
- 10 поощрительных премий по 50 рублей.

Для игровых и шахматных ПС ПЭВМ:

- 3 диплома I степени с денежными премиями по 500 рублей;
- 5 дипломов II степени с денежными премиями по 300 рублей;
- 10 дипломов III степени с денежными премиями по 100 рублей;
- 10 поощрительных премий по 50 рублей.

6.10. Для поощрения молодых авторов и молодежных коллективов, разработавших лучшие учебные, игровые и шахматные ПС ПЭВМ, учреждены 30 почетных грамот ЦК ВЛКСМ и 10 турпоездок по линии БММТ «Спутник» в социалистические страны.

6.11. Лучшие разработки, отмеченные оргкомитетом конкурса, будут рекомендованы для использования и внедрения в производство, тиражирования, экспонирования на ВДНХ СССР, публикации в журналах «Микропроцессорные средства и системы», «Изобретатель и рационализатор», «Техника — молодежи» и «НТТМ — умелец».

Телефон: 275-24-93, Москва



УДК 681.3.06

Черкасский В. С., Черных Д. В. К экономическому механизму информационной сферы народного хозяйства // Микропроцессорные средства и системы. — 1988. — № 6. — С. 3.

Рассматривается сущность информационной сферы, особенности производительных сил и производственных отношений при производстве, обмене и потреблении информации. Выдвигаются методические соображения о собственности на информацию и ее правовой охране, о минимальных и максимальных оценках стоимости информации как товара и вытекающем из них ценообразовании на впервые созданную и тиражируемую информацию.

УДК 681.326

Шереметьевский Н. Н., Долкарт В. М. Стандартизация 32-разрядных функционально-модульных систем для промышленных применений // Микропроцессорные средства и системы. — 1988. — № 6. — С. 8.

Проводится сравнительный анализ четырех стандартов, утвержденных в 1987 г. IEEE, на 32-разрядные магистрали для объединительных панелей функционально-модульных систем. Делается вывод, что стандарт 1296 должен и в нашей стране быть основным промышленным стандартом на 32-разрядную магистраль для функционально-модульных систем промышленной автоматизации.

УДК 681.325.5

Артыунов В. Г., Глухенький В. В., Лернер Р. А., Макеенок А. Н., Носенко Л. В., Страутманис Г. Ф. Цифровой процессор KM1813BE1: подготовка и отладка программ // Микропроцессорные средства и системы. — 1988. — № 6. С. 14.

Описано автономное функционально-заключенное специализированное устройство RASTR, обеспечивающее решение всего комплекса задач по разработке и проектированию ППО для однокристального цифрового процессора обработки сигналов с аналоговыми устройствами ввода-вывода KM1813BE1.

УДК 621.397

Власенко В. А., Коханов А. Б. Получение твердых копий графических и полутонных изображений // Микропроцессорные средства и системы. — 1988. — № 6. — С. 29.

Рассматривается программа вывода на печать полутонных изображений методом контактного растра с помощью АЦПУ Роботрон-6329.02.

УДК 687.3.022

Ковалев В. М., Чернис Б. З. Цветная графика реального времени для микроЭВМ типа «Электроника 60» // Микропроцессорные средства и системы. — 1988. — № 6. — С. 35.

Предлагается пакет программ цветной графики, учитывающий особенности поступления данных в темпе реального времени, обеспечивающий многооконный интерфейс и отображающий временные зависимости в виде графиков или гистограмм с линейным или логарифмическим масштабированием.

УДК 437.78 : 681.142.2(73)

Лигская А. В. Проблемы защиты авторских прав на программное обеспечение в США // Микропроцессорные средства и системы. — 1988. — № 6. — С. 39.

Рассматриваются экономические вопросы, связанные с защитой ПО как интеллектуальной собственности в США, возникшие в 80-е годы в связи с быстрым ростом рынка готовых программ, распространением незаконного копирования программ и плагината в программировании.

UDC 681.3.06

Cherkassky V. S., Chernykh D. V. On the economic mechanism of informatics industry // Microprocessor devices and systems. — 1988. — N. 6 — P. 3.

The review of nature of informatics industry, the peculiarities of productive forces and relations of production during manufacturing, exchange and consumption of information is given. The methodic approach to the problems of property of information and its defence by means of law, the maximum and minimum cost evaluations of information as one of specific goods is given. The corresponding problems of marketing prices of originally created and copied information are also considered.

UDC 681.326

Sheremetievsky N. N., Dolkart V. M. 32-bit modular systems standards for industrial applications // Microprocessor devices and systems. — 1988 — N. 6 — P. 8.

The comparison of four standards on 32-bit backplane data buses for modular systems submitted by IEEE in 1987. The authors predict 1296 standard to become main standard for 32-bit dataway for functionally-selfcontained modular industrial automation systems.

UDC 681.325.5

Artyuchov V. G., Glukhenky V. V., Lerner R. A., Makeenok A. N., Nosenko L. V., Strautmanis G. F. Digital signal processor KM1813BE1: programming and debugging // Microprocessor devices and systems. — 1988. — N. 6 — P. 14.

The stand-alone specialized development system RASTR is described. The system supports the whole technology of application software development for single-chip digital processor with analogue input-output ports type KM1813BE1.

UDC 621.397

Vlasenko L. A., Koshanov A. B. Hard-copying graphics and semi-tone images // Microprocessor devices and systems. — 1988. — N. 6 — P. 29.

The program for printing hard copies of semi-tone images using contact raster on EPSON-compatible printer Robotron CM6329.02 is reviewed.

UDC 681.3.022

Kovalev V. M., Chernis B. Z. Real-time colour graphics for "Elektronika 60" microcomputer // Microprocessor devices and systems. — 1988. — N. 6 — P. 35.

The colour graphics software kit utilizing the specifics of data traffic in realtime tasks is proposed. The kit supports multi-window interface and displays line-domain functions as X-Y plots or histograms in linear or logarithmic scales.

UDC 437.78:681.142.2(73)

Ligskaya A. V. The problem of software author rights in the USA // Microprocessor devices and systems. — 1988. — N. 6 — P. 39.

The discussion of some economic problems related with copyright on software as an intellectual property in the USA, which appeared in the 80-ties as the cause of fast software market growth, bootleg program copying and distribution, and compilative programming.

УДК 681.3.07

Банников С. Ю., Марфенко К. С., Подлепский Б. И., Торубаров С. В. Программно-аппаратный комплекс КПС-1813 для проектирования биомедицинских устройств с цифровым процессором сигналов KM1813BE1 // Микропроцессорные средства и системы. — 1988. — № 6. — С. 55.

Даны аппаратные и программные характеристики комплекса кросс-средств КПС-1813, ориентированного на работу с микроЭВМ типа ДВК и «Электроника 60» и состоящего из ассемблера-дисассемблера AS 1813, программной модели (имитатора) цифрового процессора сигналов (ЦПС) SM1813, драйвера программатора PR1813 и специального программатора.

UDC 681.3.07

Bannikov S. Yu., Marfenko K. S., Podlepetsky B. U., Torubarov S. V. A software-hardware system CPS-1813 and its application for the design of biomedical devices using digital-analog processor type KM1813BE1 // Microprocessor devices and systems. — 1988. — N. 6 — P. 55.

The hardware and software features of cross-system CPS-1813, running on LSI-11 compatible computer are described. The system consists of Assembler-Disassembler program A1813, Simulation program which immitates digital signal processor IH813, PROM programmer driver and special PROM programmer unit.

УДК 681.326

Азин С. Н., Муравьев Н. И. Модуль ввода вывода видеосигнала в память микроЭВМ «Электроника 60» // Микропроцессорные средства и системы. — 1988. — № 6. — С. 62.

Представлен простой модуль ввода с телекамеры оцифрованного видеосигнала в память микроЭВМ и вывода преобразованного в видеосигнал содержимого памяти на бытовой телевизор. Описано использование для этих целей дополнительной памяти на платах ПЗ, технические характеристики модуля, его работа, организация экранного ОЗУ в виде электронного диска. Приведены принципиальная схема модуля и текст драйвера электронного диска.

UDC 681.326

Asin S. N., Muraviev N. I. Videodigitizer and video output unit for "Electronika 60" microcomputer. // Microprocessor devices and systems. — 1988. — N. 6 — P. 62.

A simple unit providing digitization of the TV-camera signal and its fetching into computer RAM and also performing conversion of digital image stored in RAM into video displayed on the screen of home TV is described. The system accesses additional 16 KW RAM boards type ПЗ, organized as RAM disk. Technical specifications of digitizer are given. The circuit diagram of the unit and RAM disk handler program are included.

УДК 616.1.072

Водоватов Ф. Ф., Панов В. В., Сорокин А. В. Применение персонального компьютера в массовой диагностике и тренинге ритма сердца // Микропроцессорные средства и системы. — 1988. — № 6. — С. 76.

Описана методика биологической обратной связи по RR-интервалу ЭКГ как метод лечения нарушения ритма сердца и снижения психофизиологической реактивности в условиях эмоционального стресса. Измерения данных кардиоритма во время обучения обрабатываются серийно выпускаемой бытовой микроЭВМ «Микроша».

UDC 616.1.072

Vodovатов F. F., Panov V. V., Sorokin A. V. Personal computer application for wide-spread heart diagnostics and training // Microprocessor devices and systems. — 1988. — N. 6 — P. 76.

The "biological feedback" method using R-R interval in electrocardiogram is described, which helps treating heart rhythm disorder. The method is useful for reducing psychophysiological reactivity under emotional stress. ECG rhythm processing during training session is performed by "Microsha" personal comuter.

УДК 621.317.76

Глушенко К. А., Кирианки Н. В., Котыло О. Б., Юриш С. Ю. Измерители частотно-временных параметров сигналов на базе ОЭВМ серии K1816 // Микропроцессорные средства и системы. — 1988. — № 6. — С. 80.

Рассмотрен низкочастотный МП-частотомер. Приведен вариант схемной реализации, алгоритма измерения и текста программы измерения частоты для МП типа K1801.

UDC 621.317.76

Glushenko K. A., Kirianaki N. Y., Kotylo O. B., Yurish S. Yu. The time/frequency measurement with single-chip computers of K1816 family. // Microprocessor devices and systems. — 1988. — N. 6 — P. 80.

The LF frequency counter is described. The possible circuit diagram, measurement algorithm and program listing for a frequency meter built around K1816 microprocessor are given.

УДК 621.326

Белов Л. А., Бухаров Д. Г., Демидов В. М., Фомичев М. Ю. Широкодиапазонный микропроцессорный частотомер // Микропроцессорные средства и системы. — 1988. — № 6. — С. 82.

На основе микропроцессорного комплекта K580 разработан автоматический частотомер, работающий в диапазоне частот 1 мГц... 1МГц с индикацией четырех десятичных знаков частоты. Приводятся принципиальная схема и ПО микропроцессора.

UDC 621.326

Belov L. A., Bucharov D. G., Demidov V. M., Fomichev M. Yu. Wide-range microprocessor-based frequency counter. // Microprocessor devices and systems. — 1988. — N. 6 — P. 82.

The 4 digit autoranging frequency counter for 1mHz... 1MHz frequency range is developed. The device incorporates K580 LSI family chips. The circuit diagram of the counter and supporting software list are included.

Публицистика, философия, экономика, компьютеризация

- Черкасский В. С., Черных Д. В.— К экономическому механизму информационной сферы народного хозяйства 6  
 Антонюк Б. Д.— О западно-европейских программах в области информатики 3  
 Беляков Б. Н.— Анализ и синтез экономики и информатики (отклик) 6

МИКРОПРОЦЕССОРНАЯ ТЕХНИКА

- Алумян Р. С., Степанян С. О., Папян Г. Г.— МикроЭВМ с сокращенным набором команд 3  
 Артюхов В. Г., Глухенький В. Б., Лернер Р. А., Макесюк А. Н., Носенко Л. В., Страутманис Г. Ф.— Цифровой процессор КМ1813ВЕ1: подготовка и отладка программ 6  
 Бобков В. А., Горовой В. В., Лаврусенко В. А., Черняковский Д. Н.— Одноплатная микроЭВМ на основе БИС серии КР588 1  
 Бобков В. А., Чернуха Б. Н., Черняковский Д. Н., Свиридович В. С., Кушарев В. Н.— Селектор адреса КР588ВТ1 1  
 Гуревич М. Х., Кузнецов А. А., Луцкий И. Ю., Страутманис Г. Ф.— Однокристалльная ЭВМ КМ1813ВЕ1 с аналоговыми устройствами ввода-вывода 5  
 Донев В. С.— Микропроцессорный комплект БИС серии СМ600 5  
 Кобылинский А. В., Калатинцев В. М., Заика А. И.— Программируемый контроллер клавиатуры и индикации КР580ВВ79 1  
 Кушарев В. Н., Свиридович В. С., Чернуха Б. Н., Черняковский Д. Н., Бобков В. А.— Контроллер прямого доступа к памяти К588ВТ2 1  
 Кушнир В. Е., Панфилов Д. И., Шаронин С. Г.— Многофункциональный комплекс программно-аппаратных средств для семейств однокристалльных ЭВМ серии К1816 4  
 Морозов С. А., Черкай А. Д., Минкин Л. К., Семичастнов О. Л., Кротков Б. В.— Однокристалльная 4-разрядная МЭВМ КБ1013ВК7-2 3  
 Прохоров Н. Л.— Особенности архитектуры и программного обеспечения вычислительного комплекса СМ1700 2  
 Таланов В. А.— МикроЭВМ с маркой ВЭФ 5  
 Хацкевич Л. Д., Проценко И. Г.— Профессиональная персональная ЭВМ «Электроника МС0585» 2  
 Шереметьевский Н. Н., Долкарт В. М.— Стандартизация 32-разрядных функционально-модульных систем для промышленных применений 6

Периферийное оборудование

- Бронштейн Р. А., Кашкадаев В. А., Клименко С. В., Копылов С. В., Порнов С. М.— Одноплатный электронный диск для микроЭВМ «Электроника 60» и ДВК 2  
 Динес В. Л., Мартюхин В. А.— Модификация телевизионного монитора КТС ЛИУС-2 — под дисплей ВТА 2000 или РИН-609 4  
 Бронштейн Р. А., Евтехов А. С.— Электронный диск с энергонезависимым хранением информации для микроЭВМ «Электроника 60» и ДВК 2  
 Каменецкий С. В.— Способ подключения 16-контактной клавиатуры к КМ1816ВЕ48 4  
 Кузнецов С. О., Ланко А. А., Леонтьев Д. И., Матвеев О. В., Прохоров Н. Л., Раев В. К., Шотов А. Е.— Электронный диск СМ 5803 для микроЭВМ с интерфейсом «Общая шина» 2  
 Новиков Е. А., Чадаев В. Г.— Сопряжение кассетного накопителя на магнитной ленте СМ 5211 с интерфейсом МПИ 2

- Нунупаров Г. М.— Сопряжение микроЭВМ К1827ВЕ1 с внешним ППЗУ 4  
 Тарасов В. В., Сморгачев Ю. Н., Захаркин Д. Б.— Интерфейс канала общего пользования для микропроцессорных приборов 2  
 Соколов А. Н., Бремшмидт В. Г., Зайдель В. С.— Контроллер НСМД ЕС-5061 для микроЭВМ типа «Электроника 60» 4  
 Грачев А. Н., Ильин Е. П., Хухлаев Е. В.— Использование дисплея «Графит» в качестве удаленного графического терминала ИМС КАПРИ 4  
 Динес В. Л., Ардашев Н. Н., Мартюхин В. А.— Устройство сопряжения микропроцессорной системы КТС ЛИУС-2 с дисплеем РИН-609 4  
 Падирияков Ю. А., Белинский В. Т., Журило В. А.— Интерфейс накопителя на гибких магнитных дисках Потемкин М. И., Бронштейн Р. А.— Драйвер 133-миллиметровых дисководов для ЭВМ ряда ДВК 3  
 Стрижов Р. В., Пашкевич С. В.— Разрешение конфликтной ситуации при работе двух процессоров с общим полем оперативной памяти 2  
 Шубин С. Ф.— Сопряжение дисплея СМ 7204 с микроЭВМ «Электроника 60» 4

ПЕРСОНАЛЬНЫЕ КОМПЬЮТЕРЫ

- Гнездилова Г. Г., Гончаров О. А., Сенин Г. В.— К типологии компьютерных игр 5  
 Домарацкий С. Н., Шраго И. Л.— Организация экрана монитора ПЭВМ «Искра 1030» в программах автоматизации научных исследований 5  
 Иванов В. И., Клепов А. В., Костиков Д. В.— Малогабаритные персональные ЭВМ с жидкокристаллическим экраном — новый класс персональных ЭВМ 4  
 Иванов Е. А.— Совместимость ПЭВМ с помощью аппаратно-программных сопроцессоров 4  
 Попов А. Л.— Комплекс технологических дискет для ПЭВМ 5  
 Лопатин В. С., Матвеев В. М., Юрочкин А. Г.— МикроЭВМ МС 0102 с кеш-ОЗУ 6  
 Пройдаков Э. М.— Второе поколение персональных ЭВМ фирмы ИВМ 6

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

- Александров А. В.— Автоматизированная система по выбору элементной базы для технологических контроллеров 5  
 Байков А. Е., Виставкин А. Н., Катков Б. М., Мартынов А. Э.— Диалоговые средства графического взаимодействия для систем автоматизации эксперимента 4  
 Баяковский Ю. М., Михайлова Т. Н., Рыжова И. Г.— Реализация базовой графической системы для ПЭВМ 1  
 Белов А. М., Муренко Л. Л., Шестипалынов А. С.— Комплекс кросспрограмм «Электроника МИКРОСС-580/85» 2  
 Блинов Д. М.— Редактор правил 2  
 Блинов Д. М.— Редактор данных 2  
 Брябрин В. М., Блинов Д. М.— Классификационная экспертная система 2  
 Глазов А. Б., Костарев С. А., Суханова Е. В.— Быстрое преобразование Фурье на микропроцессоре КР5801К80А 4  
 Глухов В. Н., Долбилов Л. А., Дудников Е. Е., Лоозе И.— МикроДОС — адаптивная система программного обеспечения для 8-разрядных микроЭВМ 4  
 Журавский Д. А., Барков С. Н.— Операционная система СР/М с ПЗУ 5  
 Кетков Ю. Л.— Графические средства в алгоритмическом языке БЕИСИК 1  
 Кривенко А. Н.— Операционная система для ВУМС-001-01 4  
 Кипаренко В. В.— Стандартизация интерфейсов микропроцессорных систем (Приложение) 1

- Корнейчук В. И., Кочинва Э. А.—Пакет программ для решения алгебраических и трансцендентных уравнений на микроЭВМ типа ДВК2 2
- Погода А. В.—Пакет подпрограмм на ассемблере для арифметических вычислений 5
- Пройдаков Э. М.—Дизассемблер, управляемый данными 5
- Росошинский Д. А., Ковальчук-Химюк Л. А.—Операционная система реального времени для микроЭВМ 3
- Савченко А. В., Филиппов А. Н., Полетаев В. М., Муратов Д. М., Пасхин А. Е.—Комплексы сервисных программных средств для микроЭВМ 2
- Сидристый А. Ю.—Трехуровневый информационный интерпретатор командных строк 4
- Тимофеев Е. С., Васильев В. Н., Васильев Н. П.—Организация сложных файловых систем в среде ОС РАФОС 2
- Операционные системы реального времени**
- Анисимов А. А., Анисимов Г. А.—Режим разделения времени в системах с однозадочным монитором 4
- Архангельский А. Н., Орехов А. А.—Мультипрограммирование на языке Си 3
- Гальченко А. А., Самойлов В. В.—Ядро операционной системы реального времени 3
- Ключихин А. А.—Тест ОЗУ с сохранением текущей информации для систем реального времени 3
- Корнеев В. В., Тарков М. С.—Операционная система микромашиной вычислительной сети с программируемой структурой МИКРОС 4
- Языковые средства**
- Брылев С. П.—Программирование встроенных микропроцессоров с применением компилятора Паскаль/MT+ 4
- Вершингора Р. В.—Сопрограммная связь функций в языке Си 4
- Гайворонский М. А.—Простой способ адаптации развитых программных средств для работы со стандартным дисплеем микросистемы 4
- Пройдаков Э. М.—Использование макроопределений в языке ПЛ/М 6
- Таюпов А. Р.—Методы программирования аппаратуры КАМАК на языке Паскаль 6
- Филин А. Л.—Организация программ динамической структуры при разработке интегрированных систем с использованием языка высокого уровня 4
- Шилин А. Е.—Программный эмулятор команд расширенной арифметики 4
- Средства отладки**
- Бернов А. В., Порозов В. Н.—Аппаратно-программный комплекс программирования и отладки контроллера на основе микропроцессоров серии K1801 5
- Бродни В. Б., Шагури И. И.—Микропрограммируемый схемный эмулятор для отладки микропроцессорных систем 5
- Бухвостов Д. Ю., Жилияев Ю. П., Иванов А. И., Казанцев В. Ю., Печенкин А. Н.—«Цель» — простой программно-аппаратный комплекс для отладки встраиваемых управляющих микроЭВМ на базе МПК БИС серии КР580 5
- Галаган В. Г., Дереза Н. Г., Журило В. А., Некрасов Б. А., Зубрицкий Л. Л.—Программно-аппаратный комплекс программирования логических матриц и ППЗУ 2
- Гладышев В. В.—Энергонезависимое АЗУ в качестве имитатора ПЗУ 2
- Зубрицкий Л. Л.—Программно-аппаратный комплекс программирования логических матриц и ППЗУ 2
- Ерухимов П. Г., Вахитов Р. Р.—Стеид «Микротест» для диагностики одноплатного микроконтроллера «Электроника МС 2702» 5
- Конько В. В., Штурман А. И., Недорубан В. А.—Аппаратно-программный отладочный комплекс 5
- Кривцов А. Н.—Сигнатурный анализатор 5
- Муренко Л. Л.—Отладочный комплекс на базе ПЭВМ типа ДВК для однокристалльных микроЭВМ серий КМ1814, КМ1820, КМ1816 5
- Ордынский М. А., Вагреева И. Г., Нехамкин Ю. Э.—Резидентный монитор для отладки программного обеспечения 5
- Пальгин П. И., Рождественский С. М., Шагури И. И.—Микропрограммируемый комплекс для тестирования и отладки микропроцессорных систем 5
- Потапенко О. Д.—Сигнатурный анализатор 2
- Табаткин В. М.—Имитатор ПЗУ для однокристалльной ЭВМ 2
- Тарасов В. В.—Организация передачи управления монитору и прикладным программам в отладочном устройстве 5
- Щелиунов Н. Н., Дианов А. П.—Организация программного обеспечения программаторов ПЗУ 2
- Локальные сети**
- Гревцев В. В.—Средства передачи данных микроЭВМ семейства СМ1800 2
- Гыбин Е. Н., Козаренко С. В., Левин К. М.—Контроллер локальной сети 2
- Кокотов В. М., Тимофеев В. И.—Неоднородная локальная сеть в учебной аудитории 2
- Колосков М. С., Кузнецов А. Л., Кожевников Ю. Б.—Локальная сеть микро- и мини-ЭВМ 2
- Корнюшко В. Ф., Авдеев В. Н., Фролов Г. М., Жедь А. Ю.—МикроЭВМ в распределенной вычислительной системе 2
- Кузнецов С. Г., Ромашко В. М.—Межмашинная связь в двухуровневой симметричной системе ЭВМ 2
- Подвальный С. Л., Михин Ю. А., Кравец О. Я.—Локальная вычислительная сеть учебной лаборатории 2
- Межмашинная связь**
- Лебедев Ю. А., Рябов С. А., Шабанов Г. А., Рыбченко А. А.—Программно управляемый модем 2
- Рогоза В. В., Сорочинский В. В., Холоденко Ю. Н.—Устройство для организации внутрисистемной связи 2
- Обработка сигналов**
- Горшков А. Н.—Генератор импульсов, встраиваемый в микропроцессорную систему 2
- Курьев С. В., Усманов И. З.—Интегрирующий АЦП, сопрягаемый с микропроцессором 2
- Рассанов Д. Н., Ткачев В. Г., Шакиров А. А.—Простой АЦП на основе БИС программируемого таймера КР580ВИ53 2
- Овчаренко А. И.—Универсальный АЦП частотно-временных параметров с интерфейсом ЛИУС-2 2
- Ходаковский Е. А.—Счетчик событий для микропроцессорной системы 2
- Чабан С. Д., Скрыбин С. Г.—Устройство выборки и хранения аналогового сигнала 2
- Машинная графика**
- Адрианов В. Г., Сухов Д. А.—Драйвер для графопостроителя типа Н-306, Н-307 с аналоговым входом 6
- Баяковский Ю. М., Галактионов В. А., Ходулев А. Б.—Геометрические преобразования в высокопроизводительных графических станциях 3
- Безобразов В. С., Мякотин А. Е., Шишкевич А. А.—Контроллер цветного графического дисплея для персональных ЭВМ 3
- Власенко В. А., Коханов А. Ю.—Получение твердых копий графических и полутонных изображений 6
- Воробьев Н. В., Безобразов В. С.—Графический манипулятор мышь для персональных ЭВМ 3
- Гехман Б. И.—Диалоговая система на микроЭВМ

«Электроника ДЗ-28» для подготовки графических иллюстраций  
 Големинов Н. Г., Званцев А. А.—Регенеративный графический дисплей  
 Добуш Ю. Д., Старостенко О. В., Натопта Е. К.—Клавиатура на базе однокристалльной микроЭВМ КМ1816ВЕ48  
 Долгушев В. Г., Коваленко В. А., Левшина А. В.—Автоматизация чертежно-графических работ на микроЭВМ «Электроника 60»  
 Злотник Е. М., Киркоров С. И., Стежко И. К.—Графический адаптер для ППЭВМ ЕС 1840  
 Камалаягин А. А., Эгипги Э. Р.—Расширение графических возможностей микроЭВМ «Электроника ДЗ 28»  
 Кацман В. Е., Саталкин А. В.—Система визуального вывода полупроводниковых изображений на базе ПЭВМ «Искра 226»  
 Ковалев Б. М., Чернис Б. З.—Цветная графика реального времени для микроЭВМ типа «Электроника 60»  
 Кулаичев А. П.—Графический БЕИСИК для ДВКЗ  
 Лигская А. В.—Проблемы защиты авторских прав на программное обеспечение в США  
 Мирзоян С. Р.—Интерпретатор языка БЕИСИК с графическими возможностями  
 Тепляков Л. Г.—Программа получения твердой копии графического дисплея ДВКЗМ  
 Чесалин Л. С., Бернштейн М. Н., Байкин В. М., Ильин А. А., Шибер Ю. Г.—Компактная система обработки полутонной и графической информации  
 Шналько В. И.—Драйвер графического дисплея для ДВКЗ  
 Яблонский Ф. М., Усмонов У. Ю., Влахов С., Косев И., Павлов Д., Радонов Г.—Микропроцессорный комплекс для изучения средств отображения информации

## ПРИМЕНЕНИЕ МИКРОПРОЦЕССОРНЫХ СРЕДСТВ

Айзман М. И., Лобанов В. И., Митрофанов А. В., Черняков М. С., Широков А. Н.—Микропроцессорная система контроля тактофонов  
 Акинфин В. Г., Кальмансон В. А., Красовицкий В. З., Нечаев В. П., Трайнин М. М., Эпштейн Г. Р.—Автомат электронного фотонабора с микропроцессорным управлением  
 Бабаян А. В., Горьковен Ю. М., Вартанов А. С., Фрезе Г. Е.—Микропроцессорный пульт технолога — оператора  
 Бачериков В. В., Манин С. М., Томашевская А. В.—Многопроцессорный комплекс на основе микроЭВМ «Электроника 60» и миниЭВМ «Электроника 100/25»  
 Бородин С. М., Новиков Ю. В., Поддубный А. П., Томчук А. А.—Средства отображения информации для микропроцессорных систем измерения, контроля и управления  
 Богатырев В. А., Данилова Л. Д., Иванов Л. С., Каменев А. В., Куконин А. Ю., Охалкин А. В.—Устройство для построения многомашинных вычислительных систем на базе микроЭВМ «Электроника 60»  
 Гуржуенко И. В., Липидус В. Ю., Летник Л. А.—Кроссассемблер для МП-КР580 на БЕИСИКЕ  
 Дианов А. П., Щелкунов Н. Н.—Модули программирования логических схем  
 Домарацкий С. Н., Лозовой Л. Н.—Практическая реализация автоформализации профессиональных знаний при разработке диалоговых средств автоматизации  
 Жуков А. А., Кузин В. Н., Панов М. Ю., Сомов А. В., Хмелевский А. М.—Микропроцессорная автоматизированная система учета рабочего времени  
 Жуков Ю. И., Кармалин П. Н., Боровиков А. В.—Программирование ПЗУ в модуле 15УЗПП.16Кx16 в конструктиве «Электроника 60» под ОС РАФОС

Кулик О. В., Андронов М. А.—Модуль сопряжения БИС серии КР580 с магистралью КАМАК  
 Коноплев Е. А.—Режим ожидания в накопителях на гибком магнитном диске  
 Кисилевский Я. Ф., Пикусов С. Г.—Использование двойной плотности записи на НГМД «Электроника ГМД-7012» при работе с микроЭВМ «Электроника МС 1201»  
 Кочубин В. Б.—Двойная плотность записи на гибких магнитных дисках «Электроника ГМД-7012»  
 Мышкин И. Л., Щербаков А. С.—Интерфейс накопителя на гибких магнитных дисках «Электроника ГМД17012»  
 Петух А. М., Силагин А. В.—Предэкранный ввод информации в диалоговых вычислительных комплексах  
 Сасов А. Ю.—Система цифровой обработки изображений на базе микроЭВМ, совместимых с IBM PC  
 Слюсар А. Б., Мороз Ю. В., Васильев В. Г.—Цифровой регулятор скорости на основе однокристалльной микроЭВМ РВЕ035  
 Чернявский В. Е., Маханек М. М.—Устройство приоритета в базисе ПЛ/М — комплектность и быстролействие  
 Шаталов А. В.—Устройство автоматического запуска микроЭВМ «Электроника 60»

## САПР

Баталов Б. В., Русаков С. Г., Савин В. В.—Пакет прикладных программ автоматизации схемотехнического проектирования для персональных компьютеров  
 Кизуб А. В.—РИСК — реально используемая сетка конструкторов  
 Михайлов С. А.—Пакет САПР конструкторских документов  
 Семенов О. И., Бокуть В. В., Гриншпан Л. А., Малюш Я. Т., Ткачев Е. Е., Шерлин Д. Р.—Система автоматизированного проектирования мультимикропроцессорных устройств на секционированных микропроцессорах

## Устройства связи с объектами

Азин С. Н., Муравьев Н. И.—Модуль ввода-вывода видеосигнала в память микроЭВМ «Электроника 60»  
 Банников С. Ю., Марфенко К. С., Подлепецкий Б. И., Торубаров С. В.—Программно-аппаратный комплекс КПС-1813 для проектирования биомедицинских устройств с цифровым процессором сигналов КМ1813ВЕ1  
 Бокарев А. В., Кабанов А. И.—Адаптер — преобразователь адреса  
 Васильев А. В., Новиков Е. А., Ширшов В. Н., Распоян С. П., Ничков И. Ф.—Двухнаправленный интерфейс связи с микроЭВМ «Электроника ДЗ-28»  
 Вдовин С. Е., Волыничук В. Н., Зибров И. Н., Ковальчук В. Т.—Применение БИС К1802ВР2 для построения многоуровневых устройств деления  
 Груздев А. В., Федонов Е. Н.—Устройство ввода-вывода аналоговых сигналов микроЭВМ  
 Митрофанов Д. Б., Сулейманов А. Д.—Преобразователь ЦАП—АЦП для работы с процессором КР580ИК80  
 Михуткин А. В.—Контроллер КР580ВН59 в устройствах приоритетного прерывания микроЭВМ «Электроника 60»  
 Ройзен М. И., Карпушов А. Н., Бургасов А. Ю.—Система организации прямого цифрового управления на базе микроЭВМ типа «Электроника 60»  
 Омельченко В. П., Семенов С. И., Трегубов С. И.—Устройство ввода многоканальной информации в микроЭВМ «Электроника 60»

## Измерительные системы

- Белов Л. А., Бухаров Д. Г., Демидов В. М., Фомичев М. Ю.— Широкодиапазонный микропроцессорный частотомер 6
- Германенко А. В., Миньков Г. М.— Применение микроЭВМ «Искра 1256» для исследования быстропротекающих однократных процессов 1
- Глущенко К. А., Кирианаки Н. В., Котыло О. Б., Юриш С. Ю.— Измерители частотно-временных параметров сигналов на базе однокристалльных микроЭВМ серии К1816 6
- Горшков Б. Л.— Особенности построения измерительных трактов для систем автоматизации на основе ЭВМ семейства «Электроника 60» 4
- Глущенко К. А., Кудыкевич В. Б., Котыло О. Б.— Реализация функции измерения частоты в микропроцессорных системах 4
- Крянев А. С., Белянкин П. В.— Устройство сбора аналоговой информации для микроЭВМ «Электроника 60» 1
- Суранов А. Я., Госьков П. И., Якунин А. Г.— Микропроцессорная система измерения параметров бинарных изображений на основе многоэлементного фотоприемника 1
- Тимофеев Е. Ю.— Сопряжение персональных ЭВМ с приборным интерфейсом 1
- Хромов А. В., Покровский В. В.— Восьмиканальный микропроцессорный ультразвуковой дальномер 4

## Многомашинные комплексы

- Богатырев В. А., Куколин А. Ю., Иванов Л. С.— Контроллер многомашинной вычислительной системы с общей магистралью 5
- Бух-Винер П. В.— Сеть микроЭВМ в режимах реального времени 5

## В блокнот разработчика

- Беляев Б. И.— Дополнительные 7 Кбайт оперативной памяти для микроЭВМ «Электроника МС 1201.01» 5
- Жданов В. И., Бобылев В. Н., Гриль Н. Ф., Уткина Т. Г.— Автономные обучающие устройства для ЭВМ серии К1816 5
- Жуковский И. Н., Логвиненко И. В., Скрипник И. Ю., Цымбурская Л. В.— Волоконно-оптический интерфейс в адаптивной системе диагностики электроустановок 5
- Тененбаум Я. П., Туркот А. И.— Организация вычислительного процесса в управляющем вычислительном комплексе на базе микроЭВМ «Электроника 60» 5

## Медицинские приложения

- Барановский А. Ю., Неверова А. Ю., Филинов В. Н., Цветкова Т. Л.— Программное обеспечение прогноза течения и исходов язвенной болезни желудка 6
- Водоватов Ф. Ф., Панов В. В., Сорокин А. В.— Персональный компьютер в массовой диагностике и тренинге ритма сердца 6
- Кочин И. В., Берестихевский С. И., Колосова Т. В.— Компьютерная технология в социальной профилактике заболеваний 6

## УЧЕБНЫЙ ЦЕНТР

- Антоненко В. П., Ярощук Н. В., Копачевский О. Н.— Прямая передача данных ЭВМ «Электроника ДЗ-28» 3
- Барметов Ю. П., Боев С. А., Евтеев Ю. И.— Анализатор сигналов акустической эмиссии на основе микропроцессорной системы 4
- Бронштейн Р. А.— Программно-аппаратные средства расширения возможностей микроЭВМ «Электроника 60» и ДВК 4
- Геращенко В. Н., Камков А. А., Лукоянов В. Ю., Панов М. Ю.— Модульная микропроцессорная система ДРМО 3
- Геращенко В. Н., Камков А. А., Лукоянов В. Ю., Панов М. Ю.— Набор модулей и шин для построения простых микропроцессорных систем 3

- Гершкович С. К., Крынкин В. В., Востропятюв А. Ю.— Автоматизированный учебно-лабораторный комплекс Джиган В. И., Луженкий С. Г.— Модуль электрически перепрограммируемого ПЗУ для микроЭВМ семейства «Электроника 60» 4
- Добеш М., Коломиец А. В.— Аппаратные средства обеспечения пошагового режима для микропроцессорных систем на базе БИС КР580ИК80 3
- Журавский Д. А., Барков С. Н.— Организация управления расширенной памятью для процессора КР580ИК80А 4
- Курмаев О. Ф., Балабанов А. А.— Контроллер динамической памяти 1
- Панфилов Д. И., Шаронин С. Г., Яковлев С. Е.— Сопряжение ЦАП и АЦП с микропроцессорными системами 1
- Петров Л. В., Ляшев В. П., Соловьева Т. А.— Устройство сопряжения накопителя на магнитных лентах с микроЭВМ «Электроника 60» 1
- Сташкевич А. В.— Клавиатура и блок индикации для приборов на базе микропроцессора К1801ВМ1 1
- Торгов Ю. И.— Программируемый контроллер клавиатуры КР580ВВ79 1
- Чабан С. Д., Люпов Р. М.— Сенсорная клавиатура 3
- Справочная информация**
- Давыдовская В. В., Козловский В. А., Малинин А. В., Усов Г. И.— Микросхема асинхронного статического ОЗУ КР537РУ13 1
- Кулешова В. И.— Микропроцессорный комплекс БИС серии КР580 3
- Интерфейсные БИС микропроцессорного комплекта К1801 4
- Интерфейсные БИС микропроцессорного комплекта К1801: микросхема К1801ВП1-033 5
- Интерфейсные БИС микропроцессорного комплекта К1801: микросхема К1801ВП1-034 6

## Пятиугольник «МП». Форум «МП»

- Громов Г. Р.— Тридцать лет спустя 3
- Громов Г. Р.— ЭВМ и информатизация общества: старые мифы и новые проблемы 5

## Краткие сообщения

- Акимкин М. С., Виноградов А. В., Крутиков В. Н. и др.— Многоканальный анализатор оптического спектра МАС-002 2
- Байков А. Е., Выставкин А. Н., Катков Б. М., Мартынов А. Э.— Диалоговые средства графического взаимодействия для систем автоматизации эксперимента 1
- Богданов М. В.— Расширение возможностей вычислительных комплексов с быстродействующим процессором «Электроника МТ-70» 2
- Большинский С. М.— Управляющая программа АСНИ «Спектр» 2
- Булатов В. П., Иоффе А. А., Лозовский В. А. и др.— Оптический многоканальный анализатор на базе микроЭВМ «Агат» 2
- Вотинцев Н. В.— Локальная вычислительная сеть «Синхро» 2
- Евдокимов М. В., Мазалов И. Н.— Двухпроцессорная система: микроЭВМ МС 1201.01 и периферийный процессор «Электроника МТ-70» 2
- Кулик О. В., Андронов М. А.— Модуль сопряжения БИС серии КР580 с магистралью креста КАМАК 1
- Лазунова Е. Л., Fenster M. Я.— Контроллер кассетного накопителя на магнитной ленте РК-1 для микроЭВМ «Электроника 60» 4
- Мусатов А. Н., Максимович А. Б.— Параллельный интерфейс ввода-вывода для систем автоматизации эксперимента с использованным микроЭВМ «Электроника ДЗ-28» 3
- Манкевич А. Ф., Шайдоров С. Д.— Реализация ответного канала в асинхронных адаптерах дистанционной связи с интерфейсом «токовая петля» 4

# МУЛЬТИМЕТР С ЭЛЕКТРОННЫМ ПЕРЕКЛЮЧЕНИЕМ ПРЕДЕЛОВ

В. Я. Ефремов, Е. Р. Белик

Легкий и по компактности превосходящий зарубежные и отечественные аналоги мультиметр — прибор для измерения напряжения, тока и сопротивления с цифровой индикацией — разработан и внедрен на Телевизионном техническом центре им. 50-летия Октября.

Отличительная особенность мультиметра — электронный переключатель пределов и оригинальный узел согласования выходов микросхемы КР572ПВ2 (рассчитанной на работу со светодиодными индикаторами) с жидкокристаллическим индикатором.

Результат измерения выводится на 3,5 цифровой индикатор типа ИЖЦ-5—4/8 с высотой цифр 8 мм. Максимальное число, воспроизводимое на индикаторе, — 1999. На нем же указывается полярность измеряемого напряжения или тока.

Пределы измерения переключаются нажатием на один из двух псевдосенсорных переключателей. При этом запятая на индикаторе перемещается влево или вправо.

Мультиметр снабжен электронной защитой при измерении напряжений и сопротивлений. В режиме измерения токов прибор защищен плавким предохранителем. Источник питания — батарея «Корунд». Необходимость замены батареи возникает через 3—4 месяца при разрядке ее до напряжения 6,5 В. При этом свечение индикатора резко ослабевает, хотя электронная часть прибора исправно работает до снижения напряжения до 5,6 В.

Все элементы мультиметра размещены на одной печатной плате из двустороннего фольгированного стеклотекстолита толщиной 1 мм.

В приборе восемь микросхем (КР572ПВ2Б — 1, К544УД1А — 1, К140УД12 — 1, К564КП1 — 2, К564ЛА7 — 1, К564ИЕ11 — 1, К564ТР2 — 1).

В комплект документации на мультиметр с электронным переключением пределов входят: схема принципиальная (1 лист), схема печатной платы — (4 листа), техниче-



Основные технические характеристики мультиметра

Верхние пределы измерений	
постоянного напряжения, (В), тока (мА), сопротивлений	. 2, 20, 200, 2 000
переменного напряжения (В), тока (мА)	. . . . . 1, 10, 100, 1 000
Входное сопротивление, МОм	. . . . . 1
Основная погрешность при измерении	
напряжения	
постоянного, %	. . . . . 1
переменного, %	. . . . . 3
тока	
постоянного, %	. . . . . 1
переменного, %	. . . . . 3
сопротивлений, %	. . . . . 1
Диапазон частот измерения, кГц	
переменного напряжения	. . . . . 0,02 ... 20
переменного тока	. . . . . 0,02 ... 10
Падение напряжения на входе при измерении тока	
на пределах 0,1 и 1 А, В	. . . . . 2
на остальных пределах, В	. . . . . 0,2
Потребляемый ток, мА	. . . . . 5
Габариты, мм	. . . . . 130×63×25*
Масса, г	. . . . . 185

ское описание (11 листов), фотография внутренней компоновки (1 шт.).

# РАСТР — УСТРОЙСТВО ПОДГОТОВКИ И ОТЛАДКИ ПРОГРАММ ЦИФРОВОГО ПРОЦЕССОРА КМ1813ВЕ1

(К ст. Артюхова В. Г., Глухенького В. В., Лернера Р. А. и др.)

Автономное специализированное отладочное устройство РАСТР предназначено для ввода, редактирования и трансляции программ; диалоговой отладки программ на программной модели; записи программ во внутреннее ПЗУ; отладки СБИС на тестовых сигналах; комплексной отладки СБИС в реальном масштабе времени на реальных сигналах. Содержит средства, поддерживающие диалог с оператором, программирование и комплексную отладку цифрового процессора, обеспечивающие подключение к КНМЛ, АЦПУ и универсальной ЭВМ.

Программу в процессе ввода в ОЗУ можно редактировать, изменяя отдельные поля ранее введенных команд, исключая и добавляя новые команды.

РАСТР обеспечивает решение всего комплекса задач по разработке и проектированию прикладного программного обеспечения для однокристалльного процессора КМ1813ВЕ1.



Основные характеристики устройства РАСТР

Тип процессора	КР580ВМ80А, F=2 МГц
Объем ПЗУ, байт	12К×8 (3 БИС К573РФ41)
Объем ОЗУ, байт	2К×8 (4 БИС К541РУ2)
Функциональная клавиатура	30 клавиш
Тип индикатора	ИМГ-1-03
Внешние интерфейсы	ИРПР, интерфейс бытового кассетного магнитофона, интерфейс программирования ЦПОС, интерфейс испытания ЦПОС
Потребляемая мощность, Вт	70
Габаритные размеры, мм	
основного блока	480×305×405
клавиатуры	330×150×65
Масса, кг	
основного блока	20
клавиатуры	1
Диапазон рабочих температур, °С	..10... 45

**ЕСЛИ ВЫ НЕ УСПЕЛИ ПОДПИСАТЬСЯ НА НАШ ЖУРНАЛ, НАПОМИНАЕМ, ЧТО  
ПОДПИСКА НА «МП» ПОКВАРТАЛЬНАЯ**