

ЭМГ

МИКРО- ПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ

6 | 1986

ISSN 0233-4844

Комплекс микропроцессорных средств с малым энергопотреблением — на базе комплекта БИС серии К588, совместим с микроЭВМ «Электроника 60»

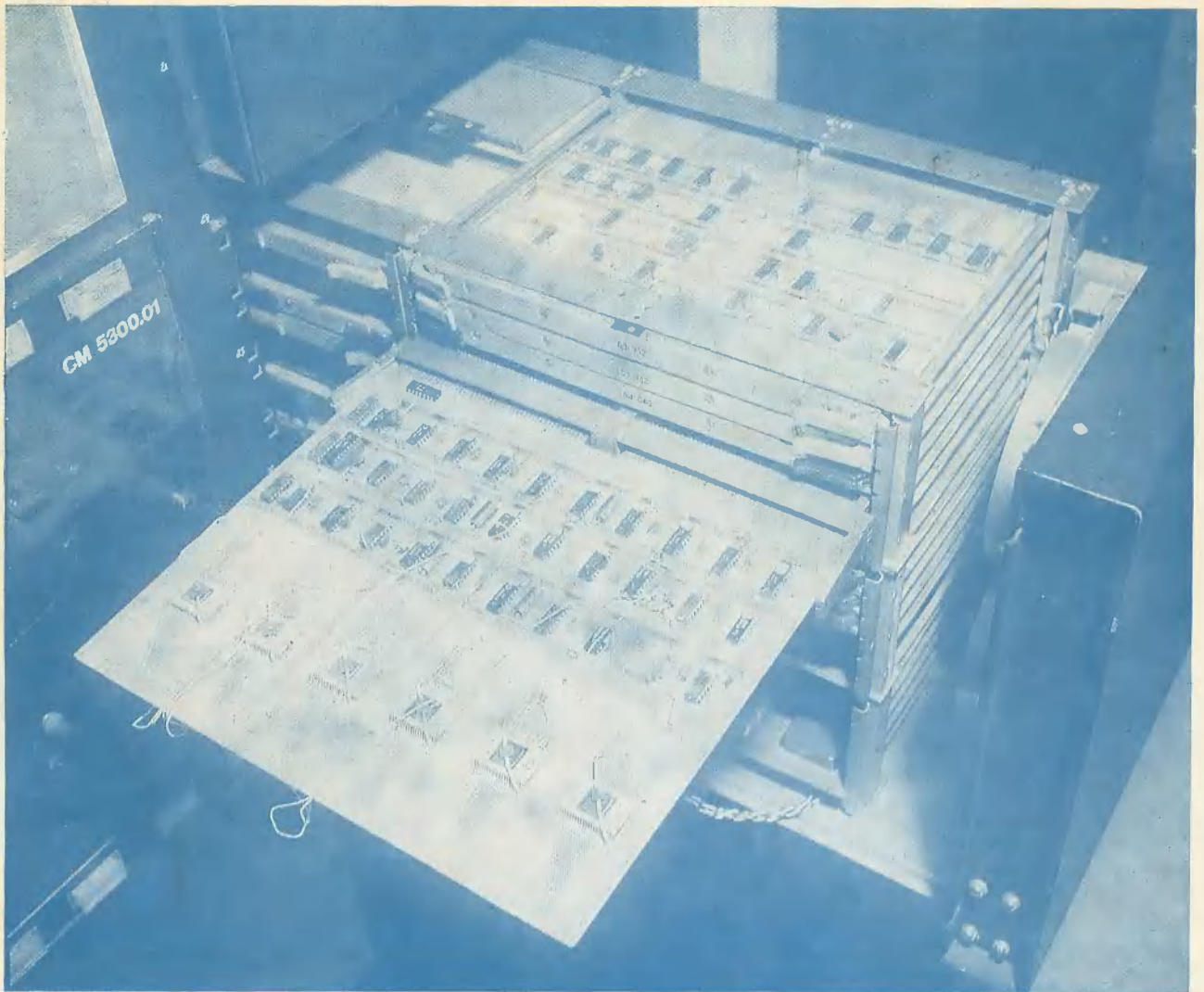
Пакеты программ, реализующих транспортные функции и систему отладки распределенного вычислительного комплекса на основе коммуникационного процессора, работающего под управлением микроЭВМ «Электроника МС 1201.1»

Оперативные запоминающие устройства серии К541 — сочетание высокого быстродействия, большой информационной емкости и относительно низкой рассеиваемой мощности

Микропроцессорная система с ограниченной программируемостью архитектуры — построена на базе процессоров с системой команд микроЭВМ «Электроника 60»; имеется возможность ее расширения однотипными модулями

Локальная сеть на базе сетевого программного обеспечения «АЛИСА»





ПРИМЕНЕНИЕ БИС К1801ВП1-035 В ИНТЕРФЕЙСНЫХ ПЛАТАХ МАЛЫХ ЛОКАЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ

(К ст. Баранова В. Г. и др.)

Вычислительные мощности программно-совместимых мини- и микроЭВМ (СМ-3, СМ-4, «Электроника 100/25», «Электроника 60М» и др.) используются часто недостаточно вследствие отсутствия быстрых и надежных каналов связи между машинами, слабой обеспеченности ЭВМ периферийными устройствами. В основном это объясняется невозможностью работы с одной ЭВМ одновременно несколькими пользователями. Выход из этого положения — объединение имеющихся средств

вычислительной техники в малую вычислительную сеть.

На кафедре «Вычислительная техника» Горьковского политехнического института им. А. А. Жданова разработана интерфейсная плата (ИП) на базе БИС К1801ВП1—035. С помощью ИП за три месяца (включая разработку и изготовление ИП) построена малая локальная вычислительная сеть типа «Звезда» с центральным комплексом на основе мини-ЭВМ «Электроника 100/25» и шестью периферийными комплексами

на основе микроЭВМ «Электроника 60М», работающая под управлением ОС Рафос-2.

Число внешних устройств, подключаемых к ИП, обусловлено максимальной плотностью монтажа. Для подключения к мини-ЭВМ большего числа периферийных устройств следует изготовить несколько подобных ИП и вставить их в гнезда БРК. Периферийные устройства связаны с ИП волоконно-оптической линией «Электроника МС4101».

ОРГАН
ГОСУДАРСТВЕННОГО
КОМИТЕТА СССР
ПО НАУКЕ И ТЕХНИКЕ

Издается с 1984 года

МПК МИКРО ПРОЦЕССОРНЫЕ СРЕДСТВА И СИСТЕМЫ

ВЫХОДИТ ШЕСТЬ РАЗ В ГОД НАУЧНО-ТЕХНИЧЕСКИЙ И ПРОИЗВОДСТВЕННЫЙ ЖУРНАЛ 6 / 1986 МОСКВА

СОДЕРЖАНИЕ МИКРОПРОЦЕССОРНАЯ ТЕХНИКА

ПЕРСОНАЛЬНЫЕ КОМПЬЮТЕРЫ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

ПРИМЕНЕНИЕ МИКРОПРОЦЕССОРНОЙ ТЕХНИКИ

УЧЕБНЫЙ ЦЕНТР

Ершов А. П.— Колонка редактора	2
Капошкин Э. П., Сухопаров А. И., Верниковский Е. А., Сержанович Д. С.— Оперативные запоминающие устройства серии К541	3
Иванов С. Н., Романов А. Ф., Чернышов Ю. Н.— Одноплатная микроЭВМ на МПК БИС серии К1810	8
Полосин А. Н., Карпинский Н. Г., Лозовой И. О., Половянюк А. И., Ургант О. В., Дябин М. И.— Учебный компьютер «Электроника УК НЦ»	14
Липаев В. В., Потапов А. И.— Длительность разработки сложных программных средств	17
Комаров И. Е., Туманов А. А.— Использование ассемблерных фрагментов при программировании на языках Бейсик-01 и Бейсик-02 ПЭВМ «Искра 226»	20
Поом К. Э., Моор А. Э., Ребане Р. В., Арулаане Т. Э. Операционная система для ПЭВМ «Искра 226»	21
Грудинин М. М., Сенченкова А. Ю.— Операционная система в ПЗУ	22
Щелкунов Н. Н., Дианов А. П.— Техника программирования 8-разрядных микроконтроллеров	23
Большинский С. М., Полтава А. Н.— Драйвер НГМД удвоенной плотности для ОС РАФОС	29
Волков С. В., Дудкин М. В., Казьмин А. И., Менн А. А., Шерстюк А. В., Кузнецов М. Н., Целяпин А. Н., Ушкевич В. В.— Программное обеспечение коммуникационных процессоров в распределенных вычислительных комплексах	30
Тарков М. С.— Организация удаленного исполнения команд монитора РАФОС в вычислительной системе МИКРОС	34
Герштейн Ю. С.— Си-реализация языка нисходящего разбора	35
Попов А. Л.— Информационно-поисковая система «Библиотека» и ее реализация на персональной ЭВМ	39
Илюкович А. А.— Информационно-поисковая система «Кадры»	40
Елинер Э. И., Клименко А. Д., Костылев Д. А.— Локальная сеть на базе сетевого программного обеспечения АЛИСА	41
Цвелодуб О. В., Щелкунов Н. Н.— Контроллер локальной вычислительной сети	47
Чмиль В. М., Ющенко Б. И.— Микропроцессорная система контроля и управления приемными устройствами связи	50
Макаров А. И.— Многопроцессорные системы с ограниченной программируемостью архитектуры	52
Баранов В. Г., Калягин С. Н., Бажанов Ю. С., Корсакова Т. А.— Применение БИС К1801ВП1-035 в интерфейсных платах малых локальных вычислительных сетей	55
Преснухин Л. Н., Белильников В. И., Волков Ю. И., Ургант О. В., Шапкин В. Г.— Адаптер локальной вычислительной сети на базе БИС К1801ВП1-065	57
Вологжанин В. А., Скворцов В. А., Слипень Н. Е.— Комплекс микропроцессорных средств для информационно-измерительных систем	59
Матвеев А. А., Пономарев Ю. П.— Микропроцессорный Модем-2400 для каналов тональной частоты	63
Солонин В. Ю.— Селекция импульсов полезного сигнала микропроцессором КР580ИК80А	69
Байков В. Д., Кабанов В. В., Попов А. М.— Мультипроцессорная организация цифровых фильтров на базе МПК БИС КР580	70
Гинзбург Б. Д.— Микропроцессорный измеритель периода сигнала	73
Кушнир В. Е., Панфилов Д. И., Шаронин С. Г.— Учебная микроЭВМ на основе однокристалльной ЭВМ КМ1816ВЕ48	75
Арсенин В. П., Воробьев А. О., Герасимович В. Н.— Применение микросхемы К1802ВВ1 для управления памятью	82
Бурочкин И. В.— Устройство параллельного обмена алфавитно-цифрового дисплея 15ИЭ-00-013 с микроЭВМ «Электроника 60М»	88
Дмитренко А. П., Старостенко О. В.— Контроллер алфавитно-цифрового индикатора на базе однокристалльной микроЭВМ	90

Главный редактор
А. П. ЕРШОВ

Редакционная
коллегия:

А. Г. Алексенко
В. В. Бойко
В. М. Брябрин
К. А. Валиев
Г. Р. Громов
(ответственный секретарь)
В. И. Иванов
М. Б. Игнатьев
А. В. Каляев
С. С. Лавров
В. В. Липаев
Б. Н. Наумов
(зам. главного редактора)
С. М. Пеленов
(зам. главного редактора)
А. К. Платонов
Д. А. Поспелов
Ю. А. Чернышев
В. А. Чиганов
И. И. Шагурин

Редакционный
совет:

Ю. Е. Антипов
Р. Л. Ашастин
Е. П. Веляхов
Н. Н. Говорун
В. И. Жильцов
Г. И. Кавалеров
И. И. Малашикин
В. А. Мясников
Ю. Е. Нестерихин
И. В. Прангишвили
Л. Н. Преснухин
В. И. Скурихин
В. Б. Смолов
Ю. М. Соломенцев
В. И. Хохлов
Н. Н. Шереметьевский

Номер подготовили:

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

Корректор Л. С. Глаголева
Технический редактор
Л. А. Горшкова

Адрес редакции: 101820,
просед Серова, 5, редакция
журнала «Микропроцессорные
средства и системы»

Телефоны 228-18-88; 221-99-26

Сдано в набор 30.10.86 Т 17486.
Подписано к печати 18.12.86.
Формат 84×138/16. Бумага № 1.
Высокая печать. Усл. печ. л. 10,08
Уч.-изд. л. 14,3. Тираж 40 000 экз.
Зак. 266. Цена 1 р. 10 к.
Орган Государственного комитета
СССР по науке и технике
Московская типография № 13
ПО «Периодика» ВО «Союзполи-
графиром» Государственного
комитета СССР по делам
издательств, полиграфии и книжной
торговли.
107005, Москва, Б-5. Денисовский
пер., д. 30

На первой странице облож-
ки — Учебный компьютер
«Электроника УК НЦ» (см. ста-
тью Полосина А. Н. и др.)

О СИЛЕ ПРИМЕРА ПАРТИЙНОЙ ПЕЧАТИ

За последние месяцы журнал «Коммунист» стал одним из наиболее читаемых журналов.

Не торопитесь считать это наблюдение дешевой похвалой, ничего не добавляющей к солидной репутации ведущего теоретического партийного органа, издаваемого миллионным тиражом.

Суть перемены в том, что любой думающий и неравнодушный читатель, независимо от партийности или должности, все чаще находит в этом журнале глубокое искреннее и неуклончивое рассуждение, вскрывающее противоречия и проблемы развития нашего общества.

Очень интересны и поучительны материалы последних выпусков журнала, посвященные обсуждению социально-экономических проблем. Видно невооруженным глазом, как призывы съезда, строки партийных документов будят мысль, раскрепощают духовные силы общества, придают полнокровный, актуальный характер научным дискуссиям.

Вызывает огромное уважение интенсивная работа мысли и чувство ответственности авторов журнала, направленные на раскрытие сущности явлений, поиск безошибочных ориентиров, преодоление субъективизма и догматического подхода. Отрадно видеть освобождение от какого бы то ни было лицемерия, когда стремление к научной истине и патристическое чувство служения обществу сливаются в нерасторжимую движущую силу, придают особую зоркость уму, укрепляют гражданский оптимизм.

Это пример, достойный всяческого подражания.

Наш журнал прожил этот год полной жизнью. Налицо предпосылки к решению главной задачи — развитию низовой инициативы, раскрытие творческого потенциала «рядовой» инженерно-конструкторской работы. Вызывает удовлетворение широта географии авторов, радуют заполненные ряды Большой политехнической аудитории, собирающей читательский актив на семинары и обсуждения. Видно по всему, что развитие информатики и вычислительной техники в СССР набирает темп.

И все же, если речь идет о наших публикациях, мы еще не достигли необходимой глубины творческого контакта с читателем. Поток «информации с мест» налаживается, однако, явно не хватает встречного потока. Нам необходимы серьезные аналитические исследования, достоверные прогнозы, надежные ориентиры, и, конечно, своевременная постановка проблем и обсуждение путей их решения.

В развитии вычислительного дела недостаточно определить номенклатуру вычислительных средств и задать объемы их производства. Даже если остаться в рамках решения технических задач, есть немало стратегических вопросов, требующих опережающего знания и выработки безошибочной установки. Периферия, заказные БИСы, массовое производство, качество, телекоммуникации и сети — все эти проблемы и многие, им аналогичные, требуют правильного целеуказания и достоверной ориентации всей научно-технической общественности.

В еще большей степени это касается вопросов программного обеспечения и применения вычислительных средств. Придание программным изделиям свойств промышленного продукта и потребительского товара, место и роль «электронной» информации в обществе, реальное повышение продуктивности трудового процесса на основе ЭВМ — уже только этого перечисления достаточно, чтобы понять серьезность проблем информатизации с проблемами развития общества в целом.

А это значит, что ведущие ученые в области информатики и вычислительной техники должны в своей области сделать то, что сейчас их коллеги — экономисты и специалисты по организации управления — делают в сфере экономики и хозяйственного механизма страны.

Президент Академии наук СССР Гурий Иванович Марчук в своих первых публичных выступлениях подчеркнул важность усиления прогностической функции научного анализа, реализации на деле предсказательной силы науки. Нет сомнения, что ученые выполняют свой долг. Остается, однако, проблема — сделать научную идею элементом общественного сознания, приблизить научное познание к реальной жизни, поднять обыденное восприятие до высот научного кругозора. В этом мы видим почетную и трудную задачу для нашего журнала, в этом мы будем стараться следовать воодушевляющему примеру журнала «Коммунист».

А. П. Ершов

УДК 681.325—181.48

Э. П. Калашкин, А. И. Сухопаров, Е. А. Верниковский, Д. С. Сержанович

ОПЕРАТИВНЫЕ ЗАПОМИНАЮЩИЕ УСТРОЙСТВА СЕРИИ К541

Микросхемы ОЗУ серии К541 изготавливаются по планарно-эпитаксиальной технологии с диэлектрической межкомпонентной изоляцией типа «Изопланар-П» с использованием сверхтонких эпитаксиальных 1,5...2,0 мкм слоев, 2 мкм проектных норм и принципов самосовмещения в сочетании с оптимизированным процессом ионного легирования. Все оперативные запоминающие устройства являются статическими с произвольной выборкой. В состав серии К541 входят микросхемы ОЗУ трех типов (табл. 1).

щую из 64 строк и 64 столбцов, дешифраторы строк и столбцов, схемы адресного и разрядного формирования, логику управления (рис. 1). Все входные и выходные сигналы соответствуют уровням ТТЛ-схем, причем выходные уровни «Лог. 1» и «Лог. 0» формируются только в режиме считывания информации. Во всех остальных режимах выход микросхемы находится в состоянии с высоким импедансом (третье состояние). Наличие третьего состояния позволяет легко объединить микросхемы по выходам.

Таблица 1

Состав серии К541

Условное обозначение микросхемы	Основное функциональное назначение	Классификационные параметры в нормальных климатических условиях			Условное обозначение корпуса
		Информационная емкость, организация, бит	Время выборки адреса, t_{AS} , нс	Удельная потребляемая мощность P_{cc} , мВт/бит	
К541РУ1 К541РУ1А	Оперативное запоминающее устройство статического типа с тремя состояниями	4096 (4096 × 1)	120 70	0,122	427.18-1
К541РУ2 К541РУ2А	То же	4096 (1024 × 4)	120 90	0,128	427.18-1
К541РУ3 К541РУ3А	То же	16384 (16384 × 1)	150 100	0,038	429.42-3

Сочетание высокого быстродействия, сравнительно низкой рассеиваемой мощности и достаточно большой информационной емкости позволяет строить на основе микросхем данной серии малогабаритные высокопроизводительные блоки оперативных и сверхоперативных запоминающих устройств для различных классов изделий вычислительной техники.

Микросхема ОЗУ К541РУ1 содержит на одном кристалле матрицу из 4096 запоминающих ячеек, состоя-

В качестве запоминающего элемента в ОЗУ выбрана сверхинтегрированная инжекционная ячейка памяти, а схемами обрамления служат схемы эмиттерно-связанной логики (ЭСЛ). Согласование внутренних ЭСЛ-уровней с внешними ТТЛ-уровнями осуществляется только во входных и выходных каскадах, что обеспечивает высокое быстродействие микросхемы ЗУ.

Схема электрическая структурная БИС ОЗУ К541РУ1 приведена на рис. 1. Режимы работы ОЗУ

Таблица 2

Таблица истинности микросхемы К541РУ1

Режим работы	Логические состояния на выходах микросхемы			
	\overline{CS}	$\overline{EWR/RD}$	\overline{DI}	\overline{DO}
Хранение информации	1	X	X	Z
Запись «Лог. 0»	0	0	0	Z
Запись «Лог. 1»	0	0	1	Z
Считывание «Лог. 0»	0	1	X	0
Считывание «Лог. 1»	0	1	X	1

Примечание. 1 — высокий уровень; 0 — низкий уровень; X — безразличное состояние; Z — третье состояние (высокий импеданс).

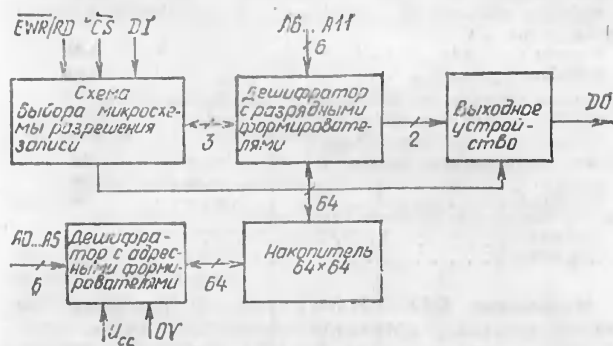


Рис. 1. Структурная электрическая схема БИС ОЗУ К541РУ1

Таблица 3
Временные параметры ОЗУ К541РУ1

Параметр	Обозначение	Норма, нс	
		К541РУ1	К541РУ1А
Время выборки адреса, не более	t_{AS}	120	70
Время выборки разрешения, не более	t_{ACS}	50	50
Время восстановления, не более	t_{RCS}	70	70
Время выборки считывания, не более	t_{ARD}	50	45
Время восстановления сигнала разрешения записи, не более	t_{RWR}	70	50
Длительность сигнала записи, не менее	t_{WR}	60	50
Время сдвига сигнала разрешения записи относительно сигнала адреса, не менее	t_{WSL}	45	30
Время сдвига сигнала разрешения записи относительно сигнала информации, не менее	t_{SUDW}	0	0
Время удержания сигнала выборки микросхемы относительно сигнала разрешения, не менее	t_{HWE}	90	90
Время удержания сигнала разрешения записи относительно сигнала выбора микросхемы, не менее	t_{HEW}	90	90
Время сохранения сигнала входной информации после сигнала разрешения записи, не менее	t_{HWI}	0	0
Время сохранения сигнала адреса относительно сигнала записи, не менее	t_{WHA}	35	20
Время цикла, не менее	t_{CY}	140	100

Таблица 4
Назначение выводов БИС К541РУ1

Вывод	Назначение
1	Выход информации D0
2...8	Выходы адреса A0...A6
9	Общий 0V
10...14	Входы адреса A7...A11
15	Вход разрешения записи-считывания EWR/RD
16	Вход выбора микросхемы CS
17	Вход информации DI
18	Напряжение источника питания U _{CC}

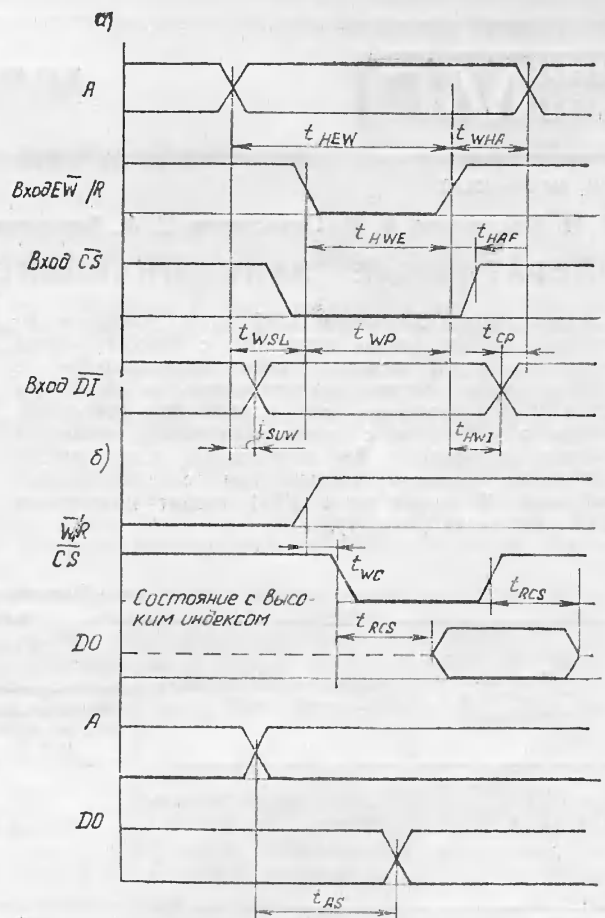


Рис. 2. Временные диаграммы работы БИС ОЗУ К541РУ1 в режимах записи (а) и считывания (б)

К541РУ1 соответствуют таблице истинности (табл. 2) и временным диаграммам, приведенным на рис. 2, а, б. Временные параметры К541РУ1 показаны в табл. 3. Назначение выводов БИС ОЗУ К541РУ1 приведено в табл. 4.

Основные статические и динамические параметры микросхемы К541РУ1 в диапазоне температур -10...+70 °С показаны ниже.

Напряжение питания U _{CC} , В	5±5%
Ток потребления I _{CC} , мА	95
Входное напряжение, В	
низкого уровня U _{TL}	0,8
высокого уровня U _{TH}	2
Входной ток, мА	
низкого уровня I _{PL}	0,45
высокого уровня I _{PH}	0,04
Выходное напряжение, В	
низкого уровня U _{OL} при I _{OL} =8 мА	не более 0,45
высокого уровня U _{OH} при I _{OH} =-5,2 мА	не менее 2,4
Длительность сигнала записи (тип), нс	60
К541РУ1	60
К541РУ1А	50
Время выборки разрешения t _{ACS} , нс	
К541РУ1	50
К541РУ1А	45

Микросхема ОЗУ К541РУ2 (рис. 3) содержит на одном кристалле матрицу запоминающих ячеек, разбитую на четыре секции по 1024 ячейки в каждой, дешифраторы строк и столбцов, схемы адресного и разрядного формирователей, логику управления,

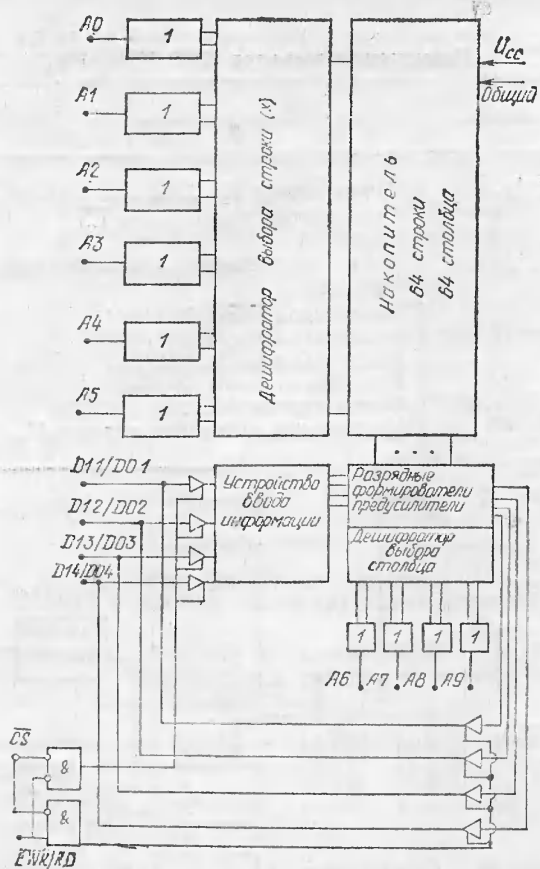


Рис. 3. Структурная электрическая схема БИС ОЗУ, K541PU2

Все входные и выходные сигналы соответствуют уровням ТТЛ-схем, причем выходные уровни «Лог. 1» и «Лог. 0» формируются только в режиме считывания информации, во всех остальных режимах выходы микросхемы находятся в состоянии с высоким выходным импедансом. Наличие объединенных входов-выходов информации позволяет работать на общую шину данных. Возможность оперирования 4-разрядными словами, высокое быстродействие — все это делает данные микросхемы особенно привлекательными для разработчиков систем на базе микропроцессорной техники. В качестве запоминающего элемента выбрана ячейка, аналогичная K541PU1.

Режимы работы ОЗУ K541PU2 отвечают таблице истинности (табл. 5) и временным диаграммам, прив-

Таблица 5

Таблица истинности микросхемы K541PU2

Режим работы	Логические состояния на выводах микросхемы		
	\overline{CS}	$\overline{EW/RD}$	DI/DO
Хранение информации	1	X	Z
Запись „Лог. 0“	0	0	0
Запись „Лог. 1“	0	0	1
Считывание „Лог. 0“	0	1	0
Считывание „Лог. 1“	0	1	1

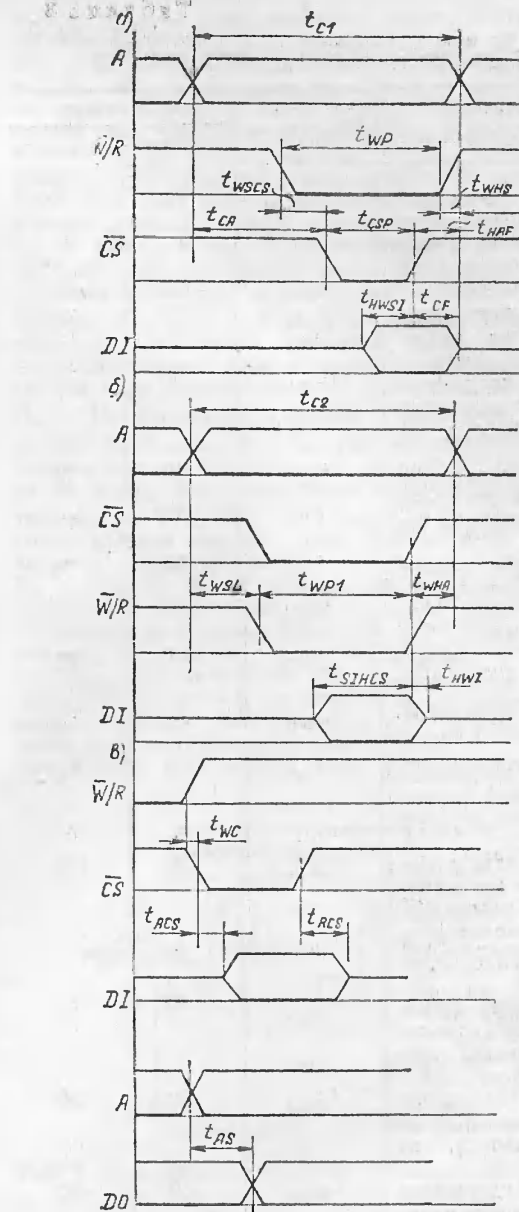


Рис. 4. Временные диаграммы работы БИС ОЗУ, K541PU2 в режимах записи (а, б) и считывания (в)

Основные статические и динамические параметры микросхемы K541PU2 в диапазоне температур $-10...+70^\circ\text{C}$ приведены ниже.

Напряжение питания U_{CC} , В	$5 \pm 5\%$
Ток потребления I_{CC} , мА	100
Входное напряжение, В	
низкого уровня U_{TL}	0,8
высокого уровня U_{TH}	2
Входной ток, мА	
низкого уровня I_{TL}	$-0,4$
высокого уровня I_{TH}	0,02
Выходное напряжение, В	
низкого уровня U_{OL} при $I_{OL}=8$ мА	0,45
высокого уровня U_{OH} при $I_{OH}=-5,2$ мА	2,4
Длительность сигнала записи t_{WP} , нс	60
Время выборки разрешения t_{ACS} , нс	40

Таблица 6-
Временные параметры ОЗУ К541РУ2

Параметр	Обозначение	Норма, нс	
		К541РУ2	К541РУ2А
Время сдвига сигнала записи относительно сигнала выбора микросхемы, не менее	t_{WCS}	0	0
Длительность сигнала выбора микросхемы, не менее	t_{CSP}	60	60
Время сохранения сигнала записи после сигнала выбора микросхемы	t_{WHS}	0	0
Время сохранения сигнала адреса после сигнала разрешения, не менее	t_{HAE}	30	30
Время сдвига сигнала разрешения относительно сигнала адреса, не менее	t_{CA}	50	50
Время удержания сигнала разрешения относительно сигнала информации, не менее	t_{HWSI}	60	60
Время сохранения входной информации после сигнала разрешения	t_{CF}	0	0
Длительность сигнала записи при эксплуатации в режиме записи относительно сигнала EWR/RD, не менее	t_{WPI}	130	130
Время сдвига сигнала записи относительно сигнала адреса, не менее	t_{WSL}	50	50
Время сохранения сигнала адреса после сигнала записи, не менее	t_{WHA}	30	30
Время удержания сигнала записи относительно сигнала входной информации, не менее	t_{SHCS}	60	60
Время сохранения сигнала входной информации после сигнала записи	t_{HWI}	0	0

денным на рис. 4, а, б, в. Временные параметры микросхемы К541РУ2 даны в табл. 6.

Назначение выводов БИС ОЗУ К541РУ2 приведено в табл. 7.

Микросхема К541РУ3 (рис. 5) выполнена в 20-выводном металло-керамическом корпусе с планарным расположением выводов. Назначение выводов приведено в табл. 8. Микросхема работает согласно таблице истинности (табл. 9) и временным диаграммам, представленным на рис. 6, а, б.

Таблица 7
Назначение выводов БИС К541РУ2

Вывод	Назначение
1...7	Входы адреса A6...A2
8	Вход выбора микросхемы \overline{CS}
9	Общий 0V
10	Вход разрешения записи-считывания EWR/RD
11	Вход-выход числа D14/D04
12	Вход-выход числа D13/D03
13	Вход-выход числа D12/D02
14	Вход-выход числа D11/D01
15...17	Входы адреса A9...A7
18	Напряжение источника питания U_{cc}

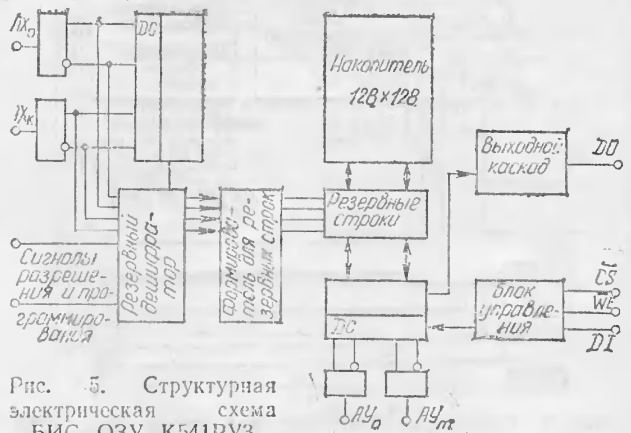


Рис. 5. Структурная электрическая схема БИС ОЗУ К541РУ3

Таблица 8
Назначение выводов БИС К541РУ3

Вывод	Назначение
1	Выход информационный D0
2...8	Входы адреса A13...A7
9	Вход адреса A0
10	Общий 0V
11...16	Входы адреса A1...A6
17	Вход разрешения записи-считывания EWR/RD
18	Вход выборки микросхемы \overline{CS}
19	Вход информационный DI
20	Напряжение источника питания U_{cc}

Таблица 9
Таблица истинности микросхемы К541РУ3

Режим работы	Логическое состояние на выводах микросхем			
	\overline{CS}	\overline{WE}	DI	DO
Хранение информации	1	X	X	Z
Запись „Лог. 0“	0	0	0	Z
Запись „Лог. 1“	0	0	1	Z
Считывание „Лог. 0“	0	1	X	0
Считывание „Лог. 1“	0	1	X	1

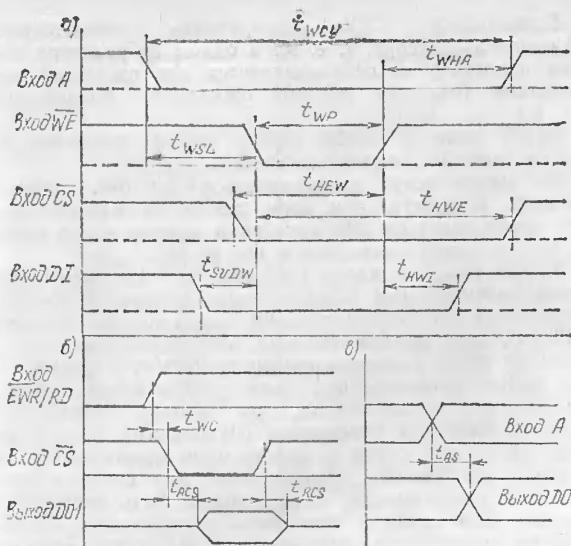


Рис. 6. Временные диаграммы работы БИС ОЗУ К541РУЗ в режимах записи (а) и считывания (б)

Основные статические и динамические параметры микросхемы К541РУЗ в диапазоне температур $-10...+70^{\circ}\text{C}$ приведены ниже.

Напряжение питания U_{CC} , В	$5 \pm 5\%$
Ток потребления I_{CC} , мА	110
Входное напряжение, В	
низкого уровня U_{TL}	0,8
высокого уровня U_{TH}	2
Входной ток, мА	
низкого уровня I_{L}	$1...0,51$
высокого уровня I_{H}	0,04
Выходное напряжение, В	
низкого уровня U_{OL} при $I_{\text{L}} = 8 \text{ мА}$	0,45
высокого уровня при $I_{\text{L}} = -5,2 \text{ мА}$	2,4
Длительность сигнала записи t_{WR} , нс	60
Время выборки разрешения t_{ACS} , нс	40

Для борьбы с возможными отказами элементов из-за дефектов, возникающих в процессе технологических операций, приводящих к снижению выхода годных, в БИС ОЗУ используются резервные элементы, которые подключаются вместо отказавших. На кристалл дополнительно введены резервный накопитель, состоя-

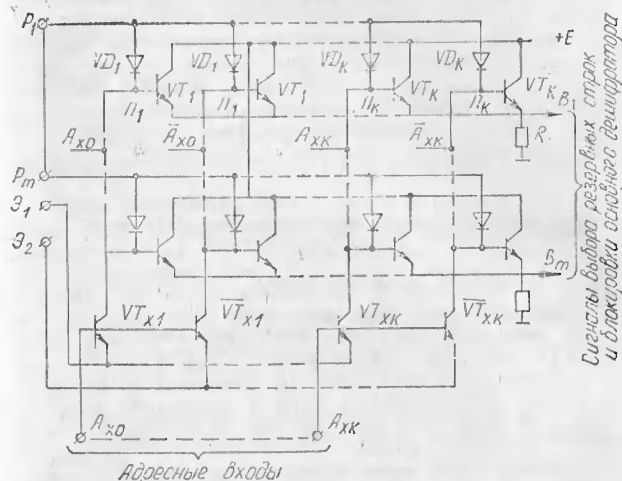


Рис. 7. Программируемый резервный дешифратор

щий из четырех строк элементов памяти (ЭП), и программируемый резервный дешифратор (рис. 7) на плавких поликремниевых вставках. Путем пережигания плавких вставок резервный дешифратор настраивается на адреса дефектных строк (строк, содержащих отказавшие ЭП). Операции по определению дефектных строк и пережиганию соответствующих плавких вставок проводятся в процессе зондовых испытаний кристаллов БИС. При поступлении на входы микросхемы ОЗУ адресов дефектных строк резервный дешифратор блокирует работу основного дешифратора и одновременно выбирает соответствующую резервную строку.

Основу резервного дешифратора составляют транзисторы $VT_1, VT_1, \dots, VT_k, VT_k$, которые образуют m -к-входовых логических элементов ИЛИ, где $m=4$ — число резервных строк, k — разрядность адреса строки. На базы транзисторов через плавкие вставки $\Pi_1, \Pi_1, \dots, \Pi_k, \Pi_k$ подаются прямые и инверсные адресные сигналы $A_{X1}, A_{X1}, \dots, A_{Xk}, A_{Xk}$. Пережигание необходимых плавких вставок для настройки резервного дешифратора на адреса дефектных строк осуществляется через транзисторы $VT_{X1}, VT_{X1}, \dots, VT_{Xk}, VT_{Xk}$ по цепи: транзистор — плавкая вставка — диод. Для этого на кристалл введены дополнительные контактные площадки: $P_1, \dots, P_m, \mathcal{E}_1, \mathcal{E}_2$, на которые подаются электрические импульсные сигналы, как при программировании ППЗУ.

Проверка кристаллов БИС ОЗУ с резервными элементами сопряжена с дополнительными временными затратами измерительного оборудования на выявление дефектных элементов, их подсчет и программирование плавких вставок. Для снижения этих затрат и увеличения производительности измерительного оборудования в БИС ОЗУ введен блок, позволяющий опреде-

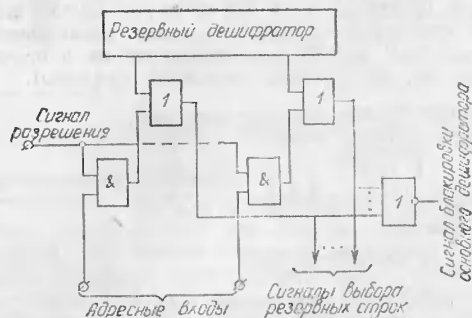


Рис. 8. Блок проверки исправности резервных элементов памяти

лять исправность резервных элементов до программирования резервных вставок (рис. 8). При подаче единичного сигнала КР и сигналов, содержащих не более одной единицы в адресе ($100 \dots 0, 010 \dots 0, 000 \dots 01$) на адресные входы, на выходах соответствующих элементов ИЛИ последовательно устанавливаются единичные сигналы. Эти сигналы выбирают соответствующие резервные строки и блокируют основной дешифратор строк. В результате происходит опрос резервных строк ЭП без программирования резервного дешифратора. Это позволяет проводить отбраковку пригодных для ремонта кристаллов, когда годные резервные элементы отсутствуют или их число недостаточно для ремонта ОЗУ.

Телефон для справок: 20-50-15, г. Минск,
Сержанович Дмитрий Иванович,
от 19 до 21 ч.
Статья поступила 19 февраля 1986 г.

ОДНОПЛАТНАЯ МИКРОЭВМ НА МПК БИС СЕРИИ К 1810

Основное назначение микроЭВМ (рис. 1) — выполнение арифметических операций при обработке больших массивов входной информации, а также выдача управляющих сигналов на внешние устройства.

Особенность одноплатной микроЭВМ на базе БИС К1810ВМ86 — увеличение ее вычислительной мощности за счет включения в схему арифметического сопроцессора. Возможности микроЭВМ расширяются с помощью стандартной шины И-41 [1], а также серий ИС (К155, К555, К565, К580 и др.).

Для связи со средствами отладки, регистрации и накопления данных необходима дополнительная микроЭВМ ввода-вывода, например, на базе БИС серии К580. Такое разделение функций управления внешними устройствами и вычислительной мощности наиболее полно реализует быстродействие БИС серий К1810 и К580.

Технические характеристики и микроЭВМ

Тазрядность, бит	16
Рабочая частота (выбирается переключателем), МГц	2,5 или 5
Объем резидентного ОЗУ, Кбайт	до 128
Объем резидентного ПЗУ, Кбайт	до 32
Внешние источники питания	+5В, 2,2 А; +12В, 20мА; +26В, 30мА (для программирования ПЗУ) 360x260
Размеры платы, мм	360x260

Память может быть размещена в любой части адресного пространства в 1М байт (в случае использования системной памяти возможно отключение резидентных ОЗУ и ПЗУ или включение их в пространство адресов, не занятых системной памятью).

Регенерация ОЗУ — «прозрачная» относительно ведущего процессора, т. е. ни в одном из режимов работы процессор не останавливается для ожидания регенерации (включая режимы ожидания, прерывания и т. п.).

ПЗУ типа К573РФ5 (РФ2) можно программировать в панельках самой микроЭВМ.

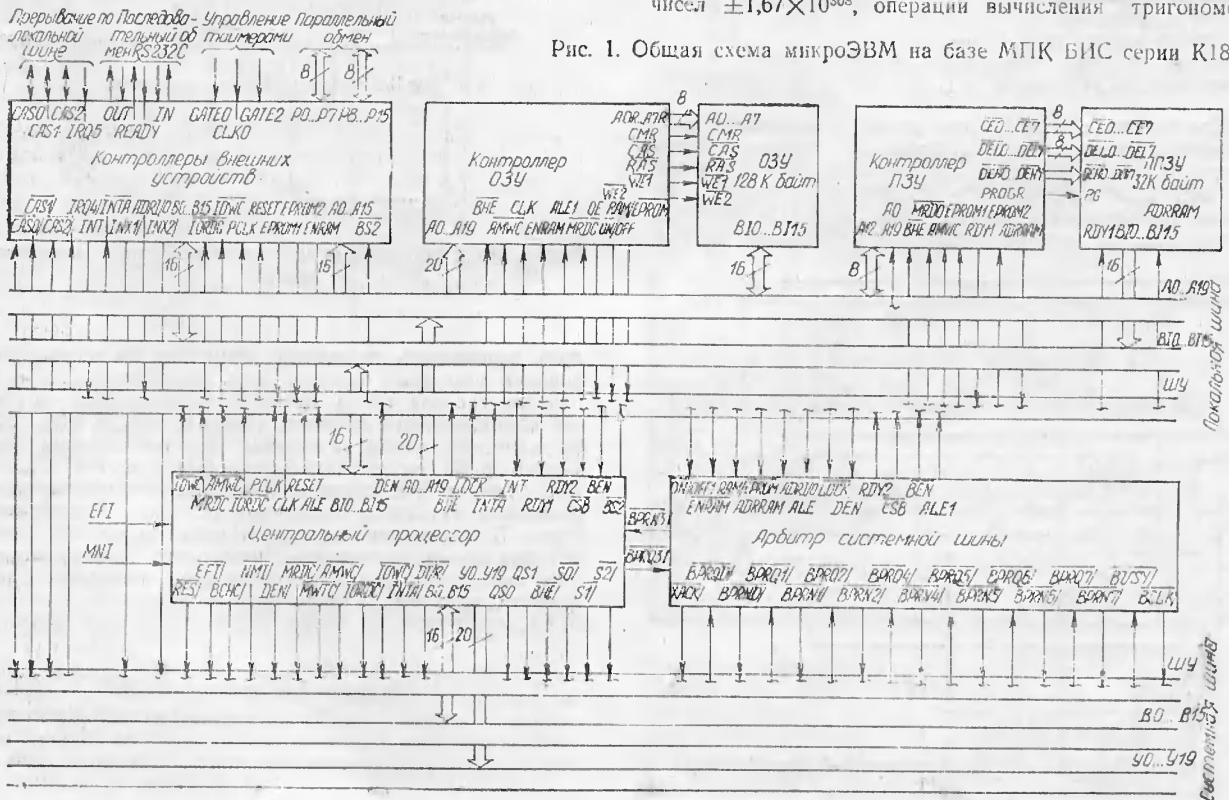
На плате могут дешифроваться все 64К адресов внешних устройств, при этом любой из адресов может использоваться для адресации портов ввода-вывода как на самой плате, так и вне ее [3].

Кроме того, на плате (см. рис. 1) предусмотрены восемь маскируемых и один немаскируемый уровень прерывания, последовательный ввод-вывод данных RS232-C, два канала таймера, обеспечивающие необходимую сетку синхронизирующих частот; третий канал таймера используется для синхронизации последовательного ввода-вывода, три канала побитного ввода-вывода: для управления внутренними функциями микроЭВМ, для связи с процессором ввода-вывода и пользователя (канал, используемый для связи с процессором ввода-вывода, также может быть запрограммирован для связи с внешними устройствами — печатающим устройством, дисплеем и т. д.), схема арбитра системной шины, обеспечивающая доступ к ресурсам системы задатчикам с любым приоритетом (0..7). При этом частота предоставления шины пропорциональна номеру приоритета задатчика.

Блок центрального процессора (рис. 2). Центральный процессор микроЭВМ, включающий в себя микропроцессор К1810ВМ86 и арифметический сопроцессор К1810ВМ87 [4], формирует сигналы управления и обработки входной информации.

Арифметический сопроцессор организует выборку всех команд в одно и то же время с основным процессором, передающим управление сопроцессору при появлении команд расширенной арифметики. Сопроцессор в дополнение к системе команд микропроцессора К1810ВМ86 выполняет 69 команд, включая операции над числами с плавающей точкой в диапазоне чисел $\pm 1,67 \times 10^{80}$, операции вычисления тригономет-

Рис. 1. Общая схема микроЭВМ на базе МПК БИС серии К1810



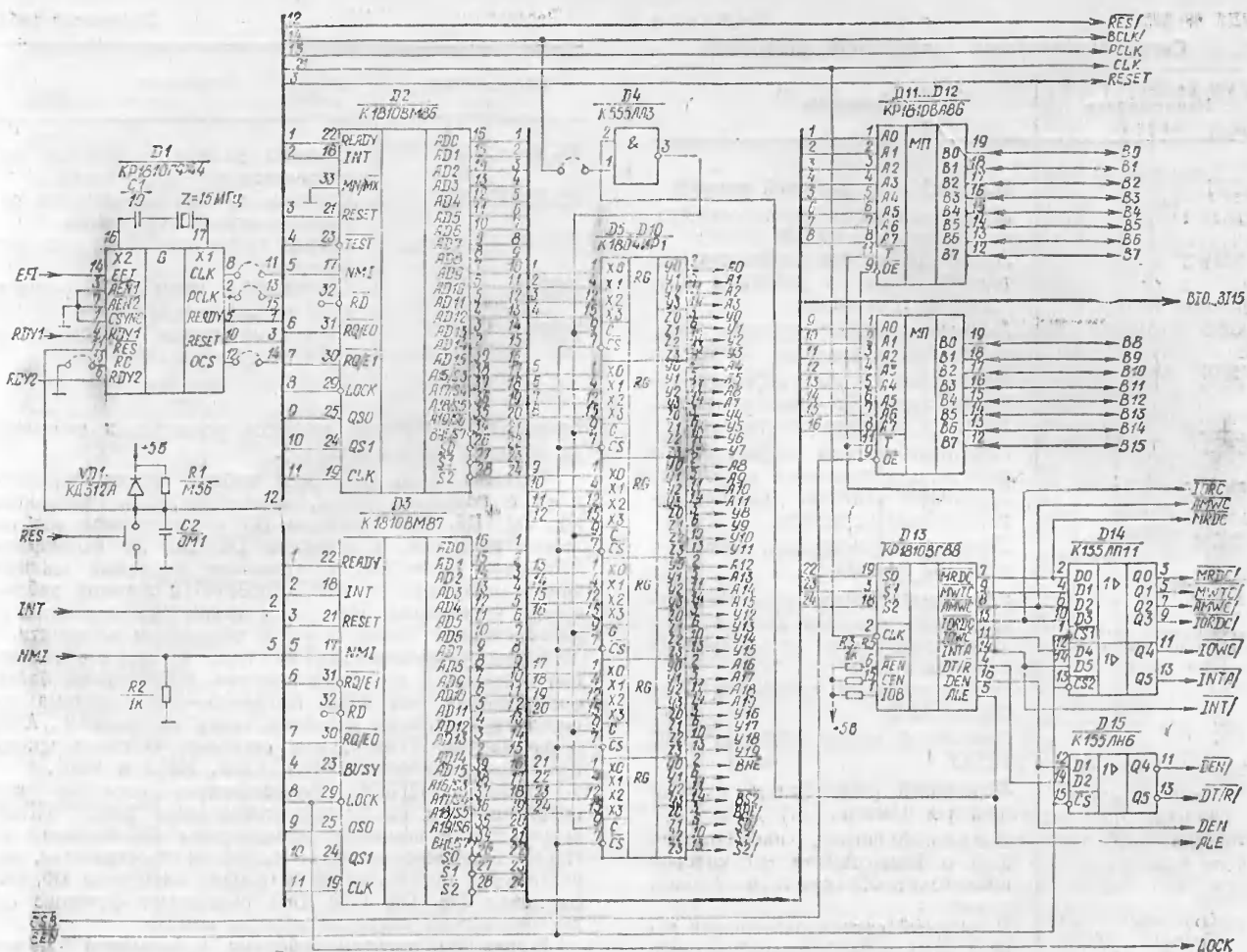


Рис. 2. Принципиальная схема блока центрального процессора

рических функций, логарифма, возведения в степень и другие операции. При работе сопроцессор использует внутренний стек из восьми регистров длиной 80 бит. Во время распознавания своей команды из общего потока команд сопроцессор вырабатывает сигнал, переводящий основной процессор в состояние ожидания.

В схему блока центрального процессора (см. рис. 2), помимо микропроцессора, включены БИС для организации буфера данных, буфера адреса и сигналов управления работой микроЭВМ, в том числе сигналов интерфейса И-41. В таблице приведено назначение вспомогательных сигналов управления данной микроЭВМ (описание сигналов микропроцессора K1810BM86 и сигналов И-41 дано в [1, 2]).

Схема управления ОЗУ (рис. 3) формирует сигналы управления ОЗУ динамического типа (до 128К байт), построенного на БИС K565PY5 или K565PY6. Схема управления использует 6 входных сигналов, шину адреса (A0...A15), формирует 8 выходных сигналов и сигналы мультиплексированных адресов ОЗУ (A0R...A7R). Адресное пространство памяти, занимаемое ОЗУ, определяется кодом, задаваемым переключками на элементе D1, D2.1. При этом сигнал разрешения обращения к ОЗУ (RAM/EPROM) формируется дешифратором (на элементе D1). Если используются БИС K565PY5, то на выходе элемента D5.4 устанавливается переключка, а если БИС K565PY6 с меньшим объемом памяти, то переключка не требуется. Переключка позволяет сформировать выходной сигнал

ON/OFF1, необходимый для работы других блоков микроЭВМ при работе с локальным ОЗУ (128К байт). Область адресного пространства ОЗУ (128К байт) определяется адресами A14...A19. К ячейкам ОЗУ обращаются сигналами AMWC, BHE, A0. Схема управления позволяет обратиться к 16-разрядному слову, к старшему или младшему байту слова (WE1, WE2). Адрес требуемой ячейки передается через коммутаторы D11...D14 побайтно. Адрес передаваемого байта формируется сигналом RAS, поступающим на входы «X1» коммутаторов. При этом младший байт адреса ячейки ОЗУ (A1...A8) передается при «Лог. 0» на данном входе, а старший байт (A9...A16) при «Лог. 1». Адреса ячеек ОЗУ передаются только в отсутствие регенерации, что идентифицируется наличием «Лог. 0» на входах «X2» коммутаторов. Адресация ОЗУ объемом 128К байт возможна при установке переключки на выходе элемента D3.

Если элементы ОЗУ — БИС K565PY6, то переключка на выходе элемента D7.2 позволяет передавать необходимые адреса A8...A14.

Ячейки ОЗУ регенерируются непрерывно при функционировании ОЗУ. При отсутствии обращения к ОЗУ на входы «X2» коммутаторов D11...D14 подается «Лог. 1», а на адресную шину ОЗУ передается содержимое счетчика адреса регенерация D10 (K561HE10). При этом каждый регенерирующий импульс RAS увеличивает на 1 содержимое счетчика D10.

Сигналы управления одноплатной микрЭВМ

Наименование	Назначение
EFI	Внешний вход тактовой частоты
RDY 1	Локальный сигнал готовности программирования ППЗУ
RDY 2	Локальный сигнал готовности системной шины к работе с процессором-задатчиком
CSB	Локальный сигнал перевода буфера данных в активное состояние
\overline{BEN}	Локальный сигнал перевода буферов адреса и сигналов управления в активное состояние
$\overline{BS2}$	Локальный сигнал обращения к памяти или внешнему устройству
PCLK	Локальная тактовая частота периферийных устройств
DEN ($\overline{DEN/}$)	Локальный (системный) сигнал передачи данных
$\overline{DT/R}$	Системный сигнал установки направления передачи данных
ENRAM	Локальный сигнал разрешения обращения к ОЗУ
ALE 1	Задержанный на время $t \approx 60$ нс сигнал ALE
\overline{OE}	Локальный сигнал выборки БИС ППЗУ
$\overline{ON/OFF 1}$	Локальный сигнал запроса обращения к памяти
RAM/ \overline{EPROM}	Локальный сигнал, информирующий о типе памяти, к которой происходит обращение в данном цикле
CMP	Локальный сигнал разрешения записи данных в буферный регистр
\overline{CAS}	Локальный сигнал выборки столбца ОЗУ
\overline{RAS}	Локальный сигнал выборки строки ОЗУ
$\overline{WE 1}$	Локальный сигнал записи старшего байта
$\overline{WE 2}$	Локальный сигнал записи младшего байта
EPROM 1	Локальный сигнал запрета программирования ППЗУ
EPROM 2	Локальный сигнал разрешения обращения к ППЗУ
ADRRAM	Локальный сигнал готовности программирования ППЗУ
$\overline{CE 0} \dots \overline{CE 7}$	Локальные сигналы чтения ППЗУ
$\overline{OEL 0} \dots \overline{OEL 7}$	Локальные сигналы разрешения записи в младший байт ППЗУ
$\overline{OEH 0} \dots \overline{OEH 7}$	Локальные сигналы разрешения записи в старший байт ППЗУ
ADRI/O	Локальный сигнал выбора адреса внешних устройств на шине
1 DCI/O ... 4 DCI/O	Локальные сигналы выборки кристалла контроллеров
$\overline{CAS 0} / \dots \overline{CAS 2} /$	Системные сигналы каскадирования контроллеров прерывания
GATE 0 ... GATE 2	Внешние сигналы разрешения счета таймеров
CLK 0	Внешний сигнал генератора счетных импульсов

Наименование	Назначение
RS 232-C-IN	Сигналы данных приемника последовательной информации
RS 232-C-OUT	Сигналы данных передатчика последовательной информации
READY I/O	Сигнал готовности данных для передачи
INX 1/	Системный сигнал обращения к ОЗУ по адресам ПЗУ
INX 2/	Системный сигнал обращения к ПЗУ по адресам ОЗУ

Временная диаграмма процесса регенерации аналогична приведенной в статье [5].

Сигналы \overline{RAS} , \overline{CAS} для выбора адреса формируются с помощью схемы, выполненной на элементах D2, D3, D5, D8. Триггеры D3 осуществляют необходимые задержки, а элементы D2, D5, D8 формируют требуемые временные соотношения, заданные техническими условиями на БИС K567PУ5 и логикой работы схемы управления. При этом исключена возможность одновременной регенерации и обращения к памяти.

Схема управления ППЗУ (рис. 4) формирует сигналы чтения и программирования ППЗУ (32К байт), построенного на БИС K573PФ5 (PФ2). Схема использует 5 входных сигналов, шину адреса (A0...A19) и формирует 27 выходных сигналов. В схеме применены элементы серий K155, K531, K555 и K561.

Чтение из ППЗУ осуществляется пословно или побайтно. При пословном чтении пара БИС ППЗУ выбирается с помощью дешифратора D6. Чтение из ППЗУ возможно только в адресном пространстве, определенном переключками на входах элементов D5, D9. Элементы D5, D9, D10, D11 выполняют функцию селектора адреса заданной области памяти.

В режиме программирования информация может записываться одновременно или раздельно в старший и младший байты ППЗУ. Соответствующие БИС выбираются с помощью дешифраторов D7 (младший байт) и D8 (старший байт). Необходимая длительность импульса программирования формируется одновибратором D2. Требуемое напряжение (+26 В) подключается переключкой и ключом (реализуемым на элементах VT1 и D4), а также сигналом от порта ввода-вывода.

Схема арбитра системной шины (рис. 5) управляет системной шиной в зависимости от наличия и приоритетов восьми уровней параллельных запросов, асинхронных по отношению к сигналу синхронизации системной шины. В состав арбитра входят приоритетный шифратор D3 для выделения задатчика с наивысшим приоритетом; дешифратор D6 для преобразования двоичного кода, поступающего с выхода шифратора, в унитарный код номера запроса с высшим приоритетом; триггеры D7.1, D7.2 для формирования сигналов RDY2, CSB, \overline{BUSY} /локальной и системной шины; логические элементы серии K155 для обеспечения логики работы арбитра.

На вход шифратора D3 арбитра поступают запросы от задатчиков с фиксированными приоритетами (старший запрос — на вход 0 шифратора, а младший — на вход 7). Номер старшего из присутствующих запросов появляется в виде кода 1—2—4 на выходе шифратора (этот код подается на вход дешифратора D6). Если запросов нет, то формируется специальный сигнал EO, блокирующий все выходы дешифратора. При наличии хотя бы одного запроса по окончании цикла системной шины и снятии сигнала \overline{BUSY} /на

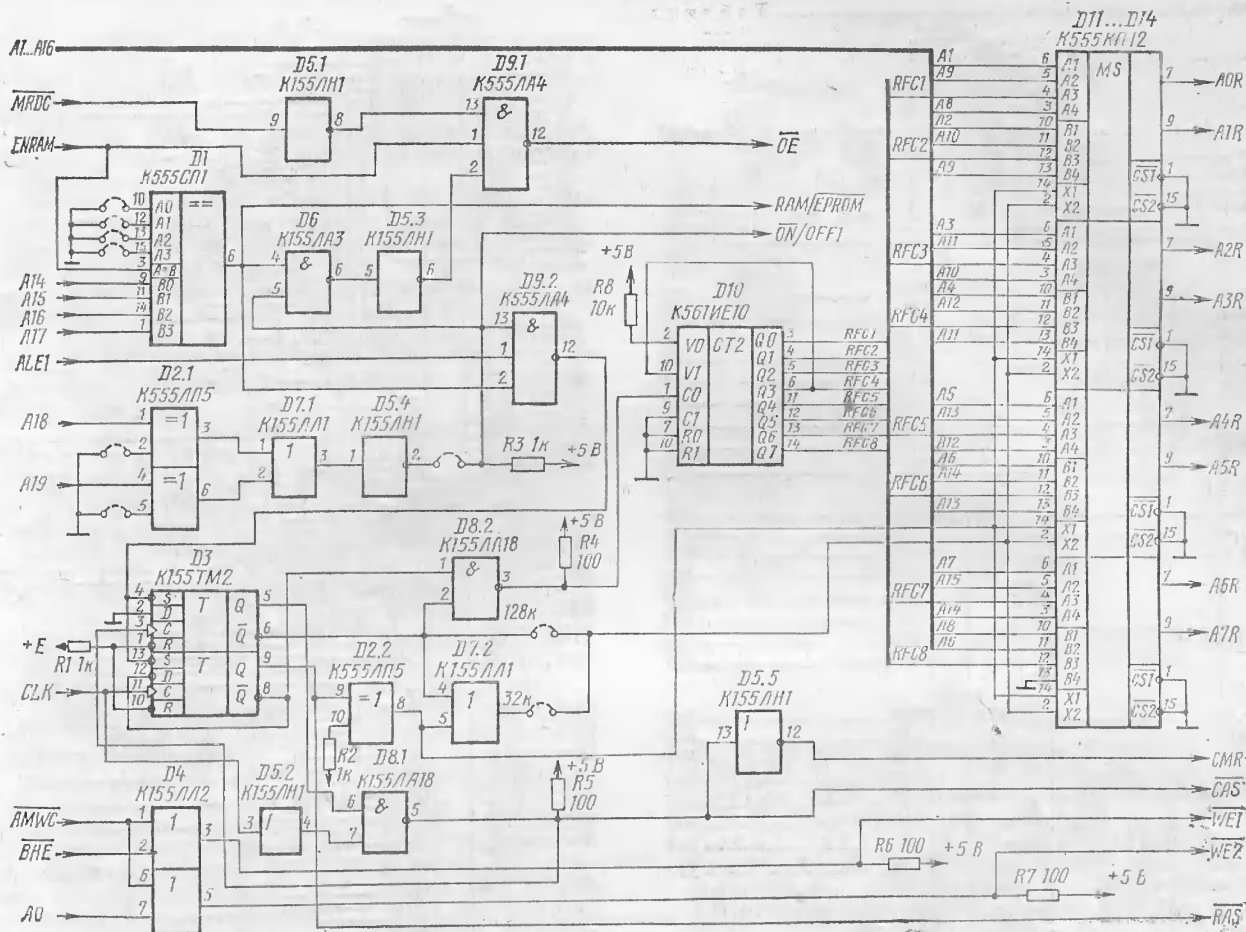


Рис. 3. Принципиальная схема управления ОЗУ.

выходе дешифратора появляется сигнал старшего запроса BPRQ \bar{N} . Системная шина передается задатчику по ближайшему фронту сигнала BCLK/. На системную шину выставляется сигнал BUSY/, а триггер D7.2* блокируется по входу «D» до окончания цикла процессора и сбрасывается окончанием сигнала DEN. Для выхода на системную шину буфера данных, адрес и управляющие сигналы переводятся в активное состояние сигналами CSB и BEN, которые являются инверсией сигнала BUSY/. Сигнал CSB формируется из сигнала BUSY/, который стробируется сигналом DEN.

Обслуженный в данном цикле системной шины задатчик может участвовать в арбитраже только через 2—3 цикла системной частоты. Это дает возможность обслужить другие запросы с более низким приоритетом. Задатчик, владеющий системной шиной, может запретить ее передачу, установив на шине сигнал LOCK.

Каждый задатчик может обратиться и к своей локальной шине: триггер D7.1 устанавливается в состояние «Лог.1» сигналом ALE1, вырабатывая сигнал готовности RDY2 для работы процессора.

При обращении задатчика к системной шине триггер D7.1 сбрасывается в состояние «Лог.0»: задатчик

* Для предупреждения зависаний триггера D7.2, связанных с несинхронностью запросов обращения по отношению к BCLK/, желательно установить второй триггер с теми же функциями.

переводится в состояние ожидания до момента удовлетворения его запроса на системную шину. Сигнал RDY2 снимается сигналом XACK/ на входе «S» триггера D7.1. Триггер D7.1 в требуемое состояние устанавливается совокупностью сигналов ON/OFF1, RAM/EPROM, ENPAM, ADDRAM, ADRI/O, анализируемых элементами D4.1 и D4.2 и определяющими логический уровень, передаваемый на вход «D» триггера D7.1.

Контроллеры внешних устройств (рис. 6) организуют:

- последовательный обмен микроЭВМ с внешними устройствами в стандарте RS232-C;
- параллельный обмен микроЭВМ с внешними устройствами;
- вывод вспомогательных сигналов для управления локальной и системной шинами (EROM1, EROM2, ENRAM, INX1/, INX2/, GATE1);
- обслуживание запросов прерывания от последовательного интерфейса (Rx RDY, Tx RDY), таймеров (OUT0, OUT1), устройств локальной (IRQ5) и системной (IRQ4/) шин;
- программно-управляемый генератор скорости передачи и приема последовательных данных (TxС, RxС), измерения и формирования временных интервалов для устройств локальной шины.

Контроллеры и процессор обмениваются информацией по двунаправленной локальной шине данных (B10... B115). Параллельный обмен информацией с внешними устройствами производится по сигналам на выводах P0...P15.

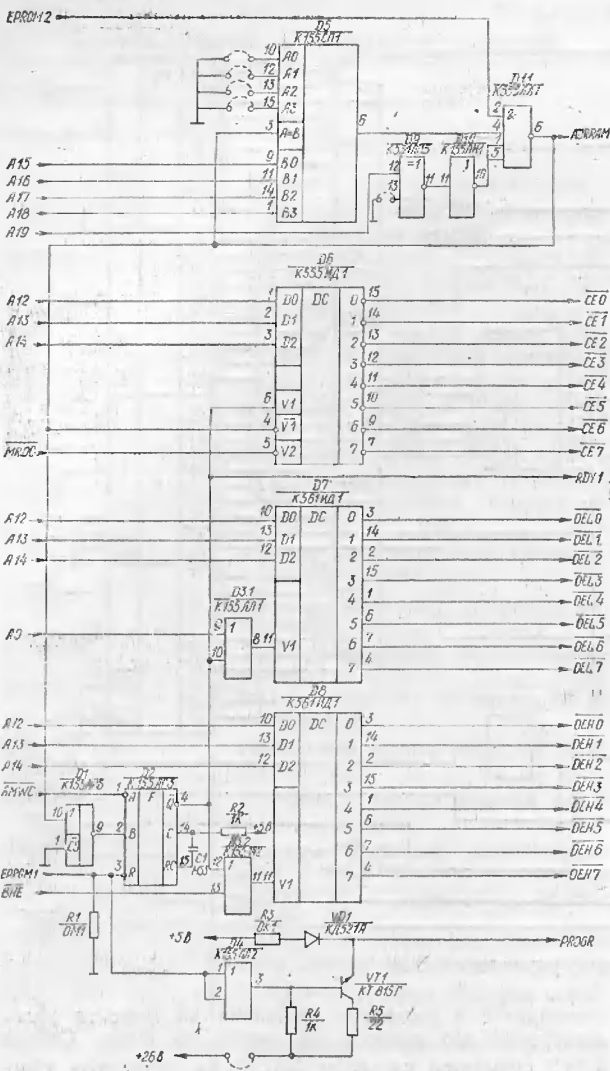


Рис. 4. Принципиальная схема управления ППЗУ

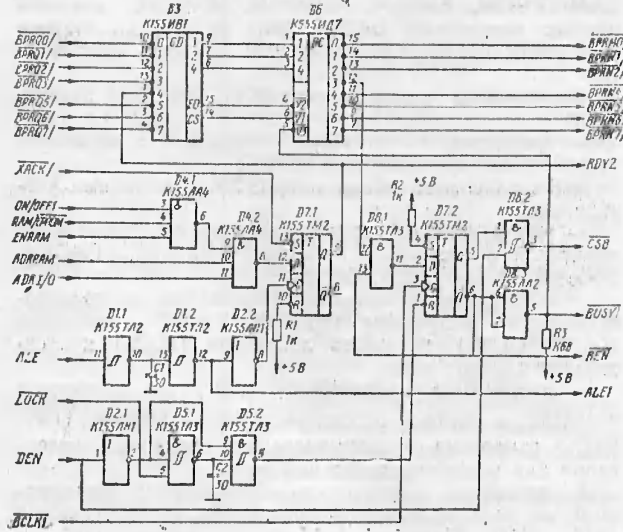


Рис. 5. Принципиальная схема арбитра системной шины

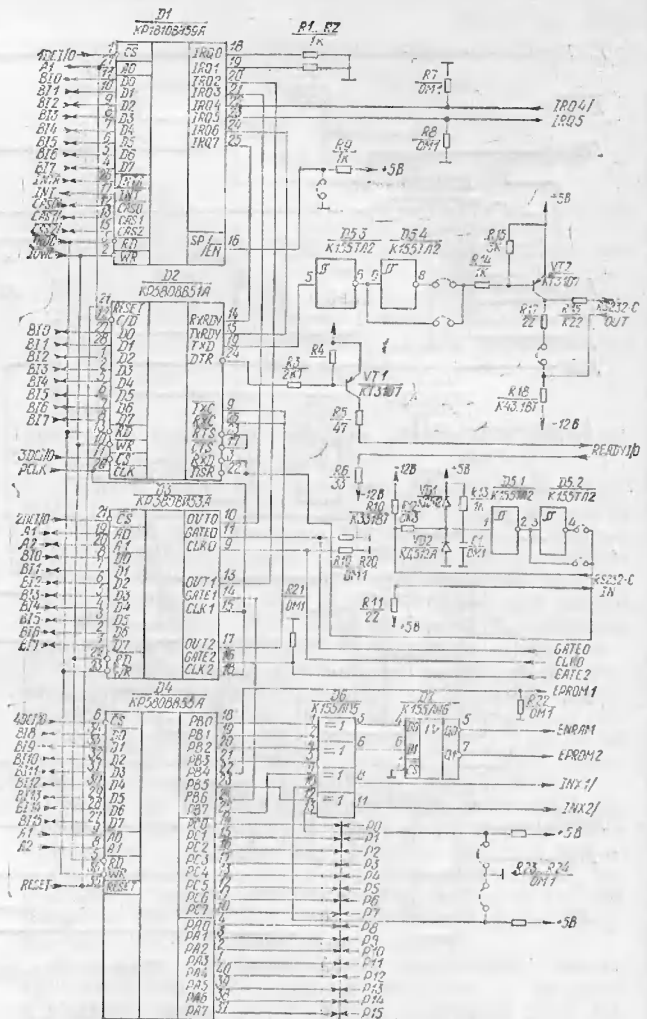


Рис. 6. Принципиальная схема контроллеров внешних устройств

Для последовательного обмена (по стандартным протоколам) необходимы:

- программируемый контроллер последовательного обмена D2 (КР580ВВ51А), формирующий требуемый протокол обмена;
- программируемый контроллер прерываний D1 (К1810ВН59А), принимающий сигналы готовности приемника (RxRDY) и готовности передатчика (TxRDY);
- программируемый таймер D3 (КР580ВН53А), формирующий сигналы, определяющие скорость передачи и приема данных (TxС, RxС);
- схема преобразования данных RS232-С.

Для инвертирования передаваемых и принимаемых сигналов нужны перемычки (на входе транзистора VT2 и выходах микросхем D5.1, D5.2). Для параллельного обмена информацией с внешними устройствами используется программируемый контроллер D4 (КР580ВВ55А). При обмене через порты А и С по сигналам P0...P15 можно подключить либо два 8-разрядных, либо одно 16-разрядное внешнее устройство. Порты А и С организуют двунаправленный асинхронно-параллельный обмен с процессором ввода-вывода.

Вспомогательные сигналы управления формируются программно и передаются через порт В контроллера D4. Эти сигналы необходимы для работы с локальным

ПЗУ (EPROM1, EPROM2), обращения к локальному ОЗУ (ENRAM), к ОЗУ по адресам ПЗУ (INX1/) и к ПЗУ по адресам ОЗУ (INX2/), управления счетчиком 1 таймера (GATE1).

Элементы D6 и D7 маскируют сигналы ENRAM, EPROM2, INX1/, INX2/ в момент включения питания и инициализации контроллеров.

Обслуживает запрос прерывания контроллер D1 (K1810BH59A). При обслуживании внешних запросов прерывания либо по локальной, либо по системной шине контроллер D1 может быть только ведущим. Кроме того, он обслуживает внутренние запросы на прерывания: от счетчиков таймеров (OUT0, OUT1) и от контроллера последовательного обмена (TxRDY, RxRDY). Для расширения количества запросов на прерывание предусмотрена перемычка на выходе SP/EN контроллера.

Формирует и измеряет временные интервалы программируемый таймер D3 (KP580BI53A). Счетчик 0 таймера управляется внешними сигналами CLK0 и GATE0. Выход OUT0 счетчика соединен со входом IRQ3 контроллера прерываний D1.

Счетчик 1 заполняется по входу CLK1 импульсами сигнала PCLK, а счет (GATE1) разрешается программно и выводится через порт В контроллера D4.

Счетчик 2 формирует сигналы тактовой частоты приема и передачи (TxC, RxC) контроллера D2.

Сигналом, поступающим на вход CLK2, служит сигнал PCLK, а разрешает счет внешний сигнал.

Дешифратор контроллеров внешних устройств (рис. 7) формирует сигналы разрешения обмена данными с контроллерами.

Адрес (A0...A15) контроллера внешнего устройства поступает на шифратор (элементы D1, D2). На выходе шифратора формируется двоичный 3-разрядный код номера контроллера, к которому осуществляется обращение. Кроме того, при выборе адреса из адресного пространства номеров контроллеров на выходе шифратора формируется сигнал ADRI/O, информирующий об обращении к контроллеру внешнего устройства локальной или системной шины.

Сформированный 3-разрядный двоичный код дешифратором D3 преобразуется в унитарный код контроллера внешнего устройства.

Операционная система совместима CP/M. МикроЭВМ служила инструментальной ЭВМ для отладки и подготовки ПО для систем управления и обработки информации, включающих встроенные вычислительные средства на базе микропроцессора K1810BM86.

Телефон для справок: 582-45-34 (Москва) — Чернышов Юрий Николаевич.

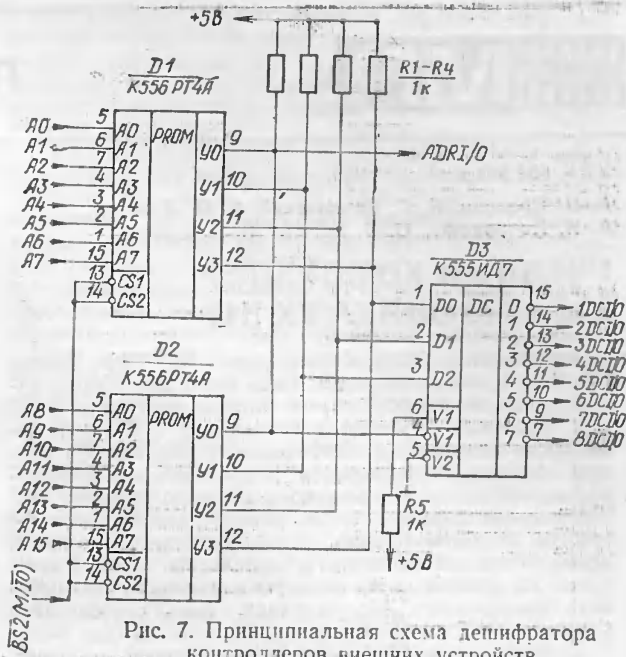


Рис. 7. Принципиальная схема дешифратора контроллеров внешних устройств

ЛИТЕРАТУРА

1. Вычислительные машины, системы, комплексы: справочник / А. П. Заморин, А. А. Мячев, Ю. П. Селиванов / Под ред. Б. Н. Наумова, В. В. Пржиялковского. — М.: Энергоатомиздат, — 1985. — 225 с.
2. Кобылинский А. В., Москалевский А. И., Темченко В. А. Однокристальный высокопроизводительный 16-разрядный микропроцессор КМ1810BM86 // Микропроцессорные средства и системы. — 1986. — № 1. — С. 28—33.
3. Описание одноплатной микроЭВМ ISBC86/12 / Пер. ВЦП № В—55766, 1981.
4. Григорьев В. Л. Программирование однокристалльных микропроцессоров. — М.: Энергоатомиздат, 1987.
5. Еремин Ю. А., Морозов А. Г. Контроллер динамического ОЗУ для микропроцессорных устройств // Микропроцессорные средства и системы. — 1986. — № 3. С. 75—77.

Статья поступила 15 июня 1985 г.

ВНИМАНИЮ ЧИТАТЕЛЕЙ!

ВИНИТИ издает с 1987 года новые выпуски сводного тома РЖ «Автоматика и вычислительная техника»: «Программное обеспечение» и «Вычислительные машины и системы».

В каждом номере выпусков будет публиковаться более 750 рефератов, аннотаций и библиографических описаний, составленных на статьи в иностранных и отечественных журналах и сборниках, а также на книги, патентные документы, депонированные рукописи и др. Основные разделы РЖ «Программное обеспечение» посвящены следующим вопросам: языки программирования; технология программирования; теория и проблематика вычислительных сетей; операционные системы; системное программное обеспечение; программное обеспечение микропроцессоров; программное обеспечение систем управления базами данных; автоматизация проектирования.

Основные разделы РЖ «Вычислительные машины и системы»: теоретические основы вычислительной техники; элементы, узлы и устройства ВМ; вычислительные машины и вычислительные системы; технология и оборудование для производства

средств вычислительной техники, автоматизация проектирования ЭВМ, диагностика средств вычислительной техники.

Периодичность — 12 номеров в год
Подписка на 1987 год производится через «Союзпечать» и общественными распространителями печати в установленные сроки.

Условия подписки:

С авторским и предметным указателями (подписка годовая):

	Индекс	Цена
1. На выпуск РЖ «Вычислительные машины и системы»:		
для учреждений и организаций	55649	24 р. 72 к.
для индивидуальных подписчиков	55650	11 р. 04 к.
2. На выпуск РЖ «Программное обеспечение»:		
для учреждений и организаций	55651	24 р. 24 к.
для индивидуальных подписчиков	55652	10 р. 68 к.

Индекс и цена указаны по каталогу «Союзпечать»: Издания органов научно-технической информации, 1987 года. Редакция.

УДК 681.322.1

А. Н. Полосин, Н. Г. Карпинский, И. О. Лозовой,
А. И. Половянюк, О. В. Ургант, М. И. Дябин

УЧЕБНЫЙ КОМПЬЮТЕР «ЭЛЕКТРОНИКА УК НЦ»

Персональная ЭВМ «Электроника УК НЦ» (далее УК НЦ) предназначена прежде всего для оборудования кабинетов вычислительной техники всех типов учебных заведений. К учебному компьютеру (УК) предъявляются сложные и противоречивые требования. Он должен обладать достоинствами профессиональной персональной ЭВМ (языки низкого и высокого уровней, дисковая операционная система, разнообразные редакторы текстов и документатора, режим работы с графикой, возможность объединения в локальную сеть). Вместе с тем УК должен иметь высокую надежность, не требовать специального обслуживания, быть относительно дешевым.

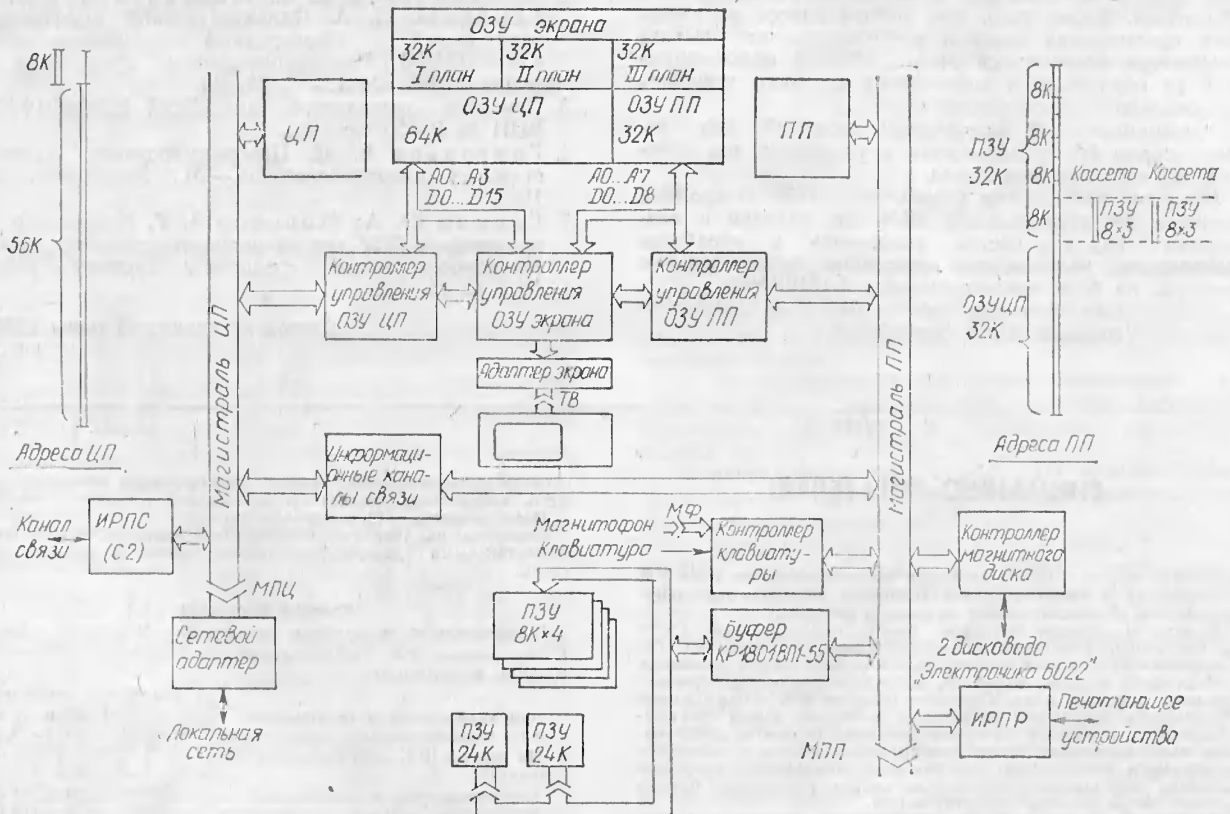
Архитектура УК НЦ — это двухмашинный комплекс, состоящий из центральной и периферийной машин, соединенных друг с другом с помощью высокоскоростных каналов связи (см. рисунок). Центральная машина (ЦМ) представляет собой универсальную микроЭВМ, состоящую из универсального 16-разрядного микропроцессора, ОЗУ емкостью 64К байт, последовательного порта ввода-вывода, адаптера локальной сети.

Микропроцессор K1801BM2 имеет производительность около 800 тыс. операций/с (типа регистр — регистр). За исключением четырех команд арифметических действий с плавающей запятой, все команды реализованы в самом микропроцессоре. Команды с плавающей точкой выполняются микропроцессором программно, в специальном режиме.

Оперативная память общей емкостью 64К байта разбита на две области. Первая, емкостью 8К байт доступна только в специальном режиме. Скрытая область используется для реализации команд с плавающей запятой, работы в режиме пульта, реализации различных режимов начального пуска ЦМ. Доступ в память осуществляется байтами или 16-разрядными словами.

Дуплексный последовательный порт ввода-вывода выполнен в соответствии со стандартом ИРПС с токовой петлей 20 мА. Скорость обмена информацией может задаваться с помощью перемычек в диапазоне 150... 57 600 бод.

Адаптер локальной сети позволяет объединить несколько УК НЦ в кольцевую локальную сеть. Обмен информацией осуществляется последовательно в соответствии со стандартом ИРПС со скоростью 57 600 бод по скрученной паре проводов. Длина одного звена сети может составлять несколько десятков метров. Предусмотрено автоматическое восстановление кольца при выключении любого входящего в состав сети УК НЦ или отключении его от сети. Предусмотрена возможность включения в сеть до 64 компьютеров.



Структурная схема ПЭВМ «Электроника УК НЦ»

Все устройства ЦМ (кроме ОЗУ) соединяются между собой с помощью 16-разрядной системной магистрали типа Q-bus. Микропроцессор осуществляет прямое обращение к ОЗУ, минуя системную магистраль. В УК НЦ предусмотрено подключение дополнительных контроллеров устройств к системной магистрали.

Периферийная машина (ПМ) является специализированной микроЭВМ и предназначена для управления стандартным набором периферийных устройств. Она выполняет функции контроллеров нескольких устройств: клавиатуры, видеомонитора, бытового кассетного магнитофона, генератора звуков, печатающего устройства, гибких магнитных дисков, кассетного ПЗУ.

В состав ПМ входят 16-разрядный микропроцессор K1801BM2, ОЗУ емкостью 32К байт, ПЗУ емкостью 32К байт, клавиатура, видеопроцессор, видеоЗУ емкостью 96К байт, интерфейс бытового кассетного магнитофона, таймер, программируемый генератор звука, программируемый параллельный порт ввода-вывода, интерфейс гибкого магнитного диска, интерфейс кассетного ПЗУ, ключи режимов.

Оперативная память емкостью 32К байт подключена непосредственно к микропроцессору. Обращение к ней возможно байтами или 16-разрядными словами. Запись или выборка слова осуществляется за два обращения в память.

Постоянная память емкостью 32К байт также подключена непосредственно к микропроцессору и предназначена для хранения операционной системы периферийной машины (ОС ПМ). Обращение к постоянной памяти возможно байтами или 16-разрядными словами. Для постоянной памяти можно использовать ПЗУ с ультрафиолетовым стиранием или масочным программированием.

Клавиатура насчитывает 88 клавиш, среди которых можно выделить несколько групп по их функциональному назначению: набор алфавитно-цифровых клавиш, дополнительная калькуляторная клавиатура, регистровые клавиши, клавиши управления курсором, клавиши программируемых ключей, управляющие клавиши.

Допускается одновременное нажатие до десяти клавиш: номера всех нажатых клавиш передаются в ПМ, фиксируется момент их отжатия. Это позволяет реализовать на УК НЦ игровые тренажеры или простой музыкальный инструмент (конечно, с использованием программируемого генератора звуков). Операционная система ПМ на основании информации о нажатых клавишах осуществляет генерацию кодов и кодовых последовательностей и передачу их в ЦМ по каналу связи. При этом осуществляется режим автоповтора с обычной или повышенной скоростью. Генерация кодов и кодовых последовательностей осуществляется в соответствии с одним из стандартных наборов КОИ-7 или КОИ-8. Кроме того практически на любую клавишу или сочетание клавиш можно назначить любой код или кодовую последовательность.

Для реализации с помощью любого монитора дисплейных функций в состав ПМ включены видеопроцессор (ВП) и видеоЗУ (ВЗУ). В качестве ТВ-монитора можно использовать любой промышленный черно-белый или цветной монитор или бытовой черно-белый или цветной телевизор, имеющих вход «Видео». Видеопроцессор записывает и корректирует информацию считывает и отображает информацию на экране ТВ-монитора в виде растрового изображения. Объем ВЗУ и телевизионный стандарт позволяют получать изображение на растре 640 × 288 точек с использованием прогрессивной развертки и кодированием точки тремя битами. В состав ВП входят восемь 5-разрядных регистров цвета. Таким образом на экране цветного ТВ-монитора можно одновременно отображать восемь цветов из палитры 32 цветов, причем каждая точка может иметь свой цвет.

Максимальная емкость экрана в алфавитно-цифровом режиме составляет 24 строки по 80 символов. Имеется режим отображения 40, 20 и 10 символов в строке. Возможно изменение масштаба и по вертикали. Это позволяет использовать в качестве ТВ-монитора малогабаритные телевизоры.

Рисунки символов задаются на растре 8 × 11 точек. Имеется полный набор рисунков символов в соответствии со стандартами КОИ-7, КОИ-8. Кроме того, предусмотрены рисунки символов табличной графики и других обозначений. По командам из ЦМ можно задавать собственные рисунки символов и присваивать им собственные коды. Символы могут иметь любой цвет и отображаться на фоне любого цвета. Дисплей отображает стандартный набор управляющих последовательностей, например перемещение курсора, стирание части экрана и т. д. В алфавитно-цифровом режиме дисплей УК НЦ полностью совместим с дисплеем персональной ПЭВМ ДВК-2М.

В графическом режиме имеется возможность рисовать основные геометрические элементы с различными параметрами: точку, маркер, отрезок, дугу, символы, закрашивать отдельные области. Для указания позиции на экране служит графический курсор, который перемещается с помощью клавиш управления курсором. Координаты курсора передаются в ЦМ в соответствии с текущим масштабом.

Для подключения к УК НЦ бытового кассетного магнитофона служит соответствующий интерфейс. Обмен информацией производится последовательным телеграфным кодом с частотным кодированием битов, в соответствии со стандартом MSX. Скорость записи 1200 или 2400 бод выбирается программно из ЦМ. Скорость чтения определяется автоматически по записанной информации.

Для управления бытовым кассетным магнитофоном, гибкими магнитными дисками и генератором звуков операционная система использует программируемый 12-разрядный таймер. Таймер может функционировать в нескольких режимах: режим отсчета интервалов времени с однократным или повторным запуском, режим замера интервала времени по внешнему событию и др. Работа с таймером осуществляется по опросу или по прерыванию. Максимальный отсчитываемый интервал времени — 16 мс. Так как таймер используется для управления несколькими устройствами, то одновременное использование этих устройств недопустимо.

Программируемый генератор звуков позволяет одновременно генерировать любое сочетание стандартных частот 60, 250, 500 Гц, 1, 8 кГц и программно генерируемой частоты. С его помощью можно создавать различные звуковые эффекты в широком частотном диапазоне.

Для подключения устройства печати или графопроектировщика в ПМ предусмотрен 24-разрядный программируемый порт ввода-вывода. Он может работать как на ввод, так и на вывод в различных программно-задаваемых режимах. Один из режимов полностью соответствует стандарту ИРПР, что позволяет подключать к нему стандартное устройство печати. Управление печатью осуществляет ОС ПМ в одном из трех режимов: печать информации, поступающей из ЦМ, отображение информации на ТВ-мониторе с параллельной печатью, сброс содержимого экрана на печать.

Для подключения накопителя на гибких магнитных дисках в ПМ предусмотрен соответствующий интерфейс, который позволяет подключать до четырех двусторонних дисков диаметром 133 мм с двойной плотностью с 40 или 80 дорожками. Емкость одной дискеты с 40 дорожками — 400К байт, с 80 дорожками — 800К байт. Интерфейс рассчитан на элементарные операции с диском, например обмен одним 16-разрядным словом. Все операции по управлению дисками осуществляет периферийный процессор в реальном времени с использо-

ванием таймера под управлением ОС ПМ. Он же отвечает за пересылку информации в память (и из памяти) ЦМ по каналу прямого доступа из ПМ в память ЦМ.

В качестве дополнительной памяти в УК НЦ используется кассетное ПЗУ (КПЗУ) емкостью от 8 до 48К байт. ОС ПМ осуществляет с помощью «окна» пересылку информации из любой области КПЗУ в ОЗУ ЦМ по каналу прямого доступа.

Так как начальная загрузка УК НЦ по включению питания может осуществляться с различных устройств: ГМД, КПЗУ, локальной сети, кассетного магнитофона, то необходимо каким-то образом указать источник загрузки. Кроме того, могут потребоваться и другие режимы начального пуска, например запуск интенсивных тестов или немедленное обращение к оператору. Для задания режима начального пуска служат ключи режимов, анализируя которые ОС ПМ выбирает тот или иной режим работы при включении питания.

Центральная и периферийная машины соединены несколькими идентичными каналами связи. Обмен информацией по ним осуществляется байтами со скоростью нескольких сотен тысяч байтов в секунду. Со стороны ЦМ они выглядят как стандартные ИРПР каналы: регистры данных в состоянии располагаются по стандартным адресам, что позволяет использовать стандартные программные средства.

Необходимость иметь несколько каналов вызвана многофункциональным назначением ПМ и желанием достигнуть полной программной совместимости на уровне операционных систем ОС ДВК и Рафос. Кроме упомянутых каналов, существует еще канал прямого доступа в память ЦМ. Обмен по нему осуществляется исключительно силами ПМ и не требует вмешательства со стороны ЦП. По этому каналу осуществляется передача основного потока информации между внешними запоминающими устройствами и ЦМ.

Принятые архитектурные решения позволили: освободить центральный процессор от управления периферийными устройствами, что существенно повысило производительность ПЭВМ;

добиться полной программной совместимости с микроЭВМ ДВК-2М и «Электроника 60» на уровне стандартных операционных систем;

существенно расширить функциональные возможности и гибкость компьютера: цветная графика, программируемые режимы работы, большой набор кодировок и символов с возможностью их быстрой смены, дополнительные внешние устройства;

в несколько раз снизить стоимость по сравнению с ДВК-2М и «Электроника 60» в аналогичной комплектации;

уменьшить габариты, потребление энергии и увеличить надежность.

Конструктивно УК НЦ выполнена в виде двух плат: на одной (системная плата) размерами 280 × 240 мм располагаются ЦМ и ПМ; на другой размерами 374 × 125 мм размещается клавиатура. Плата клавиатуры представляет собой коммутационную матрицу и не содержит радиоэлектронных компонентов. Обе платы вместе с блоком питания размещаются в пластмассовом корпусе размером 445 × 290 × 75 мм. В верхней части которого размещаются клавиатура и приемный карман КПЗУ. В задней части корпуса располагаются разъемы портов, интерфейсов и системных магистралей. В нижней части корпуса расположен подвал для установки адаптера локальной сети или другого контроллера, подключаемого к системной магистрали ЦМ.

Системная плата располагается в корпусе УК НЦ под клавиатурной платой. На клавиатурной плате помещаются клавиши, расположенные в шесть рядов. Общая масса компьютера ≤ 5 кг, потребляемая мощность 25 Вт.

Возможность применения в составе УК НЦ разнообразных периферийных устройств позволяет использовать его в режимах автономной работы с начальной за-

грузкой с кассеты или кассетного ПЗУ (режим Бейсик — машины), автономной работы под управлением стандартной дисковой операционной системы с начальной загрузкой с ГМД, в составе локальной сети класса с начальной загрузкой и последующими эпизодическими обменами информацией по сети, в качестве спутника мини-ЭВМ с подключением с помощью последовательного или параллельного порта ввода-вывода и с начальной загрузкой от мини-ЭВМ. Возможны и разнообразные гибридные варианты. При использовании черно-белого ТВ-монитора часть ВЗУ можно использовать в качестве электронного диска, снизив число обменов с ГМД или по сети.

Высокие технические характеристики, удобство в эксплуатации, полная программная совместимость с широко распространенными микроЭВМ «Электроника 60» и ДВК-2М, достаточно низкая стоимость позволяют использовать УК НЦ для обучения основам информатики, программирования и другим дисциплинам в школах, техникумах, ПТУ и высших учебных заведениях.

Телефон для справок: Москва, 534-06-02. Полосин Александр Николаевич, после 19 часов.

Статья поступила 4 июня 1986 г.

УВАЖАЕМАЯ РЕДАКЦИЯ

Вот уже больше года я получаю ваш журнал. И могу сказать, что с радостью открываю очередной номер, в каждом нахожу для себя много интересного и полезного. Однако хочу высказать ряд замечаний.

Не подлежит сомнению, что некоторые из публикуемых вами схем будут повторять. Вот тут и возникает парадокс.

С одной стороны, как указывается, схемы работоспособны и прочно себя зарекомендовали. А с другой стороны, в них наблюдаются значительные отклонения от ТУ и ОСТов на применение микросхем.

В № 3, 1985 г. была опубликована статья «Устройство обработки информации с малым энергопотреблением». Нетрудно заметить, что указанная частота генерации (8 МГц и более) задающего генератора для серии К561 (564) является сомнительной, так как время задержки элемента данной серии, при указанном напряжении питания, составляет около 100 нс и влечет за собой (при работе с частотами выше 2—4 МГц) необходимость подбора элементов, не говоря уже о взаимном согласовании реактивных сопротивлений кварцевого резонатора и элементов задающего генератора. Для большей убедительности схема была повторена. Генерация наблюдалась лишь до частот 5—6 МГц и то с разными экземплярами микросхем и варьировалось напряжение питания от 10 до 15 В. То же самое относится и к формирователю тактовых последовательностей для БИС КР580ИК80, диаграмма которого (формирователя) и так не соответствует указанной в ОСТ 11 348.917-82 (стр. 15), а из-за разброса параметров элементов может вообще оказаться непредсказуемой. Необходимо еще отметить, что выбранный способ включения РПЗУ К573РФ2 (входы CS и OE объединены) обуславливает длительность сигнала чтения не менее 650 нс, что никак не вяжется с выбранной частотой 2 МГц (длительность сигнала чтения 500 нс).

Еще один пример. В № 2 за 1986 г. в статье «Модуль процессора персональной ЭВМ «Ириша» вызывает сомнение выбранное включение БИС К580ВМ59. Как указывается в материалах фирмы «Intel» на БИС К5259 и ее отечественный аналог К580ВМ59 в ОСТ 11 348.917-82 (стр. 153), длительность третьего сигнала формируемого БИС К580ВМ59 недостаточна для ЦП КР580ИК80, вследствие чего требуется его задержка. По непонятным причинам авторы игнорировали данную рекомендацию.

С уважением, инженер А. Ю. Командный, г. Красноярск
(Ответ см. на с. 91)

УДК 681.3.06

В. В. Липаев, А. И. Потапов

ДЛИТЕЛЬНОСТЬ РАЗРАБОТКИ СЛОЖНЫХ ПРОГРАММНЫХ СРЕДСТВ

Объемы программ, реализуемых на встраиваемых микроЭВМ и персональных ЭВМ различных типов, постоянно возрастают. Разработка программных средств (ПС) как продукции производственно-технического назначения в большинстве случаев носит коллективный характер, требует значительной трудоемкости и может продолжаться несколько лет. Для прогнозирования длительности создания новых ПС необходимы данные о том, как разрабатывались уже эксплуатируемые средства. Накоплен значительный опыт разработки различных классов ПС на больших машинах. Его можно использовать при анализе программ микроЭВМ с достаточно большими ресурсами. Изучение экспериментальных характеристик длительностей разработки реальных проектов позволяет выявить основные закономерности их изменения и факторы, влияющие на продолжительность работ. Технико-экономические показатели конкретных проектов-аналогов, а также классифицированные и обобщенные данные совокупностей однородных проектов могут служить базой исходных данных для планирования и прогнозирования разработок ПС в различных условиях. Воздействие на управляемые факторы позволяет изменять длительность разработки в нужных направлениях.

Объектом изучения стали значения длительностей реальных разработок ПС и их крупных компонент, полученные методами ретроспективного и фактографического сбора технико-экономических характеристик [1]. Объем каждого из исследованных ПС изменялся от нескольких тысяч до миллиона строк исходного текста, т. е. на три порядка. Однако большинство ПС имело объемы от 10 до 200 тыс. команд. Практически все разработки не содержали программных компонент, перенесенных из предшествующих проектов. Суммарный объем ПС, учтенных при анализе, превысил 15 млн. команд.

Анализировались только коллективные разработки, в которых участвовали специалисты разной квалификации, в том числе руководители, системные аналитики и вспомогательный персонал. Вследствие этого в значительной степени нивелировались индивидуальные способности специалистов, определяющей становилась усредненная квалификация и оснащенность технологии разработки средствами автоматизации. Большие объемы ПС и коллективный характер работ обуславливают необходимость соблюдать регламентированную технологию разработки программ. Этапы системного анализа и разработки идейной части алгоритмов не учитывались. Длительность разработки исчислялась от этапа технического задания на ПС до завершения приемочных испытаний заказчиком. Этапы технологического процесса соответствуют схеме, приведенной в работе [2].

Общие особенности длительностей разработки ПС. Трудоемкость и стоимость разработок современных ПС находится в пропорциональной зависимости от объема. Диапазон изменения объемов ПС составляет величину от 10^3 до 10^7 строк, а соответствующий ему диапазон изменения длительностей разработок ПС по полному технологическому циклу как продукции производственно-технического назначения значительно меньше (от 1 до 10 лет).

Основным фактором, определяющим верхнюю и нижнюю границы, является объем программ — P_k (гипотетический график для сложных ПС реального времени приведен на рис. 1). Относительный «консерватизм» значений длительности по сравнению с трудоемкостью определяется необходимостью создавать ПС в определенные сроки: ПС должны поступать в эксплуатацию до того, как устареют цель их создания, концептуальная основа и алгоритмы. Верхний предел допустимых длительностей разработки, очевидно, не может иметь единственное значение для любых классов и объемов ПС, однако недопустимо его изменение в таком же широком диапазоне, что и объема программ. Поэтому на практике по мере возрастания объема ПС увеличиваются коллективы специалистов, что обеспечивает основной прирост необходимой трудоемкости. Чем крупнее создаваемое ПС, тем большие усилия обычно прилагаются для автоматизации и совершенствования технологии разработки.

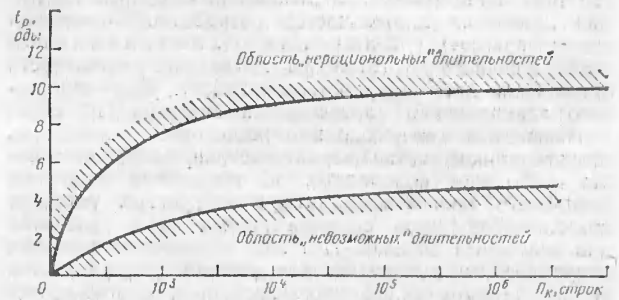


Рис. 1. Гипотетические значения областей «невозможных» и «нерациональных» длительностей разработки для сложных ПС в зависимости от объема программ P_k

Стремление ограничить длительность реальных разработок ПС приводит к объективному формированию верхнего предела, за которым распространяется зона «нерациональных» длительностей, зависящих от объема ПС (рис. 1). Даже для весьма сложных ПС, имеющих объем $P_k \sim 200$ тыс. строк, вряд ли допустима длительность разработки более 3...5 лет. Большие длительности, иногда встречающиеся на практике, обусловлены в основном низкой квалификацией разработчиков и заказчиков, недостаточной автоматизацией технологии, малым коллективом специалистов и рядом других, преимущественно организационных и технологических причин. Подобные ситуации чаще встречаются при относительно небольших разработках (10...50 тыс. строк), когда у руководителей и коллектива мал опыт их проведения, следствием чего является избыточный оптимизм в начале разработки.

Границу снизу определяет естественный технологический процесс коллективной разработки и необходимость выполнения ряда взаимосвязанных работ на последовательных этапах, которые обеспечивают получение ПС требуемого качества. Подготовка текстов программ, их тестирование, комплексирование, документирование и испытания могут проводиться только последовательно.

Попытки форсировать работы на отдельных этапах, как правило, приводят к возрастанию продолжительности других этапов. «Подключение» дополнительных специалистов увеличивает затраты на разработку и только в некоторых случаях обеспечивает сокращение сроков. Нередко увеличение числа специалистов дает обратный эффект — длительность разработки растет с увеличением затрат [3].

Под воздействием перечисленных факторов формируется объективный минимум длительностей — граница области «невозможного» (рис. 1), зависящая в первую очередь от объема разрабатываемых ПС. Нижняя граница длительностей еще более консервативна, чем верхняя [4]. Изменение этой границы возможно в небольших пределах только в процессе совершенствования технологии, повышения программной и аппаратурной оснащенности разработки, а также роста коллективной квалификации разработчиков и заказчиков.

Граница «иррациональных» длительностей имеет значения приблизительно вдвое большие, чем значения границы «невозможных» при том же объеме ПС. По данным реальных разработок может быть оценена средняя или наиболее вероятная длительность разработок ПС определенного класса при заданных условиях. Конкретное распределение длительностей зависит от исходных данных, имеющихся в базе данных технико-экономических характеристик завершенных разработок и от метода их усреднения. Ниже приведены результаты для весьма разнородных разработок, полученные при применении различных технологий. Представленные результаты имеют иллюстративный характер и служат только для выявления основных тенденций изменения длительностей разработки программ трех классов. Значения длительностей нормированы (отнесены к средней длительности разработки программ третьего класса). Для конкретного планирования длительностей создания ПО определенных классов необходимо исследовать и обобщить технико-экономические характеристики разработок более или менее однородных по технологии и другим условиям. Такие обобщения при конкретных условиях позволяют получить опорные абсолютные значения для некоторых объемов ПС. Эти абсолютные значения могут быть использованы для расчета коэффициентов нормирования с целью прогнозирования длительностей разработок на базе выявленных ниже закономерностей и реальных опорных значений для конкретных условий.

Анализ длительностей разработки ПС по данным, полученным ретроспективным методом. Исходные данные для исследования реальных длительностей разработок трех классов ПС: систем реального времени (СРВ) (управление технологическими процессами, роботами, гибкими производством и т. д.); информационно-поисковых систем (ИПС); пакетов прикладных программ и систем автоматизации разработки программного обеспечения (САРПО) достаточно подробно описаны в работе [5]. Программные средства создавались неодинаковыми по квалификации и оснащению коллективами, в разные годы и для различных задач, что привело к большому разбросу длительностей. Гистограмма (рис. 2) отражает распределение нормированных относительных значений длительностей для класса СРВ. Значения длительностей объединены по интервалам времени (годам завершения разработок) в три группы (I, II и III группы). Выявлена тенденция сокращения длительностей разработок по мере совершенствования во времени технологии создания ПС. Следует учитывать, что группа III в данном случае состоит из ПС, несколько меньших по объему по сравнению с группами I и II.

Коэффициенты вариации длительностей для трех исследованных классов ПС (СРВ, ИПС и САРПО) близки к 0,5 (0,48; 0,48 и 0,56 соответственно), что приблизительно в полтора раза меньше, чем коэффи-

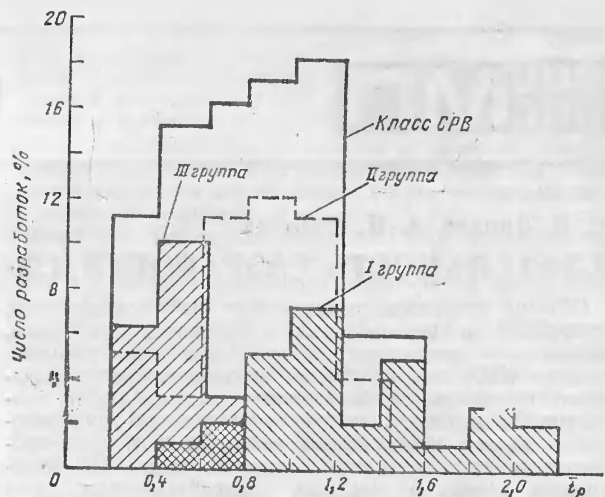


Рис. 2. Гистограмма относительных длительностей разработки ПС реального времени (штриховка справа налево — III группа, штриховка слева направо — II группа, □ — I группа, сегка — класс СРВ)

циент изменения производительности труда для тех же классов (0,84; 0,81 и 0,71). В то же время трудоемкость разработок насчитывает от нескольких человеко-лет до тысячи человеко-лет. Таким образом, несмотря на значительную неоднородность данных, изменение длительности разработки значительно меньше изменения трудоемкости и производительности труда.

Обобщенные данные по трем классам программ аппроксимировались уравнениями регрессии вида $y = a \times x^b$ по методу наименьших квадратов в зависимости от объема ПС и трудоемкости разработки. При аппроксимации в зависимости от объема программ отсутствие четкой формализации единиц измерения объема дало дополнительный разброс характеристик между различными классами программ. Программы СРВ практически полностью разрабатывались на ассемблере. Программы других двух классов ПС — на языках высокого уровня и ассемблере. Поэтому за единицу измерения объема принималось число объектных команд в программе, которое считалось эквивалентным числу операторов исходного текста на ассемблере.

В результате корреляционной обработки технико-экономических показателей получены значения относительной длительности t_p разработки ПС (см. таблицу).

Класс ПС	Относительная длительность разработки	
	в зависимости от объема ПС (P_k)	в зависимости от трудоемкости ПС (C_{1P})
СРВ	$0,208 P_k^{0,14}$	$0,304 C_{1P}^{0,24}$
ИПС	$0,092 P_k^{0,17}$	$0,166 C_{1P}^{0,34}$
САРПО	$0,186 P_k^{0,13}$	$0,316 C_{1P}^{0,24}$

Длительности разработок ПС для СРВ заметно превышают длительности для двух других классов (рис. 3). Так как часть программ САРПО и ИПС разрабатывалась на языках высокого уровня, то оценки объемов ПС для этих классов несколько ниже, чем при расчете объема по числу операторов на ассемблере. Если все объемы ПС привести к числу операторов на ассемблере, то различные длительности разработки ПС для СРВ и для двух других классов еще больше возрастает.

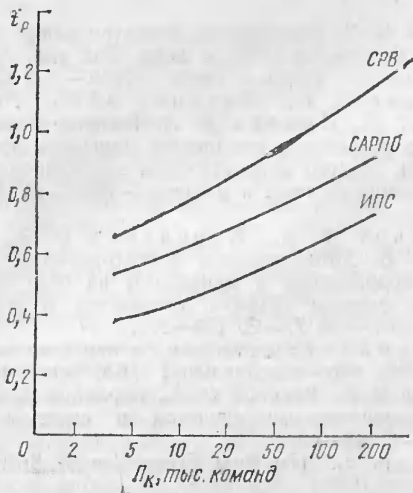


Рис. 3. Зависимость относительной длительности разработки ПС трех классов от объема программ L_k .

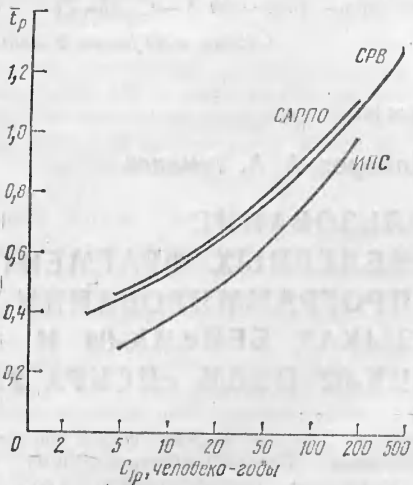


Рис. 4. Зависимость относительной длительности разработки ПС трех классов от трудоемкости непосредственной разработки программ C_{1P} .

Зависимость длительности разработки от трудоемкости (рис. 4) учитывает только трудоемкость непосредственной разработки программ C_{1P} без затрат на средству автоматизации разработки и машинное время. Показатели степеней 0,24 и 0,34 в уравнениях практически совпадают с соответствующими значениями (0,25; 0,38) для разных классов ПС, указанных в работе [4]. Этот факт подтверждает достаточно универсальный характер зависимости вида $t_p = a \sqrt[n]{C_{1P}}$ ($n = \frac{1}{b}$)

для описания связи между трудоемкостью и длительностью разработки. Относительно узкий и практически одинаковый диапазон изменения значений степени ($\sqrt[3]{C_{1P}}$ или $\sqrt[4]{C_{1P}}$) в разных исследованиях подтверждает объективный характер связи между длительностью и трудоемкостью разработки сложных ПС. Практика разработок ПС различных типов показывает, что, несмотря на большие трудности, имеются и могут применяться методы и средства управления факторами, которые позволяют разрабатывать любые ПС в рациональные сроки.

Анализ длительностей разработки крупных компонент ПС по данным, полученным фактографическим мето-

дом. Для сбора информации о процессе создания компонент и систем автоматизации разработки программ РУЗА [6] и ПРА [7] в течение всего срока работ осуществлялось технико-экономическое сопровождение проектов, ежемесячно регистрировались данные о специалистах, участвовавших в создании компонент систем, и этапы разработки. Система РУЗА реализована на ЕС ЭВМ, состоит из 21 подсистемы, каждая из которых имеет объем от 5 до 46 тыс. команд. Система может быть настроена на разработку программ любых ЭВМ и имеет, в частности, версии для микроЭВМ, построенных на микропроцессорах КР580ИК80 и КМ1810ВМ86. Общий объем системы РУЗА около 500 тыс. команд или свыше 90 тыс. операторов языка ПЛ-1 и ассемблер. Под фактографическим контролем в течение всей разработки находилось 16 подсистем, составивших более 80 % полного объема системы. Система ПРА базируется на микроЭВМ СМ-1800, также может быть настроена на разные системы команд и имеет версию, обеспечивающую разработку программ для микроЭВМ на основе микропроцессора КР580ИК80. Фактографические данные регистрировались при создании 5 из 7 крупных компонент, составивших свыше 80 % объема всех программ, или более 50 тыс. операторов ассемблера.

Различие используемых языков программирования затруднило представление полученных данных в зависимости от объема компонент ПС. Поэтому основной анализ длительности разработки проведен в зависимости от трудоемкости создания компонент (рис. 5). Определяющей длительностью создания компонент оказалась длительность разработки систем в целом. Разработка обеих систем проводилась при более или менее одинаковых условиях коллективами, близкими по квалификации. Основную часть коллективов составляли молодые специалисты, для которых разработка РУЗА и ПРА являлись первыми. Процесс разработки характеризовался значительным дефицитом машинного времени, что ограничивало возможности применения средств автоматизации разработки программ. Эти особенности следует учитывать при анализе проводимых абсолютных значений длительностей разработки.

Средняя длительность разработки компонент системы равна 3,8 года, что составляет около 70 % длительности создания всей системы при коэффициенте вариации $\sim 0,4$. Длительность разработки ряда подсистем практически соответствовала полной длительности создания системы РУЗА в целом и оказалась намного больше, чем длительность создания аналогичных по сложности автономных ПС. Даже относительно небольшие компоненты сложных ПС разрабатываются практически в сроки создания всего комплекса про-

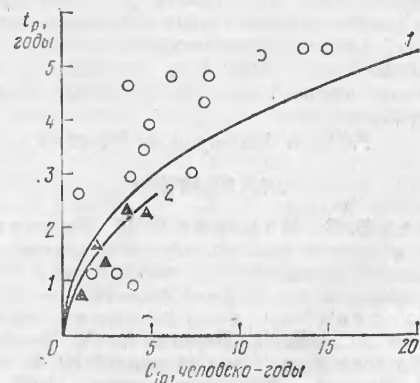


Рис. 5. Зависимость длительности разработки компонент САРПО РУЗА (О—1) и ПРА (Δ—2) от трудоемкости C_{1P} .

грамм, что определяется необходимыми сроками подготовки к комплексной отладке. Автономные ПС, имеющие объемы, близкие к объему компонент, но не входящие в более сложные системы, разрабатываются обычно в 2—3 раза быстрее.

Ниже приведены зависимости длительности разработки крупных компонент от трудоемкости для систем РУЗА и ПРА соответственно

$$t_p^{(P)} = 1,36C_{pP}^{0,5}; \quad t_p^{(П)} = 1,13C_{pP}^{0,57}.$$

Длительность разработки компонент (степень 0,5; 0,57) сильнее зависит от трудоемкости, чем длительность разработки систем в целом (степень 0,24). Это объясняется нивелировкой длительностей разработки больших ПС в целом за счет начальных этапов и комплексных работ. Кроме того отсутствовало стремление сокращать длительность разработки даже относительно небольших компонент, так как их поставка для комплексирования определялась общим графиком работ по системам. Затраты на обучение молодых специалистов также включены в приводимые данные.

Начальные этапы общесистемных работ до программирования компонент, а также комплексная отладка и испытания системы в целом занимают около 30% продолжительности разработки. Близкие результаты получены при обработке фактографических данных по этапам разработки компонент, входящих в состав систем РУЗА и ПРА. На средних этапах (программирование и автономной отладки компонент) всегда наблюдается значительное возрастание трудоемкости за счет привлечения программистов средней и низкой квалификации [8]. Начальные и конечные этапы разработки проводятся обычно небольшими группами специалистов преимущественно высокой квалификации. В результате трудоемкость средних этапов в несколько раз выше, чем крайних при более близкой по величине длительности всех этапов.

Заключение. Длительность разработки ПС во многих случаях является более важной характеристикой, чем трудоемкость и другие технико-экономические показатели. Увеличение трудоемкости разработки ПС относительно слабо (как корень третьей или четвертой степени) отражается на длительности создания сложных комплексов программ. Для полностью оригинальных разработок ПС существуют границы «невозможных» и «перациональных» длительностей, которые зависят от объема создаваемых ПС. Значения этих длительностей для фиксированного объема программ различаются в 2—3 раза. Некоторые реальные проекты выходят за границы «рациональных» длительностей, что определяется низким технологическим уровнем проектирования и другими факторами. В последнее время все большее число разработок приближается к границе «невозможных» длительностей. Преодолеть эту границу весьма трудно и наиболее эффективным методом для этого является сборочное программирование [9]. Другие методы, способные значительно уменьшать затраты, тем не менее относительно слабо влияют на длительность разработки.

Телефон для справок: 361-49-41, г. Москва.

ЛИТЕРАТУРА

1. Липаев В. В., Макаров В. И., Потапов А. И. Расчет и анализ технико-экономических показателей разработок программного обеспечения // Управляющие системы и машины.—1982.—№ 1.—С. 3—7.
2. Технологія проектирования комплексов программ АСУ / В. В. Липаев, Л. А. Серебровский, П. Г. Гаганов и др. // Под редакцией Ю. В. Асафьева и В. В. Липаева.—М.: Радио и связь.—1982.—260 С.
3. Брукс Ф. П. Как проектируются и создаются программные комплексы (мифический человек-месяц) / Пер. с англ. Под ред. А. П. Ершова.—М.: Наука.—1979.—152 С.

4. Бозм Б. У. Инженерное проектирование программного обеспечения / Пер. с англ. Под ред. А. А. Красилова.—М.: Радио и связь.—1985.—512 С.
5. Липаев В. В., Потапов А. И., Гавриловец Л. П., Имаева М. А. Технико-экономический анализ разработок комплексов программ при использовании средств автоматизации проектирования // Управляющие системы и машины.—1985.—№ 4.—С. 49—52.
6. Каганов Ф. А., Корепанов Б. А., Липаев В. В. Автоматизация проектирования программ для управляющих и микроЭВМ на базе технологической системы «Руза» // Автоматика и телемеханика.—1984.—№ 7.—С. 159—168.
7. Система автоматизации проектирования программ на базе персональных ЭВМ (система ПРА) / Липаев В. В., Каганов Ф. А., Керданов А. В. и др. // Микропроцессорные средства и системы.—1985.—№ 4.—С. 42—45.
8. Putnam L. The Real Economic of Software Development // The Economic of Information Processing.—1982.—V. 2. P. 167—176.
9. Ершов А. П. Опыт интегрального подхода к актуальной проблематике программного обеспечения // Кибернетика.—1984.—№ 3.—С. 11—21.

Статья поступила 6 июня 1986 г.

УДК 681.3.06.181.4

И. Е. Комаров, А. А. Туманов

ИСПОЛЬЗОВАНИЕ АССЕМБЛЕРНЫХ ФРАГМЕНТОВ ПРИ ПРОГРАММИРОВАНИИ НА ЯЗЫКАХ БЕЙСИК-01 И БЕЙСИК-02 ПЭВМ «ИСКРА 226»

Применение Бейсика иногда сдерживается низким быстродействием. Следуя общезвестному «правилу 95%»*, достаточно оптимизировать по скорости примерно 5% объема программы.

Языки Бейсик-01 и Бейсик-02 ЭВМ «Искра 226» допускают применение ассемблерных подпрограмм, запускаемых по оператору \boxtimes GIO', в котором задается массив данных.

Для эффективного применения ассемблерных подпрограмм можно использовать методы расширенного доступа к оперативной памяти (ОП) и ускоренного запуска ассемблерных подпрограмм в управляющей памяти (УП).

Расширенный доступ к оперативной памяти. В языке Бейсик-01 для многих программ указано в \boxtimes GIO' одного массива данных достаточно, и этот массив может служить источником данных для ассемблерного фрагмента (АФ), приемником данных или одновременно тем и другим.

Сложности возникают при необходимости обращения из АФ к нескольким объектам данных в Бейсик-программе. Простейший выход из положения — задать адресные константы, соответствующие началу каждого из объектов данных. Бейсик-01 не позволяет работать с машинными адресами, но узнать точные значения адресов можно экспериментально, используя оператор \boxtimes GIO'. При этом должна быть известна структура массивов и других объектов данных.

* Майерс Г. Надежность программного обеспечения, М.: Мир, 1980.

Пример:

```

10 DIM A Q (1000) 2, E % (25), F Q 180
20 Q GIO' HEX (FFF6), A Q ( ): STOP
30 Q GIO' HEX (FFF6), E% ( ): STOP
40 Q GIO' HEX (FFF6), F Q : STOP

```

Строка 10 задает структуру данных. Для определения адресов каждого из массивов необходимо запустить программу по RUN20, RUN30, RUN40

Однако предварительно перед запуском Бейсик-интерпретатора и приведенной программы необходимо задать останов на первой же команде АФ. Это делается по команде аппаратного загрузчика STOP 76000.

Для проведения эксперимента несущественно, какой именно АФ задается в операторе Q GIO'. Для простоты использован АФ из одной команды /BB/.

При выполнении оператора Q GIO' управление передается на адрес 76000 УП, происходит останов, и на экран выдается содержимое регистров ЭВМ.

Значение регистра B7 как раз и указывает на начало массива, заданного в соответствующем операторе Q GIO'. Так, для примера, получится следующее распределение адресов:

```

для A Q ( ) B7=174056
для E % ( ) B7=173766
для F Q B7=173474

```

Использовать полученные значения адресов можно в проектируемом АФ в виде константы или в переменной Бейсик-программы, доступной из АФ.

В языке Бейсик-02 по оператору ASMB B Q () * в УП записывается машинная программа, находящаяся в массиве B Q ().

Ускоренный запуск. При использовании данных АФ в ходе выполнения оператора Q GIO' перезапись кодов из ОП в УП достаточно длительная. В этом случае ускоренные выполнения функций, заложенных в АФ, повышается с помощью двух типов операторов Q GIO'.

Первый оператор «статический», вызывает перезапись длинного АФ в УП. В этом АФ первым словом должна быть команда /BB/, второе слово — резервное. Поэтому после перезаписи сразу происходит возврат в Бейсик-программу. Этот оператор располагается в подготовительной части программы.

Операторы второго типа располагаются в местах, где требуется выполнить некоторую функцию, заложенную в АФ. Вместо полного АФ в таких операторах задается АФ-стартер размером в одно слово. Это слово содержит команду /BP/ — переход на пущное место в «статическом» АФ. При однословном АФ-стартере оператор Q GIO' перезаписывает в УП только те машинные слова, которые в нем заданы, плюс команду /BB/.

```

В примере
100 Q GIO' HEX ( . . . . ), A ( )
200 Q GIO' HEX (80BC), B Q
300 Q GIO' HEX (00BD), C ( )

```

строка 100 задает пересылку АФ в УП, строка 200 вызывает запуск АФ с адреса 76200, строка 300 — с адреса 76400.

В языке Бейсик-02 расширенный доступ к ОП и ускоренный запуск реализуются оператором ASMB(адрес запуска > [, <список>...], где <адрес запуска> (осьмеричное число в диапазоне 0—76777) запускает машинную программу на счет; <список> — список переменных констант пользователя. Элементы списка разделяются запятыми.

Элементы списка — это исходные данные, обрабатываемые машинной программой, и результаты обработки. Оператор вычисляет параметры элементов списка (номер переменной, тип, значение или длина и адрес) и

располагает эти параметры в ОЗУ в виде списка, начальный и конечный адреса которого сформированы в ячейках 175 и 177.

№-й элемент	2-й элемент	1-й элемент
ЯЧ177	Увеличение адресов	→	ЯЧ175

Информация в ОЗУ записывается словами, причем число слов, занимаемое элементами списка, зависит от типа переменной.

Телефон для справок: 1-11-87, после 19.00., г. Смоленск.
Статья поступила 3 марта 1986 г.

УДК 681.325

К. Э. Поом, А. Э. Моор, Р. В. Ребане, Т. Э. Арулаане

ОПЕРАЦИОННАЯ СИСТЕМА ДЛЯ ПЭВМ «ИСКРА 226»

В данной работе описана реализация ОС CP/M-80 на ЭВМ «Искра 226» с использованием сопроцессора на КР580ИК80А, который присутствует в комплекте ПЭВМ «Искра 226» в виде БИФ телекоммуникации «Искра 015-85». Интерфейс с процессором ПЭВМ «Искра 226» для данного БИФ аппаратно реализован, поэтому остается дополнить объем оперативной памяти микроЭВМ на базе КР580ИК80А до 64К байт, обеспечить необходимое для ОС CP/M адресное пространство и раздельное адресное поле для команд ввода-вывода, что осуществляется добавлением к БИФ «Искра 015-85» отдельной печатной платы, занимающей соседнее место в расширителе.

В результате модернизации появляется дополнительная возможность такого применения комплекса ПЭВМ «Искра 226», где роль центральной микроЭВМ выполняет модернизированный БИФ «Искра 015-85», работающий под управлением ОС CP/M, а роль процессора ввода-вывода исполняет основной процессор ПЭВМ.

При разработке программного обеспечения выделяются две основные части: разработка BIOS для ОС CP/M в системе команд КР580ИК80А, который управляет операциями ввода-вывода для выполнения процессору ПЭВМ «Искра 226»; создание драйверов периферийных устройств в основном процессоре ПЭВМ.

Наиболее сложным при этом является драйвер накопителя на гибких магнитных дисках, что обусловлено отличием между требованиями ОС CP/M и микропрограммой БИФ «Искра 015-21», так как целесообразно использовать широко распространенную организацию НГМД ОС CP/M.

Модернизированный БИФ «Искра 015-85» является программно-совместимым со стандартным.

Реализован драйвер для НМД «Изот 5400». Кроме того, в свободном пространстве оперативной памяти ПЭВМ организован сверхпортативный аналог НГМД.

С использованием ОС CP/M-80 на ЭВМ «Искра 226» появляются возможности прямого управления любыми периферийными устройствами со стороны программы, работающей в среде ОС CP/M; отладки программ для микропроцессора КР580ИК80А и процессора связи ПЭВМ «Искра 226»; совместимости на уровне программ и носителей микроЭВМ СМ 1800; доступа к библиотеке инструментальных и программных средств. Инсталлированы программы под ОС CP/M для обработки данных и текстовой информации.

Соответствующая добавочная печатная плата разработана и эксплуатируется с 1986 г.

Адрес для справок: 492082, г. Таллин, 200006, ЭССР, а/я 1681, Центр обработки данных.

Статья поступила 9 апреля 1986 г.

М. М. Грудинин, А. Ю. Сенченкова

ОПЕРАЦИОННАЯ СИСТЕМА В ПЗУ

Операционная система, способная работать в постоянном запоминающем устройстве (ОС ПЗУ), разработана в СКТБ информационных систем Института проблем материаловедения АН УССР для микроЭВМ семейства «Электроника 60».

Система предназначена для автоматизации научных экспериментов, технологических процессов, стендовых испытаний.

МикроЭВМ «Электроника 60» с ОС ПЗУ и даже с минимальным набором периферийных устройств (ЭПМ, перфоратор и фотосчитыватель) используется в качестве персональной ЭВМ для инженерных расчетов. Перфоратор и фотосчитыватель необходимы для ввода и вывода программ при отладке. Система не использует расширенную арифметику с плавающей запятой, реализованную в процессоре М2. Таким образом, она может работать с платой «Электроника МС 1201.01» с процессором М1 и другими подобными средствами.

ОС ПЗУ включает в себя супервизор ввода-вывода; блок арифметики с плавающей запятой (48 разрядов мантиссы и 16 разрядов порядка); библиотеку стандартных подпрограмм; средства работы с внешними устройствами в режиме реального времени; калькулятороподобный язык высокого уровня.

ОС ПЗУ занимает около 8К слов, таким образом, система предоставляет пользователю 20К слов оперативной памяти. Такой небольшой объем системы при широком многообразии реализуемых функций достигается тем, что в архитектуре микроЭВМ эмулируется виртуальная многостековая микроЭВМ.

Известны операционные системы, эмулирующие стекковые компьютеры, например: ДССП [1], FORTH [2]. Опыт эксплуатации экспериментальной стековой мини-ЭВМ «Сетунь 70» [3] показал бесспорные достоинства стековой архитектуры. Однако вышеупомянутые виртуальные и реальные процессоры — двухстековые: программно доступен только операционный стек.

ОС ПЗУ имеет многостековую структуру. В языке высокого уровня программно доступны в явном виде два стека: арифметический (64-разрядный) и адресно-логический (16-разрядный). Язык высокого уровня имеет адресные средства работы со стеками, такие как: послать в стек; размножить вершину или подвершину; поменять их местами; тестировать вершину или сравнить ее с подвершиной и передать управление по результату тестирования или сравнения; печатать десятично вершину, с разрушением или без разрушения ее; переслать из арифметического стека в логический с преобразованием нормализованного числа в целое, и, наоборот, переслать из логического стека в арифметический с нормализацией и другие.

Большинство операторов безадресные, так как они совершаются над вершиной и подвершиной арифметического и адресно-логического стека. В системе имеются средства цветной графики. Цветной бытовой телевизор подключен к микроЭВМ с помощью платы КГМ (МС 4702). Семь операторов графического отображения позволяют достаточно гибко использовать площадь экрана цветного телевизора.

Неявно программисту доступны еще стеки: циклов, данных с датчиков, сохранения регистров, описаний массивов и системный стек возврата. Длина слова в каждом из этих стеков различна, поэтому ОС ПЗУ секционировала память и проводит диагностику на неполноценность отдельных сегментов памяти.

ОС ПЗУ — открытая система. Она имеет средства

модификации языка, дающие возможность разработчику создавать специализированные системы под определенный круг задач, включать в язык специализированные функции с такими именами, которые удобны пользователю. Другими словами, дает возможность создать для пользователя систему с языком по «заказу».

Язык высокого уровня предназначен, прежде всего, для пользователя-прикладника (не специалиста в области программирования), хотя удобен и для программиста.

Средства языка делятся на директивы и команды. Директивы — это вспомогательные средства пользователя. Ввод директивы (на русском языке) вызывает соответствующий диалог с пользователем. Некоторые директивы выполняются непосредственно после ввода без диалога, например: ЗАГРУЗИТЬ (ввести программу с перфоленки), ЛИСТИНГ (вывести текст программы на дисплей или ЭПМ), ВРЕМЯ (вывести текущее время), ДАТА (вывести текущую дату) и др. Набор всего слова директивы для команды необязателен — достаточно ввести три буквы.

Для ввода и отладки программы нужно ввести директиву ПРГ (программировать). Печать 0 (метка нулевого оператора) система сообщает, что редактор готов к работе. Программирование осуществляется в языке, близком к калькуляторному с польской ниверстной записью. Переменные обозначаются буквами. Константы можно вводить непосредственно в формулу. Знаки арифметических действий и элементарных функций имеют общепринятые обозначения (+, -, /, $\sqrt{\quad}$, LN, SIN, COS, TG, CTG, EXP и т. п.). В процессе ввода программы система автоматически метит каждый оператор. Метка — цифровая с двоеточием.

ОС ПЗУ располагает средствами диагностики правильности набора директив и операторов при программировании. Редактор не пропускает неправильно набранных операторов: при ошибках подается звуковой сигнал, печатает знак вопроса и возвращается в состояние «до ошибки». Это означает, что набор неправильно введенного оператора необходимо повторить, после чего редактор пропустит пользователя дальше. Если команде необходимы атрибуты или параметры, то после набора команды редактор запрашивает эти данные в диалоговом режиме. Запросы редактор печатает на русском языке, сопровождая их знаком вопроса.

В режиме программирования имеется группа команд, которые называются директивами редактора. Они выполняются немедленно после их подачи. К ним относятся: вставка, исключение, замена, листинг, выход в монитор (в режим подачи директив). При наборе программы можно редактировать вводимый текст и распечатывать на любом шаге ввода. Наборы директив редактора и команд не отличаются друг от друга.

Для выполнения программы можно в режиме редактора нажать клавишу «ПС» или в режиме монитора набрать директиву ВЫЧ (вычислить), или ВЧТ (вычислить, когда данные вводятся с терминала), или ВЫП (выполнить, когда данные не требуются или находятся в памяти). Подача команды «ПС» эквивалентна по диалогу с директивой ВЫЧ. Система спрашивает: «ЧИСЛО ЗНАКОВ ПОСЛЕ ЗАПЯТОЙ?». После ответа (от 0 до 10) задает следующий вопрос: «ГДЕ ДАННЫЕ?». Возможны такие ответы: ТЕРМИНАЛ, П/Л, ПАМЯТЬ. После ответа диалог продолжается: система запрашивает числовые значения всех входящих в формулу переменных. После этого программа выполняется, и система входит в монитор. Задавать значения переменных можно в любой удобной пользователю форме, даже в виде обыкновенных дробей. Система обрабатывает числа в диапазоне от 10^{-9863} до 10^{9863} по абсолютной величине.

Особенность редактора ОС ПЗУ в том, что он непосредственно в ходе программирования создает модуль в терминах виртуального компьютера. Эта программа

также может быть записана в ПЗУ. Команда ЛИНТИНГ генерирует текст командного модуля в текст на языке высокого уровня. Существует взаимнооднозначное соответствие между операторами языка и текстов в терминах виртуального компьютера. Для исправления выявленных ошибок набирается директива ПРОДОЛЖИТЬ, и управление передается редактору. Высвечивается следующая за последним оператором метка. Затем можно подавать редакторские команды или продолжить программирование.

Язык ОС ПЗУ назван авторами «Рось». «Комментарий» по командам и директивам занимает 12 машинописных листов. Нетрадиционные команды, например для интегрирования и дифференцирования, требуют достаточно полного описания, так как их употребление в обыкновенных дифференциальных уравнениях, в уравнениях с краевыми условиями и в уравнениях с частными производными имеет свои особенности.

Система снабжена средствами диагностики и идентификации периферийных устройств. Если устройство выключено или не готово к работе, то система после временной задержки печатает сообщение: ПЛ, ФС, АЦПУ или ВУ (внешнее устройство: АЦП или вольтметр) НЕ ГОТОВ. Такой диагностики нет в ФОДОС, RT11, RSX-11S ввиду невозможности этого при организации ввода-вывода по прерыванию с учетом приоритета устройства, т. е. если плата сопряжения с внешним устройством стоит не на своем месте, то ввода-вывода не будет. Если есть обращение к внешнему устройству, которое отсутствует, то ОС ПЗУ идентифицирует отсутствующее устройство и дает сообщение: ПЛ, ФС, АЦПУ или ВУ ОТСУТСТВУЕТ.

В настоящее время система ОС ПЗУ записана на плате ППЗУ 15УЗПП 8К×16—3.

В заключение отметим, что известные нам системы, кроме RSX-11S 4.1 и Multy Power (требует резидентную ЭВМ, подключенную по каналу связи), дают программы, не способные работать в ПЗУ (библиотечные программы не работают с ПЗУ).

ОС ПЗУ и все ее программные продукты работают в ПЗУ. На базе ОС ПЗУ разработаны и успешно функционируют три конкретные диалоговые системы автоматизации. Время разработки системы сокращено на порядок по сравнению с традиционными системами.

Адрес для справок: 252142, Киев, ул. Кржижановского, 3. СКТБ ИС Института проблем материаловедения АН УССР.

Телефон для справок: 444-02-56 (Киев).

Статья поступила 20 августа 1986 г.

УДК 681.325

Н. Н. Щелкунов, А. П. Дианов

ТЕХНИКА ПРОГРАММИРОВАНИЯ 8-РАЗРЯДНЫХ МИКРОКОНТРОЛЛЕРОВ

(Продолжение. Начало в № 5, с. 65.)

В предыдущем номере журнала представлены структурная и принципиальная схемы 8-разрядного одноплатного микроконтроллера (МК) ММС 1204, выполненного на микропроцессорных БИС серии КР580. Микроконтроллер реализует полный набор функций общесистемного профиля, обеспечивающих его широкое использование. В состав модуля ММС 1204 входит несколько программируемых устройств: два программируемых связанных адаптера (ПСА) типа КР580ВВ51, программируемый периферийный адаптер (ППА), построенный на основе КР580ВВ55, программируемый интервальный таймер (ПИТ) КР580ВВ53 и программи-

руемый контроллер прерываний (ПКП) типа КР580ВВ59 [1]. Значение и мнемоника базовых адресов этих периферийных устройств приведены в табл. 1.

Таблица 1

Определение базовых адресов

Порт	Адрес	Микросхема
PSA1	18H	ПСА1
PSA2	1AH	ПСА2
PPA	14H	ППА
TMR	1CH	Системный таймер
PKC	12H	Контроллер прерываний

Аппаратное окружение устройств в значительной степени предопределяет их функциональное назначение и режим работы. Удобно ввести стандартный низкоуровневый слой программного обеспечения (ПО), реализующий непосредственное управление аппаратурой МК. Использование такого типового программного слоя, содержащего ряд драйверов физических устройств, освобождает программиста от утомительного учета их аппаратурных особенностей. Организуется новый, не зависящий от аппаратуры, программный интерфейс управления общесистемными функциями микроконтроллера.

Подсистема ввода-вывода

Драйвер последовательного ввода-вывода. Входящие в состав МК два последовательных адаптера предназначены для организации асинхронного стартового интерфейса типа ИРПС. Минимальный драйвер каждого адаптера должен содержать три подпрограммы: инициализации, ввода и вывода по готовности.

Процедура S11N инициализирует ПСА1 для работы в асинхронном режиме с 7-битовым полем данных, контролем четности и двумя стоповыми битами. Фактор скорости устанавливается равным 1:16. Перед выдачей командного слова выбора режима SMODE (код OFAH) микросхема приводится в исходное состояние. Для этого в управляющий порт PSA1+1 выдается три нулевых байта и команда сброса SRES (код 40H). Предварительная выдача трех нулевых байтов обеспечивает перевод БИС КР580ВВ51 независимо от ее текущего состояния в рабочий режим, когда она сможет правильно отреагировать на команду программного сброса SRES. Команда SEIO (код 05H) разрешает работу одновременно как приемника, так и передатчика. После чего выполняется холостая процедура ввода по готовности S11, уничтожающая ложный символ, который может возникнуть в приемнике адаптера во время процедуры инициализации.

Следующие две процедуры условного ввода S11 и вывода S10 обеспечивают чтение и запись символа по готовности из приемника и в передатчик ПСА1 соответственно. Символ передается через аккумулятор. Признаком выполнения ввода или вывода очередного символа служит сброс флага нуля Z в слове состояния программы PSW микропроцессора. Эти три процедуры представляют собой простейший драйвер ввода-вывода ПСА1. На его основе можно построить более сложные процедуры ввода-вывода, такие как ввод символа с занятием цикла и эхо-ответом или вывод байта в шестнадцатеричном (HEX) формате и т. д. Минимальный драйвер ПСА2 имеет аналогичный вид с заменой базового адреса PSA1 на PSA2. Более подробную информацию о программировании ПСА как и всех других рассматриваемых здесь приборов можно найти в работе [2].

Драйвер параллельного ввода-вывода. Построенный на основе КР580ВВ55 параллельный адаптер ММС1204 организует 8-разрядный ввод и вывод данных, удовлетворяющий протоколу типа ИРПП. С целью стан-

дартизации функций ввода-вывода в драйвер ППА включены три процедуры: PAINI, PAI и PAO, имеющие тот же смысл, что и соответствующие процедуры драйвера ПСА. Для инициализации ППА необходимо в порт PPA+3 выдать управляющее слово выбора режима PMODE (код 0A7H) и затем разрешить ввод-вывод. Стандартизация интерфейса ввода-вывода делает независимым ПО следующего уровня от типа используемого адаптера, в качестве которого может выступать как ППА, так и любой ПСА. Появляется возможность определить ряд дополнительных виртуальных устройств ввода-вывода, удовлетворяющих данному стандарту.

Драйвер буфера памяти. Во многих случаях может оказаться полезной организация ввода не с внешнего устройства, а из памяти МК. Для этого используется драйвер буфера памяти. В драйвере определена процедура инициализации BF1WI, устанавливающая указатель BFPTR начала буфера, и процедура ввода BF1, указатель BFPTR размещается в области ОЗУ и служит для указания следующего подлежащего выборке байта памяти. При выборке байта значение указателя увеличивается на единицу. Аналогичным образом может быть определена более редкая операция вывода в память.

Драйвер устройства фиктивного вывода. Виртуальное устройство фиктивного вывода помогает осуществлять кажущийся вывод, который в действительности отсутствует. Драйвер такого устройства поддерживает единственную операцию вывода по готовности ВВО.

Организация системы ввода-вывода. Набор драйверов совместно с адаптерами образует физическую систему ввода-вывода, в составе которой можно выделить восемь устройств физического ввода или вывода. Среди них устройства :O1: и :I1: — передатчик и приемник (ПСА1), :O2: и :I2: (ПСА2), :O3: и :I3: (ППА), :BF: (буфер памяти) и :BB: (фиктивный вывод).

Над системой физического вывода надстраивается аппаратно независимая система логического вывода (рис. 1), содержащая пять логических устройств — :CO: и :CI: — образуют логическую консоль, :NO: и :NI: — формируют логический канал ввода-вывода из сети и логический принтер :LP:. Внутренний интерфейс драйверов логической системы ввода-вывода согласуется со стандартными правилами передачи параметров при вызовах физической системы. Логические устрой-

ства назначаются и переназначаются на соответствующие физические для реализации той или иной конфигурации ввода-вывода.

В мониторе MON 1200 назначенные логических устройств реализуется с помощью 15-байтовой таблицы связи, расположенной в области ОЗУ микроконтроллера (см. ниже).

CIRDY	OPG JMP	I3E0H S11	:База таблицы : :CI:=:I1: ввод с консоли
CORDY	JMP	S10	: :CO:=:O1: вывод на консоль
LPRDY	JMP	PAO	: :LP:=:O3: вывод на принтер
NIRDY	JMP	S21	: :NI:=:I2: ввод в сети
NORDY	JMP	S20	: :NO:=:O2: вывод в сеть

Драйверы логических устройств могут иметь развитый набор функций ввода-вывода. Так, поддерживаемый монитором MON 1200 драйвер консоли и принтера включает до 20 процедур, связь с которыми показана в табл. 2. Вывод на консоль может сопровождаться

Таблица 2

Связь с системой ввода-вывода

Процедура	Адрес	Назначение
CO	009FH	Вывод символа (A) на консоль
COTXT	00B5H	Вывод текстовой строки
COLF	00C7H	Перевод строки
COTLF	00C4H	Вывод текста и перевод строки
COH	06DCH	Вывод (AL) в HEX-формате
COH2	00EAH	Вывод (A) в HEX-формате
COH4	00F9H	Вывод (BC) в HEX-формате
MSGER	0104H	Вывод ERROR с кодом (A)
CI	0119H	Ввод в (A) символа с консоли
CIO	0122H	Ввод символа с эхо-выводом
HEXA	012CH	Преобразование HEX-кода (A)
CIH	0129H	Ввод в (AL) HEX-цифры
CIH2	0149H	Ввод в (A) двух HEX-цифр
CIH21	014DH	Ввод в (A) второй HEX-цифры
CIH4	015EH	Ввод в (BC) 4-х HEX-цифр
CIH41	0165H	Ввод в (BC) 3-х оставшихся цифр
COLP	01A1H	Вывод (A) на консоль и принтер
LP	0176H	Вывод символа на принтер
COPY	018BH	Включение копирования
NCOPY	0196H	Выключение копирования
S1INI	0201H	Инициализация ПСА1
S11	021AH	Ввод по готовности из ПСА1
S10	0222H	Вывод по готовности в ПСА1
S1NI	022EH	Инициализация ПСА2
S21	0244H	Ввод по готовности из ПСА2
S20	024CH	Вывод по готовности в ПСА2
PAINI	0258H	Инициализация ППА
PAI	0267H	Ввод по готовности из ППА
PAO	0271H	Вывод по готовности в ППА
BFINI	027DH	Инициализация буфера памяти
BF1	0281H	Ввод из буфера
ВВО	028FH	Фиктивный вывод

Уровень логического ввода-вывода

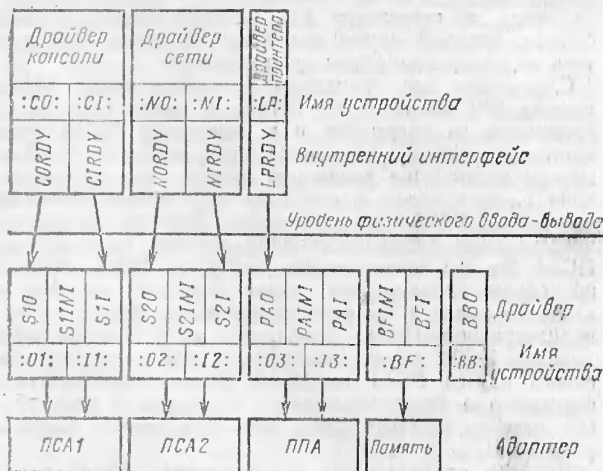


Рис. 1. Организация системы ввода-вывода микроконтроллера ММС 1204

копированием диалога на принтер. Управление копированием (его включение и выключение) осуществляется специальными процедурами COPY и NCOPY.

Драйвер системного таймера. В состав аппаратуры микроконтроллера мМС 1204 входит ряд стандартных средств поддержки работы в реальном масштабе времени [3]. Среди них два системных таймера (таймер системного времени и таймер временных интервалов) и 8-уровневая система прерываний [4]. Третий таймер резервируется как генератор скорости передачи ПСА1 и ПСА2.

Аппаратная логика мМС 1204 предусматривает строгую специализацию всех счетчиков и их резервирование исключительно для общесистемных целей. Благодаря этому удается определить простой драйвер системного таймера, содержащий четыре стандартные подпрограммы.

Драйвер поддерживает следующие процедуры: VEL — установка скорости передачи по связным каналам, CLK — установка частоты следования системных меток времени, DLINI — инициализация счетчика задержки и DELAY — запуск счетчика задержки. Процедура DLINI подготавливает счетчик СТИ для работы в режиме 0 (прерывание по окончании счета) и сбрасывает его выход в состояние с напряжением «Лог. 0» (отсутствие запроса на прерывание IR2). Процедура DELAY загружает счетчик задержки СТИ параметром, после чего он начинает счет. По окончании счета генерируется запрос на прерывание IR2 по второму уровню. Входной параметр процедур VEL, CLK и DELAY передается через 16-разрядный регистр В.

Система прерываний. Программируемый контроллер прерываний KP580BH59 обеспечивает различные режимы работы системы прерываний. Важно определить стандартный способ функционирования, для поддержки которого может быть использован ряд специальных системных подпрограмм нижнего уровня. В качестве последнего удобно выбрать наиболее широко используемый режим строгого вложения приоритетов. Находясь в рамках данного режима можно написать драйвер, включающий четыре стандартные подпрограммы.

Процедура ICINI инициализирует ПКП для работы в данном режиме. В качестве входного параметра через 16-разрядный регистр В процедура передается базовый адрес ITAB из представленной ниже таблицы векторов. Здесь ADDR N=0..7 служит указателем точки входа в подпрограмму обслуживания прерывания соответствующего уровня.

ITAB:	ORG	13 OH	
IL0:	JMP	ADDR0	; База таблицы
	DB	0	; Пользователь
	JMP	ADDR1	; Системное время
IL1:	DB	0	
	JMP	ADDR2	; Система отчета пауз
IL2:	DB	0	
	JMP	ADDR3	; Приемник ППА
IL3:	DB	0	
	JMP	ADDR4	; Приемник ПСА2
IL4:	DB	0	
	JMP	ADDR5	; Приемник ПСА1
IL5:	DB	0	
	JMP	ADDR6	; Пользователь
IL6:	DB	0	
	JMP	ADDR7	; Пользователь
IL7:	DB	0	

Следует помнить, что таблица должна располагаться по 32-байтовой границе. Инициализация включает в себя команду разрешения прерывания EI при полностью сброшенных масках ПКП (прерывания по всем уровням запрещены). Для управления маской служит специальная пара процедур MIN и MOUT, в которых регистр А используется для передачи кода маски.

При каждом включении питания или нажатии клавиши сброса монитор MON 1200 инициализирует систему прерываний с базой таблицы векторов прерываний, равной 13СОН. При этом все векторы принимают нулевое значение. Пользователь управляет средствами

Связь со средствами поддержки режима реального времени

Процедура	Адрес	Назначение
VEL	0297H	Установка скорости ПСА
CLK	02A4H	Установка системного времени
DELAY	02B1H	Счет паузы
DLINI	02BAH	Сброс счетчика паузы
ICINI	02C1H	Инициализация системы прерываний
EOI	02D4H	Команда EOI
MIN	02DBH	Чтение маски прерываний
MOUT	02DEH	Установка маски прерываний

поддержки реального времени, используя вышеуказанные процедуры, связь с которыми определена в табл. 3.

Введение низкоуровневого слоя ПО, непосредственно управляющего аппаратурой МК, позволяет организовать стандартную среду программирования, свободную от аппаратных особенностей ее реализации. Двухуровневая система ввода-вывода обладает достаточной гибкостью для создания различных физических конфигураций по вводу и выводу данных. Система поддержки реального времени обеспечивает простой интерфейс организации и управления процессами на более высоком программном уровне.

Системный монитор-отладчик

Комплект отладочных средств мDEB 1200 для микроконтроллера мМС 1204 позволяет проектировать и отлаживать прикладное ПО 8-разрядных микроконтроллеров. Комплект состоит из двух компонентов: размещаемого в ПЗУ резидентного системного монитора-отладчика мMON 1200 и исполняемого в среде ОС 1800 дистанционного загрузчика мL 1200. Комплект обеспечивает подключение и связь одноплатного МК типа мМС 1204 с МВК СМ 1800 через стандартный телеграфный канал типа ИРПС. Существующая программная версия комплекта обеспечивает чтение шестнадцатеричных (HEX) файлов в среде ОС 1800 [5], их загрузку, отладку и исполнение на прикладной микросистеме (МС), построенной на базе МК и мМС 1204.

Монитор-отладчик мMON 1200 является резидентной системной программой диалогового типа, в функции которой входят организация взаимодействия с системой ввода-вывода, ввод с консоли и исполнение ряда команд по загрузке, редактированию, отладке и запуску объектного кода для МК типа мМС 1204. Монитор расположен в ПЗУ и занимает 2К байт памяти в диапазоне 0000H...07FFH адресов. Область ОЗУ 13СОН...13FFFH размером в 64 байта резервируется под данные монитора, структура которых показана ниже.

Таблица векторов прерываний

IL0:	JMP	0	; Пользователь
	DB	0	
IL1:	JMP	0	; Системное время
	DB	0	
IL2:	JMP	0	; Система отчета пауз
	DB	0	
IL3:	JMP	0	; Приемник ППА
	DB	0	
IL4:	JMP	0	; Приемник ПСА2
	DB	0	
IL5:	JMP	0	; Приемник ПСА1
	DB	0	
IL6:	JMP	0	; Пользователь
	DB	0	
IL7:	JMP	0	; Пользователь
	DB	0	

Таблица подпрограмм ввода-вывода логических устройств

CIRDY:	JMP	S11	; Ввод с консоли
CORDY:	JMP	S10	; Вывод на консоль
LPRDY:	JMP	PAO	; Вывод на печать
NIRDY:	JMP	S21	; Ввод из сети
NORDY:	JMP	S20	; Вывод в сеть

Вспомогательные данные

SPMON	DW	0	SPMON
BFPTR:	DW	0	; Указатель : BF:
RESP1:	DW	RESP1	; Указатель прерывания 1
RESB1:	DB	RESC	; Буфер прерывания 1
RESP2:	DW	RESP2	; Указатель прерывания 2
RESB2:	DB	RESC	; Буфер прерывания 2
FLAG:	DB	0	; Флаг COPY

Таблица подпрограмм косвенного ввода-вывода через порты

IPORT:	IN	0	; Процедура ввода
	RET		
OPORT:	OUT	0	; Процедура вывода
	RET		

Таблица 4

Команды монитора MMON1200

Команда	Содержание
.MXXXX, YYYY, ZZZZ	Перемещение области XXXX...YYYY в ZZZZ
.CXXXX, YYYY, ZZZZ	Сравнение области XXXX...YYYY с ZZZZ
.FXXXX, YYYY, AA	Заполнение области XXXX...YYYY значениями AA
.DXXXX, YYYY	Дамп области памяти XXXX...YYYY
.UXXXX, YYYY	Контрольная сумма области XXXX...YYYY
.NXXXX, YYYY	Сумма и разность чисел XXXX и YYYY
.SXXXX	Пошаговый просмотр и замена байтов начиная с адреса XXXX
.XX	Вывод содержимого регистров
.XR	Вывод и замена содержимого регистра R=A, F, B, C, D, E, H, L
.YY	Вывод содержимого флажков PSW
.YF	Вывод и замена содержимого флага F=M, Z, A, P, C
.OXX, AA [BB]... <CR>	Вывод в порт XX последовательности байтов AA, BB,...
.IXX [,]... <CR>	Ввод из порта XX последовательности байтов
.R [XXXX]	Загрузка абсолютного HEX-файла со смещением XXXX
.WXXXX, YYYY	Вывод области XXXX...YYYY в HEX-формате
.EXXXX	Вывод записи EOF XXXX (конец файла) в HEX-формате
.G [XXXX] [, YYYY [, ZZZZ]]	Передача управления по PC или адресу XXXX возможно с одной или двумя точками останова YYYY и ZZZZ
.P	Включение и выключение копирования на :LP:
* строка <CR>	Комментарий

Перед началом работы МК при включении питания или нажатии клавиши RESET должна быть выполнена инициализация его аппаратных ресурсов. Определена процедура стандартной инициализации, включающая установку в исходное состояние системы ввода-вывода и средств поддержки реального времени. Описание такого начального состояния для МК ММС 1204, которое было принято в качестве типового, дано в первой части данной работы. Проверимое блока данных отображает состояние системы после инициализации.

По окончании начальной установки управление передается диалоговой процедуре монитора, которая обеспечивает ввод и исполнение 19 консольных команд (табл. 4). Команды имеют стандартную мнемонику и функциональное назначение, принятое для систем аналогичного типа [2, 6]. Команда G предусматривает запуск программ пользователя с одной или двумя точками останова, при достижении которых управление автоматически возвращается в монитор.

В данной версии монитора-отладчика входом по точке останова служит адрес 0008H (вход по команде RST1), поэтому команду RST1 не рекомендуется использовать в прикладных системах. Выполнение команды RST0 приведет к перезапуску монитора. Выполнение команд типа RST N — к передаче управления по соответствующему вектору прерывания из блока данных.

Таблица 5

Связь с монитором CM 1800

Процедура	Адрес	Назначение
MON	40H	Вход в монитор
MONER	43H	Вход в монитор по ошибке
TTS	46H	Проверка статуса ввода
CRLF	49H	Перевод строки
TTCON	4CH	Вывод текста
TTCLF	4FH	Вывод текста с переводом строки
TTI	52H	Ввод символа
TTIO	55H	Ввод символа с эхом
TTO	58H	Вывод символа
DREG	5BH	Вывод содержимого регистров
DMEM	5EH	Вывод содержимого памяти
OUTHX	61H	Вывод байта в HEX-форме
OUTH2	64H	Вывод слова в HEX-форме
INHEX	67H	Ввод байта в HEX-форме
INHX2	6AH	Ввод слова в HEX-форме
LOAD	6DH	Загрузка программы с дискеты *
BDO	70H	Открытие дискеты *
BRDR	73H	Чтение сектора в буфер *
BWDR	76H	Запись сектора из буфера *
BDDR	79H	Запись признака стертых данных *
BDC	7CH	Закрытие дискеты *
BDD	7FH	Опускание головки *
BLDR	82H	Чтение сектора *
BLDR	85H	Запись сектора *
LINK	88H	Загрузка и запуск программы *
NIBBL	8BH	Проверка HEX-цифры
MOVE	8EH	Перемещение области памяти
COMP	91H	Сравнение областей памяти
FILL	94H	Заполнение памяти константой
LPO	97H	Вывод символа на печать

* Запрещенная процедура.

```

000000: C3 54 03 00 00 00 00 00 C3 62 03 00 00 00 00 00
000010: C3 C8 13 00 00 00 00 00 C3 CC 13 00 00 00 00 00
000020: C3 D0 13 00 00 00 00 00 C3 D4 13 00 00 00 00 00
000030: C3 D8 13 00 00 00 00 00 C3 DC 13 00 00 00 00 00
000040: C3 E0 03 C3 68 03 C3 ED 13 C3 C7 00 C3 B5 00 C3
000050: C4 00 C3 19 01 C3 22 01 C3 9F 00 C3 DC 05 C3 89
000060: C4 C3 EA 00 C3 F9 00 C3 49 01 C3 5E 01 C3 9A 00
000070: C3 9A 00 C3 9A 00 C3 9A 00 C3 9A 00 C3 9A 00 C3
000080: 9A 00 C3 9A 00 C3 9A 00 C3 9A 00 C3 2C 01 C3 21
000090: 04 C3 34 04 C3 76 04 C3 70 01 3E 02 87 37 C9 F5
0000A0: CD E3 13 CA 00 3A F9 13 E6 01 C2 80 00 F1 C9
0000B0: F1 CD 70 01 C9 F5 0A B7 CA C2 00 CD 9F 60 03 C3
0000C0: 34 00 F1 C9 CD 85 00 F5 3E 0D CD 9F 00 3C C2 CD
0000D0: 00 3E 0A CD 9F 00 3C C2 D6 00 F1 C9 F5 E6 OF C6
0000E0: 90 27 CE 40 27 CD 9F 00 F1 C9 F5 OF OF OF CD
0000F0: DC 00 F1 F5 CD DC 00 F1 C9 F5 78 CD EA 00 79 CD
000100: EA 00 F1 C9 C5 01 10 01 CD B5 00 CD EA 00 C1 C9
000110: 0D 0A 45 52 52 4F 52 20 00 CD E0 13 CA 19 01 E6
000120: 7F C9 CD 19 01 CD 9F 00 C9 CD 22 01 C5 4F D6 30
000130: DA 46 01 C6 E9 DA 46 01 C6 06 F2 42 01 C6 07 DA
000140: 46 01 C6 0A B7 4F 79 C1 C9 CD 29 01 D8 C5 07 07
000150: 07 07 E6 F0 4F CD 29 01 DA 5C 01 B1 C1 C9 CD 49
000160: 01 D8 C3 69 01 CD 4D 01 D8 47 CD 49 01 D8 4F C9
000170: E5 37 F5 F1 F5 21 00 01 CD E6 13 C3 87 01 23 AF
000180: 84 C2 73 01 F1 E1 C9 F1 E1 3F C9 F5 3A F9 13 F6
000190: 01 32 F9 13 F1 C9 F5 3A F9 13 E6 FE 32 F9 13 F1
0001A0: C9 CD 9F 00 D0 CD 96 01 F5 3E 01 CD 68 03 F1 C9
0001B0: CD 49 01 D0 C3 A1 03 CD 5E 01 D0 C3 A1 03 CD 87
0001C0: 01 C5 CD E4 01 CD 87 01 C5 E1 C1 C9 CD BE 01 CD
0001D0: E4 01 C5 CD 87 01 C5 B1 C1 C9 3E 20 C3 9F 00 3E
0001E0: 3D C3 9F 00 3E 2C C3 9F 00 CD 22 01 FE 7F C0 C3
0001F0: A9 03 03 78 B1 37 C8 7D 94 7C 98 C9 F1 C1 D1 E1
000200: C9 F5 AF D3 19 D3 19 D3 19 3E 40 D3 19 3E FA D3
000210: 19 3E 05 D3 19 CD 1A 02 F1 C9 D8 19 E6 02 C8 DB
000220: 18 C9 C5 47 D8 19 E6 01 78 C1 C8 D3 18 C9 F5 AF
000230: D3 1B D3 1B D3 1B 3E 40 D3 1B 3E FA D3 1B 3E 05
000240: D3 1B F1 C9 DB 1B E6 02 C8 DB 1A C9 C5 47 DB 1B
000250: E6 01 78 C1 C8 D3 1A C9 F5 3E A7 D3 17 3E 05 D3
000260: 17 3E 0D D3 17 F1 C9 DB 16 E6 01 C8 DB 15 E6 7F
000270: C9 C5 47 DB 16 E6 08 78 C1 C8 D3 14 C9 22 F1 13
000280: C9 E5 2A F1 13 3E FF B7 7E 23 22 F1 13 E1 C9 C5
000290: 47 3E FF B7 78 C1 C9 F5 3E B6 D3 1F 79 D3 1E 78
0002A0: D3 1E F1 C9 F5 3E 34 D3 1F 79 D3 1C 78 D3 1C F1
0002B0: C9 F5 79 D3 1D 78 D3 1D F1 C9 F5 3E 70 D3 1F F1
0002C0: C9 F3 F5 3E E0 A1 F6 16 D3 12 78 D3 13 3E FF D3
0002D0: 13 F1 F8 C9 F5 3E 20 D3 12 F1 C9 DB 13 C9 D3 13
0002E0: C9 E5 D5 C5 F5 01 C0 13 CD C1 02 01 68 00 CD 97
0002F0: 02 CD 01 02 CD 2E 02 CD 58 02 01 40 9C CD A4 02
000300: CD BA 02 01 14 03 21 54 03 11 C0 13 CD 21 04 F1
000310: C1 D1 E1 C9 C3 00 00 C3 00 00 00 C3 00 00 00 00
000320: C3 00 00 00 C3 00 00 00 C3 00 00 00 C3 00 00 00
000330: C3 00 00 00 C3 1A 02 C3 22 02 C3 71 02 C3 44 02
000340: C3 4C 02 00 00 00 00 00 F3 13 CF F6 13 CF 00 DB 00
000350: C9 D3 00 C9 31 C0 15 E5 21 00 00 E3 CD E1 02 C3
000360: 6B 03 E3 2B E3 C3 6B 03 CD 04 01 E5 F5 21 06 00
000370: 39 F1 E3 E5 D6 C5 F5 21 00 00 39 22 EF 13 2A F3
000380: 13 7E FE CF C2 98 03 3A F5 13 77 2A F6 13 7E FE
000390: CF C2 98 03 3A F6 13 77 01 DA 03 CD B5 00 C3 4B
0003A0: 03 CD EC 01 97 CD 04 01 2A E1 13 F9 01 E6 03 CD
0003B0: 35 00 CD 22 01 FE 0D CA A3 03 FE 2A CA 83 05 FE
0003C0: 5B F2 A1 03 D6 A1 FA A1 03 21 EA 03 5F 16 00 19
0003D0: 19 7E 23 66 6F 01 AC 03 C5 E9 0D 0A 2A 4D 4F 4E
0003E0: 31 32 30 30 2A 00 0D 0A 2E 00 A1 03 A1 03 31 04
0003F0: 86 04 E5 07 6D 04 A7 06 29 05 66 05 A1 03 A1 03
000400: A1 03 1E 04 A1 03 47 05 8E 05 A1 03 09 07 E6 04
000410: A1 03 C6 04 A1 03 78 07 9F 05 28 06 A1 03 CD CC
000420: 01 E5 D5 C5 F5 0A 12 13 CD F2 01 FE 25 04 C3 FC
000430: 01 CD CC 01 E5 D5 C5 F5 0A EB BE CA 5C 04 CD C7
000440: 00 CD F9 00 CD DA 01 0A CD EA 00 C5 4D 44 CD DA
000450: 01 CD F9 00 CD DA 01 0A CD EA 00 C1 EB 13 CD E0
000460: 13 C2 FC 01 CD F2 01 D2 38 04 C3 FC 01 CD BE 01
000470: CD E4 01 CD B0 01 E5 D5 C5 F5 5F 7B 02 CD F2 01
000480: D2 7B 04 C3 FC 01 CD BE 01 E5 D5 C5 F5 79 E6 0F
000490: 2F C6 11 57 CD C7 00 CD F9 00 CD DA 01 0A CD EA
0004A0: 00 CD F2 01 DA FC 01 15 C2 9A 04 CD E0 13 C2 B6
0004B0: 04 16 10 C3 94 04 FE 0D CA FC 01 CD 19 01 FE 0D
0004C0: CA FC 01 C3 B1 04 CD BE 01 CD DF 01 CD D3 04 CD
0004D0: EA 00 C9 E5 D5 C5 1E 00 0A 83 5F CD F2 01 D2 D8
0004E0: 04 7B C1 D1 E1 C9 CD B7 01 CD DA 01 0A CD EA 00
0004F0: 3E 2D CD 9F 00 CD 29 01 DA 0B 05 CD 4D 01 DA 23
000500: 05 02 CD DA 01 03 03 0B C3 EC 04 FE 20 CA 05 05
000510: FE 2E CA 07 05 FE 2F D2 23 05 CD C7 00 CD F9 00
000520: C3 E9 04 69 60 C3 89 04 C9 CD BE 01 CD DF 01 C5
000530: 79 85 4F 78 8C 47 CD F9 00 C1 CD E4 01 79 95 4F
000540: 78 9C 47 CD F9 00 CD B0 01 47 CD E4 01 79 95 4F
000550: 01 CD 5D 05 CD 22 01 FE 2C CA 4E 05 C9 F5 78 32
000560: FE 13 F1 C3 FD 13 CD B0 01 47 CD DF 01 CD 7C 05
000570: CD EA 00 CD 22 01 FE 2C CA 6D 05 C9 78 32 FB 13
000580: C3 FA 13 CD 22 01 FE 0D C2 83 05 C3 AC 03 3A F9
000590: 13 EE 01 32 F9 13 C9 46 41 43 42 45 44 4C 46 CD
0005A0: 22 01 21 97 05 06 08 0E 02 BE CA BC 05 23 0C 05
0005B0: E2 A9 05 FE 38 C2 A1 03 CD F2 05 C9 26 00 69 39
0005C0: CD BA 01 7E CD EA 00 3E 2D CD 9F 00 CD 29 01 DA
0005D0: A1 03 CD 4D 01 DA A1 03 77 C3 A9 03 E5 21 04 00
0005E0: 39 E3 E5 D5 C5 F5 F5 CD F2 05 F1 F1 C1 D1 E1 33
0005F0: C3 C9 21 04 00 39 CD E7 00 01 15 06 CD DA 01 CD
000600: 85 00 CD DF 01 03 C5 4E 23 46 23 CD F9 00 C1 0A
000610: 87 C2 FC 05 C9 41 46 00 42 43 00 44 45 00 48 4C
000620: 00 53 50 00 50 43 00 00 CD 22 01 21 97 06 06 05
000630: BE 23 23 CA 63 06 23 05 C2 30 06 FE 59 C2 A1 03
000640: 21 02 00 39 01 97 06 CD DA 01 CD B5 00 CD DF 01
000650: 03 0A A6 3E 30 CA 59 06 3C CD 9F 00 03 0A E7 C2
000660: 47 06 C9 CD DA 01 EB 21 02 00 39 1A A6 3E 30 CA
000670: 73 06 3C CD 9F 00 3E 2D CD 9F 00 CD 22 01 FE 30
000680: CA 8E 06 FE 31 CA 93 06 FE 0D C2 A1 03 C9 1A 2F
000690: A6 77 C9 1A B6 77 C9 4D 00 8D 5A 00 40 41 00 10
0006A0: 50 00 04 43 00 01 00 CD 29 01 DA BB 06 CD 45 01
0006B0: DA A1 03 21 0C 00 39 71 23 70 CD 22 01 FE 2C CA
0006C0: CE 06 FE 0D C2 A1 03 01 F5 13 C5 C3 DF 06 CD 87
0006D0: 01 C5 CD 22 01 FE 2C CA E5 06 FE 0D C2 A1 03 01
0006E0: F3 13 E3 E9 06 CD B7 01 0A 32 F8 13 CE CF 02 6F
0006F0: 60 22 F6 13 C1 0A 32 F5 13 3E CF 02 69 60 22 F3
000700: 13 F1 F1 C1 D1 E1 33 C3 C9 01 00 00 CD 29 01 DA
000710: 1A 07 CD 65 01 DA A1 03 3E 0D FE 0D C2 A1 03 CD
000720: C7 00 CD 22 01 FE 3A C2 22 07 16 00 CD 61 07 5F
000730: CD A1 07 67 CD 61 07 6F 09 CD 61 07 C5 47 4B E5
000740: 1C 1D CA 51 07 CD 61 07 77 BE C2 71 07 23 C3 41
000750: 07 CD 61 07 AF B2 C2 76 07 E1 AF B1 C1 C2 22 07
000760: C9 CD 49 01 DA 6C 07 F5 82 57 F1 C9 3E F1 C3 A5
000770: 03 CE 2F C3 A5 03 3E F3 C3 A5 03 CD BE 01 CD C7
000780: 00 3E 3A CD 9F 00 11 00 10 C5 1C 15 CA 95 07 CD
000790: F2 01 D2 BA 07 16 00 C1 7B CD C8 07 78 CD C8 07
0007A0: 79 CD 07 AF CD C8 07 7A CD C8 07 03 1D C2 A9
0007B0: 07 AF 92 CD C8 07 3E 0D CD 9F 00 3E 0A CD 9F 00
0007C0: 0B CD F2 01 D2 81 07 C9 CD EA 00 82 57 C9 CD E7
0007D0: 01 CD C7 00 3E 3A CD 9F 00 1A 00 AF CD C8 07 7B
0007E0: CD C8 07 79 CD C8 07 3E 01 CD C8 07 C5 E1 03 C3
0007F0: 81 07 FF FF FF FF FF FF FF FF FF FF FF FF FF

```

Рис. 2. Дамп памяти монитора

Совместимость монитора мMON 1200 с ОС 1800 обеспечивается полным набором процедур ввода-вывода, соответствующих монитору-отладчику SM 1800 MONID [6], за исключением операций с НГМД. При этом сохранен не только протокол передачи параметров, но и точки входа в процедуры, которые организованы в виде таблицы ветвлений с базой 0040H (табл. 5). При попытке выполнения запрещенных операций с НГМД происходит возврат из соответствующей процедуры с кодом ошибки (A)=2 — устройство не готово.

При вызовах монитора-отладчика (вход по точкам MON и MONER) содержимое регистров микропроцессора, отображающее текущее состояние задачи пользователя, сохраняется в стеке системы. Затем стек используется для собственных нужд монитора (команда G), содержимое регистров пользователя автоматически восстанавливается. При инициализации системы дном растущего вниз стека служит начало блока данных монитора. Полный дамп памяти монитора мMON 1200 в HEX-формате приведен на рис. 2.

Расширение команд, поддерживаемое M1200

Команда	Содержание
.L, D, N, NAME [, XXXX]	Загрузка с N-го устройства CM 1800 HEX-файла NAME со смещением XXXX
.LP [, XXXX]	Загрузка с фотосчитывателя CM 1800 HEX-файла со смещением XXXX

CM 1800 дает возможность перенести ряд отработанных для него программных модулей на проектируемые микросистемы прикладного типа.

Телефон для справок: 408-62-44, г. Москва, МФТИ.

ЛИТЕРАТУРА

- Щелкунов Н. Н., Дианов А. П. Универсальный одноплатный микроконтроллер // Микропроцессорные средства и системы.— 1986.— № 5.— С. 65.
- Алексенко А. Г., Галицын А. А., Иванников А. Д. Проектирование радиоэлектронной аппаратуры на микропроцессорах // Программирование, типовые решения, методы отладки.— М.: Радио и связь, 1984.
- Щелкунов Н. Н., Дианов А. П. Программирование микросистем реального времени // Микропроцессорные средства и системы.— 1985.— № 4.— С. 31—35.
- Торгов Ю. И. Программируемый таймер КР580ВИ53 и его применение // Микропроцессорные средства и системы.— 1984.— № 1.— С. 77—84.
- Прохоров Н. Л., Ландау Н. Я. МикроЭВМ CM 1800 и ее программное обеспечение // Микропроцессорные средства и системы.— 1984.— № 2.— С. 28—30.
- Система программного обеспечения CM 1800. Монитор. Руководство программиста 4.072 127 ИЭ.

Статья поступила 18 июля 1986 г.

Откровения в ожидании реакции операционной системы на неправильные команды МОДУЛЯЦИИ

Еще раз проверьте, сколько часов отводят в школе алгебре и сколько — гармонии!

На вечные вопросы ЭВМ давала моментальные ответы.

Метод проб и ошибок: одни пробуют, другие исправляют ошибки.

Важно, видимо, не то, кто раньше сказал «э», а кто первый сказал: ЭВМ.

Информация к размышлению: чем больше поглощаешь информации, тем меньше остается времени на размышления.

Самые осторожные программисты — на телевидении. Кем только ни числятся нули!

Заметное преимущество ЭВМ в том, что, закончив подсчет, она не прослезится.

Нет правил для исключений.

В. Воронцов (г. Тольятти)

Разработка и отладка ПО. Массовое использование микроконтроллеров в промышленности и научных лабораториях требует развитых средств проектирования и отладки их программного обеспечения. Важно создание гибких универсальных средств и комплексов, удовлетворяющих современным требованиям простоты их использования и возможности дальнейшей модификации. В работе [1] было сказано, что для МК типа мМС 1204 в качестве центральной части такой инструментальной системы может рассматриваться серийно выпускаемый в стране МВК CM 1800. Написание программ, их трансляция и документирование для МК мМС 1204 удобно выполнять с помощью стандартных средств разработки программ, включенных в состав этого МВК. Возникает задача переноса готового объектного кода с дисков CM 1800 в память МК с целью их пробного запуска и отладки на реальной прикладной микросистеме.

Обычно для решения этой задачи в состав команд монитора включают набор директив вывода и загрузки с консоли абсолютных файлов в HEX-формате (команды W, E, R). Затем прикладной МК через консольный канал подключается к системе проектирования, которая своими средствами обеспечивает копирование выбранных файлов с дисков в канал связи в HEX-формате. Предлагается программа дистанционной загрузки мЛ 1200, обеспечивающая автоматическую загрузку HEX-модулей с дисков CM 1800 непосредственно в память МК мМС 1204. Отладка загруженной программы выполняется средствами резидентного монитора мMON 1200.

На рис. 3 приведена схема подключения МК мМС 1204 к МВК CM 1800, для чего используется консольный канал микроконтроллера и отдельный дополнительный канал CM 1800 типа ИРПС. Для активации системы вызывается программа мЛ 1200, которая иницирует канал связи и перезапускает микроконтроллер мМС 1204. После вывода заголовка программы мЛ 1200 и действительной версии мMON 1200

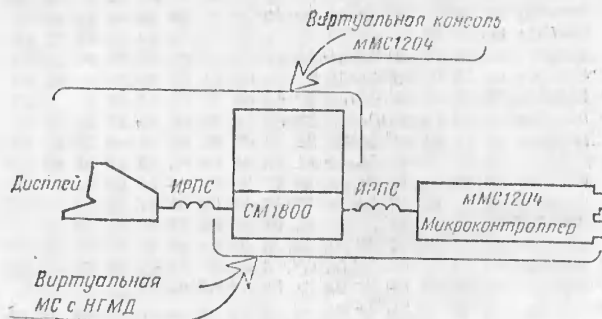


Рис. 3. Конфигурация системы отладки

консоль CM 1800 выполняет роль консоли мМС 1202. Реализуется обычный диалог с монитором мMON 1200, при котором может быть введена новая команда L (загрузка).

Программа мЛ 1200 перехватывает команду L, имеющую формат LD, N, NAME [,XXXX] или LP [,XXXX], (табл. 6), и читает HEX-файл NAME с дискового устройства номер N или фотосчитывателя CM 1800. С помощью автоматически генерируемой команды R [XXXX] считанный файл загружается в МК мМС 1204.

В целом, аппаратно-программное изделие, состоящее из трех компонент: одноплатный модуль мМС 1204, резидентный монитор-отладчик мMON 1200 и программа дистанционной загрузки мЛ 1200, представляет собой мощный комплект средств для проектирования промышленных микроконтроллеров и микросистем различного назначения. Программная совместимость с МВК

С. М. Большинский, А. Н. Полтава

ДРАЙВЕР НГМД УДВОЕННОЙ ПЛОТНОСТИ ДЛЯ ОС РАФОС

Предлагаемый драйвер используется для разработки программного обеспечения под управлением ОС Рафос на инструментальной микро-ЭВМ и позволяет вдвое увеличить емкость носителя, сохраняя неизменным время доступа к единице информации.

Применение диска с удвоенной плотностью записи наиболее эффективно для любого вычислительного комплекса типа ДВК, где в качестве системного устройства используется накопитель «Электроника 6022» и имеется контроллер НГМД для выхода на интерфейс связи с накопителем «Электроника НГМД 7012». Подготовка накопителя и носителя к работе в режиме удвоенной плотности приведена в [1]. Драйвер DS.SYS должен располагаться на системной диске, заноситься в таблицы монитора и загружаться в ОЗУ ЭВМ с помощью стартового командного файла, расширенного командами: INSTALL DS и LOAD DS. Для подключения дискового по другим адресам достаточно в тексте драйвера отредактировать строки, содержащие определения DXVEC и DXCSR или в файле DS.SYS изменить на соответствующие значения адреса вектора прерываний и регистра команд. Управление диском осуществляется контроллером, представленным регистрами команд и данных. Формат команды приведен на рис. 1.

Взаимодействие ОС Рафос с накопителем осуществляется посредством драйвера DS.SYS (рис. 2), который написан на языке MACRO и выполнен в соответствии со структурой и правилами оформления драйверов для внешних устройств этой операционной системы [2]. Обмен происходит по прерыванию с использованием Ф-обработки. После входа в драйвер из программы управления очередями определяются номер блока, с которого на устройстве начинается операция обмена; код специальной функции, которую необходимо выполнить на устройстве; номер устройства, подключенного к контроллеру, к которому относится операция обмена; количество слов, участвующих в операции обмена; адрес буфера обмена данными в задаче. Далее инициируется операция, разрешается прерывание от устройства и осуществляется возврат в основную программу, потребовавшую обмена.

Вход в программу обработки прерываний содержит две точки: непо-

средственно обработка и аннулирование операции ввода-вывода, выполняющиеся при нормальном или ненор-

мальном завершении программы, имеющей установленные в очередь к драйверу запросы ввода-вывода [3]. Непосредственному выполнению операции на устройстве предшествует алгоритм вычисления физических адресов дорожек и секторов, который определяет способ размещения информации, а следовательно, и время доступа. В процессе выполнения ал-

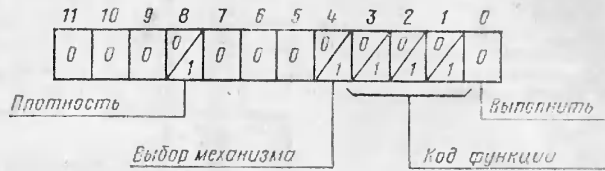


Рис. 1. Формат команды контроллера

```

-----ЗАГОЛОВОК-----
АССЕМБЛЕР MACRO-11/РАФОС
ДРАЙВЕР ДИСКА УДВОЕННОЙ ПЛОТНОСТИ
ЗАНЕСЕНИЕ ДРАЙВЕРА В ТАБЛИЦЫ МОНИТОРА:
INSTALL DS
ЗАГРУЗКА ДРАЙВЕРА В ОЗУ ЭВМ:
LOAD DS
-----ОПРЕДЕЛЕНИЯ-----
ИСПОЛЬЗУЕМЫЕ СИСТЕМНЫЕ МАКРОКОМАНДЫ
.MCALL .DRVEB, .FORK, .DREND, .DRAST, .DRFIN
СТАНДАРТНЫЕ ПАРАМЕТРЫ ГЕНЕРАЦИИ
.IIF NDF TIM*IT, TIM*IT=0 ;НЕТ ТАЙМ-АУТА
.IIF NDF MM*AT, MM*AT=0 ;НЕТ ДИСПЕТЧЕРА ПАМЯТИ
.IIF NDF ERL*E, ERL*E=0 ;НЕТ ПРОТОКОЛА ОШИБОК
.IIF NDF DX*VEC, DX*VEC == 274 ;ВЕКТОР ПРЕРЫВАНИЯ ДИСКА
.IIF NDF DX*CSR, DX*CSR == 177200 ;АДРЕС РЕГИСТРА КОМАНД ДИСКА
DXDSIZ= 995. ;РАЗМЕР УСТРОЙСТВА В БЛОКАХ
DXSTS= 102022 ;СТАТУС УСТРОЙСТВА
СТАНДАРТНЫЕ СМЕЩЕНИЯ ОТНОСИТЕЛЬНО ТРЕТЬЕГО
ЭЛЕМЕНТА ОЧЕРЕДИ
Q.BLK=0 ;НОМЕР БЛОКА НА УСТРОЙСТВЕ
Q.BUF=4 ;АДРЕС БУФЕРА ОБМЕНА
ФУНКЦИИ КОНТРОЛЛЕРА
CSGO=1 ;ВЫПОЛНИТЬ
CSFBUF= 0*2 ;ЗАПИСЬ В БУФЕР
CSBBUF= 1*2 ;ЧТЕНИЕ БУФЕРА
CSWRT= 2*2 ;ЗАПИСЬ СЕКТОРА
CSRDR= 3*2 ;ЧТЕНИЕ СЕКТОРА
CSRDS= 5*2 ;УСТАНОВКА ПЛОТНОСТИ ЗАПИСИ
CSWRTD= 6*2 ;ЧТЕНИЕ РЕГИСТРА ОШИБОК И СОСТ.
CSMAIN= 7*2 ;ЗАПИСЬ СЕКТОРА С МЕТКОЙ
CSRDIR= CSRDR+CSWRT ;ЧТЕНИЕ-ЗАПИСЬ СЕКТОРА
CSUNIT= 20 ;УСТРОЙСТВО 1
CSDONE= 40 ;УСТРОЙСТВО 2
CSINT= 100 ;РАЗРЕШИТЬ ПРЕРЫВАНИЕ
CSTR= 200 ;ГОТОВНОСТЬ
CSINIT= 40000 ;ГРУБАЯ ОШИБКА
CSERR= 100000 ;ПРИЗНАК ОШИБКИ
DADD= 100.
DBIN= 4
RETRY=8. ;КОЛИЧЕСТВО ПОВТОРЕНИЙ
SPFUNC= 100000 ;СПЕЦ. ФУНКЦИЯ
.DRVEB DX, DX*VEC, DXDSIZ, DXSTS
; ТОЧКА ВХОДА В ДРАЙВЕР ИЗ ПРОГРАММЫ УПРАВЛЕНИЯ
ОЧЕРЕДЯМИ
RXTRY: MOU #RETRY, (PC)+
.MOU .WORD 0
MOU DXCDE, R3
MOU (R3)+, R5
MOU #CSGO+CSRDR, R4
MOUB (R3)+, R1
BITB #6, CR3
BNE RMERRJ
BITB #1, (R3)+
BEQ 1*
BIS #CSUNIT, R4
1*: MOU (R3)+, R0
MOU CR3, R2

```

Рис. 2. Программа-драйвер (Продолжение см. на стр. 30)

```

BAL 2X
CMPB -(R1),R4
NEG R2
ADD PC,R1
MOVB R4,CHGTBL-(R1),R3
FOR R3
ADD R3,R4
BNT 3X
RSL R3
RSL R2
BR 4X
CLR (R0)+
MOU R2,PHYTRK
MOU #256,R2
MOU R0,BUFADR
MOU R5,RKLSN
MOU R4,RKFUN2
MOU R2,BYTCNT
RHEINT: MOU R3,INTINT
RHEINT: BIS #CSINT,ORHCSA
PC
RHEINT: JMP RHEINT
JИЗНА В ПРОГРАММЕ ОБРАБОТКИ ПРЕФИКСА
DRAST DA,5,RHEINT
RCSA: .WORD DMCSR
MOU R4,RS
JИЗМЕНА К ОБРАБОТКЕ НА Ф-УРОВНЕ
FORK DFBLK
TST (RS)+
BMT RHEINT
NEG (PC)+
INTINT: .WORD 0
MOU #256,R3
BIT #CSEBUF,RKFUN2
BNE 2X
RCD 4X
SER PC,NXTSEC
SER R0,SILOFE
MOVB (R2)+,RS
WORD C56+C6BUF
CURB RS
BCC 3X
TST RKFUN2
BPL 4X
BTT #BDD,RS
BEG MOU BUFADR,R2
JAC -(R2)
4X: JSR R0,SILOFE
MOVB (R2)+,C56+C6BUF
MOU RS,R2
JSR PC,NXTSEC
JИЗМЕНА ФИЗИЧЕСКИХ АДРЕСОВ СЕКТОРА И ДОРОЖКИ
3X: MOU (PC)+,R3

```

```

RALS: .WORD 0
PHYTRK: .WORD 0
TST RKFUN2
MOU #9,R2
CMP #15000,R3
BHI 2X
ADD #16300,R3
DEC R2
BGT 1X
MOVB R3,R2
CLR R3
SWAB R3
CMP #12,R3
RDL R3
ASL R2
ACD R2,R3
ADD R2,R3
ADD R2,R3
ASR R2
INC R2
SUB #26,R3
RDE BCC 3X
DOFUND: MOU 0
RKFUN2: .WORD 0
50: BPL 024
MOU #1,RS
TSTB 024
1X: TSTB 024
BPL RHEINT
MOU R3,RS
TSTB 024
BEG 2X
BPL RHEINT
MOU R2,RS
BR RNWAIT
DEC RHEINT
BEE MOU #CSINIT,OR4
JMP RHEINT
DNCOD,R4
BIS 11,0-(R4)
RALLD: CLR ORHCSA
RHEINT: MOU DMCSR
RHEINT: CLR DMCSR
JИЗМЕНА К ПРОГРАММЕ УПРАВЛЕНИЯ ОБРАБОТКИ
JИЗМЕНА ОБРАБОТКИ
RHEINT: JDFIN DA
NXTSEC: ADD R3,BUFADR
INC RALS

```

```

BYTCNT
RETURN
CLR
BIT #CSEBUF+SPFUN2
BNE #1,RALS
BIT #NE
TST
BPL
RRTS
SILOFE: MOU (R0)+,EFBUF
MOU (R0)+,OR4
51X: TSTB 024
BPL 51X
MOU #1,RS
MOU (PC)+,CSP
BYTCNT: .WORD 0
BEG ZFILL
CMP 024,R3
BLOS MOU R3,024
MOU BUFADR,WORD
TRBYT: TSTB 024
TRBYT: BPL
DEC 024
BGT TRBYT
MOU 024,EFBUF
14: TSTB 024
BEG 1X
BHI EFBUF
BIT (R0)+,CSP+
RIS
BYTE 022
BYTE -2+2
ORHCSA: .BYTE 022
EVEN
DFBLK: .WORD 0,0,0,0
END 2X

```

3. Программное обеспечение СМ ЭВМ. Операционная система с разделением функций РАФОС.—М.: ИНЭУМ, 1980, т. 4, кн. 5. С. 42.

Статья поступила 10 июня 1986 г.

Фрагмент из рецензии:

...Учитывая большой спрос пользователей на программы такого рода, целесообразно ввести в журнале рубрику «Системные программы». Например, системные программы-драйверы: НМД, печатающих устройств, терминального и других устройств. Здесь же следует поместить сообщения о правилах написания драйверов ОС РАФОС, ОС ДВК и других подобных операционных систем.

горятма операции обмена подтверждается плотность записи информации. Программа аннулирования имеет метку RXABRT.

Выход из драйвера осуществляется по исчерпанию счетчика слов элемента очереди монитора.

Телефон для справок: 55-71-31, 340114, СКТБ Донецкого физ.-тех. института.

ЛИТЕРАТУРА

1. Балашов Е. П., Григорьев В. Л., Петров Г. А. Микро- и мини-ЭВМ.—М.: Энергоатомиздат, 1984. С. 239—246.
2. Валикова Л. И., Вигдорчик Г. В., Воробьев А. Ю., Лукин А. А. Операционная система СМ ЭВМ РАФОС.—М.: Финансы и статистика, 1984. С. 207.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ СИСТЕМ РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ ДАННЫХ

УДК 681.3.06

С. В. Волков, М. В. Дудкин, А. И. Казьмин, А. А. Менн, А. В. Шерстюк, М. Н. Кузнецов, А. Н. Целяпин, В. В. Ушкевич

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ КОММУНИКАЦИОННЫХ ПРОЦЕССОРОВ В РАСПРЕДЕЛЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ КОМПЛЕКСАХ

Важнейшими мероприятиями последних лет по коренному изменению организационных и технологических основ производства является создание и внедрение гибких производственных систем (ГПС) на основе автоматизированного оборудования с интеллектуальными системами управления, станков с ЧПУ, промышленной робототехники, автоматических складов, транспортных систем и систем управления на базе микро- и мини-ЭВМ [1, 2].

Многие современные системы управления технологическими процессами, в том числе и ориентированные на гибкие производственные системы (ГПС), основываются на распределенных вычислительных комплексах (РВК). Среди разрабатываемых структур РВК [3] широкое распространение получили схемы с коммуникационными процессорами (рис. 1):

- иерархические РВК (рис. 1, а);
- системы связи нескольких участков

кольцевых и магистральных локальных сетей ЭВМ (рис. 1, б); коммуникационные процессоры больших ЭВМ, включенных в локальную сеть (рис. 1, в); системы автоматизации программирования и отладки программ распределенных систем управления.

В первых трех из перечисленных областей коммуникационные процессоры реализуют протоколы транспортной службы [4], управляющие передачей данных между программами пользователей, которые выполняются в различных ЭВМ РВК, а также между программами и терминалами комплекса. Некоторые области приложений выдвигают специфические требования к транспортной службе, например при передаче данных между программами, управляющими технологическим оборудованием в реальном масштабе времени. В этом случае транспортная служба должна вносить минимально возможные задержки.

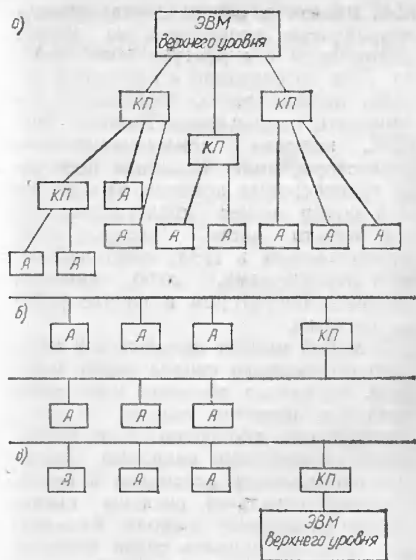


Рис. 1. Типы структур распределенных вычислительных комплексов: иерархическая (а); кольцевая и магистральная (б); большая ЭВМ в составе локальной сети (в), КП — коммуникационные процессоры, А — абоненты

Кроме того, в ряде ситуаций она должна отвечать за согласование форматов пакетов для различных операционных систем:

однозадачные мультипроцессорные ОС реального времени, которые управляют технологическим оборудованием;

многозадачные ОС, включающие развитые файловые системы и банки данных. Такие ОС необходимы для ЭВМ, решающих задачи оперативно-диспетчерского управления, календарного планирования производства.

Помимо транспортных функций коммуникационные процессоры могут выполнять роль автоматизированных рабочих мест, позволяющих с единого пульта проводить комплексную отладку программ сразу на нескольких ЭВМ. В данной работе рассматриваются два пакета программ, реализующих транспортную службу и систему отладки РВК. Эти пакеты составили программное обеспечение коммуникационного процессора АЦКС-8/16-017 (рис. 2).

Все функции процессора АЦКС реализуются под управлением микро-

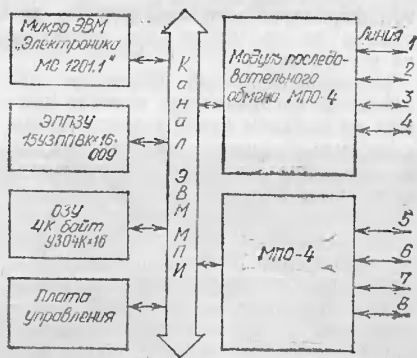


Рис. 2. Структурная схема процессора АЦКС-8/16-017

ЭВМ «Электроника МС 1201.1», программа которой находится в электрически перепрограммируемом запоминающем устройстве 15УЗПП8КХ16-009. Устройство оперативное запоминающее с батарейным питанием УЗО-4К16 емкостью 4К 16-разрядных слов служит для хранения данных, необходимых для работы микроЭВМ при отключении сетевого напряжения питания. Плата управления предназначена для согласования канала и задания режимов работы микроЭВМ.

Подключение абонентов к процессору АЦКС осуществляется с помощью модулей последовательного обмена МПО-4. Все устройства процессора АЦКС соединяются друг с другом с помощью канала ЭВМ, соответствующего интерфейсу МПИ, и выполнены конструктивно в виде модулей (кроме платы управления), аналогичных одинарным и двойным модулям микроЭВМ «Электроника 60». Питание всех модулей осуществляется от блока питания, содержащего стабилизаторы 5В, 12В и аккумуляторную батарею напряжением 4,5 В.

Модуль МПО-4 состоит из четырех независимых каналов последовательного обмена в виде токовой петли 20 мА, имеющих следующие параметры:

Режим передачи дуплексный
Разрядность передаваемого слова, бит 7,8
Контроль достоверности по четности нечетности
Скорость обмена, бит/с 300...9600
Длина линии связи из витой пары сечением 0,5 мм ² про- вода МГШВ, м до 1000

Каждый канал содержит четыре регистра: регистры состояния приемника и передатчика, регистр данных и ошибок приемника и регистр данных передатчика. Форматы регистров приведены в таблице.

Разряды Г устанавливаются в 1 в том случае, когда передатчик передает в линию символ, а приемник принимает из линии символ. Эти регистры доступны ЭВМ по чтению. Разряды РП служат для разрешения прерывания ЭВМ при появлении готовности передатчика или приемника, доступны ЭВМ по записи и чтению. Разряд ОП устанавливается в 1 при переполнении буфера приемника, ОС — при ошибке стопового элемента телеграфной посылки, а ОЧ при ошибке контроля четности в принятой посылке. Разряд О устанавливается в 1 при наличии любой из трех вышеуказанных ошибок. Все разряды ошибок доступны ЭВМ только по чтению. Разряд КП служит для контроля линии связи, который возможен как без выхода, так и с выходом на линию связи. Разряд РЛ служит для разрыва линии связи, что дает возможность ЭВМ, управляющей приемником, сообщить ЭВМ, управляющей передатчиком на другом конце линии связи, о своей неготовности принять сообщение. Буквы X в адресной части канала определяются вариантом модуля. Модулям могут присваиваться адреса в диапазоне 176500_в...176670_в, имеется возможность присвоения одному из каналов модуля адреса пульта терминала 177560_в.

Одной из проблем, стоящей перед ЭВМ, управляющей процессором АЦКС, является синхронизация своей работы с работой связанных с ней абонентов. Для решения этой проблемы в каждом канале модуля МПО-4 имеется обратная связь. В момент, когда приемник принял байт информации, ключ, связанный с сигналом готовности приемника, размыкает линию связи и по ней перестает протекать ток, соответствующий стоповым импульсам телеграфной посылки. При этом на передающей стороне приемник обратной связи не разрешает поступление сигнала готовности передатчика в ЭВМ. Когда на приемной стороне управляющая ЭВМ прочитает информацию с приемника, сигнал его готовности пролакает и ключ обратной связи замыкает линию, через которую начнет протекать

Регистр	Адрес	Разряд слова ЭВМ																
		15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00	
Регистр состояния приемника	17XXX0	—	—	—	—	—	—	—	—	—	Г	РП	—	—	—	—	—	
Регистр данных и ошибок приемника	17XXX2	О	ОП	ОС	ОЧ	—	—	—	—	—	В ₇	В ₆	В ₅	В ₄	В ₃	В ₂	В ₁	В ₀
Регистр состояния передатчика	17XXX2	—	—	—	—	—	—	—	—	—	Г	РП	—	—	—	КЛ	—	РЛ
Регистр данных передатчика	17XXX6	—	—	—	—	—	—	—	—	—	В ₇	В ₆	В ₅	В ₄	В ₃	В ₂	В ₁	В ₀

ток, соответствующий стоповым импульсам, и приемник обратной связи разрешает прохождение сигнала готовности передатчика в ЭВМ. Такая реализация обратной связи позволяет при незначительном снижении максимальной скорости обмена с большим количеством одновременно работающих с процессором АЦСК абонентов не вводить дополнительных витых пар, при передаче сигналов обратной связи. Если к процессору АЦСК подключен абонент, не имеющий аппаратуры для осуществления обратной связи, в модуле МПО-4 ее можно отключить.

Установка всех переменных режимов работы модуля МПО-4, кроме групповых адреса и вектора прерывания, осуществляется независимо по каждому каналу с помощью переключек. Для повышения помехоустойчивости линии связи модуль МПО-4 имеет вторичный источник питания для каждого канала. Программное обеспечение допускает подключение к нему оптоволоконных линий связи и адаптеров сетевых контроллеров. Коммуникационный процессор может иметь собственный системный терминал.

Для работы с данным коммуникационным процессором проведена доработка операционной системы RSX-11M [5]. Таким образом, с ним могут взаимодействовать любые ЭВМ, работающие под управлением этой ОС, например СМ-4, «Электроника 100/25», СМ 1420, СМ 1600. Для ЭВМ, управляющих технологическим оборудованием, разработана специализированная ОС реального времени, которая может быть установлена на машинах с системами команд микро-ЭВМ «Электроника 60» и И8086. В состав этой ОС включены средства, обеспечивающие обмен данными с любой ЭВМ РВК через коммуникационный процессор.

Общие принципы организации транспортной службы

Уменьшение задержек при прохождении данных через коммуникационный процессор возможно только при ограничении числа выполняемых им функций. Функции, требующие большого процессорного времени, например определение одного из нескольких допустимых маршрутов передачи данных или контроль правильности передачи, приходится перекладывать на конечных абонентов РВК. Конечные абоненты могут решать возложенные на них транспортные задачи специализированно, т. е. подготавливая решение под требования конкретной передачи данных, что в некоторых случаях значительно сокращает расходы на передачу. В рассматриваемой системе имеется несколько стандартных вариантов форматов и сообщений. Однако протоколы передачи сообщений не фиксируются. Пользователям представляется средства введения новых вариан-

тов форматов, если стандартные варианты их по каким-либо причинам не устраивают.

Коммуникационный процессор отвечает за распределение потоков данных по каналам связи в соответствии с заранее заданными маршрутами, за согласование скоростей работы смежных узлов и форматирование передаваемой информации в зависимости от типа канала связи. Тип канала связи определяют передающая среда, например оптоволоконная линия или «токовая петля», а также тип смежного узла. К любому из каналов связи коммуникационного процессора могут подключаться следующие устройства:

ЭВМ, управляющие технологическим оборудованием; коммуникационные процессоры (включение в РВК нескольких коммуникационных процессоров позволяет увеличить общее число абонентов РВК);

ЭВМ, работающие под управлением многозадачных ОС.

Последние по сравнению с остальными машинами РВК обладают существенно большим составом периферийных устройств, поэтому интенсивность передачи данных на каналах, связывающих коммуникационный процессор с многозадачными ЭВМ, значительно выше. В качестве передающей среды на таких каналах связи выгодно использовать оптоволоконные линии.

Терминалы РВК

Передача данных между двумя произвольными абонентами РВК, которые не связаны физической линией, осуществляется за счет пересылки информации через одну или несколько коммуникационных ЭВМ. Таким образом между двумя произвольными абонентами устанавливается логический канал связи. Он характеризуется парой транспортных номеров абонентов — номером отправителя информации и номером получателя. Одновременно в РВК может происходить пересылка информации по нескольким логическим каналам. При этом часто возникает ситуация, когда одна и та же физическая линия связи используется попеременно то одним, то другим логическим каналом. Коммуникационные ЭВМ динамически разделяют время физической линии между несколькими логическими каналами. Если передача идет сразу по нескольким каналам, разделяющим один физический, то скорость в каждом логическом канале пропорционально падает. Потери скорости передачи можно уменьшить путем подключения дополнительных физических связей машин. Это позволяет разработчику каждого конкретного РВК найти компромисс между быстротой и стоимостью системы.

По логическому каналу может передаваться произвольный поток дан-

ных, причем в общем случае коммуникационные процессоры не имеют информации о структуре этого потока. Для организации логического канала, прежде всего, необходимо задание его направления. Каждый узел РВК, включая коммуникационные процессоры, имеет один или несколько транспортных номеров. Транспортный номер может представлять собой просто число, которое либо «прожигается» в ПЗУ, либо набирается переключками, либо, наконец, вводится оператором в начале работы системы.

В любой момент времени для каждого физического канала связи известно, от какого абонента идет передача, т. е. известны номера, идентифицирующие абонентов. Для управления физическими каналами (задания направления, остановки и возобновления передачи) введены специальные служебные символы. Аппаратные средства каналов связи позволяют использовать только байт-ориентированные протоколы при передаче отдельных кадров информации [6]. В любом таком протоколе возникает проблема различения управляющих (служебных) символов и передаваемых данных, среди которых могут появляться коды, совпадающие с управляющими символами. В рассматриваемой системе эта проблема решена следующим образом. Как только среди информационных байтов встречается байт, совпадающий с кодом одного из служебных символов, любой передающий абонент заменяет его парой байтов: служебным символом «подстановки» и кошей заменяемого байта. В процессе приема конечным абонентом эта пара «свертывается» в один символ. Для терминалов РВК соответствующую замену байтов данных осуществляет коммуникационный процессор, непосредственно связанный с терминалом.

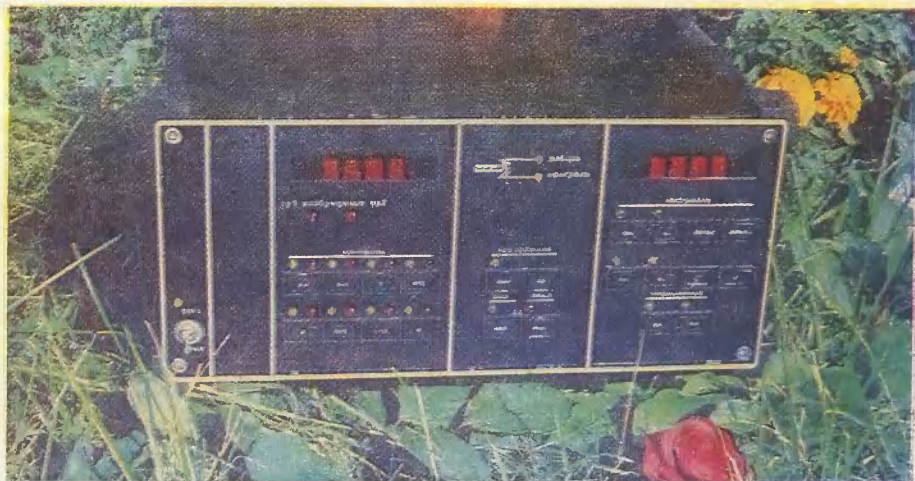
Потери времени на замену байтов будут тем меньше, чем реже встречаются в потоке информации байты, совпадающие со служебными символами. Потери можно уменьшить за счет выбора кодов служебных символов с учетом специфики передаваемой информации.

Сообщения для коммуникационной ЭВМ. Данный вид сообщений управляет работой коммуникационных машин. С помощью этих сообщений можно включать (отключать) физические каналы, устанавливать соответствие между физическими и логическими каналами, запрашивать информацию о состоянии каналов, задавать тип узлов, имеющих непосредственную линию связи с коммуникационной ЭВМ.

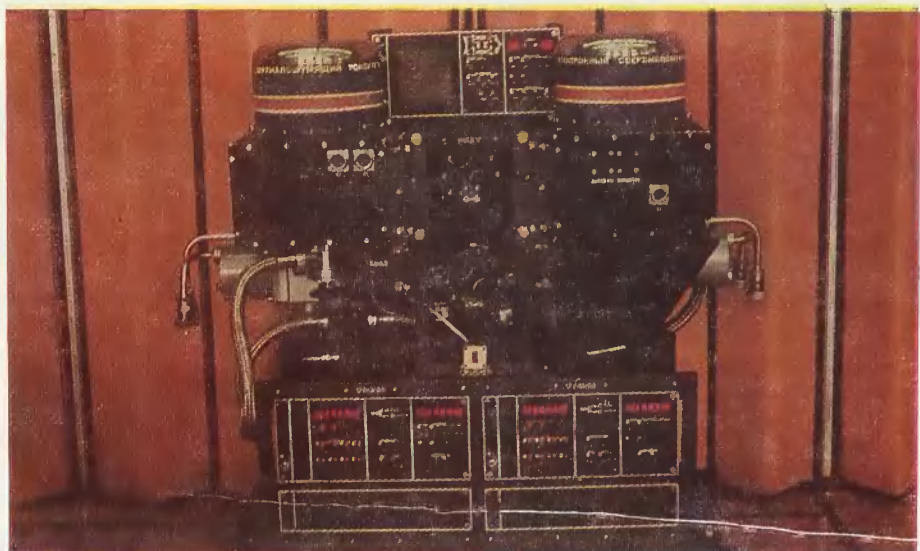
Отключение физического канала приводит к прекращению анализа его состояния коммуникационным процессором. Задание соответствия между физическим и логическим каналами связи устанавливает отображение логических каналов на физи-

МИКРОПРОЦЕССОРНАЯ СИСТЕМА КОНТРОЛЯ И УПРАВЛЕНИЯ ПРИЕМНЫМИ МАЛОШУМЯЩИМИ УСТРОЙСТВАМИ СВЯЗИ

(К ст. Чмиль В. М., Ющен-
ко Б. И.)



Блок обработки информации
системы контроля и управле-
ния МШУ



Малозумящее устройство
(МШУ) глубокого уровня ох-
лаждения



Блок управления и контроля

ЛОКАЛЬНАЯ СЕТЬ НА БАЗЕ СЕТЕВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ АЛИСА

(К ст. Елинера Е. И. и др.)



В вычислительном центре Сибирского отделения Академии наук СССР создана локальная вычислительная сеть ВЦКПСМ. Эта сеть состоит из распределенных однородных многомашинных систем (РОМСИ). Каждая РОМСИ структурно состоит из рабочей машины под-

разделения СМ24, «Электроника 100/25» или «Электроника 79» и расположенных на расстоянии до 1 км ряда периферийных микроЭВМ («Электроника 60», «Электроника МС 1212» или ДВК-2М).

Такое построение сети, объединяющее несколько РОМСИ, позво-

ляет упростить ее структуру, выделив основной объем информации структурному подразделению.

Рабочая ЭВМ РОМСИ вместе с архивной ЭВМ и сложными периферийными устройствами (НМД, НМЛ, АЦПУ) расположены в одном машинном зале.

УЧЕБНЫЙ КОМПЬЮТЕР «ЭЛЕКТРОНИКА УК НЦ»

(К ст. Полосина А. Н.)



Предназначен для оборудования кабинетов вычислительной техники всех типов учебных заведений.

«Электроника УК НЦ» — двухмашинный комплекс, состоящий из центральной и периферийной машин, соединенных высокоскоростными каналами связи. Периферийная машина выполняет функции контроллеров клавиатуры, видеомонитора, бытового кассетного магнитофона, печатающего устройства, гибкого магнитного диска, кассетного ПЗУ. Обе машины выполнены на основе микропроцессора К1801ВМ2.

Программное обеспечение полностью совместимо с ОС ДВК, ОС Рафос, RT-11.

УЧЕБНЫЙ СТЕНД «СТУДЕНТ»

Т. М. Алиев, К. Р. Алышев, Д. И. Вигдоров, Э. А. Тургиев, Г. В. Щирин

Учебный микропроцессорный стенд «Студент» предназначен для обучения студентов основным приемам объектно-ориентированного программирования, методике подключения различных устройств и систем к типовой микропроцессорной системе, изучения методов проектирования, отладки микропроцессорных контроллеров и систем управления реального времени.

Стенд представляет собой программную микроЭВМ на базе микропроцессора КР580ИК80 и комплект лабораторных модулей — «проблемных сред» — действующих моделей устройств и систем, требующих внешнего контроля и управления.

В качестве базового для стенда использовано устройство подготовки данных ЕС-9052. Из базовой модели в изделие без изменений вошли все конструктивные элементы, блок питания и клавиатура. Несложной модификацией одной платы поле отображения дисплея преобразовано в формат 32×16 знакомест для размещения адресов и полей мнемкокодов программы. В панели управления базового устройства собрана микроЭВМ.

Состав стенда «Студент»: микроЭВМ на базе БИС КР580ИК80 всего на 6 ТЭЗах ЕС ЭВМ (150×140 мм), включающая в себя ОЗУ (до 64К байт) и ПЗУ (8К байт); алфавитно-цифровой дисплей и клавиатура;

стойка для сменных модулей «проблемных сред»;

панель портов ввода-вывода и линий запросов прерывания для связи с «проблемной средой»;

набор штекерных соединений

для подключения к «проблемной среде».

Внутреннее программное обеспечение стенда включает в себя: программу-монитор, модули контроля и Быстродействие и разрядность

Число портов ввода-вывода на внешней панели

Отображение информации на экране дисплея, знакомест

тестирования, систему подготовки и отладки программ на языке мнемкокодов и библиотеку программ.

Стенд применяется в качестве автономного устройства с набором четырех «Проблемных сред» (ПС), на каждой из которых можно выполнить ряд лабораторных работ.

Первая ПС (набор импульсных генераторов и точечных индикаторов) служит для освоения техники организации циклов и маскирования, различных способов распознавания сигналов, простейшего управления.

Вторая ПС (микроэлектродвигатель с изменяемой методом широтно-импульсной модуляции скоростью управления) применима для моделирования вариантов простых систем измерений и регулирования в реальном масштабе, изучения вычислительных операций, сложных циклов и подпрограмм.

Третья ПС (набор клавиш, соединенных по матричной схеме, и многоразрядный символьный индикатор) служит для обучения методам логической обработки кодов, организации управляемых выдержек времени, формирования и просмотра массивов; для отработки программ распознавания нажатой кла-

виши и защиты от «дребезга» контактов, обеспечения требуемой временной диаграммы вывода данных на индикатор.

Четвертая ПС (динамическая модель известной задачи «Светофор» с условным отображением движения транспорта на управлении перекрестке) применяется для программирования задач массового обслуживания.

согласно параметрам МПК БИС КР580

4

32×16

Аппаратно-программные средства и конструктивные особенности стенда позволяют пользователю самому расширять набор проблемных сред с учетом принятой методики преподавания, степени подготовки и основной специальности обучаемых.

Для построения лаборатории как набора учебных стендов, подключаемых к достаточно мощному базовому вычислительному комплексу, наиболее рационально выбрать в качестве ядра системы микроЭВМ СМ-1803. Набор стендов связан с ней по радиальным четырехпроводным каналам. С этой целью предусмотрено подключение к устройству дополнительной платы последовательного интерфейса типа ИРПС.

Стенд эффективен и для профессионального использования в качестве эмулятора при разработке микропроцессорных контроллеров.

Стенд разработан на кафедре ИИВТ АЗИНЕФТЕХИМ совместно с Бакинским производственным объединением «Радиостроение». БПО «Радиостроение» планирует выпуск таких стендов в 1986/1987 гг.

За справками обращаться по адресу: 370010 Баку, проспект Ленина, 20, АЗИНЕФТЕХИМ им. М. Азизбекова, кафедра ИИВТ; тел. 98-45-09.





ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ КОММУНИКАЦИОННЫХ ПРОЦЕССОРОВ В РАСПРЕДЕЛЕННЫХ ВЫЧИСЛИТЕЛЬНЫХ КОМПЛЕКСАХ

(К ст. Волкова С. В. и др.)

Объединение производственного оборудования с интеллектуальными системами управления: станков с ЧПУ, промышленных и транспортных роботов, автоматических складов и систем управления на базе микро- и мини-ЭВМ образует гибкое автоматизированное производство (ГАП).

Центральное место в информационной структуре ГАП занимают микропроцессорные средства управления технологическим оборудованием (АСУ ТО), технологической подготовкой производства (АСТПП), технологическим процессом (АСУ ТП), гибким производственным модулем

(АСУ ПМ), качеством (АСУК), предприятием (АСУП), проектными работами (АСУПР), а также автоматическая транспортно-складская система (АТСС) и автоматизированная система оперативно-диспетчерского управления (АСОДУ).

Уровни системы управления ГАП: **первый** — управление оборудованием, станками, роботами, механизмами складов, транспортными тележками и системами, контрольно-измерительными приборами; **второй** — управление устройствами гибких производственных модулей, движением заготовок, полуфабрикатов готовых

изделий, инструмента, транспортных кассет; **третий** — управление технологическими линиями, технологическими процессами, постами контроля; **четвертый** — оперативно-диспетчерское управление и технологическая подготовка производства, управление обменом данных как внутри автоматической линии, так и с вышестоящими уровнями управления, управление транспортными устройствами и складами; **пятый** — календарное планирование основного и вспомогательного производства, управление качеством, разработками.

ческие, т. е. физический адрес линии, через которую должна осуществляться связь. В каждый момент времени для любой коммуникационной ЭВМ должно быть установлено взаимно-однозначное соответствие между логическими и физическими каналами. Однако в процессе работы это соответствие может изменяться.

Управляющие послыки могут поступать в коммуникационную ЭВМ как в виде сообщений, так и в виде командных строк с ее собственного терминала. При работе с терминала появляются дополнительные возможности динамического контроля за состоянием линий связи. На экран терминала может выводиться постоянно изменяющаяся информация, показывающая текущее закрепление физических каналов связи за логическими, типы смежных узлов, нагрузку линий за последнюю секунду и среднюю загрузку за время, прошедшее после включения системы.

Сообщения от терминалов РВК. Для многозадачных ЭВМ, работающих под управлением ОС RSX-11M, любой абонент может рассматриваться как терминал. Таких терминалов может быть несколько (до 256). Если в некоторый момент времени логический канал связи не прикреплен ни к какой задаче, выполняющейся под управлением ОС, то любые информационные символы воспринимаются особой задачей ОС RSX-11 — задачей MCR. Эта задача рассматривает поступающие символы как коды клавиш, аналогично тому, как она рассматривает коды, поступающие от непосредственно связанных с данной ЭВМ терминалов. Этот, так называемый терминальный режим линии связи, действует до тех пор, пока какая-нибудь задача, запущенная в терминальном режиме, не прикрепит к себе линию связи и не начнет режим «передачи сообщений». В терминальном режиме никакого контроля передаваемой информации не проводится, так как любая передача символа задаче MCR сопровождается эхо-символом, выводимым на экран дисплея, подключенного либо непосредственно к коммуникационной ЭВМ, либо к оконечному абоненту РВК. Для работы оконечных абонентов в терминальном режиме имеется соответствующая программная поддержка.

Сообщения между абонентами РВК. Сообщения, циркулирующие между абонентами РВК, могут быть нескольких видов, например «Загрузка оперативной памяти абонента», «Выгрузка памяти», «Проверка», «Обмен файлами», «Передача информации прикладной программ» и некоторые другие. Передача заголовка сообщения начинается тем, что инициативная сторона выдает в линию связи специальный код сообщения и ожидает ответного кода подтверждения. При получении подтверждающего символа начинается передача собственно заго-

ловка. Заголовок, как правило, состоит из двух двухбайтовых слов. Первое слово идентифицирует сообщение. К примеру, оно может содержать адрес оперативной памяти, с которого потребуются загрузить программу в ЭВМ-абонент. Второе слово содержит длину сообщения. Заголовок сопровождается байтом контрольной суммы. После получения подтверждения о правильности передачи заголовка инициативная сторона начинает передачу данных. Большие объемы данных разбиваются на пакеты фиксированной длины. Каждый пакет сопровождается контрольными байтами. Длина пакета может быть переменной в зависимости от помехозащищенности каналов связи. Существует специальный вид заголовков с нулевой длиной данных. Такие заголовки могут рассматриваться как команды для абонентов РВК.

При сбое допускается повторная передача неправильно переданного пакета данных. Все пакеты обрамляются номером логического канала, по которому должна происходить передача. Для согласования скоростей различных каналов связи в коммуникационных процессорах создаются буфера для физических каналов. Чтобы избежать переполнения этих буферов, разрешено передавать по линиям связи специальные служебные символы «запрет передачи» и «разрешение передачи». Во избежание «зависания» каналы имеют программно устанавливаемую величину максимально допустимого времени запрета передачи. По истечении этого времени любой абонент имеет право возобновить передачу данных.

Объем пакета программ, реализующего транспортную службу, оказался существенно меньшим, по сравнению с широко распространенной системой DECNET [7]. Кроме того, разработанный пакет допускает работу в реальном масштабе времени для значительно более широкого класса технологических процессов. С одной стороны, это объясняется меньшими задержками в передаче данных (в отдельных режимах на передачу байта расходуется менее десяти команд коммуникационного процессора), а с другой — тем, что данный программный пакет органично взаимодействует с операционными системами реального времени, которые устанавливаются на ЭВМ, управляющих технологическим оборудованием.

Пакет распределенной отладки программ

Два обстоятельства определяют актуальность разработки пакета отладки программ для коммуникационного процессора. Во-первых, этот процессор имеет физические каналы связи сразу с несколькими машинами. Поэтому с его помощью может быть обеспечена как отладка отдельных

программ, так и совместная отладка нескольких параллельно исполняющихся программ. Во-вторых, появляется возможность перенести основной объем отладочной системы с управляющих ЭВМ на коммуникационный процессор. Освобождаемая память в управляющих ЭВМ позволяет существенно увеличивать размеры отлаживаемых программ. Кроме того, могут быть значительно расширены функции самой отладочной системы.

Пакет отладки имеет несколько подсистем. Первая подсистема — резидентная. Она добавляется к каждой отлаживаемой программе. Можно сказать, что резидентная подсистема является компонентом системы отладки в управляющих ЭВМ, с помощью которого обеспечивается взаимодействие других подсистем пакета с отлаживаемой программой. Объем занимаемой ею памяти относительно невелик — от одного килобайта до килослова. Все остальные подсистемы пакета отладки размещаются в коммуникационной ЭВМ. Вторая подсистема предназначена для отладки программ в отдельно взятой управляющей ЭВМ. По своим функциональным возможностям эта подсистема очень близка к отладчику ODT ОС RSX-11 [8]. Дополнительно введены режимы условной трассировки, когда могут проследиваться события отлаживаемой программы, связанные с исполнением команд, изменениями области памяти, передачей (приемом) данных в другую (из другой) ЭВМ и т. п.

Подсистема совместной отладки нескольких параллельно исполняемых программ реализует следующие возможности:

отладку отдельной программы при приостановленных программах во всех остальных взаимодействующих ЭВМ; переключение с отладки одной программы на отладку программы с другой ЭВМ. Переключение сопровождается запоминанием состояния отладки (точек останова, регистров смещения и т. п.) в одной ЭВМ и восстановлением его в другой. Точки останова в отключенной от отладки программы временно удаляются;

отладку отдельной программы при исполняющихся программах в других ЭВМ. В любой момент времени пользователь может проследивать точки останова, вести условную трассировку только в одной ЭВМ. Остальные программы выполняются в обычном режиме. Система отладки только выдает на терминал диагностические сообщения в случае аварийного завершения программы;

останов программ в любой ЭВМ и повторный запуск с точки старта или с точки приостановки; продвижение всех программ или некоторой группы на одну либо несколько команд;

загрузку программ в любую укзываемую управляющую ЭВМ; Эта

возможность реализуется в тех случаях, когда коммуникационный процессор связан с многозадачной ЭВМ, где хранятся образы всех допустимых для загрузки программ;

удаление программы, которое сопровождается отключением от связи с машиной, где располагалась удаляемая программа.

Опыт эксплуатации пакета распределенной отладки показал, что достигается значительное ускорение в доведении управляющих программ. Особенно эффективно работает система отладки при больших управляющих программах, взаимодействующих друг с другом.

Заключение. Разработанное для коммуникационного процессора программное обеспечение можно использовать в РВК различной структуры — кольцевых, магистральных, иерархических. Создание средств для введения новых форматов сообщений обеспечивает довольно быстрый переход с одной структуры к другой. Дополнительное программное обеспечение управляющих и многозадачных ЭВМ позволяет подключить к коммуникационному процессору микро- и мини-ЭВМ нескольких типов. Работы по программному обеспечению коммуникационного процессора продолжаются. Предполагается, в частности, создание системы отладки для программ, написанных на языках высокого уровня.

Телефон для справок: 535-25-14, Москва.

ЛИТЕРАТУРА

1. Есин Ю. А., Лобиков Л. В., Савин А. И., Уксусов А. С. Гибкая производственная система механообработки // Электронная промышленность. — 1984. — № 8.
2. Савин А. И., Кузнецов М. Н., Уксусов А. С., Целяпин А. Н. Единство организационной структуры и структуры систем управления гибких автоматизированных производств // Микропроцессорные средства и системы. — 1985. — № 4.
3. Прангишвили И. В. Микропроцессоры и локальные сети микро-ЭВМ в распределенных системах управления. — М.: Энергоатомиздат. 1985.
4. Богуславский Л. Б. Управление потоками данных в сетях ЭВМ. — М.: Энергоатомиздат. 1984.
5. Introduction to RSX-11M and RSX-11M-PLUS, AA-L763A-TC.
6. Вейцман К. Распределенные системы мини- и микро-ЭВМ. — М.: Финансы и статистика, 1983.
7. Stuart W. DNA: the digital network architecture IEEE Trans. Commun. — 1980. — Vol. 28. — N 4.
8. ODC Reference manual, IAS/RSX-11, AA-M507A-TC.

Статья поступила 22 апреля 1986 г.

УДК 681.3.066

М. С. Тарков

ОРГАНИЗАЦИЯ УДАЛЕННОГО ИСПОЛНЕНИЯ КОМАНД МОНИТОРА РАФОС В ВЫЧИСЛИТЕЛЬНОЙ СИСТЕМЕ МИКРОС

Вычислительная система (ВС) с программируемой структурой [1—3] представляет собой совокупность элементарных машин (ЭМ), объединенных программно-управляемой из этих машин сетью линий связи. В качестве ЭМ ВС с программируемой структурой Микрос [3] выступает конфигурация из серийных средств мини-микроЭВМ, совместимая с машинами «Электроника 60М», «Электроника 100/25», СМ-3, СМ-4 и имеющая аппаратуру межмашиной связи.

Создание распределенной операционной системы ВС Микрос преследует цель сочетания режима высокоэффективного параллельного решения задач с обеспечением услуг, предоставляемых сетями ЭВМ. Все ресурсы ВС (внешние устройства, файлы и т. д.) распределены между ЭМ и одинаково доступны для всех пользователей. Базовая операционная система (БОС) ЭМ ВС Микрос содержит минимальные средства, необходимые для управления ресурсами и заданиями в системе, и является исполнительной программой, управляющие воздействия для которой вырабатываются пользователем.

В данной работе предложен способ организации взаимодействия базовой ОС ЭМ ВС Микрос со штатной ОС Рафос [4], которая позволит осуществить удаленную инициализацию исполнения системных программ ОС Рафос (транслятора, компоновщика и т. п.) в ЭМ посредством передачи в нее соответствующей команды монитора Рафос из любой ЭМ, содержащей терминал.

Использование системных программ штатной ОС Рафос предопределяет организацию программного обеспечения ЭМ на основе ФВ-монитора Рафос. Системные программы исполняются в фоновом разделе, а в оперативный раздел загружается резидент БОС, содержащий ядро ОС, программные средства организации межмашиных обменов [5, 6] и средства взаимодействия БОС с монитором Рафос. Деление БОС на резидентную и нерезидентную части вызвано необходимостью иметь объем памяти фонового раздела, достаточный для работы системных программ и трансляторов. Программы нерезидентных процессов БОС, а также процессов ОС, не входящих в состав БОС, хранятся на внешних запоминающих устройствах и вызываются по мере необходимости. Совокупность исполняемых оперативных заданий всех ЭМ

реализует сквозной протокол взаимодействия процессов.

Пользователь взаимодействует с БОС посредством программы обработки прерываний от клавиатуры, которая осуществляет запись команды монитора в специально выделенный буфер. Для идентификации записываемая в буфер команда расширяется указанием адреса ЭМ-исполнителя. Сформированное таким образом сообщение передается из ЭМ оператора по сети линий связи в ЭМ, являющуюся исполнителем данной команды. Резидент БОС по отношению к интерактивному монитору КМОН Рафос играет роль пультавого терминала. Взаимодействие резидента БОС и КМОН осуществляется следующим образом.

Резидент БОС ЭМ-исполнителя принимает команду монитора из линии связи с соседней ЭМ и записывает ее в кольцевой буфер ввода с клавиатуры терминала. Запись символов команды в кольцевой буфер реализована так же, как в резидентном мониторе Рафос. Далее резидент БОС выделяет буфер свободной памяти для записи строки символов, которые будут выдаваться на экран терминала в процессе исполнения команды монитора. Адрес первого слова буфера записывается в указатель COUNT, адрес третьего — в указатель PUT (рисунок). Сбрасывается флаг запрета прерываний терминала по выводу XEDOF и разрешаются прерывания по выводу на терминал. После этого оперативное задание обнаружит слово состояния ISTAT фонового задания и самоблокируется путем выполнения системой макрокоманды .SPND. В результате управление передается фоновому заданию, роль которого играет интерактивный монитор КМОН, интерпретирующий команду, записанную в кольцевой буфер ввода с терминала.

Блокировка фонового задания, исполняющего команду монитора, происходит по окончании вывода на экран терминала любой строки, которая была сформирована в процессе исполнения команды. Блокировка возникает при перехвате символической информации, выдаваемой системой программой Рафос на экран терминала ЭМ-исполнителя. Перехват основан на модификации программы обработки прерываний по выводу на экран терминала, входящей в состав резидентного монитора системы Рафос. Вместо команды выдачи символа на экран терминала в указанную

АСЕМБЛЕР MACRO 11-РАФОС

ПРОГРАММА ПЕРЕХВАТА СТРОКИ
 СИМВОЛОВ, ВЫВОДИМЫХ НА ТЕРМИНАЛ

ОПРЕДЕЛЕНИЯ

*IMPUR=141314 АДРЕС УКАЗАТЕЛЕЙ СМЕШАННЫХ ОБЛАСТЕЙ
 XEDOFF=150250 ФЛАГ ЗАПРЕТА ПЕРЕРЫВАНИЙ ТЕРМИНАЛА ПО ВЫВОДУ
 TTLQE=145366 УКАЗАТЕЛЬ ЭЛЕМЕНТА ОЧЕРЕДИ К ДРАЙВЕРУ ТЕРМИНАЛА
 TTPB=140030 УКАЗАТЕЛЬ АДРЕСА БУФЕРНОГО РЕГИСТРА ЭКРАНА ТЕРМИНАЛА
 PUT: WORD 0 УКАЗАТЕЛЬ ЯЧЕЙКИ В БУФЕРЕ ВЫВОДА
 COUNT: WORD 0 УКАЗАТЕЛЬ СЧЕТЧИКА СИМВОЛОВ СТРОКИ
 SYM: WORD 0 УКАЗАТЕЛЬ ПРЕДЫДУЩЕГО СИМВОЛА
 ENDMES: WORD 0 ПРИЗНАК КОНЦА ИСПОЛНЕНИЯ КОМАНДЫ МОНИТОРА

ПРОГРАММА

TYPE:	MOVB	-(R4),@PUT	СИМВОЛ В БУФЕР
	MOVB	#0,@TTPB	СБРОС ФЛАГА ГОТОВНОСТИ
	INC	PUT	PUT:=PUT+1
	INC	@COUNT	ЧИСЛО СИМВОЛОВ+1
	CMPB	@R4,#12	КОНЕЦ СТРОКИ?
	BNE	1x	НЕТ
	BR	6x	
1x:	TST	TTLQE	TTLQE > 0 ?
	BNE	7x	ДА
	CMPB	SYM,#7?	ПРЕДЫДУЩИЙ СИМВОЛ "?"?
	BNE	2x	НЕТ
	CMPB	@R4,#4?	ПРОБЕЛ?
	BNE	2x	НЕТ
	DEC	PUT	PUT:=PUT-1
	DEC	@COUNT	ЧИСЛО СИМВОЛОВ-1
	BR	7x	
2x:	CLR	ENDMES	ENDMES:=0
	CMPB	@R4,#52	ЗВЕЗДОЧКА?
	BNE	3x	
	BR	5x	
3x:	CMPB	@R4,#56	ТОЧКА?
	BNE	4x	НЕТ
	CMPB	SYM,#12	ПРЕД. СИМВОЛ "LF"?
	BNE	4x	НЕТ
	BR	5x	
4x:	CMPB	@R4,#7?	ВОПРОС. ЗНАК?
	BNE	7x	НЕТ
	CMPB	SYM,#12	ПРЕД. СИМВОЛ "LF"?
	BEQ	7x	ДА
5x:	MOV	#1,ENDMES	ENDMES:=1
6x:	MOVB	#200,@XEDOFF	ЗАПРЕТ ВЫВОДА НА ТЕРМИНАЛ (<<CTRL>>S)
	MOV	R0,-(SP)	
	MOV	#<IMPUR+2>,R0	В R0 АДРЕС СМЕШ. ОБЛАСТИ ОПЕРАТИВНОГО ЗАДАНИЯ
	CLR	@R0	СБРОС БИТА БЛОКИРОВКИ (I.STATE:=0)
	INC	34(R0)	I.SCTR:=1,SCTR+1
	MOV	<SP>+,R0	
7x:	MOVB	@R4,SYM	ЗАПОМНИТЬ СИМВОЛ
	JMP	@#150549	ВОЗВРАТ УПРАВЛЕНИЯ В МОНИТОР

монитор ожидает дополнительную информацию, необходимую для выполнения команды. Если ENDMES=0, выполнение команды монитора должно быть продолжено. Для этого оперативное задание выделяет буфер для вывода очередной символической строки, сбрасывает флаг XEDOFF и самоблокируется, выполняя SPND. Выполнение фонового задания продолжается до окончания записи в выделенный буфер вывода очередной строки.

Переданные по сети линии связи сообщения о результатах исполнения команды монитора выводятся в машину оператора на экран терминала. Если в принятом сообщении признак конца исполнения команды монитора не равен нулю, то разрешается ввод следующей команды монитора.

Предложенные программные средства реализованы в составе интерпретатора языка управления заданиями BC с программируемой структурой Микрос. Общий объем описанных здесь программ составляет около 200 команд языка Macro-11. Предложенный способ сопряжения БОС с системными программами Рафос не зависит от конкретного состава программно аппаратных средств, которые объединяют элементарные машины в систему, и может быть рекомендован для использования в широком классе вычислительных систем и сетей, построенных на базе ЭВМ, использующих FB-монитор ОС Рафос.

При выполнении работы были использованы листинги программ резидентного монитора Рафос. Приведенные в программе TYPE адреса соответствуют версии FB-монитора Рафос, поставляемого в составе программного обеспечения вычислительного комплекса 15-ВУМС-28.

Адрес для запроса дополнительной информации: 630090, Новосибирск, ул. Институтская, 4/1, Институт теоретической и прикладной механики СО АН СССР.

ЛИТЕРАТУРА

программу при инициализации БОС записывается команда передачи управления на программу TYPE (см. рисунок), входящую в состав резидента БОС.

Программа TYPE записывает символы строки, выводимой на экран терминала, в буфер, начальный адрес которого содержится в PUT. При записи каждого символа осуществляется проверка, не является ли данный символ признаком конца строки (символы «LF», «*», «.», «?»), Проверка на символы «*», «.», «?» производится, если очередь вывода терминала пуста (указатель TTLQE равен нулю). При появлении признака конца строки устанавливается флаг XEDOFF, запрещающий вывод

на терминал, и — флаг ENDMES, если выводимый символ не с «LF». Затем обнуляется слово состояния I.STATE и признака ожидания I.SCTR в смешанной области оперативного задания. В результате после возврата управления в монитор произойдет блокировка фонового задания и деблокировка оперативного задания, которое записывает значение ENDMES во второе слово буфера вывода и инициирует передачу сообщения (содержащего строку символов) по сети линий связи в машину оператора.

В ЭМ-исполнителе тестируется значение флага ENDMES после завершения вывода сообщения в соседнюю ЭМ. Если ENDMES=1, то выполнение команды монитора завершено или

- Евреннов Э. В., Хорошевский В. Г. Однородные вычислительные системы. — Новосибирск: Наука, 1978. — 318 с.
- Корнеев В. В. Архитектура вычислительных систем с программируемой структурой. — Новосибирск: Наука, 1985. — 166 с.
- Дмитриев Ю. К., Задорожний А. Ф., Корнеев В. В. Элементарная машина вычислительной системы с программируемой структурой МИКРОС // Вычислительные системы с программируемой структурой (Вычислительные системы, вып. 94). — Новосибирск, 1982. С. 16—32. (Тр. Ил-та математики СО АН СССР).

ПРЕДИСЛОВИЕ ПРЕПОДАВАТЕЛЯ

Автор этой статьи — ученик десятого класса школы № 1 п. Протвино, Московской области Юрий Герштейн. Программировать начал в 8 классе школы юных программистов под руководством Б. Б. Филимонова. За период с октября 1983 г. по май 1985 г. освоил программирование на языках Бейсик, Фортран и Си. Написал ряд игровых и прикладных программ: LIFE; движение тела в среде, противодействующей движению; движение тел в центральном поле и т. п. К настоящему моменту получил практический навык работы в операционных средах: ЭВМ DEC—SYSTEM-10; RSX-11M CM-4; UNIX V7 CM-4 и ЕС ЭВМ; ОСMSX персональных компьютеров YAMAHA.

В июне 1985 г. Юрий вместе с группой школьников проходил производственную практику в Лаборатории инструментальных средств программирования Института физики высоких энергий, а с сентября 1985 г. по май 1986 г. эта группа продолжала практику, занимаясь раз в неделю. Этот период обучения в основном был посвящен методам реализации языков программирования. Почему именно им? По нашему мнению, это направление должно быть обязатель-

ной частью фундаментальных знаний программиста-профессионала. И не только потому, что это один из основных разделов программирования, имеющих хорошее теоретическое обоснование и практическое применение, но и потому, что тексты — это в сущности единственные объекты, с которыми имеет дело программист. С помощью операций над ними он создает программы. Поэтому кажется вполне естественным требование к фундаментальным навыкам профессионала-программиста — умение автоматизировать преобразование и обработку текстов. К сожалению, этот факт находит слабое отражение при подготовке программистов в высшей школе. В результате большинство выпускников вузов по этой специальности, как правило, прекрасно ориентируются в достоинствах и недостатках заимствованных операционных систем и практически не имеют фундаментальных навыков.

Работе, представленной в настоящей статье, предшествовала учебная работа Юрия по реализации интерпретатора упрощенного Бейсика. Для этого ему был предложен язык нисходящего разбора (ЯНР), разработанный и поставленный на CM-4 для Фортрана В. В. Сиколенко. Теперь несколько слов собственно о данной работе. Прежде всего, она далеко выходит за рамки чисто учебной —

это законченный программный продукт, удовлетворяющий требованию переносимости.

Теоретической моделью ЯНР является обобщенный язык нисходящего разбора с ограниченными возвратами, подробно исследованный в классическом труде А. Ахо и Дж. Ульмана «Теория синтаксического анализа, перевода и компиляции». Основная область применения ЯНР — разработка и реализация диалоговых систем, интерпретаторов (компиляторов) с языков программирования, текстовых преобразователей. Несомненным достоинством ЯНР, особенно применительно к микропроцессорным системам, является его концептуальная простота и компактность реализации. Так, ЯНР-интерпретатор для функционирования на CM-4 требует около 1,2К байта оперативной памяти.

Реализация ЯНР-транслятора выполнена по схеме раскрутки, т. е. он реализован на ЯНР. В данном случае это тем более обосновано, так как существовала реализация ЯНР для Фортрана. В заключение хочу отметить, что реализация ЯНР выполнена автором на хорошем профессиональном уровне и совершенно самостоятельно.

С. Г. Никитин,
к.ф.-м.н., с.н.с. Института физики
высоких энергий, п. Протвино

УДК 681.325

Ю. С. Герштейн

СИ-РЕАЛИЗАЦИЯ ЯЗЫКА НИСХОДЯЩЕГО РАЗБОРА

Язык Си ориентирован в основном на создание интерпретаторов, компиляторов, драйверов внешних устройств, текстовых редакторов и других системных программ и заменяет в этих случаях язык ассемблера. Дополнение языка Си синтаксическим анализатором значительно упрощает написание программ, предназначенных для работы с текстами. Таким анализатором является язык нисходящего разбора (ЯНР). Название языка вполне отражает область применения — синтаксический анализ символьных строк. Основной объект манипуляции операторов языка — входная строка, представленная с точки зрения Си указателем на первый символ. Основная структурная единица программы на ЯНР — ЯНР-функция, состоящая из двух взаимодействующих частей:

ЯНР-части, переводимой препроцессором во внутреннее представление, интерпретируемое в дальнейшем ЯНР-интерпретатором (ЯНР-машиной);

Си-части, компилируемой обычным образом и (в общем случае) независимо от первой.

ЯНР в настоящей реализации выполнен как надязык. ЯНР-функция начинает выполняться после вызова из Си-программы ЯНР-интерпретатора. При таком вызове интерпретатору в качестве параметра передается адрес массива, в котором находится внутреннее представление соответствующей ЯНР-функции (см. рисунок).

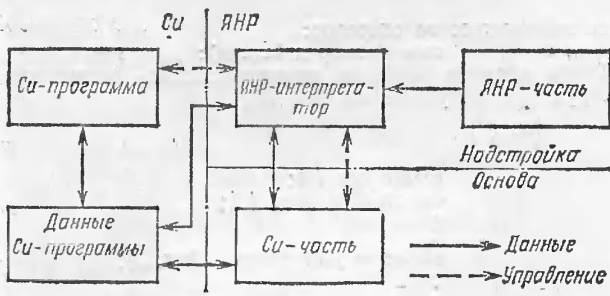
Описание языка. Рассмотрим более подробно структуру ЯНР-функции. Введем понятие статуса — характеристики, которая обеспечивает логику анализа. Статус может принимать значения TRUE, FALSE или ERROR (на языке Си этим значениям соответствуют 1, 0 и отрицательное число). В начале выполнения ЯНР-программы статус полагается равным TRUE. В процессе работы программы он может изменяться, причем ЯНР-машина контролирует эти изменения и в соответствии с ними проводит дальнейшую работу. Когда

4. Операционная система CM ЭВМ РАФОС: Справочник / Л. И. Валикова, Г. В. Вигдорчик, А. Ю. Воробьев, А. А. Лукин; Под общей ред. В. П. Семика. — М.: Финансы и статистика, 1984. — 207 с.
5. Корнеев В. В., Моныхов О. Г., Тарков М. С. Ядро операционной системы ЭМ вычислитель-

- ной системы с программируемой структурой // Однородные вычислительные системы (Вычислительные системы, вып. 90). — Новосибирск, 1981. С. 22—42. (Тр. Ин-та математики СО АН СССР).
6. Задорожный А. Ф., Корнеев В. В., Тарков М. С. Об организации коммуникаций между

процессами в вычислительной системе МИПРОС // Распределенная обработка информации (Вычислительные системы, вып. 105). — Новосибирск, 1984. С. 70—84. (Тр. Ин-та АН СО СССР).

Статья поступила 7 апреля 1986 г.



Взаимодействие различных частей Си-программы, включающей ЯНР

выполнение ЯНР-программы закончено, текущий статус считается результатом работы ЯНР-программы.

Основной операцией ЯНР является вызов символа, в результате которого вырабатывается значение статуса. Существует три типа символов: внутренние (метасимволы); внешние (действия) и встроенные. Результат вызова зависит от типа символа. Если символ внутренний, то вызывается соответствующая часть ЯНР-программы. Если внешний, то Си-подпрограмма, адрес которой был передан при инициализации.

Наиболее общая структура ЯНР-программы имеет следующий вид:

```
EXTERN <имя действия>, <имя действия> ...
<описание метасимвола>
<описание метасимвола>
```

Описание EXTERN определяет перечисленные имена как имена внешних символов и устанавливает соответствие между именем действия и его номером, совпадающим с порядковым номером действия в списке описания.

Описание метасимвола имеет вид:

```
S -> α1 β1 γ1 δ1 ...
      α2 β2 γ2 δ2 ...
      α3 β3 γ3 δ3 ...
      :
      :
```

Здесь S — определяемый метасимвол, $\alpha_i, \beta_i, \gamma_i, \delta_i$ — базовые конструкции языка или термы, отделенные друг от друга в строке пробелами. Все термы при выполнении вырабатывают значение статуса. Описание отдельного метасимвола состоит из последовательности строк-альтернатив. Схема вычисления ЯНР-машиной отдельного метасимвола при его вызове выглядит следующим образом. Вначале выполняется первый терм в первой строке данного метасимвола. Дальнейший порядок вычислений зависит от значения статуса, выработанного при вычислении терма. Если выработано значение TRUE, то ЯНР-машина переходит к вычислению следующего терма в строке. При выработке значения статуса FALSE, ЯНР-машина восстанавливает свое состояние перед началом выполнения данной альтернативы и переходит к вычислению следующей альтернативы вычисляемого метасимвола. Последнюю ситуацию в дальнейшем будем называть отказом альтернативы.

Вычисление метасимвола заканчивается в трех случаях:

по окончании вычисления последнего терма текущей строки со значением статуса TRUE; значение статуса будет также и результатом вызова данного метасимвола;

при отказе альтернативы, являющейся последней для этого метасимвола; вызов метасимвола завершится со значением статуса FALSE;

при выработке каким-либо термом значения статуса ERROR; в этом случае происходит немедленный выход из ЯНР-функции.

Простые термы. Идентификатор метасимвола или действия является термом и при выполнении интерпретируется как вызов соответствующего символа. Конструкция \uparrow (метасимвол) также является термом и означает передачу управления на указанный метасимвол. При передаче управления на метасимвол возврата к месту передачи не происходит. Очевидно, это этот терм может быть только завершающим в альтернативе.

Встроенные символы T, F, E («целое») представляет собой термы-константы, вырабатывающие соответствующие значения статуса TRUE, FALSE, ERROR. Два терма «строка символов» и «целое» предназначены для анализа входной строки. При выполнении терма «строка символов» ЯНР-машина сравнивает эту строку с началом входной строки. Если сравнение удачно, то вырабатывается значение TRUE, и указатель входной строки перемещается на символ, непосредственно следующий за совпавшим префиксом. В противном случае просто вырабатывается статус FALSE. Терм «целое» интерпретируется как сравнение этого числа с кодом первого символа входной строки.

Составной частью ЯНР-машины является стек для обмена информацией при вызове метасимволов и действий. Все термы предназначены для манипуляции со стеком и объединяются в три группы: а, б, в. Термы групп а и б вырабатывают значение статуса TRUE.

а) группа загрузки-выгрузки стека включает следующие термы:

- \square «строка символов» — загрузка в верхушку стека указателя на строку символов;
- \square «целое» — загрузка в верхушку стека целого;
- $++$ раздвоенные верхушки стека;
- $—$ выталкивание элемента из верхушки стека;

б) группа целочисленной арифметики включает арифметические операции $+$ $-$ $*$ $/$, применяемые к целым, находящимся в верхушке и предшествующем ей элементу стека;

в) термы сравнения двух элементов стека вырабатывают значения статуса TRUE или FALSE в зависимости от результатов сравнения. Эта группа включает в себя два терма:

- $=$ — проверка двух целых на равенство;
- $S=$ — проверка на совпадение двух строк, указатели которых помещены в стек.

Если произошло сравнение при вычислении этих двух термов, то сравниваемые элементы выталкиваются из стека, в противном случае выталкивается только элемент, находящийся в верхушке стека.

Составные термы. Введение составных термов не увеличивает мощность ЯНР и продиктовано только соображениями удобства. Составной терм можно получить при помощи двух конструкций.

Первая — это последовательность альтернатив, заключенная в фигурные скобки. Например:

$$S \rightarrow \alpha 1 \{ \beta 1 \gamma 1 \beta 2 \gamma 2 \} \delta 2$$

Эта запись эквивалентна следующей:

$$S \rightarrow \alpha 1 S 1 \delta 2 \\ S 1 \rightarrow \beta 1 \gamma 1 \beta 2 \gamma 2$$

Вторая — итерация, предназначенная для циклического выполнения терма, пока им будет вырабатываться значение статуса TRUE. Синтаксис итерации имеет вид: <терм>*. Следующий пример поясняет синтаксис и семантику итерации:

$$S \rightarrow \alpha 1 \beta 1 * \gamma 1,$$

или эквивалентная запись:

S → α1 S1 γ1

S1 → β1 ↑ S1

T

Интерфейс Си и ЯНР-программ. Все сказанное выше можно обобщить на любой язык программирования. Рассмотрим языкозависимые черты реализации. Из любой подпрограммы (функции), реализованной на Си, можно вызвать любую подпрограмму на ЯНР. Вызов осуществляется следующим вызовом ЯНР-интерпретатора:

```
status=janr (prog, string, ext, sv, nsave, restore);
```

Здесь status — результат работы ЯНР-программы; prog — массив байт, в котором находится внутреннее представление ЯНР-программы; string — указатель на входную строку; ext — параметр-функция, ответственная за выполнение действия по его номеру; sv, nsave — соответственно указатель на массив типа int и число элементов в нем. Этот массив будет восстанавливаться при отказах альтернатив и ошибках; restore — функция, которая будет вызываться перед восстановлением массива sv (иногда данные структурированы так, что простого восстановления массива будет недостаточно).

ЯНР-интерпретатор производит инициализирующие операции и начинает выполнять ЯНР-функцию с вычисления метасимвола, описанного в функции первым. При необходимости выполнить внешний символ (действие) ЯНР-интерпретатор производит вызов Си-функции ext, адрес которой был им получен в качестве параметра. Одновременно с вызовом функции ext интерпретатор передает ей параметр-структуру, поля которой содержат указатели на входную строку, верхушку стека, номер действия, которое необходимо выполнить, и статус.

Технология программирования на Си + ЯНР. ЯНР-функция с точки зрения синтаксиса Си представляет собой комментарий, имеющий следующий вид:

```
/*BEGIN  
<ЯНР-функция>  
END*/
```

Препроцессор ЯНР при чтении входного файла игнорирует все строки, находящиеся вне скобок /* BEGIN — END */. Поэтому ЯНР-функции могут располагаться в любом месте Си-программы, где по синтаксису допустим комментарий. Препроцессор преобразует ЯНР-функции в последовательность команд ЯНР-машины и сформированный массив выводит в текстовом виде, как последовательность целых, разделенных друг от друга запятыми. Наиболее удобно использовать такое представление ЯНР-программы для инициализации массива типа char, т. е. следующую конструкцию Си:

```
char prog [] = {  
/* BEGIN  
<ЯНР-функция>  
END */  
};
```

После преобразования данной конструкции препроцессором, а затем компилятором Си, массив prog будет содержать коды ЯНР-машины. Так как для выполнения какого-либо действия ЯНР-интерпретатор вызывает Си-функцию, которая, очевидно, для реализации действия по его номеру будет использовать оператор switch, то препроцессор для этого случая предоставляет дополнительный сервис. Везде, где после ЯНР-функции в тексте встретится конструкция:

```
/*<имя действия> */,
```

он заменяет ее на оператор:

```
case <номер действия>:
```

Таким образом наиболее предпочтительна следующая структура программы на Си + ЯНР:

```
foo ( )  
{  
extern bar ( );  
extern char prog [ ];  
:  
status = janr (prog, ..., bar, ...);  
:  
}  
char prog [ ] = {  
/*BEGIN  
<ЯНР-функция>  
END*/  
};  
bar (com)  
struct  
{  
int nrule;  
int stat;  
char * string;  
union {  
int ival;  
char * pval;  
} *jnr __ stk;  
} *com;  
{  
switch (com->nrule) {  
/*<имя действия>*/  
<тело действия>  
break;  
/*<имя действия>*/  
<тело действия>  
break;  
:  
:  
:  
}  
}
```

Ниже приведен пример интерпретатора — калькулятора арифметических действий над целыми числами.

```
char line [80];  
main ( )  
{ extern semantic ( );  
extern char com [ ];  
register  
do { gets (line);  
printf ("%s", line);  
stat = janr (com, line, semantic, 0, 0, 0);  
if (stat != 1 && stat != -2)  
printf (": syntax, error\n");  
} while (stat != -2);
```

```
char com [ ] = {  
/*BEGIN  
EXTERN WRITE, DIGIT, MPY10ADD  
calculate → "end" E(2)  
arth → "-" WRITE  
restarth → {"+", "—"} 0 term — ↑restarth  
term → {"+", "—"} ↑restarth  
resterm → {"*", "/"} term —*  
fae → "/*" fae *  
fae → "/*" fae /**  
fae → DIGIT {"DIGIT MPY10ADD"} *  
fae → "(" arth ")"  
E  
END*/  
};
```

```

semantic (p)
register struct {int nrule;
                int status;
                char *str;
                int *stack; } *p;
{ register int i;
  switch ( p -> nrule ) {
/*WRITE*/
    printf ("=% d\n", *(p->stack) );
    break;
/*MPY10ADD*/
    i = *(p->stack --);
    *(p->stack) * = 10;
    *(p->stack) + = i;
    break;
/*DIGIT*/
    if (p->status = isdigit (*p->str) )
        *(++p->stack) = *p->str++ - '0';
}

```

Архитектура ЯНР-машины. ЯНР-машина имеет двухстековую байтовую архитектуру и 20 машинных команд переменной длины, предназначенных для работы со стеком, входной строкой и структурой программы. Первый байт команды содержит код, остальные — информационную часть. Кроме стека, непосредственно доступного на уровне ЯНР-функции, ЯНР-машина поддерживает стек, используемый для запоминания текущего состояния машины и последующего возврата к нему. Необходимость в этом возникает в командах вызова символа и при отказах альтернатив.

В настоящее время система реализована на ЭВМ СМ-4 под управлением ОС RSX-11M. ЯНР-интерпретатор занимает около 1,2К байт оперативной памяти. На расшифровку ЯНР-команды и передачу управления соответствующему блоку интерпретации затрачивается 5—6 машинных команд. Для переноса ЯНР на другую ЭВМ достаточно изменить единственный оператор, задающий длину машинного слова.

Статья поступила 15 мая 1986 г.

УДК 002 : 658.012.01.56

А. Л. Попов

ИНФОРМАЦИОННО-ПОИСКОВАЯ СИСТЕМА «БИБЛИОТЕКА» И ЕЕ РЕАЛИЗАЦИЯ НА ПЕРСОНАЛЬНОЙ ЭВМ

Информационно-поисковая система (ИПС) «Библиотека» используется для накопления и хранения текстовой информации. Она облегчает подготовку и обработку деловых бумаг, является удобным средством хранения и поиска текстов программ системного, общего и прикладного назначения с соответствующими инструкциями для пользователя. Возможности ИПС предусматривают регистрацию источников информации на бланках саморасширяемой картотеки, запись информации из этих источников в информационные файлы, поиск и последующий вывод на дисплей и устройство печати необходимых бланков из картотеки и информационных файлов.

В качестве базовой операционной системы применена дисковая операционная система Рафос, совместимая с распространенными вычислительными комплексами ДВК, ВУМС, СМ ЭВМ (СМ-3, СМ-4). Управляющая программа разработана на языке Бейсик. Все программное обеспечение ИПС размещается на двух гиб-

ких магнитных дисках: первый диск «Система» содержит управляющую операционную систему (ОС Рафос), интерпретатор языка Бейсик (BASIC.SAV), управляющую программу (BIBL.BAS), краткую инструкцию по запуску системы (BIBL.TXT). Второй диск «Картотека»/«Библиотека» содержит картотеку, информационные файлы, командный файл запуска системы, файл хранения номера последнего бланка в картотеке (примерный объем картотеки тысяча бланков).

Эксплуатация системы начинается со стандартного запуска операционной системы с диска «Система», установленного в ячейку НГМД, после чего производится запуск командного файла с диска «Картотека»/«Библиотека». Последующее обращение к ИПС ведется в режиме диалога. Диалог с ЭВМ ориентирован на пользователя, не имеющего специальной подготовки в области вычислительной техники.

Пример протокола диалога ВАС ПРИВЕТСТВУЕТ ИНФОРМАЦИОННО-ПОИСКОВАЯ СИСТЕМА

ЗАПУСК ПРОГРАММЫ СИСТЕМЫ: ПОСЛЕ ПОЯВЛЕНИЯ ТОЧКИ /./ ВВЕСТИ С КЛАВИАТУРЫ: @ BIBL

ЗАПУСК СИСТЕМЫ: ПОСЛЕ ПОЯВЛЕНИЯ /READY/ ВВЕСТИ С КЛАВИАТУРЫ:

RUN SY:BIBL

ДАТА (Ч. М. Г.): Ф. И. О.; КООРДИНАТЫ:

РЕЖИМ РАБОТЫ:

1— ЗАПИСЬ.

2— ПОИСК, ЧТЕНИЕ, ПЕЧАТЬ.

3— ОКОНЧАНИЕ РАБОТЫ.

УКАЖИТЕ НОМЕР РЕЖИМА: 1

1— РЕЖИМ ЗАПИСИ ИНФОРМАЦИИ

ОТКРЫВАЮ КАРТОТЕКУ

ЖДИТЕ

КАРТОТЕКА ОТКРЫТА

ВЫХОД ИЗ РЕЖИМА: — РАЗДЕЛ: — (ПРОЧЕРК) (BK)

ЗАПОЛНИТЕ, ПОЖАЛУЙСТА, БЛАНК НОМЕР —1.

РАЗДЕЛ: ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА.

ТЕМА: ИНФОРМАЦИОННО-ПОИСКОВАЯ СИСТЕМА.

ИСТОЧНИК: УПРАВЛЯЮЩАЯ ПРОГРАММА. ОПИСАНИЕ.

ПРИМЕНЕНИЕ: ЗАПИСЬ ИНФОРМАЦИИ.

ОБЕСПЕЧЕНИЕ: ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ.

ПРИЗНАКИ: ХРАНЕНИЕ ИНФОРМАЦИИ ПРОИЗВОЛЬНОГО ТИПА.

УСТАНОВИТЕ ДИСК БИБЛИОТЕКА В ПРАВЫЙ ДИСКОВОД

ДЛЯ ЗАПИСИ ИНФОРМАЦИИ.

ВЫ ГОТОВЫ? (ДА — Д)? Д

ЖДИТЕ

ИНФОРМАЦИОННЫЙ ФАЙЛ —1. LST

ВВОДИТЕ СТРАНИЦУ ИНФОРМАЦИИ — КОНЕЦ ВВОДА — (BK)

ПЕРЕД НАЧАЛОМ ВВОДА ИНФОРМАЦИИ (20 СТРОК),

УСТАНОВИТЕ НЕОБХОДИМЫЙ РЕГИСТР (РУС/ЛАТ).

СТРАНИЦА НОМЕР —1

?

?

?

?

ЖДИТЕ

УСТАНОВИТЕ ДИСК КАРТОТЕКА В ПРАВЫЙ ДИСКОВОД.

ВЫ ГОТОВЫ? (ДА — Д)? Д

ВЫХОД ИЗ РЕЖИМА: — РАЗДЕЛ: — (ПРОЧЕРК) (BK)

ЗАПОЛНИТЕ, ПОЖАЛУЙСТА, БЛАНК НОМЕР —2.

РАЗДЕЛ: —

ЖДИТЕ

Программа организована так, что после окончания работы в любом из режимов пользователь автоматически попадает в основную часть программы и может выбрать другой режим.

* Запись информации пользователя.

Реализация управляющей программы на языке Бейсик позволяет изменять определители бланка карточки и обеспечивает возможность настройки ИПС на конкретные требования пользователя. Ввод информации в режиме ЗАПИСЬ не требует предварительной систематизации. Вывод в режиме ПОИСК, ЧТЕНИЕ, ПЕЧАТЬ определяется формой заполнения бланка поиска, систематизирует полученную информацию.

Возможности записи графической информации ограничены набором знаков клавиатуры и построчным методом ввода информации. Однако есть возможность построения упрощенных рисунков, чертежей, схем и т. п. в режиме экранного редактора в заранее резервированном с помощью ИПС файле.

ИПС «Библиотека» практически не имеет ограничений в сфере обработки и поиска текстовой информации. Она может работать на ЭВМ с дисковой системой внешней памяти и развитыми операционными системами типа: Рафос, Фодос, ОС ДВК.

Комплекты поставки ИПС «Библиотека» может быть поставлена в одном из следующих вариантов: на одном ГМД заказчика (200 мм) полный состав ИПС, включая ОС, управляющую программу, карточку и библиотеку;

на двух ГМД (200 мм): диск 1 — ОС и управляющая программа, диск 2 — карточка и библиотека; листинги исходных текстов: управляющая программа, вспомогательные программы ИПС, библиотека.

Каждый вариант поставки содержит краткую отпечатанную инструкцию по использованию.

Телефон для справок и предложений: 162-91-11 (до 21.00) Попов Александр Леонидович, Москва.

Статья поступила 21 октября 1985 г.

УДК 681.3.06

А. А. Илюкович

ИНФОРМАЦИОННО-ПОИСКОВАЯ СИСТЕМА «КАДРЫ»

Информационно-поисковая система (ИПС) «Кадры» построена на базе микроЭВМ «Электроника 60» с набором внешних устройств: до трех накопителей на магнитных лентах (НМЛ) ИЗОТ 5003; два видеотерминала (дисплеи) 15ИЭ00-013, накопитель на магнитных дисках «Электроника ГМД 7012», печатающие устройства DZM-180 или Роботрон 1160.

Пакет прикладных программ системы позволяет обрабатывать в пакетном и диалоговом режимах информацию кадровых служб организаций с численностью работающих до 15 тыс. человек. Максимальное время ожидания ответа на любой запрос практически не превышает времени полного сканирования информационного массива на МЛ. В реальных условиях, при численности работающих до 3 тыс. человек, время ожидания ответов на запросы, как правило, не превышает 3, 4 мин. При численности работающих от 3 до 15 тысяч время ожидания ответа на отдельные запросы может составить 15...20 мин. Для размещения массивов данных можно использовать НМД ЕС 5061, что значительно сократит время ответа на любые запросы. Так, например, время ответа на запросы справочного характера составляет 3, 4 с., на запросы, связанные с обобщением информации, не более 3 мин. Максимальное время для обучения правилам пользования системой составляет не более 2—3 дней.

Состав и характеристики показателей (реквизитов), образующих информационную базу ИПС, определяются самим пользователем. Стандартный вариант — сведения личной карточки сотрудника. Внутримашинная база состоит из двух информационных массивов данных: КАДРЫ, содержащего сведения о работающих

сотрудниках, и КАДРУВ — со сведениями об уволенных сотрудниках.

При корректировке массива КАДРЫ изменения могут вноситься непосредственно в личную карточку сотрудника, высвеченную на экране дисплея. В случае увольнения сотрудника информация автоматически переносится из массива КАДРЫ в КАДРУВ. В составе рассматриваемой ИПС имеются программные средства, позволяющие выдавать на печать имеющуюся информацию в виде форм статистической отчетности.

Структура запроса состоит из четырех частей: текста, формы, логического выражения, служебных признаков, разделенных символом \square . В тексте запроса задается заголовок, под которым будет выдаваться видеодиаграмма или таблица на печатающем устройстве. Форма определяет вид выходных данных: видеодиаграммы, сводные таблицы или сообщения о количественном составе организации. Логическое выражение состоит из левой части, разделителя и правой части. Логический элемент представляет собой идентификатор реквизита и его значение, соединенные логическими операциями: равно ($=$), больше ($>$), меньше ($<$), не равно (\neq), не больше (\geq), не меньше (\leq). Левая часть логического выражения (включает один или несколько логических элементов, соединенных союзами «и», «или»), идентифицирует записи массива, которые будут подвергнуты обработке. Правая часть — используется только при запросах на корректировку информации, содержит перечисления корректируемых реквизитов, их новые значения, логическую операцию « \Rightarrow » и союз «и». Разделителем между левой и правой частями запроса служит союз «ТО». Таким образом любой запрос в общем случае имеет следующий вид:

$$P_1^1 \text{ I знач } 1 \text{ (или) } P_2^1 \text{ I знач } 2 \text{ (или) } \dots P_n^1 \text{ I знач } n \text{ ТО} \\ P_1^2 = \text{знач } 1 \quad P_2^2 = \text{знач } 2 \dots P_m^2 = \text{знач } m$$

где P_j^i — идентификаторы реквизитов, $j=1,2, I \in J$ (J — множество индексов идентификаторов; I — логическая операция; знач I — I -й реквизит).

В четвертой части запроса задаются служебные признаки: П — печати, У — работы с массивом КАДРУВ, З — каталогизации. Признак печати указывается при необходимости вывода видеодиаграммы на печатающее устройство. Сводные таблицы распечатываются независимо от наличия служебного признака. Если нет признака У, то все данные берутся из массива КАДРЫ. Признак каталогизации указывает, что запросу присвоен определенный порядковый номер и он будет помещен в библиотеку стандартных запросов (БСЗ). При работе со стандартными запросами признаки П и У игнорируются. В БСЗ рекомендуется заносить часто используемые запросы. В структуре запроса могут отсутствовать отдельные части. Если нет текста, то выходные данные выводятся без заголовка и формы — выдается только необходимое сообщение о количественном составе. Логическое выражение может отсутствовать только при запросе на формирование сводных таблиц.

Программное обеспечение разработано на языке ассемблера под управлением операционной системы Рафос и может быть использовано для обработки любой информации, хранящейся в справочниках, каталогах, таблицах, карточках.

ИПС «Кадры» и технические средства, обеспечивающие подключение накопителей на магнитных носителях к микроЭВМ «Электроника 60», разработаны в Белорусском филиале Всесоюзного научно-исследовательского и проектно-технологического института по механизации и автоматизации учета и отчетности в народном хозяйстве (ВНИПИУчет) ЦСУ СССР.

220070, г. Минск, пр. Партизанский, 14, телефон для справок: 46-00-13.

Статья поступила 15 декабря 1985 г.

УДК 681.3.

Э. И. Елинер, А. Д. Клименко, Д. А. Костылев

ЛОКАЛЬНАЯ СЕТЬ НА БАЗЕ СЕТЕВОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ АЛИСА

В составе систем автоматизации хозяйственной и административной деятельности используется большое число ЭВМ, что неизбежно приводит к объединению их в сети разного масштаба. Особенно велика потребность в создании локальных вычислительных сетей (ЛВС) мини- и микроЭВМ.

В настоящее время в массовом пользовании находятся мини- и микроЭВМ с наиболее употребляемыми объемами ОЗУ (см. табл. 1).

В табл. 2 представлен примерный набор серийно выпускаемых средств связи для мини- и микроЭВМ серий «Электроника» и СМ, предназначенных в основном для подключения терминалов (за исключением МПД-ПСА и АДС-С) и имеющих максимальную скорость передачи 9600 бод (за исключением ИРПР). Использование таких интерфейсов в локальных сетях приводит к некоторым ограничениям области применения последних, особенно в системах управления, работающих в режиме реального времени.

Большинство из перечисленных в табл. 2 интерфейсов позволяют использовать сетевое программное обеспечение (СПО) DECNET фирмы DEC США. DECNET — достаточно мощная и хорошо развитая система с обширным набором сервисных средств и возможностью работы под управлением практически всех операционных систем (ОС) фирмы DEC. Кроме того, СПО DECNET может обеспечивать функционирование как локальных, так и глобальных сетей. Немаловажное достоинство сети DECNET — наличие встраиваемых средств связи с сетями других фирм, в частности IBM, и возможность поддержки сетевого протокола X25, являющегося международным стандартом. Поэтому область применения СПО DECNET чрезвычайно широка.

Однако необходимо отметить, что использование СПО DECNET в локальных сетях (в особенности однородных, ориентированных на обслуживание технологических и других процессов в режиме реального времени) имеет ряд ограничений, вытекающих непосредственно из перечисленных достоинств этого СПО. Например, универсальность сети (поддержание различных ОС и протоколов) влечет за собой по меньшей мере два ограничения: сложный интерфейс с пользователем и большой объем оперативной памяти. Поскольку сеть DECNET рассчитана на работу под управлением многих ОС и проектировалась в начале 70-х годов, ее интерфейс с пользователем не позволяет применять стандартные средства и утилиты ОС для доступа к сети. Таким образом, пользователь, работающий ранее на ЭВМ без сетевой поддержки, должен изучать специальные команды и правила доступа для работы с сетью.

Таким образом, для развития систем автоматизации требуются более высокоскоростные и качественно новые аппаратные сетевые интерфейсы; актуально создание менее универсального и более ориентированного на работу в рамках производственных систем и локальных сетей СПО, имеющего существенно более простой интерфейс с пользователем, по сравнению с СПО DECNET.

Ведутся разработки различных локальных сетей, одной из которых является однородная локальная сеть на базе ЭВМ СМ и «Электроника» (ВЦКПСМ), созданная в ВЦ СО АН СССР.

Разработка ЛВС ВЦКПСМ велась с учетом следующих основных технических условий: ориентация сети на решение производственных и технологических задач (АСУП и АСУТП) — управление в реальном масштабе времени; простота интерфейса системных средств с пользователем и возможностью применения сетевых средств без модификации уже написанных пользователем программ; системные программные средства, включая ОС, должны занимать не более трети оперативной памяти ЭВМ; обеспечение поддержки не только вновь разрабатываемых, но и серийно выпускаемых средств связи; разрабатываемые аппаратные сетевые интерфейсы должны иметь скорость обмена 10...100 кбайт/с и более; доступность элементной базы и среды передачи для вновь разрабатываемых интерфейсов.

Таблица 1

Совместимые микро- и миниЭВМ

1985		1986	
ЭВМ	ОЗУ, К байт	ЭВМ	ОЗУ, К байт
«Электроника 60»	56	«Электроника 60»	56
ДВК-2М	56	ДВК-3	56
СМ-1300	56	«Электроника МС1211»	248
«Электроника 100/25»	248	БК-0010	24
СМ-4	248	«Электроника 85»	248
СМ-1420	248	СМ-1300.01	248
«Электроника 79»	256	«Электроника 79»	2048

Таблица 2

Коммуникационные устройства

Интерфейс	Скорость передачи информации (макс.), кбайт/с	Расстояние между подключаемым оборудованием (макс.), км	Число ЭВМ
ИРПР	10...30	0,015	2
ИРПС (БС АДС, БИ, УРО)	1	1	2
Адаптер АДС-С	1	1...2	2
Мультиплексор СМ 8514	1	1...2	17
Мультиплексор МДП-ПСА	1	1...2	17
Мультиплексор МПА-1	1	1...2	17
Мультиплексор МПД-А	1	1...2	9

Примечание: среда передачи — витая пара.

Архитектура сети ВЦКПСМ

При организации локальной сети предприятия необходимо учитывать движение потоков информации. Структура подразделений практически любой организации строится по функциональному признаку со стремлением замкнуть поток информации внутри подразделения, отдавая вовне только необходимый минимум (рис. 1, а), составляющий 10...20%. Рассмотрим модель локальной сети с точки зрения распределения потоков данных (рис. 2), которая наиболее естественно ложится на уже имеющуюся иерархию структурных подразделений организации. При этом, будем считать функциональным уровнем модели набор компонент, имеющих похожую структуру и содержание потоков данных.

На уровне I оборудование является конечной ступенью вычислительной техники, непосредственно обращенной к пользователю (АЦ-дисплей, графический дисплей, АЦПУ, графопостроитель) или к объекту (крейт КАМАКА, робот и т. д.). Отличительная черта этого уровня — в некоторых случаях прикладное программное обеспечение реализовано автономно, т. е. вне рамок универсальных ОС.

На уровне II персональная или управляющая микро-ЭВМ (периферийная) со своей ОС позволяет выполнять такие функции, как подготовка программ, группо-

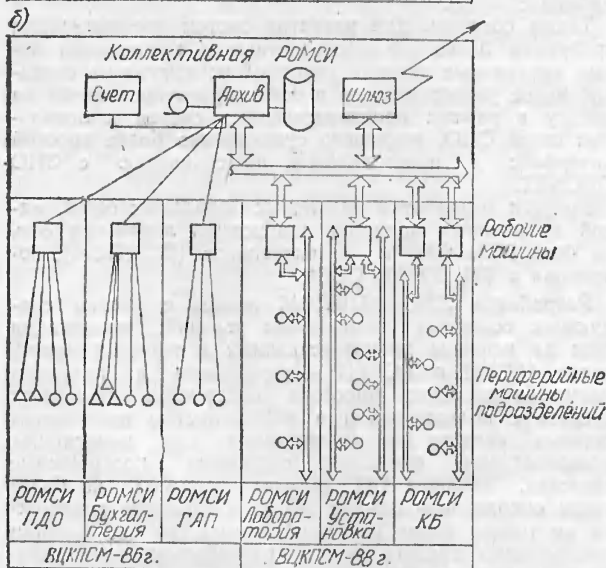
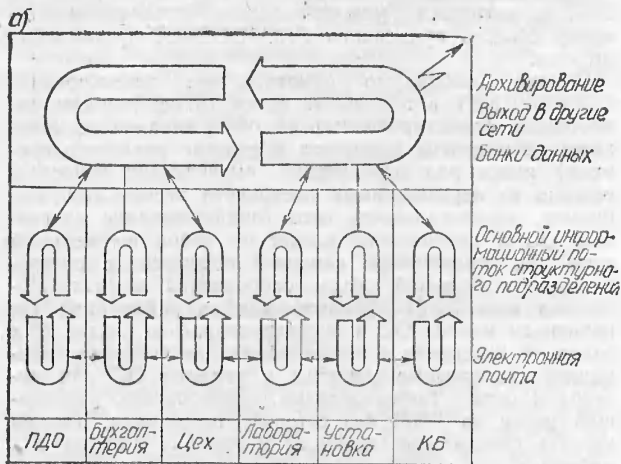


Рис. 1. Распределение информационных потоков и архитектура сети ВЦКПСМ на базе РОМСИ.

вое управление станками, сбор и первичная обработка данных и т. п.

Средства связи между уровнями I—II при этом в ряде случаев могут не иметь больших скоростей и выполняться в соответствии со стандартами ИРПС, ИРПР и стык С-2.

На уровне III мини-ЭВМ (рабочие) имеют внешнюю память и выполняют функции управления подразделениями, обработки данных, а также предоставления внешней памяти и вычислительных ресурсов периферийным ЭВМ. Производительность мини-ЭВМ СМ-4 позволяет объединять не более 5...8 микроЭВМ второго уровня. В зависимости от объема задач, решаемых подразделением, определяется число рабочих машин, которые могут объединяться с помощью линий связи.

При подключении ЭВМ второго уровня к ЭВМ третьего уровня требуются скорости передачи информации на порядок больше. Это связано с большими объемами передаваемой информации, с необходимостью быстрой реакции при использовании ЭВМ второго и третьего уровней в управлении в режиме РВ.

На уровне IV коллективные ЭВМ состоят из наиболее производительных вычислительных машин или машин с большими накопителями на магнитных дисках и лентах. Функционально этот уровень предназначен для надежного архивирования пользовательской информации в рамках всего предприятия, проведения счетных работ в пакетном режиме; он может содержать машины-«шлюзы» для выхода на мощные ЭВМ других семейств или в другие сети.

При выборе типа подключения ЭВМ третьего уровня в ЭВМ четвертого уровня необходимо учитывать большой объем пересылаемой информации и большую частоту обращений со стороны третьего уровня, а также сложность возникающих на этом участке взаимосвязей различных ЭВМ.

Распределенная однородная многомашинная система. Анализируя рассмотренную модель, приходим к выводу, что минимальной конфигурацией, необходимой для успешного обслуживания подразделения, является конфигурация распределенной однородной многомашинной системы (РОМСИ), состоящая из уровней II—III или I—III, поскольку потоки данных в ней могут быть практически замкнуты. С точки зрения пользователя, РОМСИ — это система, принадлежащая данному подразделению и способная автономно выполнять какой-то определенный набор функций. В каждом подразделении возможно существование различных РОМСИ с различной конфигурацией и сочетанием машин (см. рис. 1, б). С точки зрения сетевых функций, РОМСИ представляет собой локальную сеть произвольной конфигурации, в состав которой входят, как минимум, одна рабочая ЭВМ, имеющая внешнюю память, и несколько периферийных ЭВМ (возможно без внешней памяти), обслуживаемых единым СПО.

Организация эксплуатации сети ВЦКПСМ. Сеть ВЦКПСМ — набор РОМСИ, принадлежащих отдельным подразделениям, объединенных в общую сеть, включающую одну или несколько архивных, счетных и, возможно, «шлюзовых» ЭВМ общего пользования, и использующих СПО АЛИСА. Необходимо отметить, что для удобства обслуживания, увеличения надежности и т. д. все рабочие, архивная и счетная машины размещаются в едином зале. Только такое объединение позволяет проводить единую политику по отношению к вычислительной технике предприятия и резко повышает надежность ее использования. Установка всех мини-ЭВМ в одном зале ни в коем случае не должна их обезличивать. На ВЦ СО АН СССР в настоящее время вводится в строй локальная сеть ВЦКПСМ-1 (рис. 3). Как видно из рисунка, это традиционная древовидная иерархическая структура. Подобные сети на базе сетевых интерфейсов DL — Ки/Си и СПО АЛИСА в разном объеме уже действуют во многих организа-

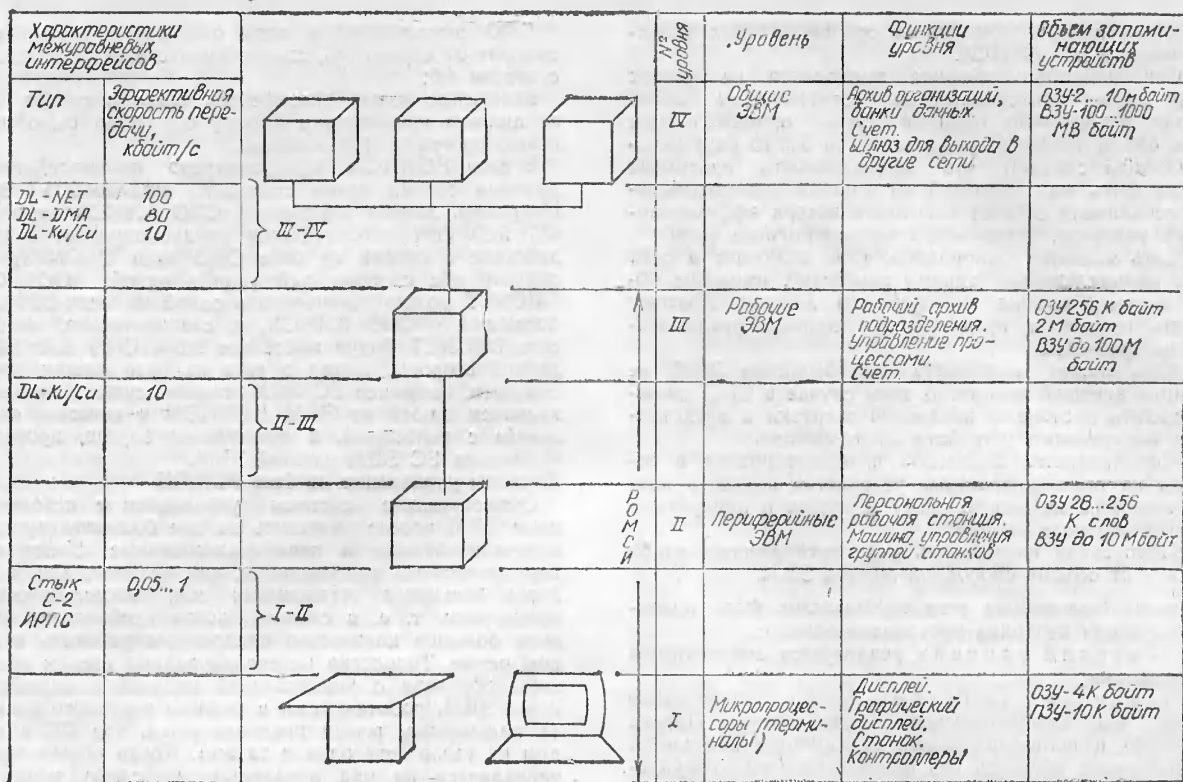


Рис. 2. Функциональная модель сети ВЦКПСМ

ниях СО АН, а также в других организациях. Сетевой интерфейс DL — Ки/Си (СПИ-15) серийно выпускается Министерством приборостроения. Сетевое программное обеспечение ЛВС ВЦКПСМ — АЛИСА

АЛИСА — это СПО, разработанное ВЦ СО АН СССР и Ленинградским СКБ тяжелых и уникальных станков Минстанкопрома для построения распределенных систем автоматизации и управления на базе мини-ЭВМ и микроЭВМ серий СМ и «Электроника».

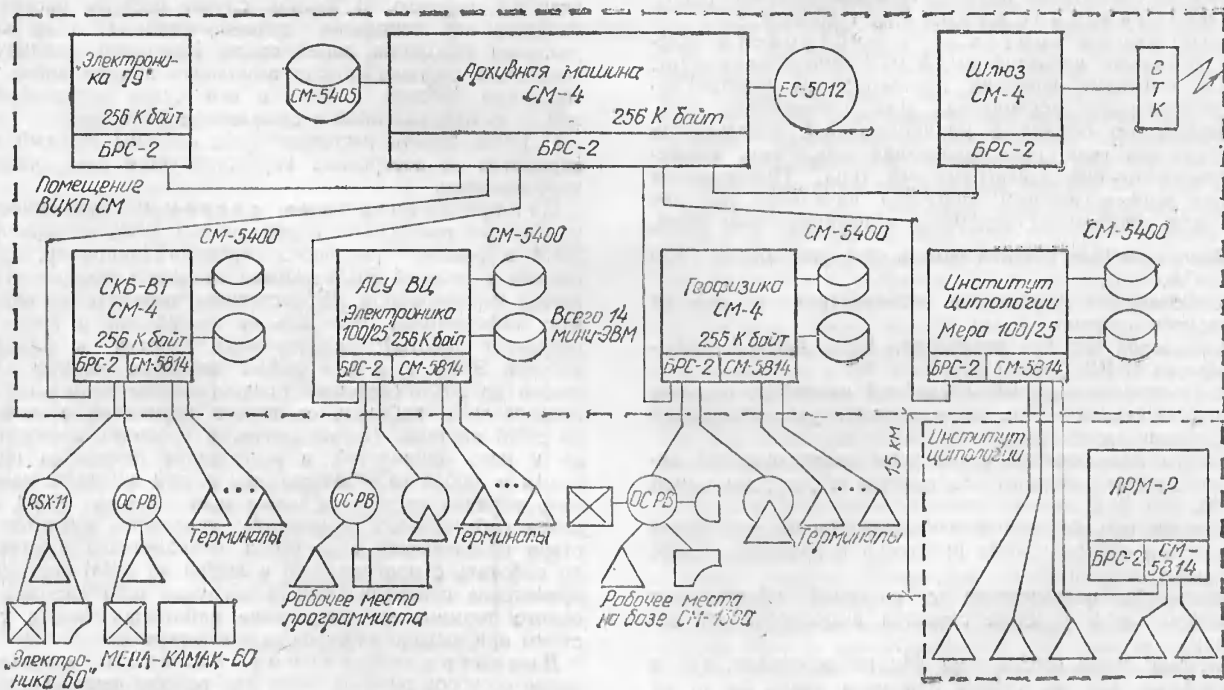


Рис. 3. Локальная сеть ВЦКПСМ

Рассмотрим предпосылки, на основе которых разрабатывалось СПО АЛИСА.

1. Основной объем трафика приходится на участок между двумя непосредственно соединенными ЭВМ. Участки, содержащие один и более промежуточных узлов, объем трафика существенно (в 5... 10 раз) меньше. Отсюда следует, что эффективность протокола должна быть максимальной на участке непосредственного соединения за счет некоторой потери эффективности на участках, содержащих промежуточные узлы.

2. Сеть является однородной, т. е. включает в себя ЭВМ, принадлежащие одному семейству, имеющие общую систему команд и форматы данных. Поэтому уровень протокола представления данных реализовывать не требуется.

3. Сеть может содержать периферийные ЭВМ, не имеющие внешней памяти. В этом случае в СПО должны входить протоколы начальной загрузки и представления виртуальных устройств ввода-вывода.

4. Сеть является локальной и ориентирована в основном на высокоскоростные устройства передачи данных, имеющие надежную среду передачи и аппаратные средства контроля ошибок.

5. Пропускная способность узла сети должна слабо зависеть от объема свободной памяти ЭВМ.

В результате оценки этих предпосылок была принята следующая иерархия протоколов обмена:

физический уровень реализуется аппаратными интерфейсами;

управление каналом — только обнаружение ошибок без автоматического восстановления. Потоки служебной и пользовательской информации разделены. Это позволяет обойтись без промежуточного буферирования. Максимальный размер передачи — 8К байт;

управление потоком — каждая физическая линия делится на фиксированное (до восьми) число логических каналов;

маршрутизация — не требуется на участке непосредственного соединения;

управление сессией — на каждом из логических каналов есть возможность синхронизации и слежения за состоянием пользовательских процессов;

представление данных не требуется;

протоколы виртуальных устройств (диски, терминалы, магнитофоны, АЦПУ, виртуальная сеть). Так выглядит иерархия протоколов на участке непосредственного соединения. Для обеспечения межпрограммного обмена с маршрутизацией используется виртуальная сеть, представляющая собой сеть коммутации сообщений джиттаграммного типа. Применяются также вспомогательный протокол начальной загрузки ОС или автономных программ в периферийные ЭВМ.

Цели, которые преследовались при разработке СПО АЛИСА:

1. Обеспечить управление аппаратурой в режиме реального времени.

2. Базовой ОС для реализации СПО АЛИСА должна быть ОС РВ (RSX-11M).

3. Обеспечить простой и удобный интерфейс системы с пользователем. При этом должны удовлетворяться следующие требования:

доступ пользователя к ресурсам распределенной системы не отличается от его доступа к ресурсам одной ЭВМ;

пользователь без явной необходимости не занимается вопросами синхронизации процессов в различных ЭВМ системы;

интерфейс пользователя с системой обеспечивает ему простые и удобные средства выполнения его задач;

система легко расширяема как по вертикали, так и по горизонтали, не требуя при этом какой бы то ни было модификации программного обеспечения, как пользовательского, так и системного;

СПО выполняется в виде отдельного уровня и не зависит от версии ОС, совместимой с RSX-11M, начиная с версии 4.0;

включение новых интерфейсов связи в состав сети не должно приводить к какому бы то ни было изменению программ пользователя.

В сети ВЦКПСМ предусмотрено взаимодействие с другими сетями путем создания «шлюзовых» ЭВМ. Например, данная реализация СПО АЛИСА на основе ОС RSX-11M автоматически предусматривает взаимодействие с сетями на базе СПО типа DECNET, поскольку обе системы работают в одной ОС. СПО DECNET можно установить в одной из мини-ЭВМ, работающей с СПО АЛИСА, и, следовательно, выход в сеть DECNET будут иметь все ЭВМ СПО АЛИСА. С другой стороны, выход в сети на базе машин других семейств, например ЕС ЭВМ, также осуществляется выделением любой из ЭВМ ВЦКПСМ в качестве «шлюзовой» с эмуляцией, в простейшем случае, протокола терминала ЕС ЭВМ на этой ЭВМ.

Системы управления на базе РОМСИ

Существующие системы управления с использованием ЭВМ можно разделить на две большие группы — централизованные и децентрализованные. Централизованные системы используются, как правило, для управления большими установками или технологическими процессами, т. е. в случаях, когда необходимо заставить большое количество аппаратуры работать в едином ритме. Топология централизованных систем обычно звездобразная с управляющей рабочей и периферийными ЭВМ, работающими в режиме реального времени (в дальнейшем всюду предполагается, что ОС в каждом из узлов сети одна и та же). Когда общая задача распадается на ряд локальных и слабо зависимых друг от друга подзадач, выгоднее применять децентрализованные системы.

Типичный пример децентрализованной системы управления — система управления рабочими местами экспериментатора или инженера-проектировщика. Топология децентрализованных систем не столь сильно тяготеет к звездобразной, хотя такая тенденция и просматривается, при использовании выделенной ЭВМ с внешней памятью. В данном случае система рассматривается как топология потоков данных, а не как реальная топология линий связи. Например, реальную топологию системы можно выполнить в виде шины, а топология потоков данных в ней будет звездобразной, т. е. направленной к выделенной ЭВМ.

С точки зрения распределенной системы каждый из вариантов ее построения характеризуется следующими требованиями.

Централизованные системы: возможность управления процессами периферийных ЭВМ из рабочей ЭВМ в режиме реального времени (например, программа в рабочей ЭВМ должна запустить программу в любой периферийной ЭВМ системы, передать ей входную информацию, отследить ее завершение и принять результат работы); равноправный доступ к файлам рабочей ЭВМ, т. е. все файлы являются общими для любой из ЭВМ системы; предоставление возможности пользователю работать со своего терминала с любой из ЭВМ системы (вмешиваться, в пределах имеющихся у него привилегий, в выполнение любой из программ в любой из ЭВМ системы и, при желании, изменить порядок ее работы, даже если данная ЭВМ не имеет собственного терминала); упрощение взаимодействия пользователя с системой, позволяющей одинаково работать с программами в любой из ЭВМ системы; проведение централизованной загрузки всей системы с одного терминала; сохранение работоспособности системы при выходе из строя ее элементов.

Децентрализованные системы: использование ресурсов рабочей ЭВМ для работы периферийной ЭВМ; защита данных и программ, принадлежащих каждой из периферийных ЭВМ, друг от друга; хранение

ние данных и программ, принадлежащих любой из периферийных ЭВМ, в архиве; работа с терминала периферийной ЭВМ в качестве пользователя рабочей ЭВМ; временное сохранение работоспособности периферийной ЭВМ при выходе из строя рабочей ЭВМ.

Сравнивая перечисленные требования, видно, что децентрализованная система является частным случаем централизованной. В распределенной системе могут существовать узлы, работающие в режиме как централизованной, так и децентрализованной систем. Как правило, в любой реальной системе управления требуется именно комбинация таких возможностей, поскольку всегда существуют элементы, работающие автономно и не требующие постоянной синхронизации.

При создании распределенных систем на базе локальных сетей с ОС реального времени и развитой структурой, таких, например, как RSX-11M фирмы DEC США, существует концепция, позволяющая реализовать большинство из перечисленных выше требований. Это так называемая виртуализация внешних устройств (рис. 4): в одной из ЭВМ системы создается программный имитатор внешнего устройства, имеющий интерфейс, не отличающийся от интерфейса с соответствующим реальным устройством. Этот программный имитатор, называемый также виртуальным устройством, принимает запросы на ввод-вывод от программ в местной ЭВМ и по сети транслирует их в ту ЭВМ, где находится соответствующее реальное устройство. Далее

запрос выполняется на реальном устройстве и его результат транслируется обратно в ЭВМ, в которой находится программа, выдавшая запрос. Запрос завершается от имени программного имитатора.

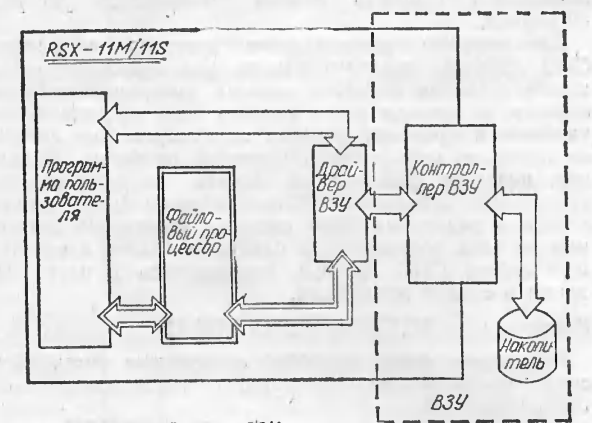
Достоинство такой концепции: простота интерфейса с сетью и однородность СПО для сетей с различной структурой. Виртуальное устройство можно назначать не только на реальное, но и на такое же виртуальное устройство. Это позволяет осуществлять доступ к устройствам сети без применения специальных средств маршрутизации. Наиболее важное достоинство такой системы — возможность использования в распределенной системе уже существующего системного и прикладного программного обеспечения для обычных систем без какой бы то ни было модификации.

Чтобы понять, какие преимущества обеспечивает применение РОМСИ, сравним характеристики систем управления, одна из которых является распределенной, а другая реализована в одной ЭВМ. При этом распределенная система обеспечивает следующие преимущества: получение меньших времен реакции системы управления за счет вынесения процессов (работающих в режиме реального времени) на периферию; повышение надежности системы в целом за счет того, что при выходе из строя периферийных ЭВМ система может сохранять работоспособность; возможность автономной работы периферийных ЭВМ; повышение суммарной вычислительной мощности системы при параллельном выполнении процессов.

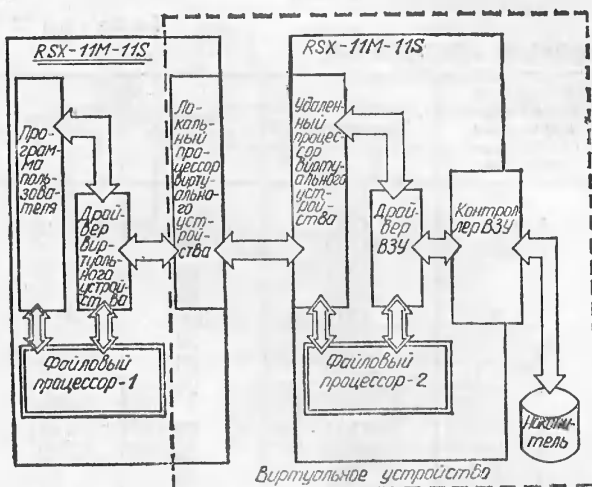
Кроме того, существуют преимущества реализации системы с использованием концепции виртуальных устройств. По сравнению с традиционными методами реализации систем управления, системы с виртуальными устройствами обеспечивают следующие преимущества: уменьшение сложности интерфейса с распределенной системой в целом при использовании уже существующего в любой ОС интерфейса с внешними устройствами; аппаратное наращивание системы без изменения программного обеспечения, поскольку число узлов системы определяется только содержимым таблиц управления устройствами; гибкая перестройка алгоритмов управления и режимов работы, потому что независимость от конфигурации являлась одной из целей при создании системы; использование существующего системного программного обеспечения, так как система реализована на базе одной из наиболее распространенных систем реального времени RSX-11M и, следовательно, можно использовать все написанное для этой системы программного обеспечения; независимость СПО распределенной системы от ОС, позволяющей простую адаптацию его к новым версиям ОС RSX-11M.

Интерфейс с пользователем. Для пользователя однопользовательской системы на базе RSX-11M при переходе на работу в РОМСИ появляются всего две новые команды: разгрузка и загрузка ОС в периферийную ЭВМ. В остальном он работает так же, как и в однопользовательской системе. Кроме того, в состав программного обеспечения распределенной системы входит специальный режим работы с ОС RSX-11M, позволяющий ему работать с командами ОС на уровне экранного редактора. Этот режим дает возможность пользователю набрать и хранить некоторое число команд системы с их экранном редактированием и выполнением в любом удобном ему порядке. Он обеспечивает также автоматическую генерацию последовательности команд для подготовки программ и выполнения этих команд в автоматическом режиме.

Объем памяти. Суммарный объем памяти, занимаемый СПО АЛИСА и ОС RSX-11M в периферийной ЭВМ системы, не имеющей диспетчера памяти («Электроника 60») и ориентированной на подготовку программ, составляет 10К слов. Если периферийная ЭВМ является узлом в сети с маршрутизацией, то суммарный объем памяти под системное программное обеспе-



Принцип работы с ВЗУ



Принцип работы с усиленным ВЗУ

Рис. 4. Виртуализация внешних устройств

чение увеличивается до 13...14К слов. Если периферийная ЭВМ ориентирована на специализированное использование, не требующее полных возможностей системы, объем памяти уменьшается до 7...8К слов.

Скорость обмена определяется скоростью работы линии связи. Потери скоростей из-за программных издержек существенны только для линии связи с обменом по программному каналу. Среднее время реакции на обработку прерывания составляет около 500 мкс. Следовательно, если аппаратная скорость канала составляет менее 500 мкс на передачу единицы информации (байт или слово), то реальная скорость не превысит 2К байт для каналов с байтовым обменом и 4К байт для каналов со словным обменом вне зависимости от аппаратной скорости канала. Для быстрых интерфейсов, работающих по программному каналу, дальнейшее увеличение скорости связано с введением оптимизации и уменьшением числа прерываний для передачи порции информации. Если канал связи работает в режиме прямого доступа к памяти, скорость обмена определяется только аппаратной скоростью канала, программные издержки составляют доли процента. В качестве примера можно привести измерение скорости передачи для интерфейсов, работающих по программному каналу: DL-11 — аппаратная скорость 9600 бод, скорость передачи при размере блока 512 байт — 700...800 байт/с; DL-Ки/Си — аппаратная скорость 500 кбод, скорость передачи при размере блока 512 байт — 10 000...20 000 байт/с (словный обмен).

Использование стандартного ПО. Программное обеспечение системы связи реализовано таким образом, что позволяет избежать каких бы то ни было модификаций не только пользовательского, но и системного программного обеспечения RSX-11M. Это значит, что благодаря применению виртуальных устройств пользователь может работать в распределенной системе точно так же, как и в обычной одомашинной системе. Если, например, программа, находящаяся в периферийной ЭВМ, работала с дисками, которые были подключены к этой же машине, то в распределенной системе на периферийной машине она будет работать точно так же, используя диски рабочей ЭВМ.

Перспективы. Развитие СПО АЛИСА продолжается в следующих направлениях: развитие подсистем виртуальных устройств. Использование подсистемы виртуальных дисков для доступа к реальным дискам любой ЭВМ, входящей в сеть; упрощение и повышение уровня сервиса для интерфейса с пользователем; поддержка других ОС (RT-11, UNIX, PC/DOC); поддерж-

ка новых интерфейсов связи; развитие возможностей для динамического распределения ресурсов в рамках ВЦ предприятия.

Самым важным из этих направлений, на наш взгляд, является повышение уровня сервиса для пользователя. Отсутствие удобного и простого в запоминании интерфейса с пользователем — недостаток многих хороших систем, и разработчики СПО АЛИСА уделяют этой проблеме много внимания. Одним из путей развития в этом направлении является создание для пользователя возможности применять свой, наиболее удобный для его целей интерфейс с системой. Другая возможность — это расширение уже существующего интерфейса для повышения его мобильности и появления средств самообучения.

Что касается поддержки новых интерфейсов связи, то в СПО АЛИСА процедура программирования нового интерфейса локализована так, что программы пользователя и все системное программное обеспечение более высоких уровней не претерпевает при этом никаких изменений. Для новых интерфейсов связи необходимо наличие возможности работы в режиме прямого доступа к памяти. Это повышает работоспособность системы в десятки раз, приводит к значительному уменьшению загрузки процессоров системы и уменьшает время реакции на события в режиме реального времени. Разработка одного из таких интерфейсов (DL — DMA) проведена на ВЦ СО АН СССР. С его появлением скорость обмена увеличилась до 70...80 кбайт/с.

Динамическое распределение ресурсов в рамках СПО АЛИСА позволит решить ряд проблем, касающихся создания пользовательского программного обеспечения, и прежде всего сделать этот процесс более удобным и простым, скрывая от пользователя детали, не имеющие отношения к основной проблеме. Например, появится возможность писать пользовательские программы, отдельные части которых будут выполняться в различных ЭВМ системы. Появление некоторых из этих возможностей следует ожидать в следующей версии СПО АЛИСА, находящейся в настоящее время в стадии разработки.

Развитие сети ВЦКПСМ — ВЦКПСМ-II

Рассмотрим более подробно аппаратные интерфейсы связи, чтобы точнее представить техническую базу ЛВС ВЦКПСМ.

Сеть ВЦКПСМ-II строится на базе РОМСИ, основанных на шинной структуре с применением метода

Таблица 3

Характеристики сетевых интерфейсов ЛВС ВЦКПСМ

Тип	Среда передачи	Скорость передачи информации, кбайт/с	Расстояние между подключаемым оборудованием, км	Конструктив	Число элементов	Число ЭВМ
ИРПР	Витая пара	20	0,015	ОШ	60	2
ИРПР	То же	20	0,015	МПИ	60	2
ИРПС (БС, АДС, БИ)	" "	1	1	ОШ	60	2
ИРПС (УПО)	" "	1	1	МПИ	60	2
Мультиплексор CM 8514	" "	1	1...2	ОШ	300	17
СПИ-15 (DL-Ки/Си)	2PK	20	1	ОШ	90	2
СПИ-15 (DL-Ки/Си)	2PK	10	1	МПИ	60	2
СПИ-15 (DL-Ки/Си)	2PK	20	1	Евростандарт	90	2
Разрабатываемые						
DL-DMA-U	1PK	80	1	ОШ	90	2
DL-DMA-Q	1PK	80	1	МПИ	70	2
DL-NET-U	1PK	4...60	1	ОШ	80	32
DL-NET-Q	1PK	4...60	1	МПИ	80	32
DL-NET-C2	1PK	4...60	1	МПИ	80	32

доступа с передачей маркера. В качестве шины используется 50-омный коаксиальный кабель (РК) с параллельным подключением сетевых интерфейсов. Скорость передачи равна 1 Мбит/с.

Предполагается, что сеть ВЦКПСМ-II вначале будет включать три типа сетевых интерфейсов: сетевой интерфейс для подключения машин с шиной ОШ (DL-NET-U) — СМ-4, «Электроника 100/25», «Электроника 79», СМ-1700, СМ-1600, СМ-1300, СМ-1300.01; сетевой интерфейс для подключения машин с шиной МПИ (DL-NET-Q) — «Электроника 60», «Электроника МС 1211», ДВК-2М, ДВК-3, ДВК-4; станция для подключения к сети до восьми устройств с интерфейсом стык С2 (DL-NET-C2).

Все сетевые интерфейсы сети ВЦКПСМ строятся на основе однокристального микропроцессора К1801ВМ2, имеющего систему команд, совместимую с системой команд вычислительных машин СМ и «Электроника». Функциональные схемы сетевых интерфейсов ЛВС ВЦКПСМ-II приведены на рис. 5.

Использование в сетевых интерфейсах DL-NET U, DL-NET-Q и DL-NET-C2 сети ВЦКПСМ-II мощного микропроцессора с системой команд, аналогичной системе команд ЭВМ «Электроника» и СМ, позволит существенно разгрузить процессор ЭВМ и переложить на сетевой интерфейс выполнение большого объема функ-

ции по управлению сетью и взаимодействию с ОС ЭВМ.

Метод доступа с передачей маркера позволяет: обеспечивать при большой нагрузке высокую степень использования шины; предоставить равные права всем машинам сети РОМСИ; предоставить гарантированное время доступа к сети; с помощью изменений программы управления сетью создавать РОМСИ с различной необходимой ее пользователям интенсивностью передачи.

При построении локальной сети можно воспользоваться тем, что потоки информации в организации имеют тенденцию к локализации в структурных подразделениях. На наш взгляд, использование одного моноканала для объединения всех пользователей организации в единую сеть не является целесообразным. Разбиение сети на несколько связанных между собой подсетей подразделений дает возможность получать высокие эффективные скорости при значительно меньших аппаратных затратах. При этом упрощается сетевой интерфейс, можно использовать для каналов связи обычный РК кабель и сократить длину некоторых сегментов сети, а также число подключаемых машин на одном элементе сети может быть ограничено.

Все эти факторы приводят к тому, что становится возможным уже сегодня создать дешевую и эффективную локальную сеть на имеющейся в стране элементной базе.

Телефон для справок: 35-11-66, г. Новосибирск.
Статья поступила 23 января 1986 г.

УДК 621.394.74

О. В. Цвелодуб, Н. Н. Щелкунов

КОНТРОЛЛЕР ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ

Рассматривается сеть шинной конфигурации с пропорциональным доступом к каналу, согласно которому каждый абонент в порядке строгой очереди получает временной интервал (сегмент) переменной длины. При сегментированном управлении доступом предполагается динамическое распределение ресурсов сети, когда абонент занимает канал только при необходимости передать пакет, а благодаря строгой очередности передачи пакетов время ожидания доступа для каждого абонента ограничено. В сети используется децентрализованный принцип синхронизации по концу передаваемого пакета.

Предлагается контроллер для локальной сети магистрального типа, в функции которого входит формирование пакета, контроль ошибок при приеме, управление доступом к каналу и обменом на физическом уровне.

Архитектура контроллера. В состав контроллера (рис. 1) входят 8-разрядный микропроцессор (МП), контроллер прерываний (КП), периферийный связной адаптер (ПСА) КР580ВВ51, программируемый интервальный таймер (ПИТ) КР58СВИ53, а также модули ОЗУ, ПЗУ и устройства сопряжения с абонентом.

Цикл обмена информацией между абонентами происходит следующим образом. Подготовленное для отправки сообщение абонент записывает в находящийся в ОЗУ выходной буфер контроллера и указывает ему направление передачи. Контроллер формирует пакет, добавляя заголовок со служебной информацией и контрольную сумму. Затем он в своем временном сегменте в последовательном формате передает пакет через ПСА в сеть.

Принятый пакет освобождается от заголовка и помещается во входной буфер контроллера, а его адрес и размеры указываются абоненту. Если при приеме пакета была обнаружена ошибка, то в управляющем блоке входного буфера, отражающем его состояние,

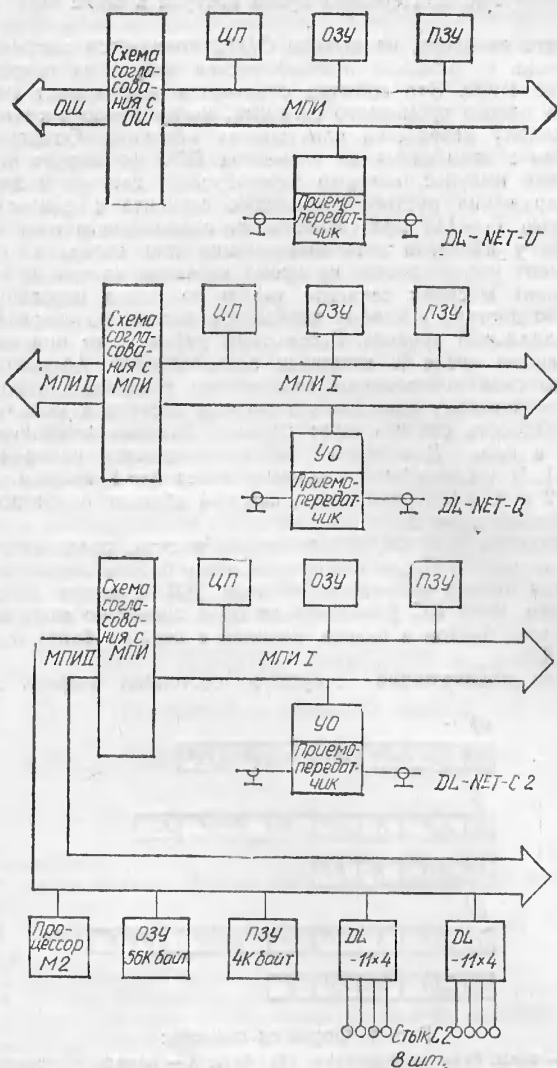


Рис. 5. Функциональные схемы сетевых интерфейсов

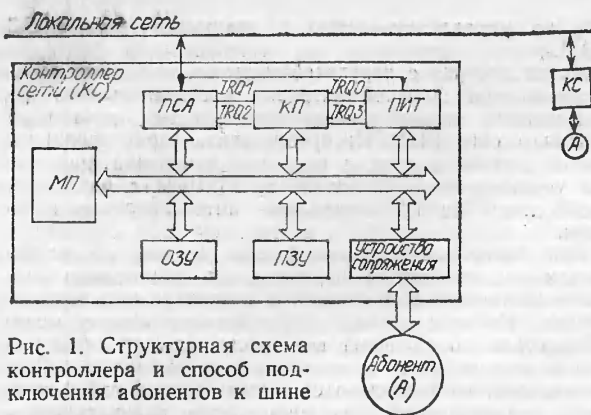


Рис. 1. Структурная схема контроллера и способ подключения абонентов к шине

устанавливается флаг ошибки. Тип абонента может быть произведен: от НГМД и ЭЛТ-контроллера до микро- и мини-ЭВМ, а взаимодействие с ним определяется устройством сопряжения.

Контроллер можно построить на базе системы пользователя с разделяемыми ресурсами. Тогда для его реализации необходимы лишь ПИТ и ПСА, подключаемые к внутрисистемной магистрали пользователя (рис. 2). ПИТ позволяет полностью без МП осуществить синхронизацию пересылок и определить границы временных сегментов. В этом случае ПСА при приеме и передаче символа обслуживается по прерываниям ИРQ2 и ИРQ1 соответственно. Освободившееся время работы МП можно использовать для реализации прикладных функций абонента.

Физический уровень. Сетевая шина (см. рис. 2), объединяющая выходы всех контроллеров по схеме «монтажное ИЛИ», представляет собой одну экранированную сигнальную линию. Абоненты располагаются в пределах нескольких смежных лабораторий.

Адаптер ПСА программируется для работы в асинхронном режиме с временем передачи одного бита T , равным 16 периодам сигнала на ТХС входе. Передает-

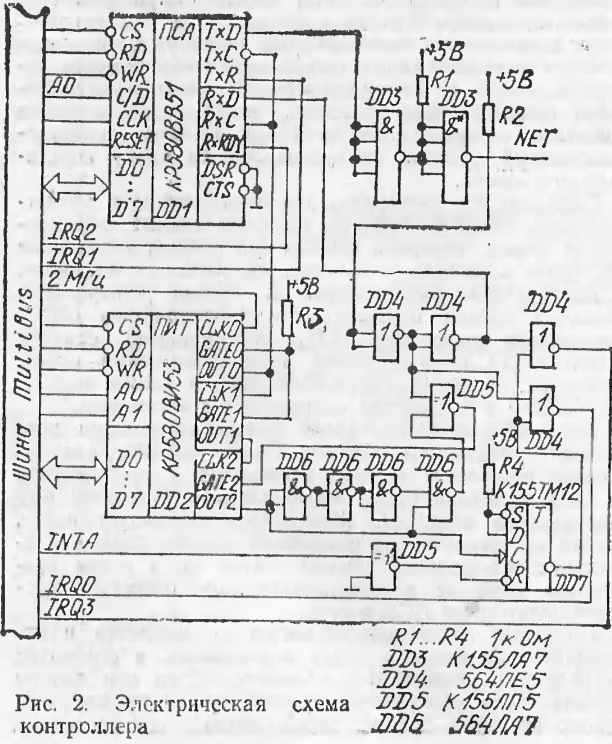


Рис. 2. Электрическая схема контроллера

- R1...R4 1к Ом
- DD3 K155/PA7
- DD4 564/IE5
- DD5 K155/IN5
- DD6 564/PA7

ся старт-бит, восемь бит данных и один стоп-бит [2]. Счетчики 0 и 1 ПИТ программируются для работы в режиме 2 (генератор скорости). При этом счетчик 0 задает опорную частоту приема и передачи для ПСА, равную 153,6 кГц, что соответствует скорости передачи данных 9,6 кбод. Счетчик 1 загружается числом 16, чтобы период сигнала на его выходе был равен T .

Счетчик 2 настраивается на работу в режиме 1 (программируемый одновибратор). Его управляющий вход GATE2 фактически через инвертор подсоединен к шине, и каждый принимаемый старт-бит будет перезапускать счет. В счетчик загружается число 11. Тогда в течение передачи всего пакета на выходе OUT2 будет низкий уровень напряжения (рис. 3). Когда передача

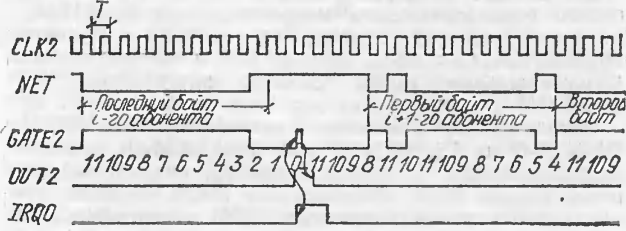


Рис. 3. Передача права доступа к шине

пакета окончена, на выходе OUT2 появляется высокий уровень напряжения и генерируется запрос на прерывание ИРQ0. Это событие отмечается как момент начала нового временного сегмента, когда право доступа к каналу передается следующему абоненту. Одновременно с этим схема на элементах DD6 формирует короткий импульс, который перезапускает счетчик 2 для обнаружения пустого временного сегмента с длительностью $T_0 = 11T$ или сегмента, не содержащего пакета. Если у абонента есть информация для передачи, то сегмент увеличивается на время передачи пакета. Если абонент в своем сегменте так и не начал передачу, право доступа к каналу передается дальше по очереди.

Канальный уровень. Рассмотрим работу сети при неизменном числе N активных пользователей. Абоненты через свои контроллеры подключены к общей сетевой шине, каждому известна его очередь доступа к каналу. Очередность соответствует порядку включения абонентов в сеть. Для этого им присваивается номер k , $k=1...N$ (включившийся первым, имеет $k=1$, вторым — $k=2$ и т. д.). Кроме этого, каждый абонент имеет постоянный адрес A .

Форматы пакетов, передаваемых в сети, представлены на рис. 4. Число информационных байтов ограничивается только разрядностью поля ДЛ и может быть любым. Байт КС дополняет до нуля сумму по модулю 256 всех байтов в пакете, начиная с первого байта поля ДЛ.

Для поддержания текущего состояния очереди и,

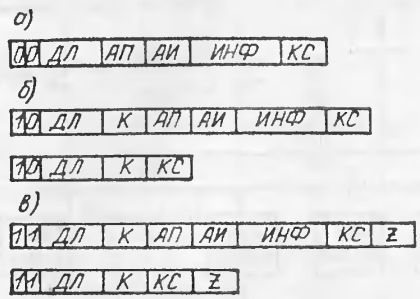


Рис. 4. Форматы пакетов:

ДЛ — число байт информации (14 бит); К — номер источника (8 бит); АП — адрес приемника (8 бит); АИ — адрес источника (8 бит); ИНФ — передаваемая информация (ДЛx8 бит); КС — контрольная сумма (8 бит); Z — байт с содержимым ООИ

своего места в ней все контроллеры отслеживают две переменные n и m . В начале каждого сегмента эти переменные анализируются, затем одна или обе модифицируются. В любой момент времени каждый контроллер имеет свою уникальную комбинацию значений n и m . Право доступа принадлежит абоненту с $n=1$. Если у него $m \neq 1$ и есть информация для передачи, он может послать ее простым пакетом с форматом, представленным на рис. 4,а. Пакеты с форматами, показанными на рис. 4,б, в, называются флаг-пакетами (ФП). Они используются для безусловной синхронизации контроллеров и посылаются абонентом с $n=m=1$ независимо от того, есть у него информация для передачи или нет. Если ФП не содержит информации, то в поле ДЛ записываются нули. ФП с форматом (см. рис. 4,в) посылается абонентом, который включился в сеть последним (т. е. с $k=N$) и содержит в себе информацию о числе активных пользователей сети. Тип формата однозначно определяется двумя первыми битами пакета. Поле z необходимо для бесконфликтного включения в сеть новых абонентов.

Основной алгоритм работы контроллера. По прерыванию $IRQ0$ (конец сегмента):

1. Если $n \neq 1$, то выполнить $n := n - 1$, затем разрешить прием данных через ПСА. Выход из прерывания.

2. При $n=1$ и $m \neq 1$ выполнить $m := m - 1$, $n := N$. Если есть информация для передачи, то сформировать простой пакет и перейти к п. 4, если информации нет, то выход из прерывания.

3. При $n=m=1$ установить $n=N$, $m=N+1$, сформировать флаг-пакет, проверить, не выключился ли кто-либо из сети.

4. Разрешить передачу данных через ПСА. Выход из прерывания.

По прерыванию $IRQ1$ (передатчик пуст):

1. Загрузить ПСА очередной байт пакета.
2. Если весь пакет уже передан, то запретить передачу данных через ПСА.

3. Выход из прерывания.

По прерыванию $IRQ2$ (символ принят) — прием пакета:

1. Принять поля ДЛ, АП и К, если они есть.
2. Проверить, не выключился ли кто-либо из сети.
3. По полю АП проанализировать, кому адресован пакет. Если адрес АП совпадает с адресом абонента, то вести прием пакета до его окончания.
4. Запретить прием данных через ПСА.

Немного изменив алгоритм, легко организовать обмен с квитированием, посылая подтверждение безопоздочного приема непосредственно после принятого пакета. Очередность пуска ФП при $N=3$ поясняется на рис. 5, где изображен случай фактического бездействия всех пользователей.

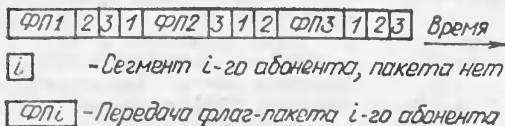


Рис. 5. Пример минимальной загрузки сети

Включение / выключение абонентов из сети. При включении в сеть абонент выполняет следующий алгоритм:

1. Выдержать паузу, пропорциональную своему адресу.

2. Дождаться посылки ФП с форматом рис. 4,а.

3. Проанализировать его поле ДЛ и, ведя отсчет принятым байтам, зафиксировать момент времени t начала передачи z -байта.

4. В момент времени $t+T_1$ начать передачу z -байта, причем $3T < T_1 < 8T$.

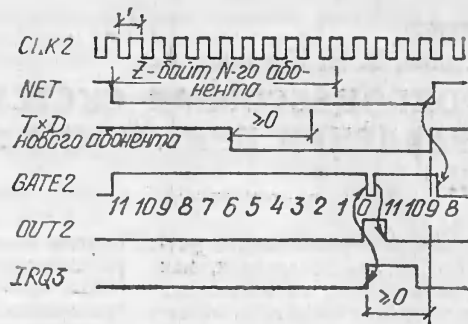


Рис. 6. Включение в сеть нового абонента

5. Инициировать у себя переменные N, k, n, m : $N = k = n = N + 1, m = 1$. При этом сегмент абонента с $k=N$ не удлиняется, но все контроллеры вместо прерывания $IRQ0$ активизируют запрос на прерывание $IRQ3$, так как в момент смены сегментов в сети будет низкий уровень сигнала (рис. 6). Физически тип прерывания определяется схемой на элементах DD4 (см. рис. 2). По прерыванию $IRQ3$ каждый контроллер инкрементирует у себя N и m , а основной алгоритм не выполняется. На этом процедура включения в сеть заканчивается.

Если при выполнении п. 2 абонент в течение T_2 не обнаружил изменение уровня сигнала в шине, то это означает, что в сети никакие устройства не работают, и он получает номер $k=1$. $T_2 = (N_{max} + 1) \times T_0$, где N_{max} — максимально возможное число пользователей сети. Пункт 1 обеспечивает бесконфликтное включение абонентов в сеть при ее перезапуске, например, при кратковременном отсутствии питания.

Выключаются абоненты из сети асинхронно, поэтому факт их выключения обнаруживается не сразу. Пусть отключился i -й абонент ($i \neq N$). Первым это заметит $i+1$ -й абонент, когда придет его очередь посылать ФП и выяснится, что предыдущий ФП передал не i -й абонент. Тогда он у себя уменьшает на единицу числа N, k, n, m , а ФП передает, как будто ничего не произошло. Остальные абоненты также контролируют очередность передачи ФП и, заметив пропуск ФП с $k=i$, выполняют $N := N - 1$, а абоненты с $k > i$ также $k := k - 1$. Отключение абонента с $k=N$ первым заметит абонент с $k=1$. Дальнейшая конкретизация алгоритма реализуется в кодах микропроцессора.

Заключение. Наиболее близкой по своей организации к данной сети является локальная сеть фирмы Motorola [3]. Основное отличие состоит в том, что флаг-пакет посылается в $N+1$ раз реже и при низкой загрузке канала значительно сокращается время ожидания доступа. Поэтому данная сеть эффективна при использовании в масштабе нескольких лабораторий с общим числом абонентов, не превышающим 30. Максимальное число пользователей сети зависит от выбора T_0 и T_1 . Так как сеть полностью децентрализована, она остается работоспособной при выходе из строя любого числа абонентов.

Телефон для справок: 408-62-44, г. Москва

ЛИТЕРАТУРА

1. Прангишвили И. В., Подлазов В. С., Стецюра Г. Г. Локальные микропроцессорные вычислительные сети. — М.: Наука, 1984. — С. 66—77.
2. Балашов Е. П., Григорьев В. Л., Петров Г. А. Микро- и мини-ЭВМ. — Л.: Энергоатомиздат, 1984. — С. 135—161.
3. Скавеззи Д. Сегментированное управление доступом узлов к ресурсам локальной сети // Электроника. — 1981. — Т. 54. — № 12. — С. 51—59.

Статья поступила 22 мая 1985 г.

В. М. Чмиль, Б. И. Ющенко

МИКРОПРОЦЕССОРНАЯ СИСТЕМА КОНТРОЛЯ И УПРАВЛЕНИЯ ПРИЕМНЫМИ УСТРОЙСТВАМИ СВЯЗИ

Маломощные усилительные устройства (МШУ) располагаются, как правило, на антеннах, на расстоянии 50...70 м от пульта оператора, поэтому недоступны в процессе сеанса связи. Основные параметры МШУ должны подстраиваться непосредственно перед сеансом связи, а контроль работоспособности вестись непрерывно. При обнаружении неисправности решение о переходе на резерв должно быть принято за 0,3...0,5 с. Эти требования являются определяющими при выборе системы контроля и управления (СКИУ) основным и резервным МШУ.

Наиболее информативным является контроль основных параметров МШУ: температуры шума и коэффициента усиления, однако требования к времени наработки на отказ (не менее 20 000 ч) и надежность элементов позволяют создать СКИУ с контролем только косвенных параметров (~40 параметров). Рассматриваемая система представляет собой специализированное средство контроля и управления на базе микропроцессорной БИС. МШУ является устройством с 100%-ным резервированием (два идентичных канала), следовательно, должна быть обеспечена устойчивая работа каждого канала в отдельности. Поэтому целесообразно создавать идентичные СКИУ для обоих каналов. С учетом необходимости дистанционного управления каналами и передачи информации о работоспособности и причинах

выхода из строя на пульт оператора система контроля и управления МШУ приняла вид двухуровневой трехпроцессорной СКИУ со специализированными микроконтроллерами на базе БИС серии К580 (рис. 1).

Структура и принцип работы

Основным устройством системы контроля и управления МШУ является блок обработки информации (БОИ), обеспечивающий сбор аналоговой информации от датчиков, преобразование информации в цифровой вид, анализ состояния МШУ по заданным критериям, выдачу информации оператору в блок управления и индикации (БУИ), выдачу команд на исполнительные элементы МШУ (волноводные переключатели, микрокриогенная система, ЦАП в устройстве предварительной обработки (УПО)).

БОИ представляет собой специализированный микроконтроллер, выполненный на микропроцессорных БИС серии К580 (без шинных усилителей). Блок содержит ПЗУ емкостью 12К байт на микросхемах К573РФ5, ОЗУ емкостью 2К байт — К132РУ2А, ЭРПЗУ — К1601РР1. В ЭРПЗУ запоминаются необходимые уровни изменяющихся в процессе настройки параметров МШУ (необходимость настройки МШУ перед сеансом связи определяется свойствами собственно МШУ и не зависит от системы контроля). Это позволяет восстанавливать их после выключения питания

или отказа устройства. Связь БОИ с УПО и БУИ производится по параллельному и последовательному интерфейсам, собранным в виде отдельных плат на микросхемах серий К580 и К133. Связь БОИ с микрокриогенной системой (МКС) и внешними волноводными переключателями осуществляется через соответствующие интерфейсные платы. В состав БОИ включены: АЦП с программно-управляемым усилителем постоянного тока, позволяющие нормировать разнородные аналоговые сигналы от датчиков; специализированный пульт управления с кнопками-индикаторами на светодиодах и знакосинтезирующих светодиодных матрицах. Наличие собственного индикатора контролируемых параметров и специализированного пульта управления параметрами позволяет оператору выполнять настройку и визуальный контроль состояния МШУ непосредственно с БОИ без применения стандартной измерительной аппаратуры при проведении профилактических работ.

Устройство первичной обработки, состоящее из четырех ЦАП, коммутатора и параллельного интерфейса связи с БОИ, представляет собой устройство связи с объектом (собственно МШУ). Выполняет функции преобразования цифровых управляемых сигналов в аналоговые и осуществляет последовательный опрос датчиков с выводом аналоговых сигналов через коммутатор на микросхемах серии К543КНЗ на АЦП БОИ.

Комплекс, состоящий из одного канала МШУ, УПО и БОИ, может функционировать автономно, обеспечивая контроль и управление работой МШУ. Однако, с учетом потребности оценки работоспособности для обеспечения возможности управления МШУ на расстоянии 50...70 метров, в МШУ включен блок управления и индикации (БУИ), представляющий собой специализированный микроконтроллер, состоящий из плат МП, ПЗУ, ОЗУ и последовательного интерфейса связи (аналогичных БОИ), специализированного пульта управления (аналогичного пульта управления блока БОИ с учетом управления двумя каналами) и индикатора на ИМГ со специализированным интерфейсом связи. Основные задачи, решаемые БУИ: взаимный обмен информацией между БОИ основного и резервного каналов при автоматическом режиме работы изделия, обеспечение возможности общения оператора с МШУ при управлении и получении информации о работоспособности изделия в процессе эксплуатации либо о причинах выхода из строя узлов и блоков.

При пониженных требованиях к массогабаритным характеристикам и условиям эксплуатации блок может быть заменен любой стандартной микроЭВМ, поддерживающей круглосуточный режим работы, но при этом

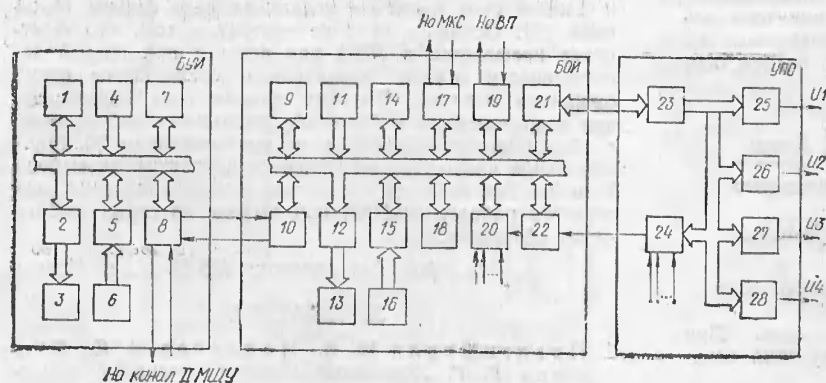


Рис. 1. Структурная схема канала СКИУ:

1, 9 — плата микропроцессора; 2 — интерфейс ИМГ; 3 — индикатор матричный газоразрядный (ИМГ); 4, 11 — ПЗУ; 5 — интерфейс управления БУИ; 6 — пульт управления БУИ; 7, 14 — ОЗУ; 8, 13 — последовательный интер-

фейс связи; 12 — интерфейс индикатора; 13 — индикатор БОИ; 15 — интерфейс управления БОИ; 16 — пульт управления БОИ; 17 — интерфейс связи с МКС; 18 — ЭРПЗУ; 19 — интерфейс связи с волноводными переключателями; 20 — АЦП; 21, 23 — параллельный интерфейс связи; 22 — усилитель постоянного тока; 24 — аналоговый коммутатор; 25, 26, 27, 28 — ЦАП

потребуется разработка дополнительного программного обеспечения и усложнится работа оператора.

Программное обеспечение

Для минимизации объема результирующей программы работы БОИ и БУИ программирование выполнено на языке ассемблера. Программа работы БОИ и БУИ включает основ-

ную программу, выполняющую функции диспетчера, и несколько подпрограмм (рис. 2, 3). Основные программы БОИ и БУИ состоят из самостоятельных блоков и операций перехода на специализированные подпрограммы, выполняющие следующие функции:

«Контроль пульта» — обзор состояния органов управления пульта в каналах и передача управления на соответствующую подпрограмму обслуживания органов управления пульта либо возврат системы при отсутствии команды с пульта управления к последующей операции;

«ЦАП» — посылка информации в четыре 10-разрядные ЦАП;

«Обзор» — установка аналогового коммутатора УПО в положение для последовательного измерения параметров, вызов подпрограммы АЦП, измерение;

«АЦП» — запуск АЦП, измерение и усреднение результатов измерения. Периодически производит коррекцию дрейфа нулевого уровня и коэффициента усиления;

«Измерение» — устанавливает путь прохождения аналогового сигнала через аналоговые мультиплексоры и необходимый коэффициент усиления программно-управляемого усилителя постоянного тока;

«Сравнение» — сравнение величины параметра с границами допусков. При выходе за границы в ОЗУ записывается признак аварийности параметра, который воспринимается подпрограммой «Индикация» и выводится на индикацию;

«Индикация» — вывод на индикацию величины контролируемого параметра, задаваемого с пульта управления или подпрограммы сравнения;

«ВП» — переключение входного и выходного волноводных переключателей (переключение на резервный канал). Вызывается либо подпрограммой «Контроль пульта», либо в процессе выполнения программы;

«Анализ» — прием аварийного сигнала основного и резервного каналов, сравнение состояния каналов (исправен — неисправен резервный канал при аварии основного; неисправен — исправен основной канал при аварии резервного);

«Таймер» — вспомогательная подпрограмма, вызываемая каждые 10 с. Предназначена для подсчета времени наработки, периодического обновления информации ПЗУ. Определяет необходимые задержки при включении и выключении МКС.

Ряд подпрограмм предназначен для обслуживания органов управления пульта при непосредственной работе оператора с подпрограммой «Контроль пульта». Подпрограммы «ПУ1», «ПУ2», «СМ», «РН» выводят на индикацию и устанавливают в необходимые положения порты связи с управляемыми элементами для проведе-

ния операций настройки. «Большее», «меньшее» регулирует увеличение или уменьшение настраиваемого параметра. «Запись» производит запись в ЭРПЗУ установленной при настройке величины параметра. «Сброс» устанавливает выходное напряжение ЦАПов в соответствии с данными, записанными в ЭРПЗУ. «У1», «У2», «ТХ», «ТС», «ДВ», «ТН», «ТК» передают управление подпрограмме «Индикация» заданного параметра. «ГН1», «ГН2», «МКС» осуществляют последовательность операций включения и выключения соответствующих узлов и блоков. «Индикация отключения» — включение и выключение вывода текущей индикации на пульт управления работой во взаимосвязи с подпрограммой «Индикация».

Достаточно большое число специализированных подпрограмм позволяет оптимизировать объем программ БОИ и БУИ, добиться гибкости при доработке программ в случае модернизации изделий, максимально унифицировать программное обеспечение БОИ и БУИ. Создание автоматической СКИУ на базе микропроцессорных БИС позволило решить проблему контроля параметров МШУ в процессе эксплуатации, снизить эксплуатационные затраты на контроль работоспособности и трудоемкость профилактических работ, получить новые качества МШУ в целом: прогнозирование отказов, устойчивость к пропаданию напряжения питания, снижение потерь информации при выходе из строя одного из каналов приема.

Разработка специализированного микроконтроллера дает возможность существенно снизить массогабаритные характеристики СКИУ МШУ в целом, решить задачу обеспечения помехоустойчивости, что в достаточной степени затруднено при применении универсальных средств микропроцессорной техники. Специализированные пульты управления позволяют снизить требования к квалификации обслуживающего персонала, что при широком применении указанных устройств имеет немаловажное значение.

Телефон для справок: 488-61-28,
г. Киев, Чмель Владимир Моисеевич,
после 19 часов.

Статья поступила 18 августа 1985 г.

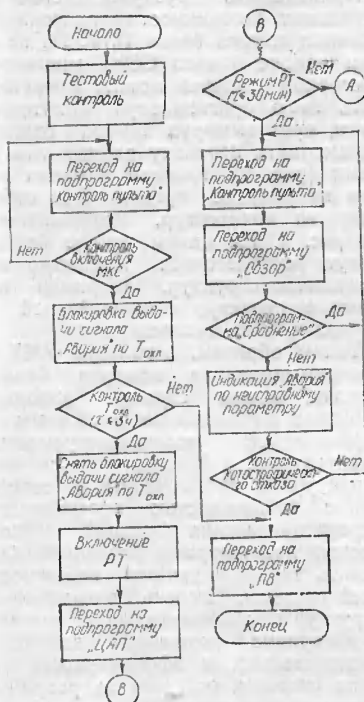


Рис. 2. Алгоритм работы БОИ

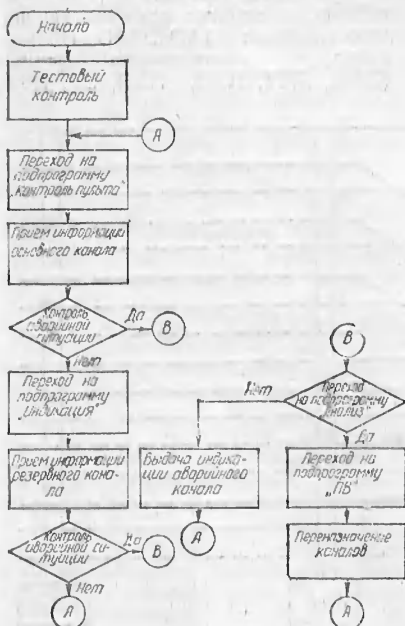


Рис. 3. Алгоритм работы БУИ

А. И. Макаров

МНОГОПРОЦЕССОРНЫЕ СИСТЕМЫ С ОГРАНИЧЕННОЙ ПРОГРАММИРУЕМОСТЬЮ АРХИТЕКТУРЫ

Предлагаемая многопроцессорная система предназначена для цифровой обработки сигналов, разработки систем управления высокочастотными объектами и т. п., где производительность и (или) надежность одной микроЭВМ недостаточна, а использование большой ЭВМ нежелательно или невозможно. При ее разработке использованы процессоры с развитой, распространенной системой команд и предусмотрена возможность простого расширения системы и изменения ее архитектуры для решения другого класса задач или повышения надежности. Основу ее составляют процессоры с системой команд микроЭВМ «Электроника 60», оперативная память, организованная в виде банков по 4К слов с программируемым выбором их, и универсальные магистральные коммутаторы (УМК), имеющие по четыре двунаправленные шины и коммутирующие шину адреса/

данных и основные управляющие сигналы процессоров.

В многопроцессорной системе с магистральной архитектурой (рис. 1, а) каждый процессор (Пр) работает с двумя своими банками ОЗУ через коммутатор (К), причем один из процессоров может работать через общую магистраль с банками ОЗУ других процессоров. Недостаток такой архитектуры — несколько процессоров не могут занимать общую магистраль одновременно.

В многопроцессорной системе с конвейерной архитектурой (рис. 1, б) обрабатываемые данные записываются в ОЗУ через коммутатор К1. По его заполнении коммутатор организует запись данных в ОЗУ, а процессор Пр2 обрабатывает данные из ОЗУ_н и записывает в ОЗУ_н следующего коммутатора. После обработки данных в ОЗУ_н процессор Пр2 начинает обработку данных в ОЗУ_н и переписывает их в ОЗУ_н следующего коммутатора. При этом запись данных должна осуществляться уже в ОЗУ_н, а процессор Пр3 должен начать обработку данных из ОЗУ_н, подключенного к коммутатору К2, и т. д. Передача данных между удаленными процессорами неэффективна. Для этого необходимо остановить промежуточные процессоры и через последовательно соединенные коммутаторы подключить необходимый банк ОЗУ к нужному процессору.

В многопроцессорной системе с иерархической архитектурой (рис. 1, в) каждый из процессоров эффективно

работает со своими ОЗУ и ОЗУ процессора с более высокой иерархией. Обращение любого процессора к произвольному банку ОЗУ возможно через коммутаторы с более высокой иерархией и для этого требуется, при необходимости, остановка соответствующих процессоров. Это, естественно, снижает эффективность системы в целом.

Приведенные структуры систем не отражают возможности программирования номера банка ОЗУ. На каждом уровне банков ОЗУ многопроцессорной системы можно организовать свою оригинальную архитектуру и, программируя номера банков, осуществлять передачу данных между ними (рис. 2). Каждая ломаная линия на рисунке представляет собой одну из архитектур, изображенной на рис. 1. На одном уровне банков могут располагаться несколько законченных структур, например несколько структур с конвейерной и иерархической архитектурами.

Таким образом, используя УМК и программирование номеров банков памяти, можно строить довольно сложные вычислительные системы с ограниченной программируемостью архитектуры. Введение в систему структурной избыточности в сочетании с возможностью программного перераспределения функций, выполняемых процессорами, позволяет обеспечить также заданную надежность всей системы. Степень адекватности структуры вычислительной системы и алгоритма решаемой задачи, а следовательно, и эффективность работы системы, остается за разработчиком архитектуры [1].

Основу УМК (рис. 3) составляют четыре БИС байтовых магистральных коммутаторов К583ХЛ1 [2]. БИС D1 и D3 осуществляют коммутацию шин адреса/данных 1AD...4AD, D5 коммутирует управляющие сигналы (ОБМ, ДЧТ, ДЗП, ПЗП, ВУ, ПРР)

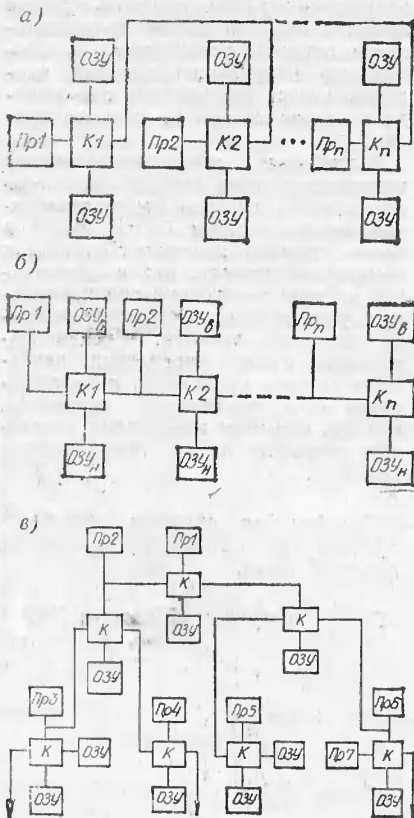


Рис. 1. Структурные схемы многопроцессорных систем: с магистральной (а), конвейерной (б) и иерархической (в) архитектурами

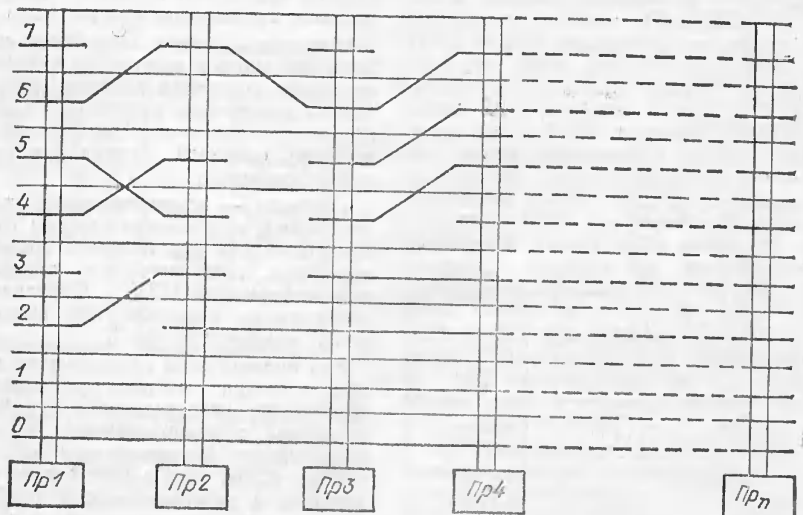


Рис. 2. Организация структур на уровне банков

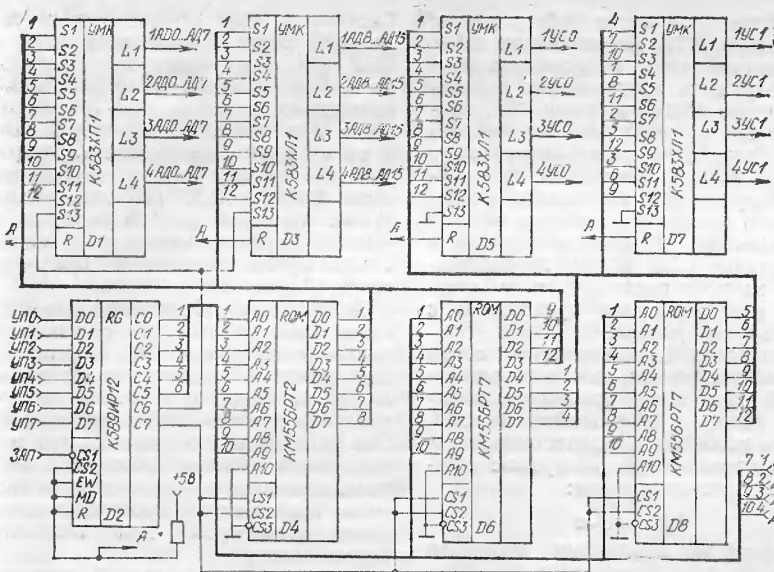


Рис. 3. Принципиальная схема УМК

активного устройства канала (1УС0 ... 4УС0), а D7 — управляющие сигналы (ОТВ, ЗПР, АИЦ, ПВС) пассивного устройства канала (1УС1 ... 4УС1). Направление передачи сигналов БИС D7 противоположно направлению сигналов БИС D5. Изменение направления передачи обеспечено перекоммутацией разрядов (S1 ... S13) слова микрокоманды. Управляются коммутаторы через дешифратор, выполненный на трех ПЗУ КМ556РТ7. Регистр D2 служит для хранения управляющего слова УП0 ... УП7 коммутатора. Изменение направления передачи шины адреса/данных при выполнении цикла чтения активным устройством осуществляется дешифратором под управлением сигналов 1ДЧТ ... 4ДЧТ, подключенных к старшим разрядам ПЗУ.

Возможные варианты коммутации магистралей приведены на рис. 4. Магистралы на этом рисунке условно изображены ребрами прямоугольника, а направление передачи информации — стрелками. Разряды УП0 ... УП4 управляющего слова определяют соединения магистралей. Активное устройство (3...5 строки) может записывать данные параллельно в два или три ОЗУ. Шестой разряд управляющего слова позволяет инвертировать направления соединений, изображенных на рисунке (направление передачи противоположно направлению стрелок). В этом случае при соединениях, приведенных в 3-й и 4-й строках рисунка, данные из двух ОЗУ объединяются по схеме И или ИЛИ в зависимости от 7-го разряда управляющего слова. При этом управляющие сигналы магистралей всегда объединяются по схеме ИЛИ. Соединения, приведенные в 5-й строке, позволяют, в зависимости от УП6, осуществлять объединение

данных из трех ОЗУ по схеме ИЛИ или мажоритировать их. Выполнение тремя процессорами одинаковой программы с последующим мажоритированием результата удобно использовать для повышения надежности вычислений. Восьмой разряд управляющего слова позволяет запретить передачу данных через коммутатор во всех направлениях.

Используя систему микрокоманд БИС К583ХЛ1 [2], таблицу прошивки ПЗУ составить просто. При ограничении выполняемых соединений дешифратор легко реализуется на небольшом числе ИС средней степени интеграции. При разработке УМК необходимо помнить, что все выходы БИС К583ХЛ1 и КР556РТ7 выполнены с открытым коллектором, поэтому требуются нагрузочные ре-

зисторы (на схеме не показаны). Питание БИС К583ХЛ1 осуществляется током 170 мА от источника +5 В через резисторы 22 Ом с номинальной мощностью не менее 1 Вт.

Конструктивно УМК целесообразно выполнить на одной плате с одним или двумя банками ОЗУ для сокращения числа выводов с платы. Для программирования номера банка в схему ОЗУ необходимо ввести регистр вместо переключек, определяющих этот номер. При использовании БИС ОЗУ КР537РУ9, контроллера ЗУ КР588ВГ2 и 8-разрядных регистров УМК с одним банком ОЗУ реализуется на 16 корпусах. Еще один банк ОЗУ требует дополнительно семь корпусов. Управление УМК и ОЗУ осуществляется одним 16-разрядным словом. Его младший байт управляет УМК, а три или шесть разрядов старшего байта — номерами банков ОЗУ.

Управление работой всей вычислительной системы с произвольной ее конфигурацией может выполнять любой из процессоров этой системы. При достаточно большом числе процессоров функции управления системой целесообразно поручить отдельному процессору. Управляющий процессор должен иметь интерфейс для записи управляющего слова в УМК с ОЗУ, начального и последующих запусков процессоров и обслуживания сообщений процессоров о завершении ими программ или фрагментов программ. Каждый из процессоров системы должен иметь возможность сообщить управляющему процессору эту информацию.

Управляющее слово в УМК записывается с ОЗУ через интерфейс с программным режимом работы. Начальный пуск процессоров системы осуществляется аналогичным интерфейсом, разрешая прохождение сигнала ПОСТ в процессоры. Одно 16-разрядное слово может управлять

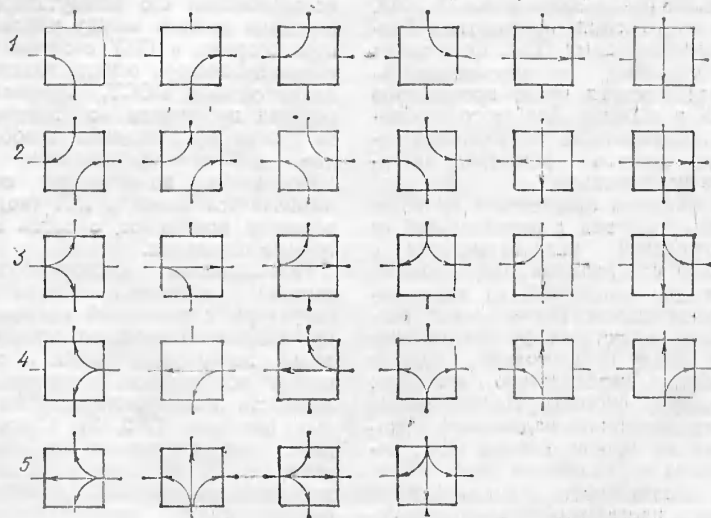


Рис. 4. Варианты коммутации магистралей

начальным пуском 16 процессоров, который целесообразно осуществлять в нулевом режиме через 24-ю ячейку ОЗУ. Повторный пуск удобно выполнить по прерыванию по внешнему событию (таймер) через 100-ю ячейку ОЗУ. Для повторного пуска и его запоминания не требуются регистры. Одним 16-разрядным словом также можно управлять повторным пуском 16 процессоров.

Для сообщения процессорам о завершении выполнения программы управляющий процессор должен иметь развитую систему прерываний, выполненную, например, на БИС К589ИК14 по схеме, приведенной в статье [3]. При организации запросов на прерывание используются сигналы СБРОС процессоров системы. Если нет возможности использовать этот сигнал (процессор, например, обслуживает внешние устройства, требующие сброса), необходимо изготовить программный интерфейс с одним регистром, доступный на запись со стороны процессора системы и на чтение со стороны управляющего процессора. В этом случае через регистр можно передавать и дополнительную информацию, например, условное имя выполненной программы или другие сведения. При этом возможно управление системой не только по прерываниям, но и по флагу.

Конструктивно интерфейсы обслуживания УМК, ОЗУ, начального и повторного пусков процессоров и прерываний целесообразно выполнить на одной плате, применив для соединения их с каналом управляющий процессор УМК, с помощью которого можно передавать управление системой другим процессорам. При использовании двух БИС селектора адреса КР588ВГЗ и одной БИС шифратора приоритета прерываний К589ИК14 плата интерфейсов управляющего процессора позволяет применить в вычислительной системе до восьми процессоров и до 13 УМК с соответствующим количеством банков коммутируемых ОЗУ. Если таких плат несколько, то пропорционально увеличивается число процессоров и УМК в системе. Для этого необходимо предусмотреть возможность изменения адресов регистров интерфейсов переключками.

Программное обеспечение вычислительной системы с произвольной ее конфигурацией разрабатывается с помощью стандартных операционных систем для микроЭВМ с аналогичным процессором. Программист, распределяя задачи или фрагменты программ между процессорами, должен разработать необходимые для них архитектуру системы. Целесообразна магистральная или радиальная архитектура на уровне банков ОЗУ, содержащих управляющие программы. Это позволит просто загружать и отлаживать программы каждого процессора автономно, подключив к нему необходимые банки ОЗУ и внешние

устройства и запретив работу других процессоров. При эксплуатации такая архитектура дает возможность просто загружать программы в процессоры через общее внешнее ЗУ, подключенное к управляющему процессору. Разработку программного обеспечения полезно начинать с программы обслуживания прерываний управляющего процессора, которая определяет порядок и последовательность коммутации ОЗУ между процессорами и запуска последних, т. е. алгоритм взаимодействия процессоров с памятью во времени.

Программы процессоров в своем оформлении имеют несколько особенностей. При пуске процессоры начинают работу с 24-й ячейки памяти. Из нее необходимо осуществить переход к тривиальной программе ожидания повторного пуска:

```
WAIT
MET:BR.—2
```

Выполняя эту программу процессор будет ожидать прерывания по внешнему событию (таймеру) для выполнения основной программы. В 100-й ячейке памяти должен содержаться адрес этой программы, которая начинается с команд

```
PROG: NOP
      MOV≠2, MET
      MOV≠2, PROG
```

модифицирующих начало управляющей программы и программы ожидания, а заканчивается

```
MOV≠140, PROG
MOV≠176, MET
RESET
RTI
```

Первые две команды восстановят в первоначальном виде программы, третья сообщит управляющему процессору об окончании работы сигналом СБРОС, а последняя возобновит работу программы ожидания с последней команды.

Модификация программы нужна только в тех случаях, когда необходима остановка процессора для использования его коммутатора для передачи данных между удаленными процессорами и ОЗУ системы. Остановка процессора осуществляется подачей сигнала ПОСТ, вызывающего переход процессора по прерыванию на программу ожидания; возобновление работы — прерыванием через 100-ю ячейку, по которому сначала выполняется команда RTI (код 2) в основной программе, а затем в программе ожидания.

Предлагаемая многопроцессорная система позволяет организовать структуры с различной архитектурой из небольшого числа однотипных модулей, легко расширяется и обеспечивает возможность повышения надежности и живучести системы в целом (см. рис. 1, 2, 3). Система из двух, трех процессоров на базе БИС серии К1801 [4] при меньшем потреблении энергии имеет большую информационную производительность, чем однопроцессорная микроЭВМ на базе БИС серии К1804 или К1802.

Система из трех процессоров на базе БИС серии К1801, двух УМК и ОЗУ 16К слов реализуется примерно на 80 корпусах и имеет производительность около $1,4 \cdot 10^6$ операций/с при энергопотреблении 21 Вт. В то же время микроЭВМ «Электроника 60» на базе одного процессора серии К1804 с ОЗУ 16К слов реализуется примерно на 140 корпусах и имеет производительность около 1 млн. простых операций/с при энергопотреблении более 30 Вт.

Программное обеспечение для системы легко разработать с использованием стандартных ОС, широко используемых в вычислительных комплексах на базе микроЭВМ «Электроника 60» и мини-ЭВМ серии СМ. Оно незначительно усложняется при увеличении числа процессоров в системе, причем производительность системы практически пропорциональна числу процессоров для широкого класса задач.

Недостаток предлагаемой системы: большинство модулей, используемых в ней, не стандартны и не выпускаются промышленностью.

Телефон для справок: 69-42-17,
г. Минск

ЛИТЕРАТУРА

1. Каляев А. В. Многопроцессорные системы с программируемой архитектурой.— М.: Радио и связь, 1984.
2. Микропроцессорные комплекты БИС на основе интегральной логической логики / Под ред. Э. П. Калошкина.— М.: Радио и связь, 1984.
3. Лукьянов Д. А. КР580 — автоматизация без проблем! // Микропроцессорные средства и системы.— 1985.— № 1.
4. Филиппов С. А., Майдыковский И. В., Борщенко Ю. И., Зубов Ю. В. Применение однокристалльного микропроцессора К1801ВМ1 в автономных системах сбора и обработки информации // Микропроцессорные средства и системы.— 1985.— № 1.

Статья поступила 4 марта 1986 г.

В. Г. Баранов, С. Н. Калягин, Ю. С. Бажанов, Т. А. Корсакова

ПРИМЕНЕНИЕ БИС К1801ВП1-035 В ИНТЕРФЕЙСНЫХ ПЛАТАХ МАЛЫХ ЛОКАЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ

Вычислительные мощности программно-совместимых мини- и микроЭВМ (СМ-3, СМ-4, «Электроника 100/25», «Электроника 60М» и др.) используются часто недостаточно вследствие отсутствия быстрых и надежных каналов связи между машинами, слабой обеспеченности ЭВМ периферийными устройствами. Выход из этого положения — объединение имеющихся средств вычислительной техники в малую локальную вычислительную сеть (МЛВС).

На кафедре «Вычислительная техника» Горьковского политехнического института им. А. А. Жданова разработана интерфейсная плата (ИП) на базе БИС К1801ВП1-035. С помощью ИП за три месяца (включая разработку и изготовление ИП) построена малая локальная вычислительная сеть типа «Звезда» с центральным комплексом на основе мини-ЭВМ «Электроника 100/25» и шестью периферийными комплексами на основе микроЭВМ «Электроника 60М», работающая под управлением ОС РАФОС-II.

Описание интерфейсных БИС

Основу ИП составляют шесть специализированных БИС ИРПС К1801ВП1-035. Именно они преобразуют параллельную информацию, поступающую из канала мини-ЭВМ, в последовательную и наоборот, в соответствии с требованием ОСТ 11 305.916.

Микросхема К1801ВП1-035 обеспечивает по последовательному каналу:
— обмен данными при тактовой частоте 4608 кГц со скоростями 50, 75, 100, 150, 200, 300, 600, 1200, 2400, 4800, 9600, 19200, 57600 бод;

— прием и выдачу послыбки последовательного кода в форматах 5, 7 или 8 информационных бит;

— формирование 2 стоповых бит, причем в формате 5 бит — 1,5 стоповых бит;

— формирование и контроль бита паритета (целости или нечетности), а также работу без бита паритета.

Микросхема содержит два регистра состояния (приемника и передатчика) и два регистра данных (приемника и передатчика), а также два источника адресов векторов прерывания (приемника и передатчика).

Функциональная схема ИП

При проектировании ИП наиболее сложно было согласовать интерфейс типа «Общая шина», присущий мини-ЭВМ «Электроника 100/25», и интер-

фейс типа Q-bus, на который ориентирована БИС ИРПС.

Для корректной работы ИП необходимо полное согласование сигналов управления БИС ИРПС и «Общей шины» как в режимах обмена данными с программным опросом регистров состояния БИС, так и в режиме прерывания.

Адрес и данные с канала мини-ЭВМ через каналные приемники поступают на мультиплексор адресованных (рис. 1). Выход мультиплексора образует внутреннюю шину адресованных ИП, к которой параллельно подключены шесть БИС ИРПС. Адрес с каналных приемников поступает и на адресный селектор, образуя сигналы BS1...BS6, выбирающие одну из шести БИС ИРПС. Адреса регистров БИС задаются ПЗУ АДР (К155РЕЗ), вставленным в контактирующее устройство на интерфейсной плате, и в любой момент могут быть изменены по желанию пользователя.

Блок согласования сигналов управления и блок прерываний преобразуют сигналы синхронизации, управления и прерывания в соответствии с требованиями, предъявляемыми к каждому типу интерфейса. Выходные сигналы этих блоков поступают параллельно на соответствующие входы и выходы всех шести БИС ИРПС.

Сигнал разрешения прерывания IAKI поступает с блока прерываний и проходит последовательно через все БИС ИРПС — выход разрешения

ТАКО предыдущей БИС соединен со входом разрешения прерывания IAKI последующей.

Микросхема, затребовавшая прерывание, прекращает прохождение сигнала разрешения прерывания через себя к последующим БИС. Таким образом, разрешаются конфликты между интерфейсными БИС, одновременно запросившими прерывание. Приоритет последней в этом ряду БИС — естественно, самый низкий. Кроме того, в каждом приемопередатчике приоритет запроса от приемника конструктивно заложен выше запроса от передатчика.

В микросхеме К1801ВП1-035 адреса внутренних регистров и адреса векторов прерывания (АВП) — сменные по группам. По желанию разработчика, можно задавать одну из четырех групп с внутренними, либо внешними источниками адресов регистров ИРПС и векторов прерывания. В описываемой разработке используется четвертая группа с внешним источником адресов векторов прерывания, хранящихся в ПЗУ АВП.

Сигналы с выходов ТАКО DD1...DD5 поступают на адресные входы А1...А5 ПЗУ АВП (К155РЕЗ), где записаны старшие разряды (А3...А7) адресов векторов прерывания. Во время прерывания сигнал IAKI, проходя последовательно через микросхемы ИРПС вплоть до БИС, потребовавшей прерывания, образует код адреса источника АВП. Этот код поступает на адресные входы ПЗУ АВП, выбирая требуемый в данный момент АВП из ПЗУ, который передается через каналные передатчики в канал мини-ЭВМ («Общая шина»). Разряд А2 АВП подается на каналные передатчики непосредственно из БИС ИРПС, уточняя АВП (передатчика или приемника). Микросхема ПЗУ АВП также вставлена в кон-

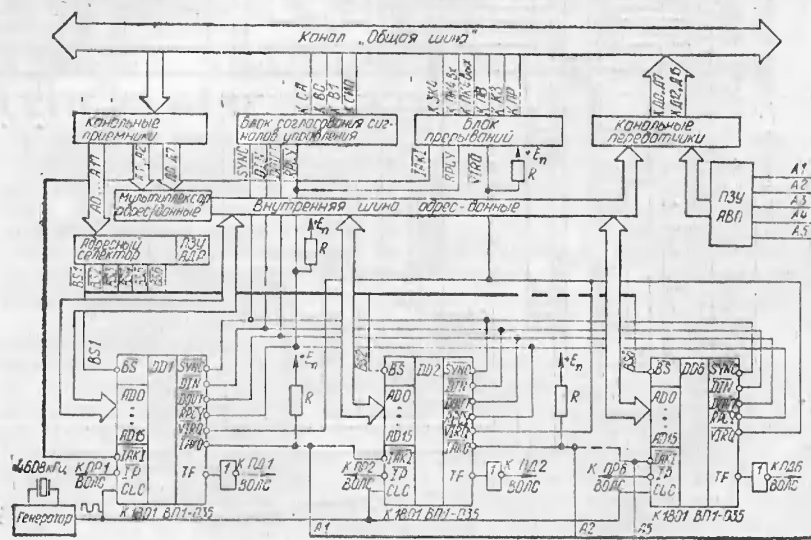


Рис. 1. Функциональная схема интерфейсной платы

тактирующее устройство на интерфейсной плате, что позволяет пользователю при необходимости сменить АВП заменой ПЗУ.

Импульсы частотой 4608 кГц с кварцевого генератора поступают на входы СLC микросхем ИРПС, синхронизируя работу передатчиков и приемников и задавая требуемую скорость передачи.

Информация в последовательном коде поступает с выходов передатчиков ТF через буферные вентили на волоконно-оптическую линию связи (ВОЛС), а с ВОЛС — на входы приемников БИС.

Принципиальная схема

Входы AD0...AD7, AD12, AD15 всех интерфейсных БИС DD31...DD36 объединены во внутреннюю шину адреса-данных (рис. 2) с выходом через шинные передатчики DD28, DD29 на шину данных канала мини-ЭВМ. Адрес и данные, поступающие из канала мини-ЭВМ, через каналные приемники DD4...DD6 приходят на мультиплексор (DD12, DD13), выходы которого объединены с внутренней шиной ИП.

Адресный селектор выполнен на элементах DD1...DD3, DD7, DD8.1,

DD9...DD11. Сигналы выборки с выходов B0...B5 элемента DD11 поступают на входы B3 каждого приемопередатчика, выбирая требуемую в данный момент БИС.

Блок согласования сигналов управления выполнен на элементах DD14; DD15; DD16.1,2; DD17; DD18.1,2; DD19.1,2. Одновибратор DD14 задерживает сигналы DIN, DOUT на 70...90 нс по отношению к сигналу SYNC и совместно с триггером DD15 осуществляет временное разделение адреса-данных с канала мини-ЭВМ.

Емкость С3 задерживает сигнал DOUT на 25...50 нс по отношению к заднему фронту сигнала BS во время записи данных в БИС ИРПС. Это делается для устранения временных рассогласований данных на внутренней шине (надежность выполнения указанной операции повышается).

Блок прерываний реализует функции по захвату канала на уровне ЗК4 и согласованию протоколов обмена управляющими сигналами между процессором «Электроника 100/25» и БИС ИРПС.

Кварцевый генератор на элементе DD13 синхронизирует работу интерфейсных БИС.

Режимы работы приемопередатчи-

ков задаются переключками E1...E8. При данном положении переключек (рис. 2) скорость обмена — 9600 бод, формат посылки — 8 бит, прием и выдача посылок без бита паритета.

Как было сказано выше, адреса внутренних регистров приемопередатчиков, а также адреса векторов прерываний определяются содержимым ПЗУ DD11 и DD25 в контактирующих устройствах на ИП. Схема адресного селектора (рис. 2) задает адресное пространство регистров ИРПС: 776400—776776 (восьмеричный код).

При программировании ПЗУ АДР (DD11) необходимо выбрать шесть желаемых рабочих адресов ПЗУ, соответствующих разрядам A3...A7, и записать по этим адресам коды выборки (слова с записанным «Лог.0» в каждом из них). Этот «Лог.0» и будет сигналом выборки, поступающим на вход BS каждого из приемопередатчиков. Заметим, что коды выборки в ПЗУ не должны повторяться. По остальным разрядам в ПЗУ АДР должны быть записаны «Лог.1».

Адреса векторов прерывания могут быть практически любые из пространства 000—376 (исключая системные АВП). Отметим, что в ПЗУ DD25 коды АВП имеет смысл запи-

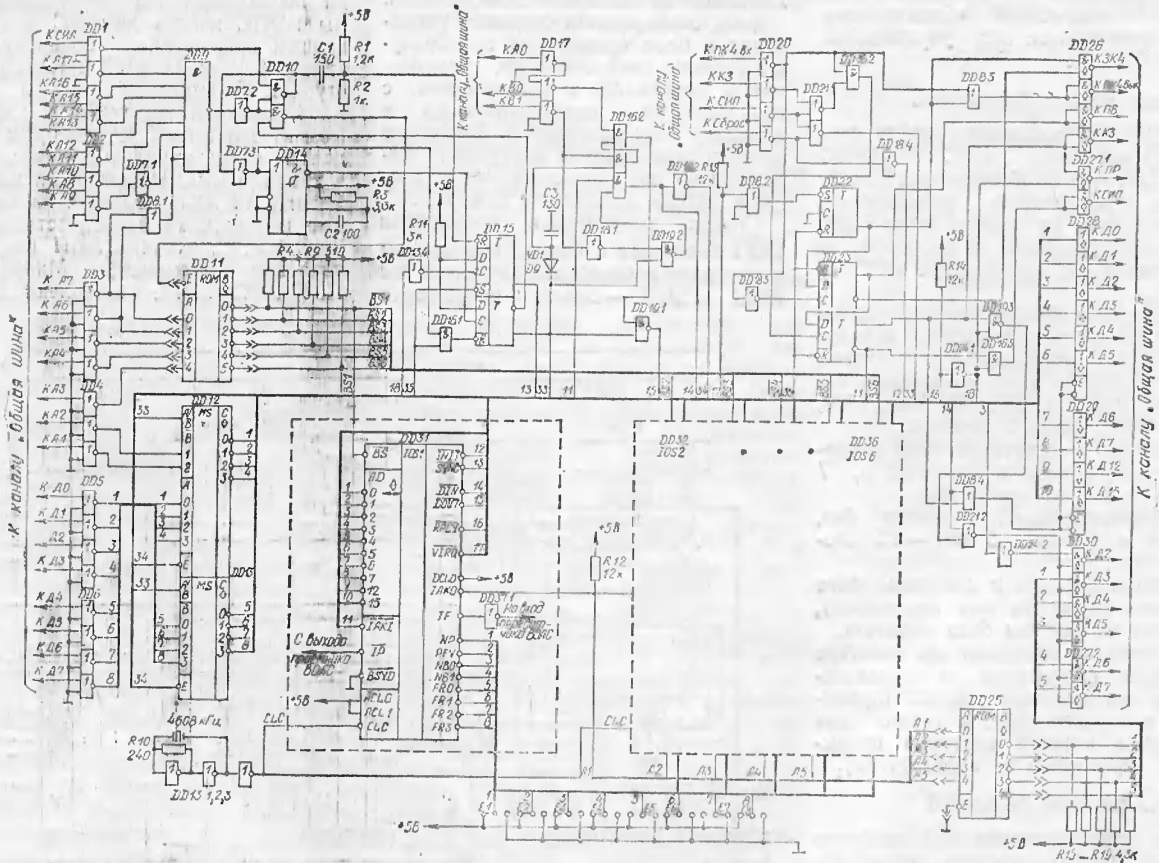


Рис. 2. Принципиальная схема интерфейсной платы

сывать лишь в следующие рабочие адреса (восьмеричный код): 37, 17, 07, 03, 01, 00, принадлежащие соответственно DD31... DD36.

Программная поддержка малой локальной вычислительной сети

ОС РАФОС-II позволяет подключить в сеть через ИП наряду с мини и микроЭВМ и отдельные терминалы, работающие с последовательной линией связи (например, дисплей 15ИЭ—00—013).

Для каждого периферийного комплекса на дисках центральной машины отводится виртуальный файл, служащий системным посетителем ПК. На нем размещаются управляющий монитор, драйверы внешних устройств, трансляторы, системные библиотеки, а также программы пользователя. В оперативную память ПК загружается ядро SJ- или FB-монитора вместе с DD-драйвером линии связи. Со стороны центрального комплекса многопользовательский TS-монитор и DISK-программа связи с дисками [3] управляют МЛВС.

Информацию обрабатывает процессор ПК, причем пользователь вправе использовать все команды монитора и системные программы ОС (например, программы копирования и обработки файлов, работы с редакторами текстов и т. д.).

Работая с периферийной машиной, пользователь может через программу эмуляции терминала выйти на диалог с TS-монитором центральной машины. Вернуться к работе с монитором периферийной машины также нетрудно.

Система почтовых ящиков ОС РАФОС-II ведет обмен сообщениями между пользователями, а также между задачами (на разных ПК) без участия человека, что весьма удобно при организации распределенной обработки данных.

Время загрузки ядра SJ-монитора в память процессора ПК при скорости обмена 9600 бод — 50...60 с. При частом обмене большими объемами информации между периферийной и центральной ЭВМ целесообразно скорость обмена повысить до 19200 или 57600 бод. В этом случае время обмена сопоставимо с реакцией гибких дисков.

Конструкция ИП

Конструктивно ИП выполнена (рис. 3) в виде стандартного модуля (250×280 мм) и вставляется для работы в свободное гнездо блока расширения канала СМ БРК наряду с контроллерами других внешних устройств.

Питание платы — от источника питания +5 В БРК, ток потребления — 1,5 А, что на порядок меньше по сравнению с аналогичными устройствами, выпускаемыми промышленностью.

Число внешних устройств, подключаемых к ИП, обусловлено максимальной плотностью монтажа, до-

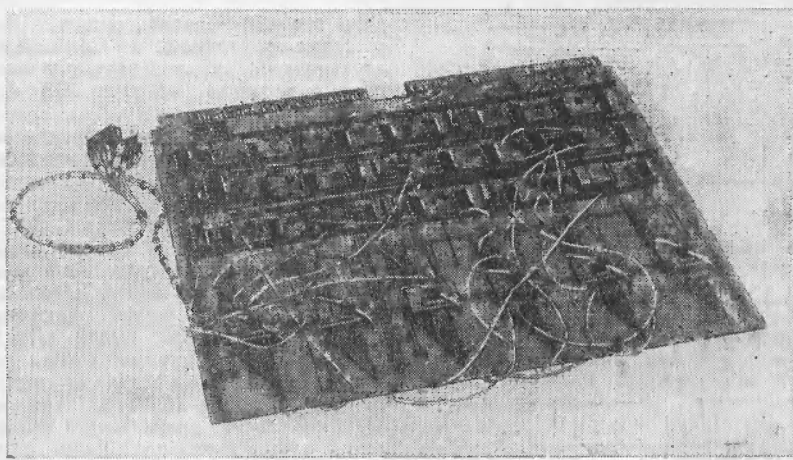


Рис. 3. Интерфейсная плата для организации малой локальной вычислительной сети

стигнутой при изготовлении платы в лабораторных условиях. Отметим, что ИП представляет собой функционально и конструктивно законченный интерфейсный узел. Для подключения к мини-ЭВМ большого числа периферийных устройств следует изготовить несколько подобных ИП и вставить их в гнезда БРК. Периферийные устройства связаны с ИП волоконно-оптической линией «Электроника МС 4101» [1]. Удаленность периферийных устройств от центральной ЭВМ зависит от длины оптического кабеля (50—300 м). Если линия связи — витая пара, то каждую БИС ИРПС необходимо снабдить схемой выдачи токовых посылок в линию связи и обратного их приема [2].

Телефоны для справок: 36-82-28 (сл.), 66-88-11 (д.). — *Калягин Сергей Николаевич.*

УДК 681.3.03

Л. И. Преснухин, В. И. Белильников, Ю. И. Волков, О. В. Ургант, В. Г. Шапкин

АДАПТЕР ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ НА БАЗЕ БИС К1801ВП1-065

В настоящее время основным способом коллективизации вычислительных ресурсов является их объединение в вычислительные сети с топологией типа кольцо, звезда и др. Кольцевые локальные вычислительные сети (КЛВС) не получили широкого распространения из-за отсутствия специальной элементной базы для построения сетевых адаптеров (СА). Применение ИС малой и средней степени интеграции для реализации СА приводит к существенному увеличению объема устройства.

Использование в СА БИС К1801ВП1-065 и передача части функций по выполнению протокола сети на программное обеспечение позволяет упростить его схему (рис. 1). СА способен обеспечить работоспособности и «живучести» КЛВС:

ЛИТЕРАТУРА

1. Опволоконные системы передачи данных // Микропроцессорные средства и системы. — 1984. — № 4.
2. Корольков А. А., Раденко М. Е., Сеньков В. К. Применение БИС КР580ВВ51 для реализации последовательных интерфейсов микропроцессорных систем // Микропроцессорные средства и системы. — 1985. — № 1. — С. 82—85.
3. Операционная система для организации высокопроизводительных систем с разделением функций РАФОС-II. Локальные многомашинные системы. Руководство системного программиста. Техническое описание.

Статья поступила 21 июня 1986 г.

поддерживает целостность КЛВС (при отключении питания, физическом выключении СА из КЛВС, сбое программы, обслуживающей КЛВС и т. д.), когда СА отключается от сети (замкнув свой вход с выходом) [1]. Информация при этом проходит через устройства коммутации, не нарушая режима работы других СА в КЛВС [2];

использует дешевую среду передачи информации в КЛВС — витая пара или телефонный провод. В работе [3] приводится пример применения витой пары для передачи цифровых сигналов со скоростью до 10 Мбит/с; формирует протокол сети программно с жесткой синхронизацией работы программы и СА. Кроме того, СА КЛВС имеет свой адрес [4], а также гальваническую развязку от дру-

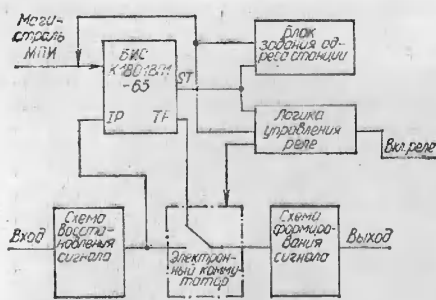


Рис. 1. Схема СА

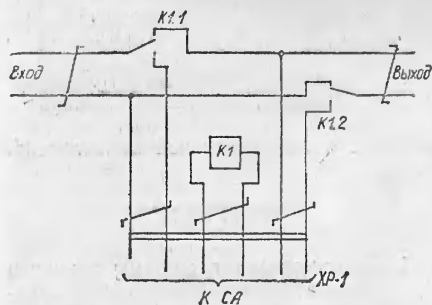


Рис. 2. Схема поддержания целостности кольца

гих устройств сети: коммутационные устройства СА КЛВС без встроенных источников питания [1].

При выключении СА (выключении питания или разъема ХР1) не нарушается целостность КЛВС (рис. 2). Среда передачи информации и реле К1, установленные в местах подключения СА, не должны отключаться во время работы. Сигналы управления реле К1 формируются в СА. Логика управления должна обеспечивать автоматическое отключение реле через 1...2 с после того, как программа перестала к нему обращаться.

При программном формировании протокола работы КЛВС необходимо учитывать логику работы контроллеров некоторых периферийных устройств (например, дисковод 6022), запрещающих прерывания на интер-

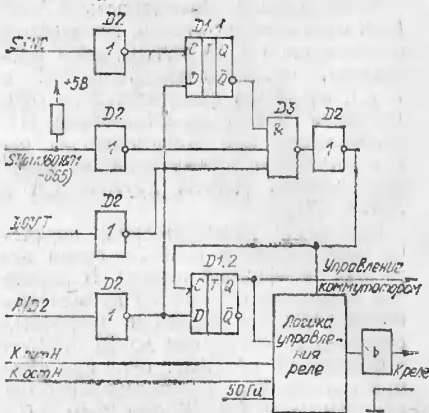


Рис. 3. Схема сопряжения с логикой управления

вале времени чтения/записи. Для решения этой задачи в СА введен электронный ключ, переключающий вход с выходом внутри СА (см. рис. 1).

Электронный ключ управляется однобитовым программно доступным регистром (рис. 3), расположенным во втором разряде регистра управления приемником (триггер D1.2). Импульс записи (ИЗ) на его вход поступает в случае, если идет обращение к БИС K1801BP1-065 (сигнал ST); в адресной части цикла во втором разряде был выдан «Лог.0» (выход D1.1) и активен сигнал управления выводом данных системной магистрали (D0U). По этому импульсу запоминается состояние второго разряда шины адреса данных в триггере D1.2. Импульс записи и состояния выхода D1.2 используется в схеме управления реле. ИЗ указывает на то, что идет работа с КЛВС и сетевая программа функционирует. Выход D1.2 активен при включении электронного ключа; сигнал ИЗ в этот момент может отсутствовать.

Отсутствие сигнала ИЗ при неактивном выходе D1.2 более 1 с означает, что сетевая программа выгружена из памяти и реле следует отключить. Отключение реле вызывает также сигналы сбоя питания (КПИТН) и остановка процессора (КОСТН).

Для гальванической развязки (рис. 4, а) применен импульсный трансформатор. Однако сигнал TF (выходной сигнал передатчика) имеет постоянную составляющую, которая не передается через трансформатор. Из временной диаграммы (рис. 4, б) видно, что передаются только фронты сигнала TF. Каждому фронту соответствует импульс нужной полярности и фиксированной длительности. Отношение длительности к паузе, а также характеристики трансформатора и номиналы резисторов R3 и R4 таковы, что амплитуда помех, обусловленных переходными процессами, не превышает 5% от амплитуды импульса. На приемной части сигнал восстанавливается. Схема восстановления (рис. 4, в) представляет собой детектор, выделяющий положитель-

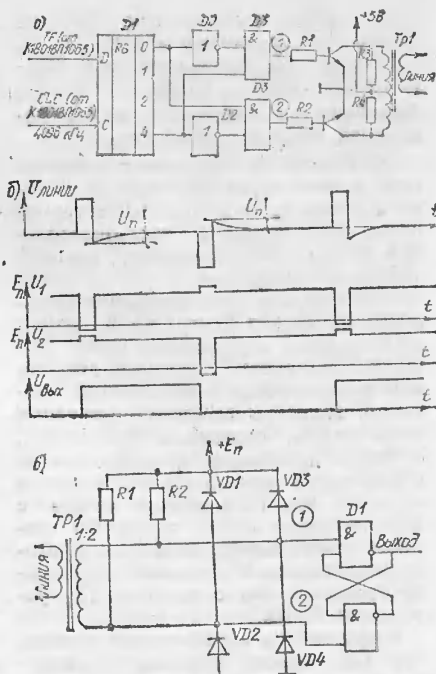


Рис. 4. Прием сигнала из линии:

а — схема гальванической развязки передатчика, б — временная диаграмма работы передатчика, в — схема гальванической развязки и восстановления сигнала в приемнике

K1801BP1-065 разряды A/D 8...11, A/D 13, 14 в двунаправленной шине адреса/данных являются либо входами БИС, либо не используются вообще, что позволяет вывести в эти разряды индивидуальный адрес станции, который будет считываться со всех четырех регистров. Усилитель D1 начинает работать, когда на входах D2 сформируются сигналы: опознавания адреса ST (вырабатываемый БИС K1801BP1-065 для синхронизации обмена системной магистралью), SYNC и управления выводом данных DIN. Формируемый таким образом адрес станции позволяет объединить в КЛВС до 63 станций.

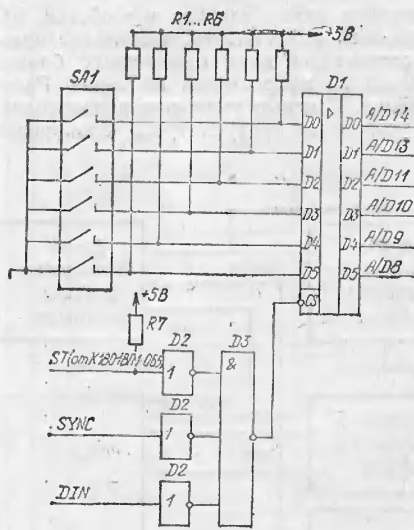
Сетевой адаптер размещается на полуплате интерфейсного модуля УКНЦ. Основные технические данные СА:

Габаритные размеры, мм	145×100×15
Масса, г, не более	150
Потребляемая мощность, Вт, не более	2,5 (ток не более 0,5 А)
Принцип передачи данных	последовательный, байтами, с контролем по четности
Физический канал связи	витая пара МГШВ-0,35
Максимальная скорость передачи, бит/с	57 000
Сохраняется работоспособность при изменении напряжения питания в пределах 5% от номинала	

ные и отрицательные импульсы, и R-S-триггер, восстанавливающий форму сигнала. Для улучшения фронтов принимаемых импульсов, в качестве элементов D1 применены вентили с триггерами Шмидта на входе.

Индивидуальный адрес каждой станции устанавливается переключателем SA1 (рис. 5). В БИС

В программном обеспечении предусмотрена ширококвотельная загрузка, использование периферийных средств других машин, «почта» и другие сервисные возможности. Совместимость программного обеспечения СА достигается на уровне драйвера сети. При наличии в одной из ЭВМ КЛВС быстродействующих дисковых



устройств с объемом памяти более 4 М байт, на ЭВМ, включенных в сеть и имеющих только терминал в СА, можно организовать доступ к виртуальным дискам, работая в операционной системе RT-11.

Кольцевая локальная вычислительная сеть, построенная на базе рассмотренного СА, может объединять машины типа ДВК-2М, ДВК-1, БК-0010 и др. с магистралью, аналогичной магистрали «Электроника 60». КЛВС можно использовать в учебных классах, в системах АСУ, САПР как терминальную сеть и т. п.

Телефон для справок 534-64-96.
г. Москва

Рис. 5. Схема формирования адреса станции

МИКРОПРОЦЕССОРЫ В СИСТЕМАХ СВЯЗИ

УДК 681.322

В. А. Вологжанин, В. А. Скворцов, Н. Е. Слизень

КОМПЛЕКС МИКРОПРОЦЕССОРНЫХ СРЕДСТВ ДЛЯ ИНФОРМАЦИОННО- ИЗМЕРИТЕЛЬНЫХ СИСТЕМ

Развитие микропроцессорных средств вычислительной техники (МСВТ) и расширение сферы их применения сопровождается расширением круга решаемых задач, которые ставятся перед современными информационно-измерительными системами (ИИС). Разработанный комплекс микропроцессорных средств построен по магистрально-модульному принципу, обеспечивающему гибкость, агрегируемость, алгоритмическое, схемотехническое единство и сравнительную простоту модификации и развития. Реализация этого принципа предусматривает унификацию аппаратных и программных средств, что позволяет создавать различные прикладные системы из модулей, отвечающих требованиям совместимости по программному, конструктивному, интерфейсному признакам. Система команд комплекса соответствует ОСТ 11.305.903-80. Связь между модулями осуществляется по стандартному интерфейсу МПИ.

Модули представляют собой печатные платы размером 170 × 240 мм, объединенные системным соединителем СМП49. Максимальное число модулей на магистрали — 20. Среди них вычислительные модули (ВМ), реализующие функции управления магистралью, хранение и обработку поступающей информации, контроль за работоспособностью системы. Интерфейсные модули (ИМ), осуществляющие связь ВМ с внешней средой и управление реальными объектами. Сервисные модули (СМ) для настройки, отладки и ремонта системы. Программные модули (ПМ), представляющие собой пакеты прикладных программ, реализующие алгоритм работы комплекса для конкретных исполнений. Дополнительные модули (ДМ), существенно расширяющие возможности и улучшающие характеристики ИИС. Базовый состав комплекса представлен в таблице. Число и тип используемых модулей определяется потребителем,

ЛИТЕРАТУРА

1. Хапдин Х. Дж. Исследования фирмы ИВМ по структурам локальных сетей для конторы будущего // Электроника. 1982. № 7. С. 3—5.
 2. Стриттер Э., Шустек Л. Локальная сеть, объединяющая персональные компьютеры в многоабонентскую многофункциональную систему // Электроника. 1981. № 12. С. 44—51.
 3. Абрамсон П. Метод передачи биполярного кода с возвратом к нулю по линиям с трансформаторными связями // Электроника. 1984. № 6. С. 67—73.
 4. Дэвис Д., Барбер Д., Прайс У., Соломагидес С. Вычислительные сети и сетевые протоколы. — М.: Мир, 1982. С. 369—459.
- Статья поступила 11 июня 1986 г.

Центральный процессор разработан на основе КМДП БИС серии К588 с системой команд и интерфейсом, совместимыми с системой команд и интерфейсом микроЭВМ «Электроника 60» [1]. Он управляет распределением времени использования канала внешними устройствами, выполняет арифметико-логические операции, обеспечивает обработку прерываний и прямой доступ в память.

Базовый состав комплекса аппаратно-программных средств

Тип модуля	Функциональное назначение модуля
Вычислительный (3 модуля)	Центральный процессор Оперативное запоминающее устройство Перепрограммируемое запоминающее устройство
Интерфейсный (10 модулей)	Параллельный интерфейс Последовательный интерфейс Связь с устройством пользования Управление перфолоточными устройствами ввода-вывода Отсчет времени Аналоговый ввод-вывод Ввод релейных сигналов Контроллер модема Контроллер накопителя на магнитной ленте Контроллер канала общего пользования
Сервисный (1 модуль)	Имитатор канала
Программный (3 модуля)	Пакет тестовых программ Пакет прикладных программ Загрузчики

Оперативное запоминающее устройство на 8К шестнадцатиразрядных слов выполнено на микросхемах К537РУ2. В состав одного модуля ОЗУ входят два идентичных блока (рис. 1). Модуль работает в режимах записи и чтения 16-разрядных слов или байтов (рис. 2).

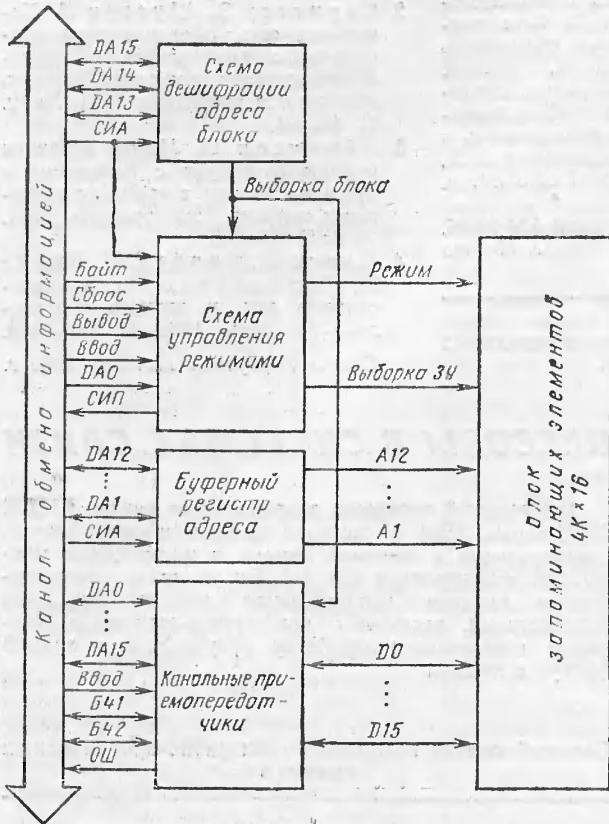


Рис. 1. Структурная схема блока оперативной памяти

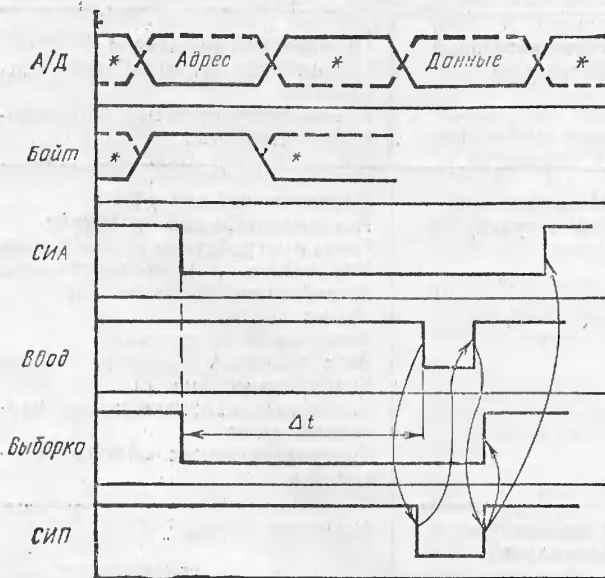


Рис. 2. Временная диаграмма модуля ОЗУ в режиме «ввод»: * — уровень сигнала не имеет значения; Δt — время сокращения цикла ввода

Перепрограммируемое запоминающее устройство обладает информационной емкостью 8К шестнадцатиразрядных слов и состоит из двух одинаковых блоков (рис. 3), построенных на микросхемах К573РФ2. Работает в режиме чтения. С целью уменьшения потребляемой мощности в устройстве предусмотрена блокировка

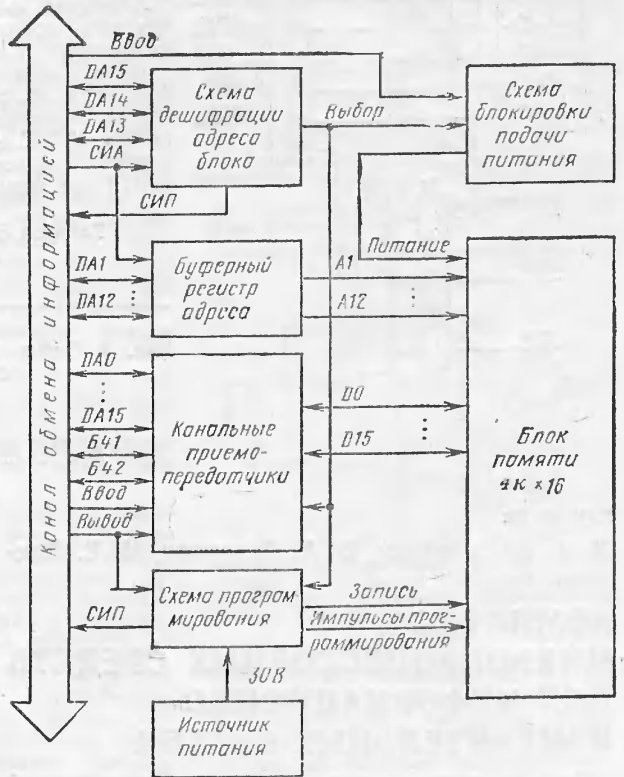


Рис. 3. Структурная схема блока постоянной памяти

подачи питания на микросхемы в момент отсутствия обращения. Имеется возможность программирования микросхем памяти непосредственно в составе модуля.

Интерфейсные модули обеспечивают связь ВМ с внешними устройствами (ВУ) и оборудованием. Обмен информацией производится с помощью стандартных циклов обращения к каналу [2]. Общая структурная схема ИМ показана на рис. 4. Выделенная часть схемы аналогична для всех модулей, а схема выработки сигналов управления ВУ определяет тип ИМ. Обмен данными с ВУ осуществляется посредством программных операций с опросом флага или выполненном прерывании. Элементная база ИМ — микросхемы серий К588, К564.

Параллельный интерфейс (ИРПР8/ИРПР16) предназначен для согласования канала типа МПИ с радиальным параллельным 8 (16)-разрядным интерфейсом. Уровни выходных сигналов соответствуют ТТЛ-логике. Тип логики — «положительная» или «отрицательная» — задается пользователем. Ввод и вывод данных происходят в асинхронном режиме под управлением сигналов «Запрос» и «Строб» при наличии активного сигнала «Готовность» от ВУ.

Последовательный интерфейс предназначен для сопряжения вычислительных модулей с ВУ, имеющими стык ИРПС. В основе работы модуля лежит принцип преобразования параллельного кода, принятого из канала, в последовательный и передача его по двухпроводной линии к удаленному устройству, а также прием

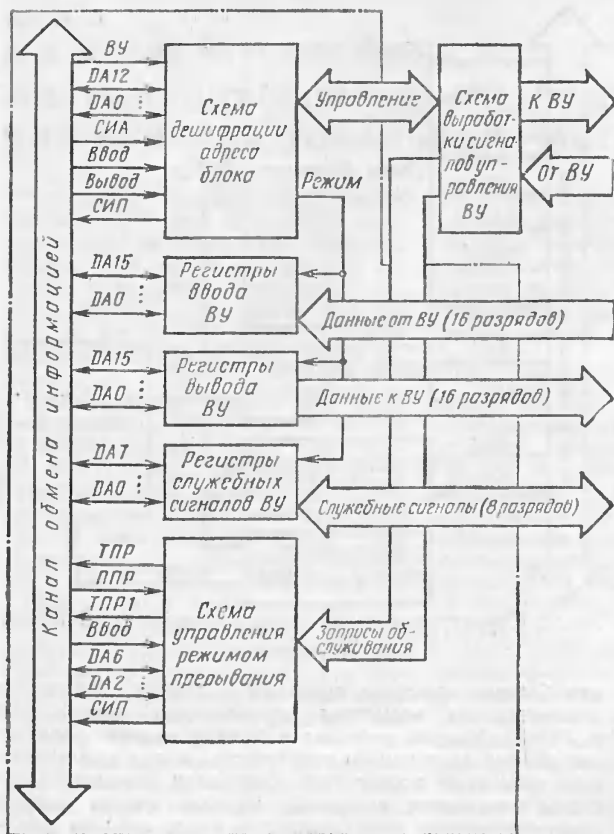


Рис. 4. Структурная схема интерфейсного модуля

последовательного кода, переданного по другой двухпроводной линии, обратное преобразование его в параллельный и передача последнего в канал. Формат посылки — стартовый бит, 5..8 битов данных, 1, 2 стоповых бита — задается пользователем с помощью соответствующих переключек. Максимальная скорость обмена — 9600 бит/с. Преобразование последовательного кода в параллельный и обратно осуществляется асинхронным приемопередатчиком оконечной аппаратуры 1002 ХЛ1. Модуль предоставляет пользователю возможность выхода в локальную вычислительную сеть.

Модуль связи с устройствами пользователя используется для ввода и вывода двоичной информации по 16 линиям связи. Ввод информации может производиться в режиме прерывания по фиксированному вектору, что дает возможность осуществлять контроль изменения состояния входной информации без программного опроса модуля. Такой режим позволяет повысить суммарную производительность ЭВМ за счет подключения нескольких блоков.

Модуль управления перфоленточными устройствами ввода-вывода обеспечивает связь ЭВМ со считывателем с перфоленты FS-1501 и перфоратором ПЛ-150. Временные диаграммы, адреса регистров и векторов прерывания блока соответствуют диаграммам и адресам аналоговых устройств, входящих в комплект микроЭВМ «Электроника 60».

Блок отсчета времени обеспечивает отсчет реального времени в режимах «Таймер», «Одновибратор», «Часы». В режиме «Таймер» модуль вырабатывает периодически повторяющийся сигнал К ПРТ Н — прерывание от таймера, используемый для организации прерывания программы с адресом вектора 100. Период выработки сигнала К ПРТ Н (500 мкс, 1 мс, 10 мс, 100 мс, 1 с, 6 с) устанавливается программно.

В режиме «Одновибратор» модуль обеспечивает организацию прерывания программы по окончании заданного интервала времени. Адрес вектора прерывания в режиме «Одновибратор» устанавливается пользователем с помощью переключек на плате модуля. Интервал времени задается программно числом импульсов тактовой частоты, записываемых в регистр установки. Прерывание программы происходит при равенстве числа импульсов в счетчике импульсов тактовой частоты числу импульсов, записанному в регистр установки. Максимальное число импульсов — 255. Счет импульсов начинается с момента записи разрешения прерывания. Период импульсов тактовой частоты (500 мкс, 1 мс, 10 мс, 100 мс) устанавливается программно независимо от установки периода выработки сигнала К ПРТ Н.

В режиме «Часы» модуль выполняет счет реального времени от 00 часов 00 минут до 159 часов 59 минут. Режимы «Таймер», «Одновибратор», «Часы» имеют независимую установку и могут использоваться одновременно.

Модуль аналогового ввода-вывода обеспечивает ввод и преобразование в цифровую форму восьми аналоговых сигналов, а также вывод двух аналоговых сигналов. В состав модуля входят 8-канальный мультиплексор входных аналоговых сигналов, 12-разрядный АЦП поразрядного уравнивания и два 10-разрядных ЦАП.

Модуль ввода релейных сигналов обеспечивает прерывание программы, кодирование и передачу номера аварийного датчика при поступлении сигнала аварии от любого из 256 датчиков релейных сигналов.

Контроллер модема выполняет сопряжение с аппаратурой передачи данных типа модем 1200 КН4 по стыку С2. Модуль обеспечивает управление линиями стыка С2 групп С2-100 и С2-200 для устройств автоматического вызова, осуществляет подключение к аппаратуре передачи данных, вызов удаленного абонента, обмен информацией и выдачу признаков ошибок в принимаемом сообщении. Для отладки программного обеспечения и контроля исправности модуля предусмотрен режим «Шлейф», при котором передаваемое сообщение поступает во входной регистр модуля.

Контроллер накопителя на магнитной ленте предназначен для управления, записи и чтения информации с накопителя на кассетной магнитной ленте СМ 5204. Контроллер обеспечивает выполнение следующих функций: запись вперед, воспроизведение вперед-назад, поиск вперед-назад, перемотка вперед-назад. Модуль КНМЛ содержит схему фазового кодирования и декодирования информации.

Контроллер канала общего пользования предназначен для управления программируемыми устройствами с байт-последовательным и бит-параллельным обменом информацией по каналу общего пользования системы интерфейса ГОСТ 26.003-80. К одному модулю контроллера можно подключить до 14 внешних устройств. Модуль выполняет следующие функции:

- синхронизация приема СИ1;
- синхронизация источника СИ1;
- приемник П4;
- источник И8;
- контроллер К1...К4, К25.

Сервисные модули. Имитатор канала (рис. 5) необходим для тестирования и настройки интерфейсных модулей и запоминающих устройств с помощью статических сигналов [3]. Временная задержка между сигналами в последовательности определяется оператором. Состояние канальных шин на каждом шаге тестирования передается на схемы индикации. Если центральный процессор нельзя удалить из системы на время тестирования и настройки модулей, то имитатор отключает его от канала, имитируя режим прямого доступа в память.

Дополнительные модули. Расширитель интерфейса (рис. 6) используется для согласования двух

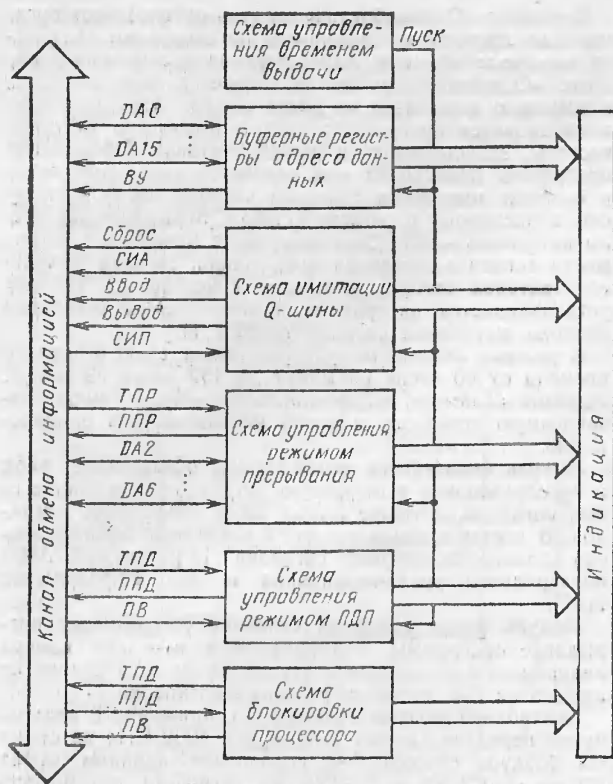


Рис. 5. Структурная схема имитатора канала

каналов типа МПИ с уровнями, соответствующими КМОП-схемам. Это дает возможность увеличить длину канала при подключении к магистрали более 20 модулей. Применяемая схема согласования каналов позволяет сохранить его быстродействие.

Преобразователь уровней. Структурная схема преобразователя аналогична структурной схеме модуля расширителя интерфейса. Схема согласования каналов в данном случае обеспечивает уровни, соответствующие ТТЛ-схемам. Использование преобразователя позволяет увеличить информационную производительность системы за счет подключения более быстродействующих процессоров, выполненных на ТТЛ-элементах.

Регистровое запоминающее устройство (рис. 7) применяется в случае использования системы в качестве управляющего вычислительного комплекса для хране-

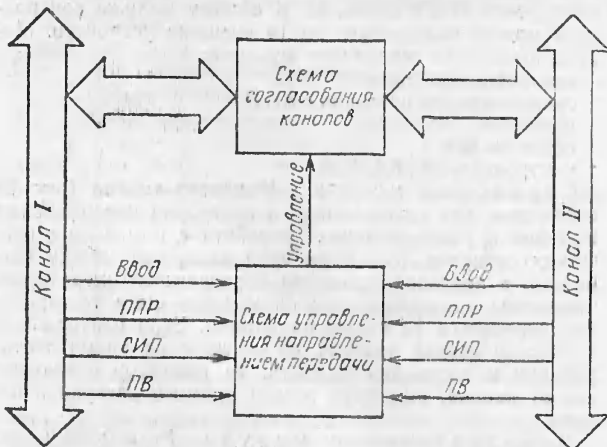


Рис. 6. Структурная схема расширителя интерфейса

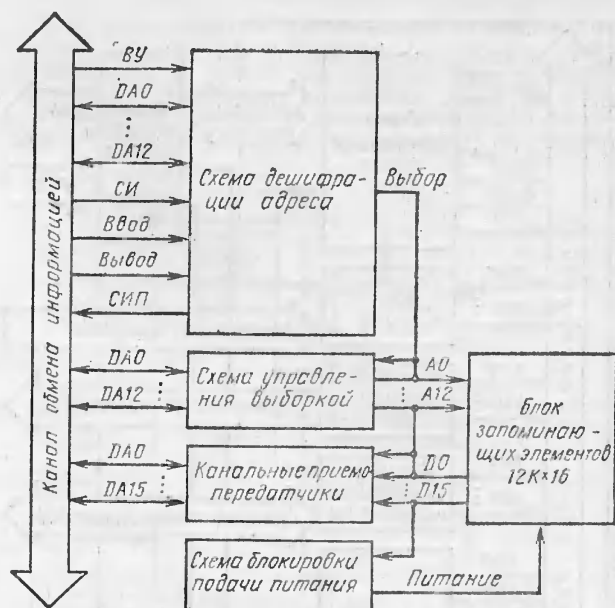


Рис. 7. Структурная схема регистрового запоминающего устройства

ния больших массивов программ и данных. В качестве запоминающих элементов применяются микросхемы К573РФ2. Модуль работает в режиме чтения. Увеличение объема адресуемого пространства блока запоминающих элементов достигается благодаря использованию схемы управления выборкой: адресом ячейки памяти являются данные, передаваемые в цикле «вывод» модулю РПЗУ.

Программные модули. Пакет тестовых программ состоит из тестов проверки работоспособности аппаратных модулей комплекса микропроцессорных устройств в автоматическом и диалоговом режимах с выводом результатов на дисплей или печатающее устройство.

Программы начальной загрузки обеспечивают загрузку операционной системы через локальную вычислительную сеть или пакета прикладных программ с любого заданного внешнего устройства или памяти в автоматическом или интерактивном режиме.

Пакет прикладных программ включает, как отдельные компоненты, управляющие программы для некоторых вариантов использования описываемого комплекса аппаратно-программных средств.

Рассмотренный комплекс аппаратно-программных средств позволяет строить микропроцессорные устройства различной конфигурации в соответствии с конкретными задачами потребителя методом комплексирования модулей, в частности информационно-измерительные системы связи.

Телефон для справок: 269-80-35, Ленинград, Скворцов Владимир Алексеевич, после 19.00.

ЛИТЕРАТУРА

1. Черняковский Д. Н., Шиллер В. А., Юровский А. А. Процессор с системой команд и интерфейсом микроЭВМ «Электроника 60» на основе БИС К 588 // Электронная промышленность. — 1983. — Вып. 9. — С. 11—13.
2. Конов Е. В., Тихомиров С. Н., Черняковский Д. Н., Шиллер В. А. Модуль ОЗУ с унифицированным интерфейсом на основе БИС К 588 // Электронная промышленность. — 1983. — Вып. 9. —
3. Дж. Кофрон. Технические средства микропроцессорных систем. — М.: Мир, 1983. — 344 с.

Статья поступила 22 ноября 1985 г.

МИКРОПРОЦЕССОРНЫЙ МОДЕМ-2400 ДЛЯ КАНАЛОВ ТОНАЛЬНОЙ ЧАСТОТЫ

Устройство преобразования сигналов (модем) — один из наиболее важных элементов аппаратуры передачи данных. Оно предназначено для сопряжения цифрового оконечного оборудования с каналом связи. Его работа во многом определяет технические характеристики всего тракта передачи.

Существующие модемы для передачи данных разрабатывались исходя из требований локальных сетей связи и строились как совокупность функционально различных устройств, собранных в модульную конструкцию, а сами функциональные узлы выполнялись на аналоговых элементах с малой степенью интеграции.

При создании следующего поколения аппаратуры передачи данных требуется существенно сократить габаритно-весовые показатели при одновременном значительном улучшении технических и эксплуатационных характеристик модемов.

Для создания модемов с относительно высокими скоростями (2400 бит/с и выше) целесообразно использовать стандартные БИС и МПК БИС универсального применения (за счет микропрограммного управления выполняемыми операциями). Подобный модем содержит специализированный процессор для решения задач преобразования и обработки сигналов,

снабженный на стыке с каналом связи аналого-цифровым и цифро-аналоговым преобразователями. Микропрограммное управление достаточно просто изменяет скорость передачи информации, вид модуляции и алгоритмы приема заменой пакета программы.

В частности, полностью цифровой модем на БИС имеет по сравнению с аналоговым вариантом значительно меньшие габариты, массу и потребляемую мощность [1—4].

Авторами разработано микропроцессорное цифровое устройство преобразования сигналов (ЦУПС-2400) со скоростью передачи 2400 бит/с на основе МПК БИС серии К589 [5, 6]. Микропроцессорный вычислитель для ЦУПС-2400

ЦУПС-2400 предназначен для передачи двоичных символов со скоростью 2400 бит/с по четырехпроводным каналам тональной частоты (ТЧ) кабельных, спутниковых и радиорелейных линий связи. ЦУПС-2400 выполнен в соответствии с рекомендациями V26 и V26 бис МККТТ: в передатчике ЦУПС-2400 использована двухкратная относительная фазовая модуляция (ДОФМ); приемник ЦУПС-2400 реализует автокорреляционный метод приема.

Передатчик модема (рис. 1, а) состоит из кодера, ставящего в со-

ответствие группам данных координаты сигнала по синфазной и квадратурной осям; формирующего фильтра, обеспечивающего нужную форму спектра элементарного сигнала в основной полосе частот; преобразователя спектра (модулятора) и цифро-аналогового преобразователя ЦАП.

Формирующий фильтр — нерекурсивный цифровой фильтр, порядок которого определяется выбранной длительностью формирования и частотой дискретизации сигнала. Спектр преобразуется в цифровом виде умножением выборки выходного сигнала фильтра на взятые в отсчетные моменты значения двух синхронных несущих колебаний, находящихся в квадратуре.

Приемник модема (рис. 1, б) включает в себя аналого-цифровой преобразователь (АЦП), преобразователь Гильберта (ПГ) и блоки цифровой обработки сигнала: преобразователь спектра (ПС), решающее правило, демодулятор и устройства синхронизации несущего и тактового колебаний. Основные функции цифрового модема таковы:

1) Цифровая фильтрация (дискретная свертка) $Y_n = \sum_{k=-N}^N C_k X_{n-k}$. В об-

щем случае все входящие в соотношение величины — комплексные. Коэффициенты C_k определяют импульсную реакцию (двумерного) цифрового фильтра, а X и Y — входную и выходную последовательность соответственно. Соотношение описывает формирование спектра сигнала на передающей стороне, предварительную фильтрацию на приеме и цифровой преобразователь Гильберта.

2) Перенос спектра в цифровой форме (умножением комплексного сигнала на синхронные отсчеты несущей).

3) Демодуляция сигнала в оптимальном приемнике состоит в вычислении скалярных произведений принятого вектора сигнала и всех возможных переданных с последующим выбором наибольшего произведения.

Точность и скорость выполнения этих трех функций определяются числом и расположением значений вектора модулированного сигнала, его допустимой мощностью, характером помех в канале, а также количеством операций, выполняемых при обработке сигнала.

Главные особенности вычислителя — работа в реальном времени, большой удельный вес арифметических операций, единая синхронная процедура ввода-вывода, сохранение постоянного набора программ для данного типа аппаратуры.

Функциональная схема ЦУПС-2400 (рис. 2) содержит микропроцессорный вычислитель (МПВ) объединенной

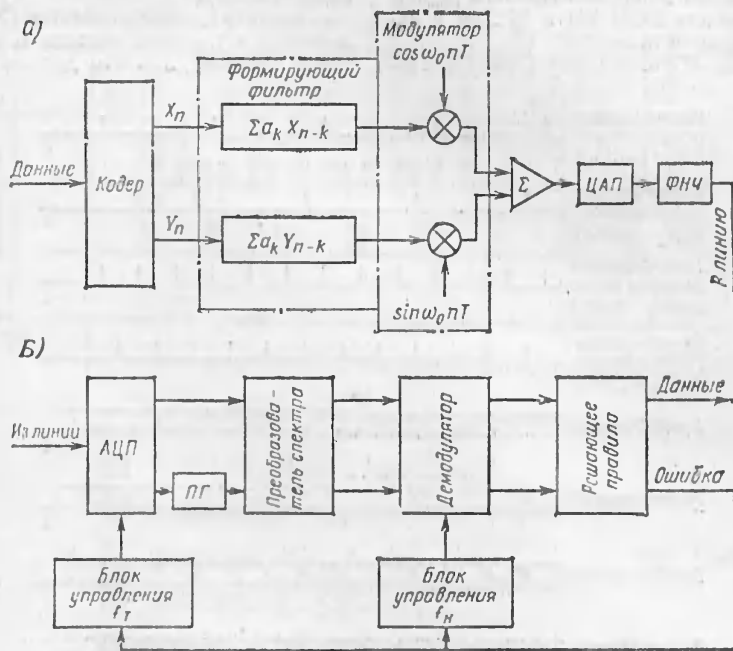


Рис. 1. Структурная схема цифрового модема ЦУПС-2400

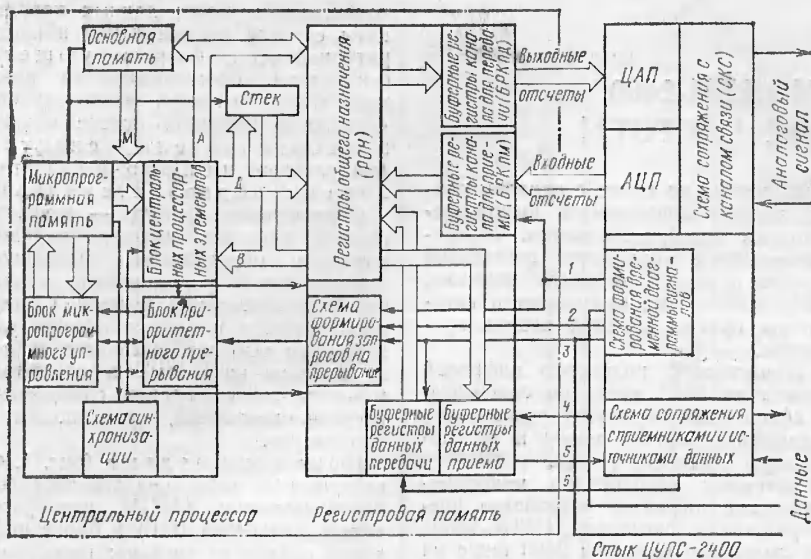


Рис. 2. Функциональная схема ЦУПС-2400

пунктирной линией) и стык. МПВ — пассивное устройство, выполняющее функции приема-передачи под воздействием внешних (по отношению к МПВ) сигналов синхронизации. В интервалах времени, свободных от выполнения указанных выше функций, МПВ ведет самоконтроль за счет резерва машинного времени. МПВ и стык связаны.

Цель «Выходные отсчеты» связывает выход АЦП с буферными регистрами канала для приема (БРКПд). БРКПд — 12-разрядный параллельный регистр. Запись в него идет под управлением ЦП. Разрядность цепи «Выходные отсчеты» определяется разрядностью ЦАП (если она не превышает 12).

Цель «Входные отсчеты» связывает выход АЦП с буферными регистрами канала для приема (БРКПм). БРКПм — 12-разрядный параллельный регистр. Запись в него идет под воздействием импульса по цепи «Синхронизация входных отсчетов». ЦП управляет только считыванием из БРКПм. Разрядность цепи «Входные отсчеты» определяется разрядностью АЦП (если она не превышает 12).

Цель «Синхронизация входных отсчетов» (1) управляет записью в БРКПм и выработкой запроса на прерывание на 5-м уровне приоритета. Цель «Синхронизация выходных отсчетов» (2) управляет выработкой запроса на прерывание на 4-м уровне приоритета.

Цель «Синхронизация передаваемой БП» (3) управляет записью в БРДПд (БП — битовая последовательность) и выработкой запроса на прерывание на 3-м уровне приоритета. БРДПд — 2-разрядный сдвигающий регистр.

Биты передаваемой БП последовательно поступают на вход БРДПд и записываются в его младший разряд

под воздействием импульса по цепи «Синхронизация передаваемой БП». Этот же импульс сдвигает код в БРДПд. Со стороны ЦП производится считывание из БРДПд в параллельном коде.

Цель «Синхронизация принимаемой БП» (4) управляет сдвигами кода в БРДПм. В результате этого биты принимаемой БП в последовательном коде выдаются на схему сопряжения с приемниками и источниками данных (СПИД) и далее на приемник данных. БРДПм — 12-разрядный регистр. После каждого двенадцатого сдвига вырабатывается сигнал, управляющий переносом из промежуточного регистра области РОН РП в БРДПм в параллельном коде.

Цель «Принимаемая БП» (5) свя-

зывает выход БРДПм со схемой СПИД (цель последовательная).

Цель «Передаваемая БП» (6) также последовательная. Она связывает схему СПИД с входом БРДПд.

Схема формирования временной диаграммы (СФВД) обеспечивает сигналы синхронизации в соответствии с временной диаграммой сигналов между МПВ и стыком модема (рис. 3). На основе сигналов синхронизации, поступающих в МПВ, вырабатываются запросы на прерывание, под воздействием которых запускаются соответствующие программы ЦУПС-2400. Для правильной обработки необходимо привязать отсчеты и битовые последовательности (БП) к сигналам синхронизации и сигналам синхронизации друг к другу:

— для привязки передаваемой последовательности БП к сигналу в источник данных подается сигнал частоты f_1 ;

— для привязки выходных отсчетов к сигналу частоты f_2 МПВ выдает отсчеты при передаче в БРКПд с некоторой задержкой (обусловленной системой прерывания);

— частота f_1 вырабатывается СФВД делением в три раза частоты f_3 , т. е. данные сигналы жестко связаны по фазе;

— для привязки входных отсчетов к сигналу частоты f_4 схемой стыка в СФВД должна содержаться схема тактовой синхронизации принимаемого сигнала;

— привязка принимаемой БП к сигналу частоты f_2 обеспечивается МПВ. В этом случае задержка обусловлена параметрами схемы БРДПм. Сигнал f_2 может быть подан на приемник данных;

— частота f_2 вырабатывается СФВД делением в три раза частоты f_4 (тактовой частоты), так что данные сиг-

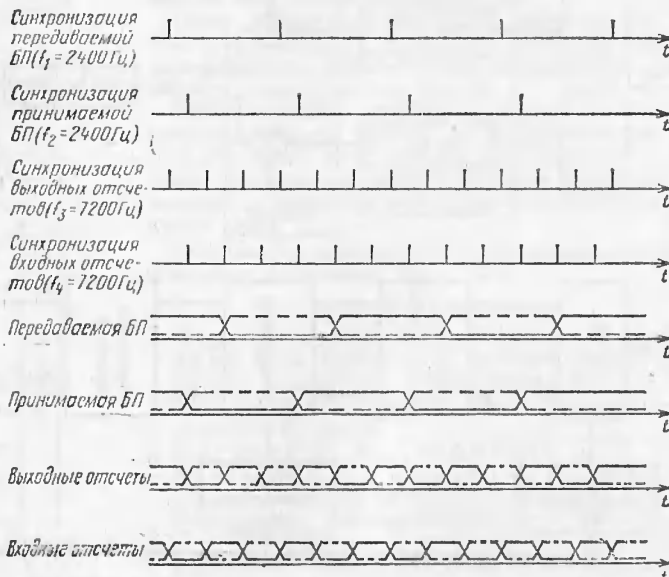


Рис. 3. Временная диаграмма сигналов между МПВ и стыком модема

валы также жестко связаны по фазе; — частоты f_3 и f_4 могут быть как жестко связанными по фазе (например, на определенной стадии испытаний), так и не связанными по фазе (для обеспечения независимости приема и передачи).

Основные алгоритмы функционирования ЦУПС-2400 для удобства пользования представлены в системе команд МПВ в виде следующих программ: 1 — подготовка исходных данных для программ 2, 3, 7, 6; 2 — накопление входных отсчетов на приеме, запоминание группы из шести отсчетов и выработка запроса на запуск программы 3; 3 — вычисление уравнения свертки, формирование принятого сигнала дибита (так называется сигнал, содержащий два бита (0 и/или 1) информации, следующие друг за другом во времени) и подготовка исходных значений сигналов для последующего цикла работы на приеме; 4 — формирование и запись в БРД очередного дибита на приеме; 5 — вывод очередного бита в цепь «Принимаемая БП»; 6 — преобразование исходного дибита на передаче, запоминание преобразованного дибита, формирование первого отсчета и подготовка данных для следующего

цикла работы программы 6 и программы 7; 7 — вычисление пяти последующих выходных отсчетов на передаче и запись этих отсчетов и первого отсчета в БРКд.

Все указанные программы распределены по уровням приоритета: уровень 0 — программа 1 и тест МПВ; 2 — программа 3; 3 — программа 6; 4 — программы 7; 5 — программа 2; 7 — программа фиксации аварийного состояния; уровни 1 и 6 прерываний оставлены свободными.

Алгоритмы передачи и приема ЦУПС-2400 реализуются в соответствии с указанными уровнями приоритетов (рис. 4).

ЦУПС-2400 может работать в нескольких режимах.

В режиме отладки и диагностических проверок к МПВ должен быть подключен специальный пульт с органами управления и индикации для визуального контроля правильности выполнения микропрограмм ЦУПС-2400.

В режим автономной работы ЦУПС-2400 входит после включения. Вначале выполняется программа подготовки исходных данных для программ приема и передачи ЦУПС-2400, после нее циклы теста

МПВ повторяются до тех пор, пока включен данный режим.

Режим работы в реальном масштабе времени (РМВ) — основной режим работы ЦУПС-2400 (разрешается обслуживание запросов на прерывание). Так как тест МПВ выполняется на 0-м уровне, то любой запрос, вырабатываемый СФЗП, имеет более высокий приоритет и, следовательно, прерывает тест. В свою очередь, запросы имеют одинаковый приоритет. Это иногда вызывает прерывание программы, ранее прервавшей выполнение менее приоритетной программы (см. рис. 4).

В режим фиксации аварийного состояния МПВ переходит, когда тестом определяется ошибка в выполнении какого-либо контрольного примера или когда на микропрограммном уровне находится неправильный переход в микропрограмме. В режиме фиксации аварийного состояния устанавливается 7-й (наивысший) уровень приоритета, о чем сигнализирует специальная индикация на панели платы процессора. Так как у всех запросов, поступающих из СФЗП, приоритет более низкий, то они в данном режиме не обслуживаются.

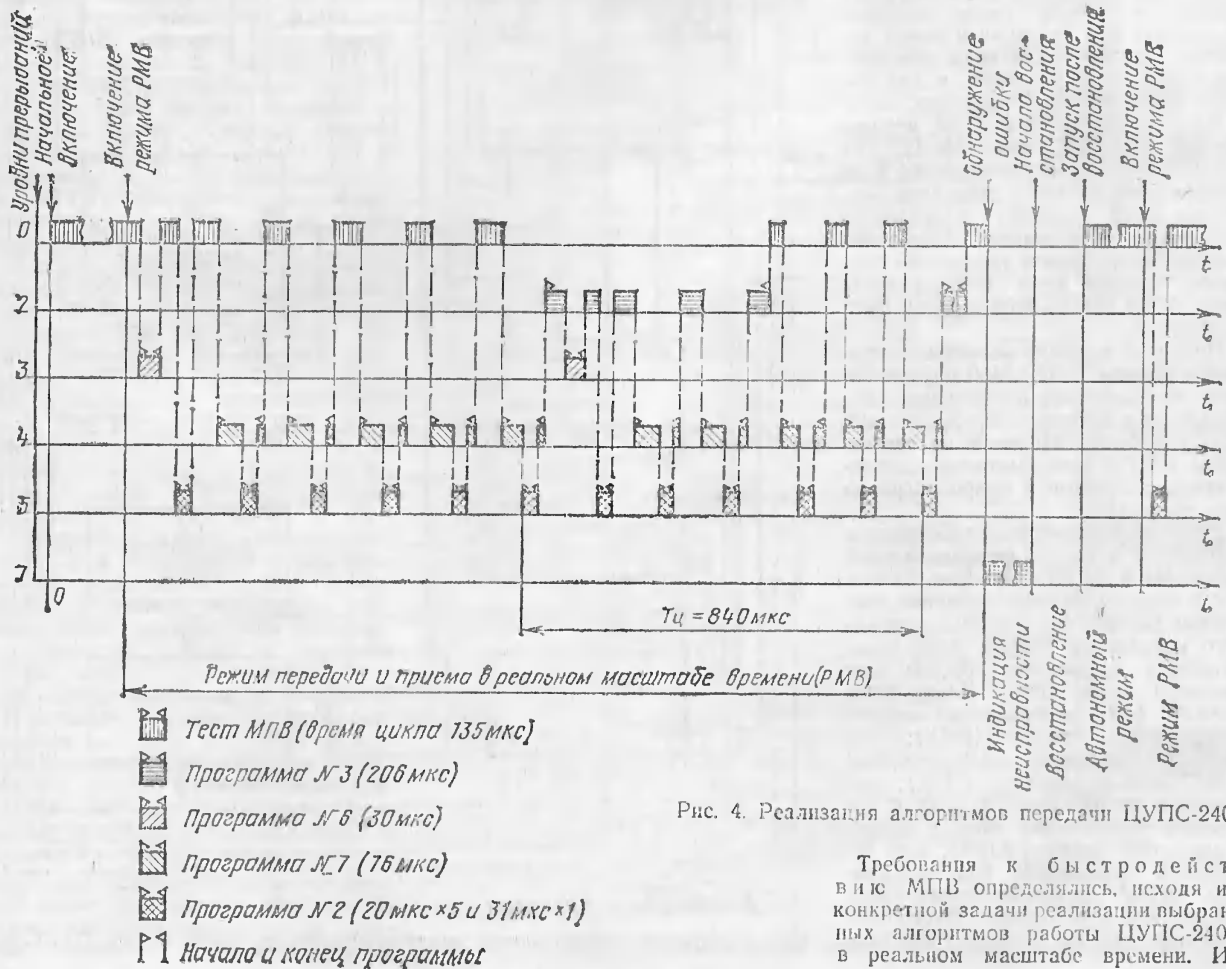


Рис. 4. Реализация алгоритмов передачи ЦУПС-2400

Требования к быстродействию МПВ определялись исходя из конкретной задачи реализации выбранных алгоритмов работы ЦУПС-2400 в реальном масштабе времени. Из

временной диаграммы (рис. 4) видно, что для правильного функционирования ЦУПС-2400 требуется, чтобы все критические по времени программы выполнялись за время цикла $T_{ц} = 840$ мкс. Если учесть, что общее количество команд, которые должны выполняться за это время, равно приблизительно 250, то среднее быстродействие МПВ должно быть не менее 1,5...2 млн. операций/с.

При выборе разрядности МПВ учитывались точность представления исходных данных; экономия слов команды для уменьшения количества обращения к памяти; разрядность используемых в ЦУПС-2400 ЦАП и АЦП; градации разрядности, обусловленные выбранной элементной базой.

Учитывая, что константы и параметры в исходных данных для алгоритмов ЦУПС-2400 заданы с точностью 10^{-3} , для ее сохранения при использовании двоичного кода в МПВ требуется использовать 10 разрядов. Кроме того, один разряд требуется для определения знака вычислений. При разработке системы команд для данного МПВ разрядность слова команды была выбрана равной разрядности операнда. Для уменьшения количества обращений к памяти было решено в первом слове команды совместить код операции и номер регистра, при этом для кода операции достаточно 6...7 разрядов, а для номера регистра — 5...6 разрядов.

Разрядность ЦАП и АЦП, используемых в ЦУПС-2400, — 10...12. Так как в набор микропроцессорного комплекта К589 входят 2-разрядные элементы ЦПЭ, поэтому разрядность процессора, построенного на этих элементах, может иметь количество разрядов, кратное двум. Итак, разрядность МПВ ЦУПС-2400 должна быть равной 12.

Приятный принцип реализации программ работы ЦУПС-2400 обуславливает использование конфигурации вычислителя с внешними прерываниями. Для обработки запросов на прерывание в МПВ предусмотрены соответствующие схемные и микропрограммные средства.

Для удовлетворения рассмотренных требований в состав функциональной схемы МПВ ЦУПС-2400 (рис. 5) входят 6 центральных процессорных элементов (ЦПЭ); блок микропрограммного управления (БМУ); блок приоритетного прерывания (БПП) (все элементы серии К589). Кроме того, должны быть использованы микропрограммная память (ПЗУ); память для хранения микропрограмм ЦУПС-2400 и констант (ПЗУ); для хранения оперативных данных и буферного запоминания данных обмена с внешними цепями (ОЗУ); для сохранения информации при прерываниях (ОЗУ стекового типа). Для обеспечения необходимого быстродействия все эти виды памяти должны

иметь малый цикл обращения. К пересчисленным основным можно добавить дополнительные блоки синхронизации и управления.

Некоторые операции над входными отсчетами и привязка работы МПВ к сигналам стыков С1 и С2 реализованы за счет регистровой памяти (РП). Внутренние регистры ЦПЭ сохраняют специальное стековое ОЗУ (СТ).

Дадим краткое пояснение работы функциональной схемы МПВ [7]. Программа, заложенная в основную

память, поступает в ЦПЭ через буферный регистр на шину М. Код операции и маску ЦПЭ получают из микропрограммной памяти (МКП). На схему ускоренного переноса задействованы выходы X, Y, C, ЦПЭ. Управление шинами адреса и данных ЦПЭ отсутствует, поэтому входы ВА, ВД находятся в состоянии разрешения. Из ЦПЭ может поступать адрес ОП или РП, а к коду памяти происходит обращение — определяет дешифратор адреса. Из МКП выходит шина управления памятью, вклю-

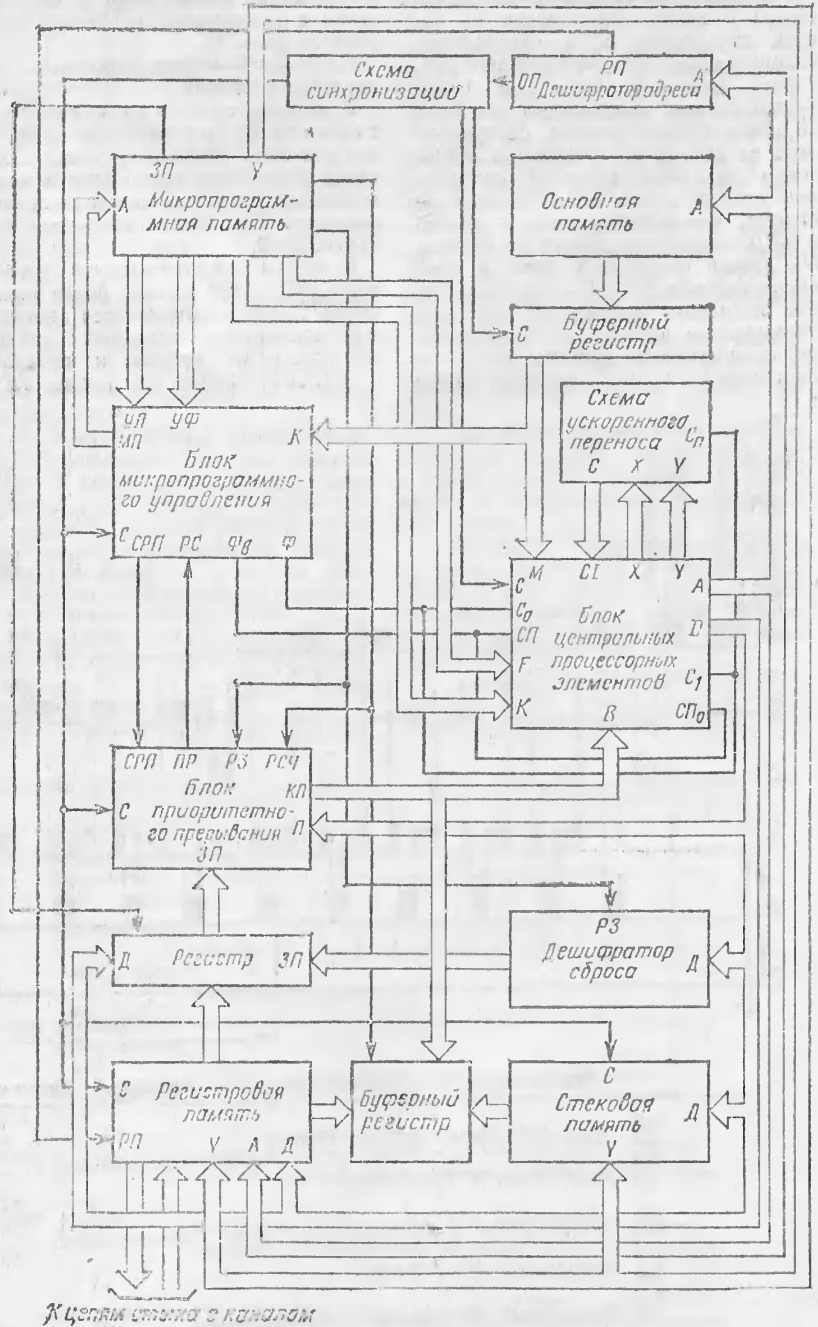


Рис. 5. Функциональная схема микропроцессорного вычислителя ЦУПС-2400

чающая сигналы: считывание (СЧ), запись (ЗП), прерывание из стека (СЧС), запись в стек (ЗПС).

По шине данных информация из ЦПЭ поступает в РП, СТ, блок приоритетного прерывания (БПП), регистр запросов на прерывание. Запросы на прерывание устанавливаются в регистре ЗП как из РП, так и программно, причем РП устанавливает уровни 3, 4 и 5, а программно устанавливаются уровни 2 и 7. БПП выдает уровень принятого запроса на В-шину по сигналу РСЧ из МКП, при этом буферный регистр СТ и РП переводится в третье состояние. Запросы на прерывание сбрасываются с помощью дешифратора сброса ЗП по сигналу РЗ из МКП, у которого адресные шины подключены к шине данных. Блок микропрограммного управления (БМУ) получает код операции из ОП, для чего использует семь старших разрядов слова ОП. Выходы адреса строки переводятся в третье состояние сигналами ПР из БПП при поступлении нового, более приоритетного уровня прерывания. Адрес из БМУ передается в МКП, которая выдает 32-разрядное слово микрокоманды, включающее в себя биты управления адресом (УА) и управления флагом (УФ), поступающие обратно в БМУ. Ввод флага соединен с С₀ старшего ЦПЭ. Вывод флага соединен с С₁ старшего ЦПЭ. Для сохранения информации из ОП в последующих циклах использован буферный регистр, который стробируется при условии выбора ОП и считывания (СЧ). Так как данные на выходе ОП появляются с задержкой, строб подается через 70 нс после синхримпульса.

Для синхронизации элементов МПВ используется схема синхронизации (СС). Схема синхронизации выдает синхримпульсы каждый такт для БМУ, БПП, СТ и РП. На ЦПЭ синхримпульсы проходят, если отсутствует блокировка синхримпульса ЦПЭ, для чего служит бит слова микрокоманды. Схема синхронизации выдает также строб на буферный регистр М-шины. Частота следования синхримпульсов — 4 МГц.

Регистровая память обеспечивает сопряжение вычислителя с устройством ввода-вывода на канал и на оборудование данных, запоминание промежуточных значений при выполнении программ приема и передачи ЦУПС-2400.

В соответствии с этим в составе РП можно выделить несколько отдельных регистров или групп регистров, имеющих индивидуальное управление. Согласно функциональной схеме РП и СТ (рис. 6) входная информация поступает с выходов АЦП в регистр Р₀. Частота поступления информации $f_{\text{АЦП}} = 7,2$ кГц. Очередной отчет записывается под воздействием импульса синхронизации входных отсчетов (сигнал «Входной

строб»). Одновременно с записью выработывается запрос прерывания, сигнал «ЗП», подаваемый в процессор. Информация считывается по инициативе процессора на шину В. Разрядность входной информации — 12 бит.

Первая группа регистров (Р₁...Р₆ и Р₁₂...Р₁₅) обеспечивает запоминание промежуточных значений при выполнении программ приема и передачи модема. Все эти регистры адресуются как ячейки памяти, работающие на запись и считывание. Входная информация поступает с Д-шины (разрядность — 12 бит). Запись и считывание производится при наличии адреса регистра и сигналов

«Запись» и «Считывание». Выход — через буферный регистр на В-шину.

Во вторую группу регистров входят Р₇, Р₈, Р₉; регистр Р₇ адресуется как ячейка РП, работающая на запись и считывание (разрядность — 12 бит). Вход — с шины Д, выход на В-шину — через буферный регистр. Регистр Р₈ — сдвиговой. Входная информация при сдвиге — состояние цепи «Передаваемая БП» в момент поступления импульса сдвига. Строб сдвига поступает по цепи «Синхронизация передаваемой БП». Регистр Р₉ адресуется как ячейка РП, работающая на запись и считывание (разрядность — 2 бита). Вход —

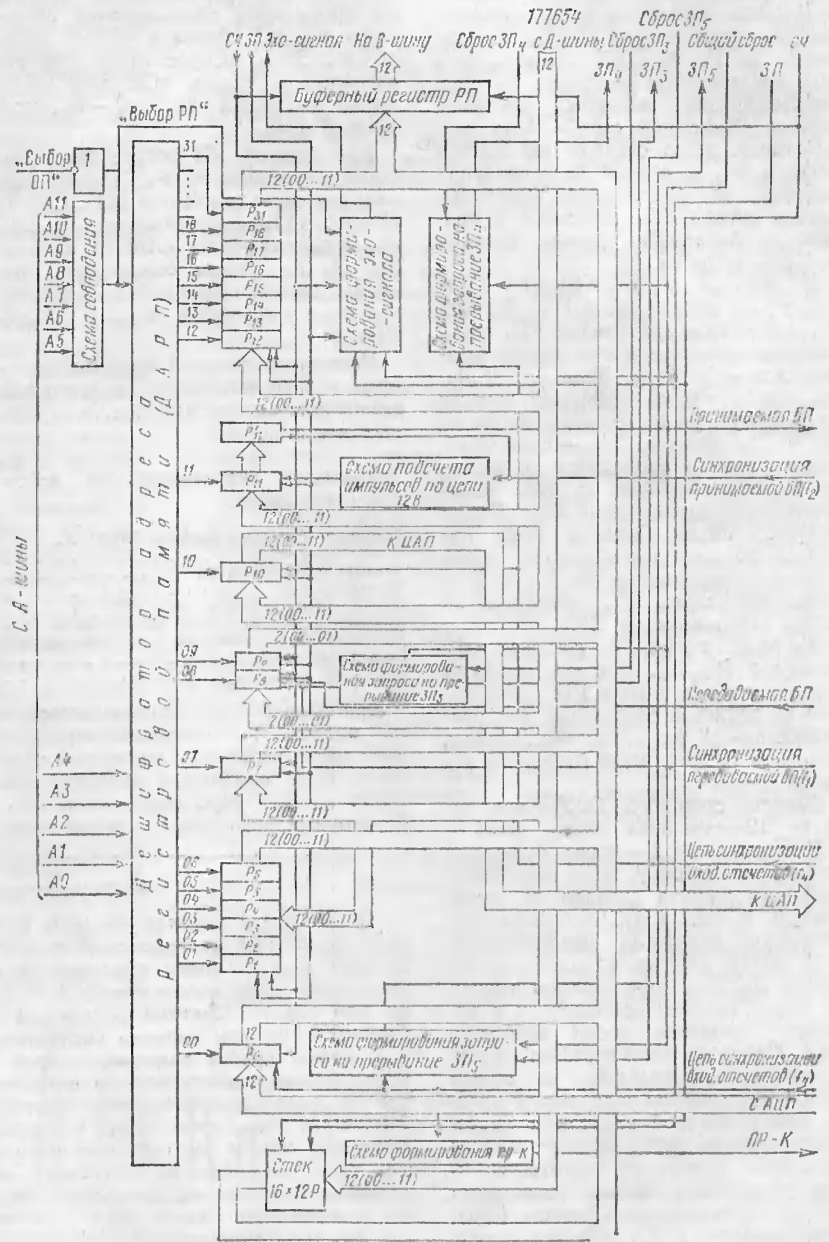


Рис. 6. Функциональная схема регистровой памяти и стекового ОЗУ.

с шины D, выход — через буферный регистр на В-шину. При заполнении регистра R_8 схема формирования запроса прерывания формирует сигнал ЗП₃, который поступает в МПВ. Активный уровень «ЗП₃» — «Лог. 0». Сдвиг в регистре R_8 происходит по заднему фронту сигнала «Синхронизация передаваемой БП». Регистр R_9 адресуется как ячейка памяти РП, работающая на запись и считывание. Разрядность — 2 бита. Вход — с Д-шины, выход — через буферный регистр на В-шину.

В третью группу регистров входит R_{10} , который адресуется как ячейка РП, работающая на запись (разрядность — 12 бит). Вход — с Д-шины, выход на цифро-аналоговый преобразователь (ЦАП). Считывание из R_{10} в ЦАП производится под воздействием импульса по цепи синхронизации выходных отсчетов, сигнал «Выходной строб». Под воздействием этого сигнала включается регистр R_{10} и схемой формирования запроса на прерывание вырабатывается сигнал ЗП₄ (подается в процессор). Активный уровень сигнала «ЗП₄» — «Лог. 0».

В четвертую группу регистров входят R_{11} и R'_{11} . Регистр R_{11} адресуется как ячейка РП, работающая на запись и считывание (разрядность — 12 бит). Вход — с Д-шины, выход — через буферный регистр на В-шину и параллельным кодом на R_{11} и R'_{11} . Регистр R'_{11} — сдвиговый. Сдвиги вправо проводятся по заднему фронту импульса по цепи «Синхронизация принимаемой БП». Схема подсчета после каждого 12-го импульса по цепи вырабатывает строб переписи информации из R_{11} в R'_{11} . Младший разряд R'_{11} выводится на выход «Принимаемая БП».

Пятую группу регистров образуют $R_{12}...R_{31}$. Все эти регистры адресуются как ячейки РП, работающие на запись и считывание (их разрядность — 12 бит). Вход — с Д-шины, выход на В-шину через буферный регистр.

Емкость стекового ОЗУ — шестнадцать 12-разрядных чисел. Вход — с Д-шины, выход — через буферный регистр на В-шину. При появлении переднего фронта сигнала на входе «Запись в стек (ЗПС)» содержимое адресного указателя увеличивается на 1. Затем данные, находящиеся на входах $D_0...D_{11}$, записываются в ячейку, адрес которой образовался в адресном указателе после прибавления 1. При появлении переднего фронта сигнала «Считывание из стека» (СИС) считывается содержимое ячейки, номер которой записан в адресном указателе, затем из содержимого адресного указателя вычитается 1. Для управления всеми регистрами формируются сигналы «Запись» (ЗП), «Считывание» (СЧ), «Запись стек», «Считывание стек» с помощью формирователей импульсов, выполненных

на D-триггерах. Эти сигналы формируются при наличии данных, поступающих с процессора, и синхронимпульса.

Выходы всех регистров имеют элементы с открытым коллектором, объединены по выходу в схему «ИЛИ» и соединены со входами буферного регистра. Информация строится в буферном регистре сигналом, сформированным дизъюнкцией сигналов СЧ или СЧС. Буферный регистр сохраняет информацию, записанную в него при считывании из любого регистра РП независимо от других сигналов управления. Выход буферного регистра управляется сигналом УПР МБР. Схемы формирования запроса на прерывание сбрасываются сигналами «Общий сброс» и «Сброс ЗП₃», «Сброс ЗП₄», «Сброс ЗП₅». Все эти сигналы, а также «Сигналы ЗП» имеют активный уровень — потенциальный ноль.

Дешифратор РП обеспечивает выборку регистров $R_0...R_{31}$ только при наличии сигнала «Выбор РП». Сигнал «Выбор РП» вырабатывается в случае появления всех «Лог. 0» на входах $A_5...A_{11}$ схемы совпадения. Выходной сигнал «Выбор ОП» — инверсный по отношению к сигналу «Выбор РП».

Микропроцессорный вычислитель — эффективное аппаратное средство для реализации алгоритмов среднескоростных модемов.

Технические характеристики действующего макета

Скорость передачи данных, бит/с	2400
Место модуляции	ДОФМ
Частота модуляции, Гц	1200
Метод приема	Автокорреляционный
Сдвиг частот в канале связи, Гц	Около 10
Среднее время наработки на отказ, ч	Около 4000
Мощность, потребляемая от источников 5 В постоянного тока, ВА	20
Габаритные размеры, мм	300x150x220
Масса, кг	Около 1

При разработке микропроцессорного модема на скорость передачи 2400 бит/с по каналу тональной частоты был изготовлен действующий макет на базе БИС серии К589. Лабораторные испытания макета под-

твердили эффективность функционирования модема при микропрограммной реализации его алгоритмов.

Заключение

Микропроцессорный ЦУПС-2400 бит/с для канала тональной частоты на элементной базе 4-го поколения обеспечивает существенное сокращение массо-габаритных размеров, а также на порядок уменьшает энергопотребление и стоимость разработки технических средств передачи данных.

Телефон для справок — 287-23-56.
Николаев Владимир Евгеньевич. После 19⁰⁰.

ЛИТЕРАТУРА

- Harris H., All digital 9600 BPS LST modem // IEEE NTC, 1974. San-Diego, Calif., 1974, F 1.
- Семейство вычислительных элементов серии Интел 3000. — ПИП СССР, перевод № 8850 — М. — 1978.
- Gerwen P. J. van, Verhoeckx N.A.M., Essen M. A. van, Shijders F.A.M. Microprocessor, implementation of high-speed data modems // IEEE Trans. — Com-25. — 1977. — № 2.
- Астанкович К. Ф., Буянов В. Ф., Егоров В. А. и др. Цифровой модем 9600 для каналов ТЧ. — М.: Электросвязь. — 1984. — № 3.
- Микропроцессорный комплект БИС ТТЛ с диодами Шоттки. Серия К589. Справочные данные. 1977.
- Мизин И. А., Муравьев С. К., Матвеев А. А. Микропроцессорное цифровое устройство преобразования сигналов (ЦУПС-2400) //

Сб. научных статей № 2. — Рига, РВВАИУ им. Я. Алксниса, 1983.
7. Козлов В. И., Матвеев А. А. Микропроцессорный вычислитель для ЦУПС-2400 // Там же.
Статья поступила 30 апреля 1986 г.

Информационное сообщение

В Киевском политехническом институте на кафедре автоматизации экспериментальных исследований разработана многокритериальная система для отладки программного обеспечения микропроцессора KP5801K80. В качестве базовой ЭВМ используются М-6000, СМ-1. Операционные системы — ДОС РВ или RTE-11. Система рассчитана на видеотерминалы типа ДМ-2000 или СИД-1000, но при наличии соответствующих драйверов возможно применение любых других видеотерминалов. Отладочная система обеспечивает одновременную работу восьми пользователей и включает в себя удобный в работе диалоговый редактор, допускающий кадровое или строчное редактирование, кросс-транслятор и отладчик. При отладке микропроцессорных программ объем имитируемой памяти равен 12К байт, и при желании может быть расширен до 64К байт. Система позволяет также подготовить и отладить прикладные программы на алгоритмических языках Фортран, Алгол и мнемокод. Планируется включить в систему кросс-транслятор и отладчик для микропроцессора KM1810VM86.

За справками обращаться по адресу: 252056, Киев, пр. Победы, 37. Кафедра АЭИ, т. Федиву В. В. Телефон 441-97-67.

АППАРАТНО-ПРОГРАММНЫЕ СРЕДСТВА ОБРАБОТКИ СИГНАЛОВ НА БАЗЕ БИС КР580

УДК 681.325.5—181.4

В. Ю. Солонин

СЕЛЕКЦИЯ ИМПУЛЬСОВ ПОЛЕЗНОГО СИГНАЛА МИКРОПРОЦЕССОРОМ КР580ИК80А

Нетрадиционное использование микропроцессора (МП) в амплитудных селекторах резко повышает помехозащищенность, так как можно подавить помехи с амплитудой как меньше, так и больше полезного сигнала. МП адаптирует верхнее и нижнее пороговые напряжения к плавно (инерционно) изменяющейся амплитуде полезного сигнала и скорости ее изменения. По критерию инерционности изменения амплитуды полезного сигнала и происходит выборка полезных импульсов. Импульс полезен, если его амплитуда отличается от амплитуды предыдущего полезного импульса не больше чем на величину допуска, определяемого микропроцессором по скорости изменения амплитуды полезного сигнала. МП-селектор более помехозащищен по сравнению с полностью аналоговым селектором*, выбирающим импульсы полезного сигнала по этому же критерию, так как уменьшает промежуток между пороговыми напряжениями при уменьшении разности амплитуд двух соседних импульсов полезного сигнала. Цифровое селектирование по сравнению с аналоговым более точно устанавливает верхнее и нижнее пороговые напряжения, исключает возможность их самопроизвольного снижения при отсутствии полезных импульсов, а значит, уменьшает промежуток между порогами, уменьшив этим вероятность ошибок.

Импульсы (рис. 1), вершины которых попадают в промежуток между верхним (Н) и нижним (L) пороговыми напряжениями, воспринимаются МП-селектором как импульсы полезного сигнала, а остальные импульсы — как помехи. На временной диаграмме видна адаптация пороговых напряжений Н, L к амплитуде полезных импульсов и адаптация промежутка Δ между пороговыми напряжениями Н, L к скорости изменения амплитуды полезного сигнала. В момент прихода полезного импульса 1 разность пороговых напряжений Н, L (промежуток Δ между ними) меньше, чем в момент прихода полезного импульса 2. Это вызвано меньшей разницей в амплитудах двух соседних импульсов полезного сигнала.

Для наибольшей наглядности цифровой селекции импульсов полезного сигнала по критерию инерционности изменения их амплитуды выбран микропроцессор КР580ИК80А. Он позволяет обойтись без ОЗУ, так как имеет достаточное количество регистров для хранения данных, необходимых для селекции импульсов,

* Солонин В. Ю. А. С. СССР № 978366. — Оpubл. в Б. И., 1982, № 44, с. 252.

Время между вершиной полезного импульса и изменением пороговых напряжений Н, L — это время выполнения микропроцессором цикла программы (180 тактов микропроцессора). Если длительность такта — 0,4 мкс, то вся программа (рис. 2) выполняется за 72 мкс (минимально допустимое время между вершинами двух соседних импульсов полезного сигнала). Все импульсы, поступившие после полезного импульса раньше, чем через 72 мкс, расцениваются как помехи и подавляются.

Для повышения быстродействия селекции следует реализовать на более быстродействующей цифровой элементной базе или на основе аналоговой вычислительной машины.

Представленная в цифровом виде амплитуда каждого принятого с порта 1 (PORT 1) полезного импульса или помехи (строка 5) сравнивается с нижним пороговым напряжением L (строка 7) и верхним пороговым напряжением Н (строка 10). Если амплитуда импульса меньше нижнего L или больше верхнего Н пороговых напряжений, то это помеха. Пороги Н, L остаются неизменными и вводится амплитуда очередного импульса (переход командами строк 9, 14 на метку В2).

Если амплитуда принятого импульса больше нижнего L, но меньше верхнего Н пороговых напряжений, то это полезный импульс. В PORT 2 вводится число 1 (строки 33, 34), обозначающее, что принятый импульс полезный. После приема импульса полезного сигнала устанавливаются (корректируются) пороговые напряжения Н, L. Для этого путем переноса в регистр принятый импульс представляется предыдущим, а предыдущий импульс — поступившим перед предыдущим (строки 15, 18). Рассчитывается величина С — половина разности между верхним и нижним пороговыми напряжениями (строки 20—26) по формуле $C = |(D - E) \cdot K|$, где D — амплитуда предыдущего полезного импульса; E — амплитуда полезного импульса, поступившего перед предыдущим импульсом полезного сигнала; K — коэффициент запаса, выбираемый по предварительному анализу принимаемого сигнала так, чтобы последующий полезный импульс не попал своей вершиной за пороговые напряжения и не был подавлен. Это должно учитываться даже, если последующий полезный импульс отличается по амплитуде от предыдущего больше, чем предыдущий от импульса, поступившего перед предыдущим. В описываемой программе коэффициент запаса принят равным 2.

Затем рассчитываются новые значения верхнего ($H = D + C$) и нижнего порогового напряжения ($L = D - C$).

Недостаток преобразователя напряжения — код заключается в дискретизации аналоговых величин амплитуд импульсов. Кодирование приводит к возможности нулевого значения разности D—E, при котором сливаются (сливаются в один) пороговые напряжения Н, L. В этом случае разница Δ между ними становится равной нулю, в результате чего подавляются как по-

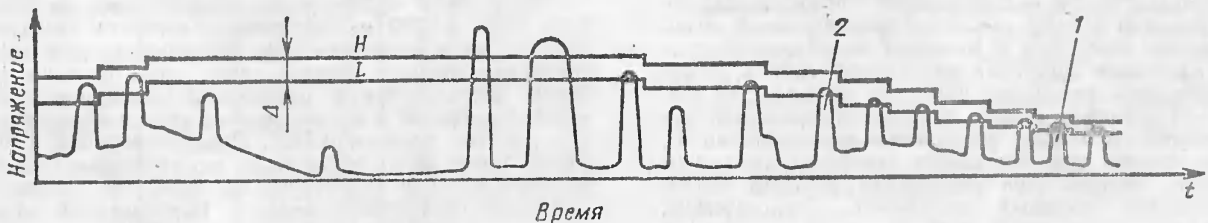


Рис. 1. Временная диаграмма импульсного сигнала

1 B1:	IN	PORT 1	;ПРИЕМ КОДА АМПЛИТУДЫ ИМПУЛЬСА
2			;ИЗ ПОРТА 1
3	MOV	E,A	
4	JMP	B4	
5 B2:	IN	PORT 1	;ПРИЕМ КОДА АМПЛИТУДЫ ИМПУЛЬСА
6			;ИЗ ПОРТА 1
7	CMP	L	;СРАВНЕНИЕ АМПЛИТУДЫ ИМПУЛЬСА С
8			;НИЖНИМ ПОРОГОВЫМ НАПРЯЖЕНИЕМ L
9	JM	E2	
10	CMP	H	;СРАВНЕНИЕ АМПЛИТУДЫ ИМПУЛЬСА С
11			;ВЕРХНИМ ПОРОГОВЫМ НАПРЯЖЕНИЕМ H
12	JZ	B3	;ИМПУЛЬС, ПОЛЕЗНЫЙ ДАЖЕ ПРИ РАВЕНСТВЕ
13			;ПОРОГОВОМУ НАПРЯЖЕНИЮ H
14	JP	E2	
15 B3:	MOV	E,D	;ПРЕДЫДУЩИЙ ИМПУЛЬС ПРЕДСТАВЛЯЕТСЯ
16			;ИМПУЛЬСОМ, ДОСТУПИВШИМ ПЕРЕД
17			;ПРЕДЫДУЩИМ
18 B4:	MOV	D,A	;ПРИНЯТЫЙ ИМПУЛЬС ПРЕДСТАВЛЯЕТСЯ
19			;ПРЕДЫДУЩИМ
20	SUB	E	;РАСЧЕТ D - E
21	JP	B5	
22	MOV	A,E	
23	SUB	D	
24 B5:	JNZ	B6	;ЕСЛИ A = 0, ТО D - E + 1
25	LNR	A	
26 B6:	RLC		;УМНОЖЕНИЕ НА КОЭФФИЦИЕНТ ЗАПАСА 2
27	MOV	C,A	;РАСЧЕТ ВЕРХНЕГО ПОРОГА H = D + C
28	ADD	D	
29	MOV	H,A	
30	MOV	A,D	;РАСЧЕТ НИЖНЕГО ПОРОГА L = D - C
31	SUB	C	
32	MOV	L,A	
33	MVI	C,01H	;ВЫВОД В ПОРТ 2 ЧИСЛА 1, УКАЗЫВАЮ-
34	OUT	PORT 2	;ЩЕГО, ЧТО ПРИНЯТЫЙ ИМПУЛЬС ЗАДЕЛЕТСЯ
35			;ИМПУЛЬСОМ ПОЛЕЗНОГО СИГНАЛА
36	JMP	B2	
37	END		

Рис. 2. Программа цифровой селекции полезного сигнала

мехи, так и полезные импульсы. Чтобы нейтрализовать этот недостаток, необходимо к разности D—E прибавить 1, если она равна нулю (строки 24, 25), т. е. принудительно минимально раздвинуть пороги измерения амплитуды.

Таким образом, установлены пороговые напряжения нижнего L меньше, а верхнего H больше амплитуды D предыдущего полезного импульса на величину C, пропорциональную (K—коэффициент пропорциональности) разности |D—E| амплитуд предыдущего D и поступившего перед ним E полезных импульсов. Достигнута адаптация пороговых напряжений H, L к скорости изменения амплитуды полезного сигнала. Это значительно повышает степень подавления помех, так как промежуток Δ между пороговыми напряжениями H, L уменьшается при замедлении изменения амплитуды полезного сигнала (при уменьшении разности амплитуд соседних полезных импульсов). Следовательно, вероятность попадания в этот промежуток импульсов помех уменьшена до минимума.

По амплитуде самого первого импульса программы (строки 1—4) устанавливаются пороговые напряжения H, L. Возможна начальная установка пороговых напряжений H, L и другими путями (например, непосредственно задавая величины H, L в начале программы).

В регистрах H, L, D, E, C микропроцессора, работающего по данной программе, хранятся одноименные, описанные величины H, L, D, E, C, представленные двузначными числами.

Данная селекция импульсов применима в аппаратуре связи и аппаратуре, использующей акустические сигналы в навигационном оборудовании и в других областях радиоэлектроники, где уровень помех высок.

Быстродействие выполнения данной программы микропроцессором КР580ИК80А достаточно для обработки телеграфных, телеграфных сигналов и команд управления, принимаемых с каналов связи (проводных и беспроводных). В радиолокаторах и эхолотах, а также в радионавигационном оборудовании описанный микропроцессорный селектор можно использовать при частоте поступления полезных импульсов до 14 кГц.

Адрес для справок: 245780, Сумская область, г. Котопол, пл. 50 лет Октября, д. 6, кв. 2. Телефон 3-24-82.

Статья поступила 30 апреля 1986 г.

УДК 681.323 : 621.391.266

В. Д. Байков, В. В. Кабанов, А. М. Попов

МУЛЬТИПРОЦЕССОРНАЯ ОРГАНИЗАЦИЯ ЦИФРОВЫХ ФИЛЬТРОВ НА БАЗЕ МПК БИС КР580

Цифровая фильтрация — одно из эффективных применений микропроцессорных систем. Микропроцессорная реализация цифрового фильтра по сравнению с иными аппаратными решениями дает дополнительные преимущества (главным образом возможность программного изменения характеристик и алгоритмов обработки [1]). Программная реализация цифровых фильтров на 8-разрядных микропроцессорах связана с большим количеством умножений слов двойной длины, приходящихся на выходной отсчет. В особенности это касается реализации фильтров с конечной импульсной характеристикой (КИХ-фильтров), описываемых уравнением свертки

$$y(n) = \sum_{m=0}^{N-1} h_m x(n-m),$$

где $y(n)$, $x(n)$ — отсчеты выходного и входного сигналов соответственно, h_m — отсчеты импульсной характеристики. В этой связи целесообразно разработать алгоритмы и структуры, позволяющие использовать универсальные 8-разрядные микропроцессоры для вычисления свертки при обработке сигналов в реальном масштабе времени.

Предлагаемая мультипроцессорная система на базе МПК БИС КР580 и эффективный алгоритм вычисления свертки в данной системе без использования операций над словами двойной длины ориентированы на задачи исследования и разработки алгоритмов цифровой фильтрации с использованием системы числения в остаточных классах (СОК). Применение МПК БИС КР580 в этих целях обусловлено его универсальностью, широким набором периферийных БИС и наличием апробированных средств отладки. Предлагаемый алгоритм, использующий арифметику в СОК, — это модификация метода работы [2].

Дифференциатор (N=32, разрядность исходных коэффициентов 12 бит)

Исходный алгоритм		Модификация в СОК			
m	h _m	$\Delta_j^{(2)}$	$\Delta_j^{(2)}$ mod 127	$\Delta_j^{(2)}$ mod 255	$\Delta_j^{(2)}$ mod 256
0	-5	0	0	0	0
1	7	0	0	0	0
2	-3	0	0	0	0
3	3	0	0	0	0
4	-4	0	0	0	0
5	4	0	0	0	0
6	-5	0	0	0	1
7	6	1	0	1	1
8	-8	1	0	1	1
9	10	1	0	1	1
10	-14	1	1	1	1
11	21	3	1	1	1
12	-34	6	2	1	2
13	67	20	3	6	5
14	-187	97	6	20	19
15	1660	1379	10	21	20

Как известно, в СОК целое число представляется набором вычетов x_i по выбранным взаимно простым основаниям (модулям) $p_i, i=1, r$ [3]. Диапазон чисел, единственным образом представимых в СОК, определяется произведением оснований. В отличие от позиционных систем счисления цифры x_1, x_2, \dots, x_r представления числа x в СОК независимы друг от друга. Это позволяет в соответствующей структуре организовать параллельное вычисление уравнения свертки по каждому из модулей на отдельных микропроцессорах [4, 5]. Использование СОК повышает быстрдействие и точность фильтрации, поскольку арифметические операции выполняются в СОК без округлений.

При обработке сигналов в рассматриваемой системе интересны прежде всего алгоритмы вычисления свертки по модулю p_i и преобразования чисел из СОК в двоичный позиционный код (ДПК).

Алгоритм вычисления свертки [2] использует разности переставленных коэффициентов (РПК). Рассмотрим модификацию этого алгоритма в СОК на примере полосового фильтра (табл. 1). Исходные коэффициенты фильтра взяты из [6] и округлены до 9 двоичных разрядов. Для нахождения РПК в СОК определяются вычеты коэффициентов h_m по модулям $p_i, i=1, r$. В табл. 1 приведен пример для модуля 256. Найденные вычеты упорядочиваются в соответствии с возрастанием цифр по модулю p_i . Коэффициенты g_k связаны с исходными коэффициентами перестановкой индексов: $m=\pi_1(k)$. Для переставленных коэффициентов g_k определяются первые разности $\Delta_k^{(1)}$, которые, в свою очередь, также упорядочиваются. Переставленные первые разности $\Delta_1^{(1)}$ связаны с $\Delta_k^{(1)}$ перестановкой индексов: $k=\pi_2(1)$. Аналогично можно определить вторые разности $\Delta_1^{(2)}$ и переставленные вторые разности $\Delta_j^{(2)}, 1=\pi_3(j)$.

Вычисления проводятся при расчете фильтра для всех модулей. Использование РПК снижает разрядность чисел, представляющих коэффициенты фильтра, и уменьшает число самих коэффициентов, поскольку некоторые из них обращаются в нуль.

Преимущества предлагаемой модификации по сравнению с [2] — отпадают операции над знаками входных отсчетов и отсутствуют затруднения при реализации дифференциаторов и широкополосных фильтров нижних частот. Для этих типов фильтров

исходный алгоритм незначительно сокращает разрядность наибольшего коэффициента. С помощью табл. 2 можно сравнить два алгоритма на примере широкополосного дифференциатора:

Выражение для свертки в СОК через переставленные разности второго порядка имеет вид $y_1(n) = \sum_{j=N-N_2+1}^{N-N_2+1} \Delta_{j,1}^{(2)} S_{j,1}(n)$, где $N_{2,1}$ — число ненулевых $\Delta_{j,1}^{(2)}$, а величины $S_{j,1}(n)$ получены из вычетов входного сигнала двукратным суммированием и перестановками [2]. Индекс i указывает на выполнение операций по модулю p_i .

Свертка вычисляется в мультипроцессорной системе (рис. 1). Процессоры МП1...МП3 вычисляются по мо-

Таблица 1

Полосовой фильтр (N=32, разрядность исходных коэффициентов 9 бит)

m	h _m	h _m mod 256	g _k	$\Delta_k^{(1)}$	$\Delta_1^{(1)}$	$\Delta_1^{(2)}$	$\Delta_j^{(2)}$
0	-3	253	0	0	0	0	0
1	0	0	1	1	1	1	0
2	4	4	3	2	1	0	0
3	-4	252	4	1	1	0	0
4	7	7	6	2	1	0	0
5	1	1	7	1	1	0	0
6	-10	246	34	27	2	1	1
7	3	3	43	9	2	0	1
8	-21	235	104	61	5	3	2
9	6	6	156	52	9	4	3
10	34	34	189	33	11	2	4
11	-5	251	235	46	27	16	6
12	43	43	246	11	33	6	6
13	-67	189	251	5	46	13	9
14	-152	104	252	1	52	6	13
15	156	156	253	1	61	9	16

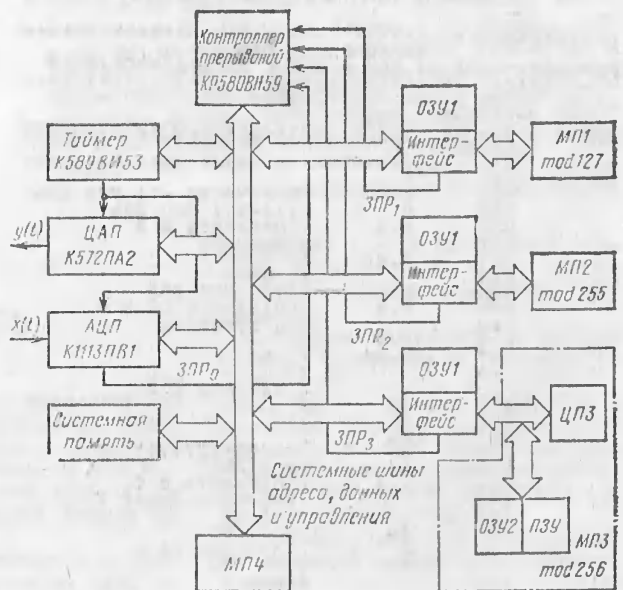


Рис. 1. Структурная схема мультипроцессорной системы для цифровой фильтрации

дулям $p_1=127$, $p_2=255$, $p_3=256$ соответственно. Это обусловлено особенностями системы команд микропроцессора КР580ИК80А и удобством преобразования чисел из ДПК в СОК и обратно. Выбранные в качестве модулей p_1 и p_2 числа имеют вид 2^n-1 . Правила выполнения арифметических операций по таким модулям совпадают с арифметикой p -разрядных чисел в обратном коде. В процессоре КР580ИК80А операции сложения и вычитания по модулям 127 и 255 выполняются после анализа флагов S и CY соответствующей коррекцией результата. Хотя каждый из процессоров МП1...МП3 8-разрядный, общая эквивалентная разрядность вычислителя (23 бита) определяется произведением модулей.

Выходные отсчеты $y(n)$, представленные в СОК, преобразуются процессором МП4 на основе Δ -алгоритма [7]. Одновременно делением на p_1 выполняется масштабирование с тем, чтобы окончательный результат мог быть представлен в двухбайтовом формате.

Числа из СОК в ДПК переводятся программно (рис. 2). МП4 может выполнять также программы, непосредственно не связанные с вычислением свертки (например, его можно загрузить расчетом новых коэффициентов фильтра). Особенность структуры (см. рис. 1) — организация связи процессора МП4 с процессорами МП1...МП3 через двухпортовые ОЗУ1. Процессорами МП1...МП3 в ОЗУ1 выводятся вычисленные отсчеты, МП4 размещает в них отсчеты входного сигнала, там же хранятся управляющие байты. Реализация ОЗУ1 на БИС К1802ИР1 обеспечивает наиболее простой интерфейс и упрощает синхронизацию процессоров. Емкость каждого ОЗУ1 — 64 байт.

В состав каждого из процессоров МП1...МП3 входят БИС КР580ИК80А, КР580ВК38, ОЗУ2 для запоминания массивов на промежуточных этапах вычисления и хранения перестановок, а также ПЗУ (512 байт) с программами фильтрации и загрузки в ОЗУ2 РПК и перестановок.

Алгоритмы обработки информации и взаимодействия процессоров в системе представлены на рис. 3. Система синхронизируется программируемым таймером КР580ВИ53. Его сигналами запускается АЦП К1113ПВ1 и стробуруется выходной регистр ЦАП К572ПА2. После окончания преобразования АЦП сигналом ЗПР₀ прерывает МП4. По прерыванию МП4 запускает процессоры МП1...МП3 на обработку отсчета

```

*****
***** ПОДПРОГРАММА CONV *****
***** ПЕРЕВОД ЧИСЕЛ ИЗ СОК В ДПК С МАСШТАБИРОВАНИЕМ *****
***** ВХОД: ЧИСЛО В СОК: (Y1, Y2, Y3) *****
***** ВЫХОД: ДВОИЧНОЕ ЧИСЛО: Y=U3*256+U2, U3<255, U2<256 *****
***** B=U2, A=U3 *****
***** CONV: *****
XRA A ; В А ЧИСЛО -Y1 MOD 256
SUB B ; (Y2-Y1) MOD 256
ADD B, A ; СОХРАНИТЬ В В
RAR ; УМНОЖИТЬ
MVI A, 0Fh ; НА
RAR ; 127 MOD 256
SUB B, A ; СОХРАНИТЬ U2 В В
RAR ; U2 УМНОЖИТЬ
MOV H, A ; НА
MVI A, 0Fh ; НА
RAR ; 127 MOD 255
ADD H, B ;
ACI B ;
SUB B, B ;
SBI C ;
ADD C, A ; V=(U2*127+Y1)
ACI C ; MOD 255
MOV C, A ; СОХРАНИТЬ В С
CMA ; ИНВЕРСИЯ ЗНАКА V
ADD L, A ; (Y3-V)
ACI L ; MOD 255
ADD A, A ; УМНОЖИТЬ НА 2
ACT A ; MOD 255
CMA ; A=U3
RET

```

Рис. 2. Подпрограмма перевода чисел из СОК в ДПК с масштабированием

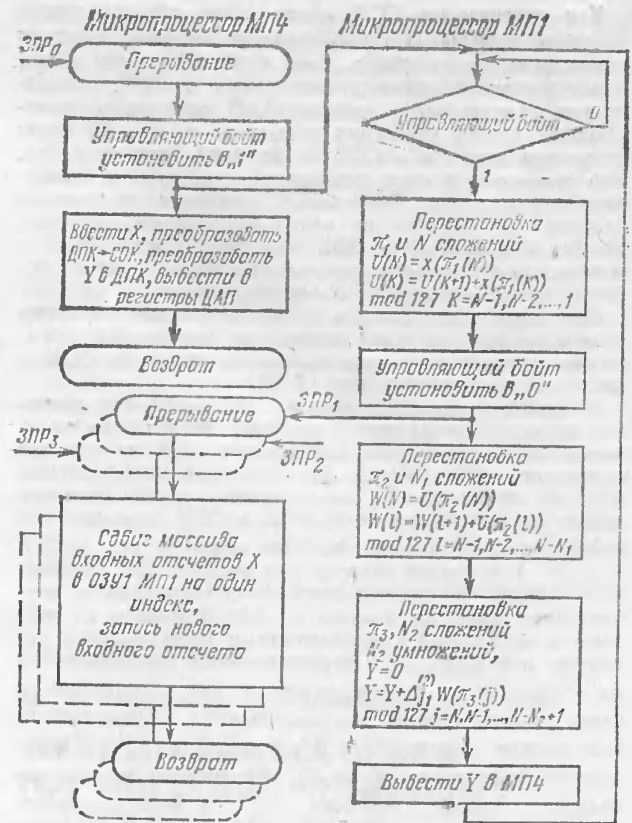


Рис. 3. Алгоритм обработки информации и взаимодействия процессоров МП1...МП4 (см. рис. 1)

$x(n-1)$, вводит из регистров АЦП отсчет $x(n)$ и преобразует его в СОК, считывает из ОЗУ1 микропроцессоры МП1...МП3 выходной отсчет $y(n-2)$ в модулярном представлении, преобразует его из СОК в ДПК и выводит в регистр ЦАП. Признак начала обработки очередного отсчета для МП1...МП3 — единичное состояние управляющих байтов в соответствующих ОЗУ1. В соответствии с рис. 3 и табл. 1, 2 для вычисления выходного отсчета в микропроцессоре МП_i ($i=1, 2, 3$) требуется выполнить N сложений в цикле по k , $N_{1,i}$ сложений в цикле по l и $N_{2,i}$ сложений и умножений в цикле по j , причем умножаются 8-битовые слова на малоразрядные (3..5 бит) коэффициенты. Программа (рис. 4), реализующая алгоритм обработки для МП1 (модуль 127), выполняется в три этапа.

На первом этапе процессоры МП1...МП3 используют свои ОЗУ1 для считывания в соответствии с перестановкой π_1 входных отсчетов, суммирования их и записи сумм в соответствующую зону ОЗУ2. После выполнения первого этапа i -й процессор изменяет на нулевое состояние управляющего байта в соответствующем ОЗУ1. При этом схемой интерфейса формируется сигнал прерывания ЗПР_i. По прерыванию процессор МП4 осуществляет сдвиг входных отсчетов в ОЗУ1 МП_i на один индекс и заносит в освободившуюся ячейку входной отсчет $x(n)$, после чего снимает запрос ЗПР_i.

Процессоры МП1...МП3 выполняют второй и третий этапы алгоритма фильтрации одновременно с обработкой прерываний процессором МП4, используя только ОЗУ2 (рис. 3). После завершения вычислений i -м процессором значение $y_1(n-1)$ заносится в ОЗУ1, а процессор переходит в режим ожидания, непрерывно анализируя состояние управляющего байта.

Частота дискретизации в системе определяется по наибольшему из двух интервалов времени. Первый ин-

```

***** ПЕРИОДЫ ПЕРИОДА ПЕРИОДА ПЕРИОДА *****
***** ВЫЧИСЛЕНИЕ СВЕРТКИ МЕТОДОМ РПК ПО МОДУЛЮ 127 *****
*****
SAVEB EQU 200H ;АДРЕС УПРАВЛЯЮЩЕГО БАЙТА
SAVEY EQU 202H ;АДРЕС ЗАПИСИ ВЫХОДНОГО ОТСЧЕТА
SAVEZ EQU 204H ;АДРЕС ПЕРЕСТАНОВКИ 3 И КОЭФФИЦИЕНТОВ
PERB1 EQU 206H ;АДРЕС ПЕРЕСТАНОВКИ 1
SAVE1 EQU 208H ;АДРЕС ЗОНЫ ЗАПИСИ 1
PERB2 EQU 20AH ;АДРЕС ПЕРЕСТАНОВКИ 2
SAVE2 EQU 20CH ;АДРЕС ПЕРЕСТАНОВКИ 2
CONB7 EQU 200FH ;СТАРОЙ БАЙТ АДРЕСА И ЧИСЛО КОЭФФИЦИЕНТ

FLTR1: NH1 D,01H ;КОНСТАНТА ДЛЯ СЛОЖЕНИЯ ПО MOD 127
        LDA 200H ;УКАЗАТЬ УПРАВЛЯЮЩИЙ БАЙТ
        AND 01H ;ПЕЧАТЬ "0" - ОБРАБОТКА
        OR 02H ;ИНАЧЕ - ОБИВАНИЕ
        LSR ;АДРЕС ПЕРЕСТАНОВКИ 1
        LSR ;АДРЕС ПЕРЕСТАНОВКИ 1
        AND 01H ;АДРЕС ЗАПИСИ
        JMP ;УКАЗАТЬ А
;СОХРАНИТЬ МЛАДШИЙ БАЙТ В E
        MOV 200H ;ПЕРЕСТАНОВКА 1
        AND 01H ;СЛОЖЕНИЕ С СОДЕРЖИМЫМ ОЗУ1
        AND 01H ;ПО MOD 127
        OR 02H ;ЗАПИСЬ В ЗОНУ 1 ОЗУ2
        MOV 200H ;НОВЫЙ АДРЕС ЗАПИСИ
        AND 01H ;НОВЫЙ АДРЕС ПЕРЕСТАНОВКИ
        OR 02H ;ВОССТАВЛЕНИЕ E
        OR 02H ;ПРОДОЛЖИТЬ, ЕСЛИ НЕ ВСЕ
        JMP ;ИНАЧЕ ОЧИСТИТЬ A

;АДРЕС УПРАВЛЯЮЩЕГО БАЙТА В "0", ПЕРЕДАНИЕ МЛА
        AND 01H ;АДРЕС ПЕРЕСТАНОВКИ
        AND 01H ;АДРЕС 2-ОЙ ЗОНЫ ЗАПИСИ
        OR 02H ;СЛОЖЕНИЕ С СОДЕРЖИМЫМ ЗОНЫ ОЗУ2
        AND 01H ;MOD 127
        OR 02H ;ЗАПИСЬ В ЗОНУ 2 ОЗУ2
        MOV 200H ;ПРОДОЛЖИТЬ, ЕСЛИ НЕ ВСЕ
        AND 01H ;ОЧИСТИТЬ A
        AND 01H ;СТАРОЙ БАЙТ АДРЕСА, С=№
        AND 01H ;АДРЕС ПЕРЕСТАНОВКИ 3 И КОЭФФИЦИЕНТОВ
        AND 01H ;КОЭФФИЦИЕНТ В L-МЛАДШИЙ БАЙТ АДР.
        AND 01H ;АДРЕС В HL
        AND 01H ;УМНОЖЕНИЕ НА КОЭФФИЦИЕНТ
        AND 01H ;MOD 127
        OR 02H ;УМНОЖЕНИЕ НЕ ЗАКОНЧЕНО - ПРОДОЛЖИТЬ
        AND 01H ;ИНАЧЕ НОВЫЙ КОЭФФИЦИЕНТ
        AND 01H ;КОЭФФИЦИЕНТЫ НЕ ВСЕ - ПРОДОЛЖИТЬ
        AND 01H ;ИНАЧЕ ВЫВЕСТИ Y В ОЗУ1
        AND 01H ;ИЖ ОЖИДАНИИ НОВОГО ОТСЧЕТА

```

Рис. 4. Программа вычисления свертки методом РПК по модулю 127

тервал — время обработки одного отсчета микропроцессором МП1, а второй складывается из времени, необходимого для выполнения первых N сложений в цикле по k в МП1 (рис. 3) и суммарного времени обработки прерываний по сигналам ЗПРi ($i=1, 2, 3$) процессором МП4. Например, для фильтра из табл. 1 программа (рис. 4) вычисляет выходной отсчет за 1,9 мс при тактовой частоте процессора 2,5 МГц, однако для выполнения 32 сложений и для сдвига массивов входных отсчетов в трех ОЗУ1 требуется 2,1 мс. Последнее значение соответствует частоте дискретизации 476 Гц и зависит только от тактовой частоты процессора и порядка фильтра.

Особенности предлагаемого способа организации цифровых КИХ-фильтров на универсальных микропроцессорах: использование СОК и вытекающий от-

сюда параллелизм структуры, практическое отсутствие в предлагаемом алгоритме вычисления свертки «длиных» операций умножения. Умножения на $\Delta_j^{(2)}$, необходимые на третьем этапе вычислений (рис. 3) для БИС КР580ИК80А, выгоднее выполнять последовательным суммированием. Программы при этом выполняются более быстро, чем по специальным подпрограммам умножения 8-разрядных чисел. Выигрыш в быстродействии за счет исключения умножений значительно превосходит потери из-за выполнения операций по модулю, отличным от 256. И, наконец, процессоры взаимодействуют так, что преобразование систем счисления и сдвиг отсчетов выполняются параллельно с обработкой. Это дополнительно повышает быстродействие в два раза.

При традиционной программной реализации КИХ-фильтров в позиционной системе счисления основные временные затраты связаны с умножениями. При разрядности операндов 10–16 бит время умножения для КР580ИК80А — 450...800 мкс [1, 8]. Таким образом, предлагаемый алгоритм вычисления свертки (рис. 4) с использованием модулярной арифметики в 7–13 раз повышает быстродействие программной реализации цифровых КИХ-фильтров на универсальных 8-разрядных микропроцессорах.

Телефоны для справок: 234-25-03, 234-84-04 (Ленинград).

ЛИТЕРАТУРА

- Алексенко А. Г., Галицын А. А., Иванов А. Д. Проектирование радиоэлектронной аппаратуры на микропроцессорах. — М.: Радио и связь, 1984. — 272 с.
- Na Kayama K. Permuted difference coefficient realisation of FIR digital filters // IEEE Trans.— Vol. ASSP—30. 1982.— № 2. p. 269—278.
- Акушский И. Я., Юдицкий Д. И. Машинная арифметика в остаточных классах. — М.: Сов. радио, 1968. — 439 с.
- Петренко А. И., Бублик С. А. Построение устройств обработки сигналов на микропроцессорах // Изв. вузов. Радиоэлектроника. — 1981.— Т. 24.— № 6.
- Soderstrand M. A., Vernia C. Microprocessor controlled development system for adaptive filtering using parallel processing and residue number arithmetic // Int. J. of Mini and Microcomputers. 1981.— Vol. 3.— № 3.— p. 39—43.
- Рабинер Л., Голд Б. Теория и применение цифровых обработки сигналов. — М.: Мир, 1978. — 848 с.
- Амербаев В. М. Теоретические основы машинной арифметики. — Алма-Ата: Наука, 1976. — 324 с.
- Цифровые фильтры и устройства обработки сигналов на интегральных микросхемах // Под ред. Б. Ф. Выскоцкого. — М.: Радио и связь, 1984. — 216 с.

Статья поступила 4 апреля 1986 г.

УДК 681.325.5—181.4

В. Д. Гинзбург

МИКРОПРОЦЕССОРНЫЙ ИЗМЕРИТЕЛЬ ПЕРИОДА СИГНАЛА

Сигналы датчиков с частотным выходом широко используются в качестве источников информации для микропроцессорных систем обработки информации и управления. Период сигнала обычно измеряют аппаратно с помощью счетчиков и других цифровых схем.

При способе измерения периода низкочастотного сигнала (до 16 кГц) с относительной погрешностью $2 \cdot 10^{-5}$ и разрешающей способностью 40 нс,

когда функцию сета выполняет сам микропроцессор КР580ВМ80А (МП), аппаратные затраты ограничены (рис. 1) триггером Шмитта для нормализации измеряемого сигнала, D-триггером и инвертором (общие затраты ≈ 1 корпус).

Суть метода заключается в подаче измеряемого сигнала на вход запроса прерывания МП. В промежутках между прерываниями МП с помощью бесконечного цикла ведет

внутренний счет в регистровой паре (D,E) с частотой, в 10 раз меньше частоты тактового генератора системы.

Измерения начинаются по сигналу «Пуск». Количество n периодов T_x измеряемого сигнала, в течение которых идет измерение, определено в программе заранее (в приведенной программе $n=250$), исходя из максимального (но без переполнения) заполнения регистровой пары (D,E) при максимальном периоде T_x .

В промежутках между измерениями процессор находится в состоянии останова.

Обработка прерывания заключается в определении его номера и принятии решения: если прерывание первое

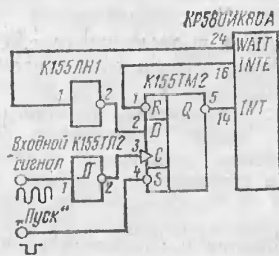


Рис. 1. Схема подключения измеряемого сигнала к микропроцессору

по счету (от сигнала «Пуск»), то ожидается следующее («фиктивный» счет); если второе, то счет начинается; если третье или последующие, то счет продолжается; если прерывание ($n+2$)-е, то счет прекращается и МП переходит к обработке результатов.

Текущая сумма каждый раз корректируется с учетом времени, затраченного на обработку прерывания.

Реализованная система выдает вектор прерывания, и его значение — RST3, RST4 или RST5 (рис. 2) определяется режимом работы измерителя (задается положением переключателя на его лицевой панели).

Период T_x в относительных единицах определяется подпрограммой TIME, вызываемой из программы, начинающихся в точках обработки прерываний. Абсолютная погрешность измерения — единица счета в регистровой паре (D,E), а относительная погрешность (если регистровая пара (D,E) близка к заполнению) $\approx 2 \cdot 10^{-5}$.

Полученный от подпрограммы TIME результат (содержимое регистровой пары (D,E)) программы используют для необходимых расчетов. В частности, период колебаний в абсолютных единицах $T = \tau K N / n$, где τ — период колебаний генератора МП-системы; K — число тактов МП, за которые текущая сумма в (D,E) возрастает на 1 (в программе $K=10$); N — конечная величина в (D,E); n — заданное число периодов колебаний измеряемого сигнала ($n=250$).

Содержание программ PROG1, PROG2, PROG3 (рис. 2) зависит от конкретного применения измерителя.

Программы PROG1 и PROG2 в реализованной системе запоминают результат работы подпрограммы TIME в ячейках оперативной памяти, затем рассчитывают по приведенной формуле абсолютную величину периода колебаний, преобразовывают результат в двоично-десятичную форму и выдают его на цифровой 6-разрядный индикатор.

Программа PROG3 использует результаты, полученные от подпрограммы TIME во всех трех режимах работы измерителя, проводит необходимые расчеты и результат, преобразованный в двоично-десятичную форму, выдает на цифровой индикатор.

Телефоны для справок: 210-42-73 (р.), 213-18-32 (д.) — Ленинград.

Статья поступила 20 марта 1986 г.

АССЕМБЛЕР / СПО СМ 1800

 * TIMER - ОПРЕДЕЛЕНИЕ ПЕРИОДА НИЗКОЧАСТОТНОГО СИГНАЛА

ИСПОЛЬЗУЕМЫЕ РЕГИСТРЫ N - ЧИСЛО ПЕРИОДОВ ИЗМЕРЯЕМОГО СИГНАЛА
 (N,L) - АДРЕС ПОДПРОГРАММЫ ROT
 (БЕСКОНЕЧНЫЙ ЦИКЛ)

		(D,E) - ТЕКУЩАЯ СУММА	
	ORG	0000H;	
	JMP	INIT;	ПЕРЕХОД К ПОДПРОГРАММЕ ИНИЦИАЛИЗАЦИИ
RST3:	ORG	0010H;	
	CALL	TIME;	ВЫЗОВ П/П ИЗМЕРЕНИЯ ПЕРИОДА T_x
	JMP	PROG1;	ПЕРЕХОД К ОБРАБОТКЕ РЕЗУЛЬТАТОВ
RST4:	ORG	0020H;	
	CALL	TIME;	ВЫЗОВ П/П ИЗМЕРЕНИЯ ПЕРИОДА T_x
	JMP	PROG2;	ПЕРЕХОД К ОБРАБОТКЕ РЕЗУЛЬТАТОВ
RST5:	ORG	0020H;	
	CALL	TIME;	ВЫЗОВ П/П ИЗМЕРЕНИЯ ПЕРИОДА T_x
	JMP	PROG3;	ПЕРЕХОД К ОБРАБОТКЕ РЕЗУЛЬТАТОВ
INIT:	ORG	0040H;	ПОДПРОГРАММА ИНИЦИАЛИЗАЦИИ
	LXI N, ROT;		В (N,L) АДРЕС НАЧАЛА БЕСКОНЕЧ. ЦИКЛА
	MVI A, 0F0H;		ИСХОД. ЧИСЛО ДЛЯ СЧЕТА ПЕРИОДОВ T_x
			ОСТАЛЬНЫЕ КОМАНДЫ ИНИЦИАЛИЗАЦИИ

	EI		РАЗРЕШЕНИЕ ПРЕРЫВАНИЯ
	HLT		КОНЕЦ ИНИЦИАЛИЗАЦИИ, ОСТАНОВ ИП.
			ОЖИДАНИЕ СИГНАЛА "ПУСК"
ROT:	INX D		БЕСКОНЕЧНЫЙ ЦИКЛ ВНУТРЕННЕГО СЧЕТА;
	PCML		ИНКРЕМЕНТ РЕГИСТРОВОЙ ПАРЫ (D,E)
			ВОЗВРАТ К ПРЕДЫДУЩЕЙ КОМАНДЕ
TIME:	INR A		П/П ИЗМЕРЕНИЯ ПЕРИОДА T_x
	CPI	0FFH	ЭТО ПЕРВЫЙ ЗАПРОС ПРЕРЫВАНИЯ?
	JNZ	M1	ЕСЛИ НЕТ, ТО ПЕРЕХОД НА M1
	EI		ЕСЛИ ДА, ТО РАЗРЕШЕНИЕ
			СЛЕДУЮЩЕГО ПРЕРЫВАНИЯ
FICT:	PCML		И ПЕРЕХОД К "ФИКТИВНОМУ" СЧЕТУ
M1:	CPI	0FH	ЭТО ВТОРОЙ ЗАПРОС ПРЕРЫВАНИЯ?
	JNZ	M2	ЕСЛИ НЕТ, ТО ПЕРЕХОД НА M2
	INX SP		ЕСЛИ ДА, ТО КОРРЕКЦИЯ УКАЗАТЕЛЯ
	INX SP		СТЕКА
	INX SP		
	INX SP		
	LXI D, 0020H;		НАЧАЛЬНАЯ КОРРЕКЦИЯ (D,E)
	EI		РАЗРЕШЕНИЕ СЛЕДУЮЩЕГО ПРЕРЫВАНИЯ
ESIN:	PCML		НАЧАЛО ВНУТРЕННЕГО СЧЕТА ИМПУЛЬСОВ
			ОБРАЗЦОВОЙ ЧАСТОТЫ - ЕДИНИЦА СЧЕТА ЗА
			10 ТАКТОВ РАБОТЫ МИКРОПРОЦЕССОРА
M2:	CPI	250	ЭТО ПОСЛЕДНЕЕ ПРЕРЫВАНИЕ, Т.Е. ЗАФИКСИ-
			РОВАННО ЗАДАННОЕ (ЗДЕСЬ - 250) ЧИСЛО
			КОЛЕБАНИЙ ИЗМЕРЯЕМОЙ ЧАСТОТЫ?
	JZ	M3	ЕСЛИ ДА, ТО ПЕРЕХОД НА M3
	INX SP		ЕСЛИ НЕТ, ТО: КОРРЕКЦИЯ
	INX SP		УКАЗАТЕЛЯ СТЕКА,
	XCML		КОРРЕКЦИЯ ТЕКУЩЕЙ
	LXI D, 0000H;		СУММЫ В (D,E),
	DAD B		
	XCML		
	NOP		ПАУЗА
	EI		РАЗРЕШЕНИЕ СЛЕДУЮЩЕГО ПРЕРЫВАНИЯ,
CONT:	RET		ПРОДОЛЖЕНИЕ ВНУТРЕННЕГО СЧЕТА С
			НОВОЙ КОМАНДЫ, НА КОТОРОЙ СМ ПРЕРВАЕТСЯ
M3:	INX SP		СЧЕТ ЗАКАНЧИВАЕТСЯ -
	INX SP		КОРРЕКЦИЯ УКАЗАТЕЛЯ СТЕКА,
	INX SP		
	INX SP		
FIN:	RET		КОНЕЦ П/П "TIME", ПЕРЕХОД К ОБРАБОТКЕ
			РЕЗУЛЬТАТОВ
PROG1:			
-----			ОБРАБОТКА РЕЗУЛЬТАТОВ ИЗМЕРЕНИЯ

	HLT		ОСТАНОВ ИП, ОЖИДАНИЕ СЛЕДУЮЩЕГО
			СИГНАЛА "ПУСК"
PROG2:			
-----			ОБРАБОТКА РЕЗУЛЬТАТОВ ИЗМЕРЕНИЯ

	HLT		ОСТАНОВ ИП, ОЖИДАНИЕ СЛЕДУЮЩЕГО
			СИГНАЛА "ПУСК"
PROG3:			
-----			ОБРАБОТКА РЕЗУЛЬТАТОВ ИЗМЕРЕНИЯ

	HLT		ОСТАНОВ ИП, ОЖИДАНИЕ СЛЕДУЮЩЕГО
			СИГНАЛА "ПУСК"

Рис. 2. Программа работы микропроцессорной системы

УДК 681.322.1+681.325.5

В. Е. Кушнир, Д. И. Панфилов, С. Г. Шаронин

УЧЕБНАЯ МИКРОЭВМ НА ОСНОВЕ ОДНОКРИСТАЛЛЬНОЙ ЭВМ КМ1816ВЕ43

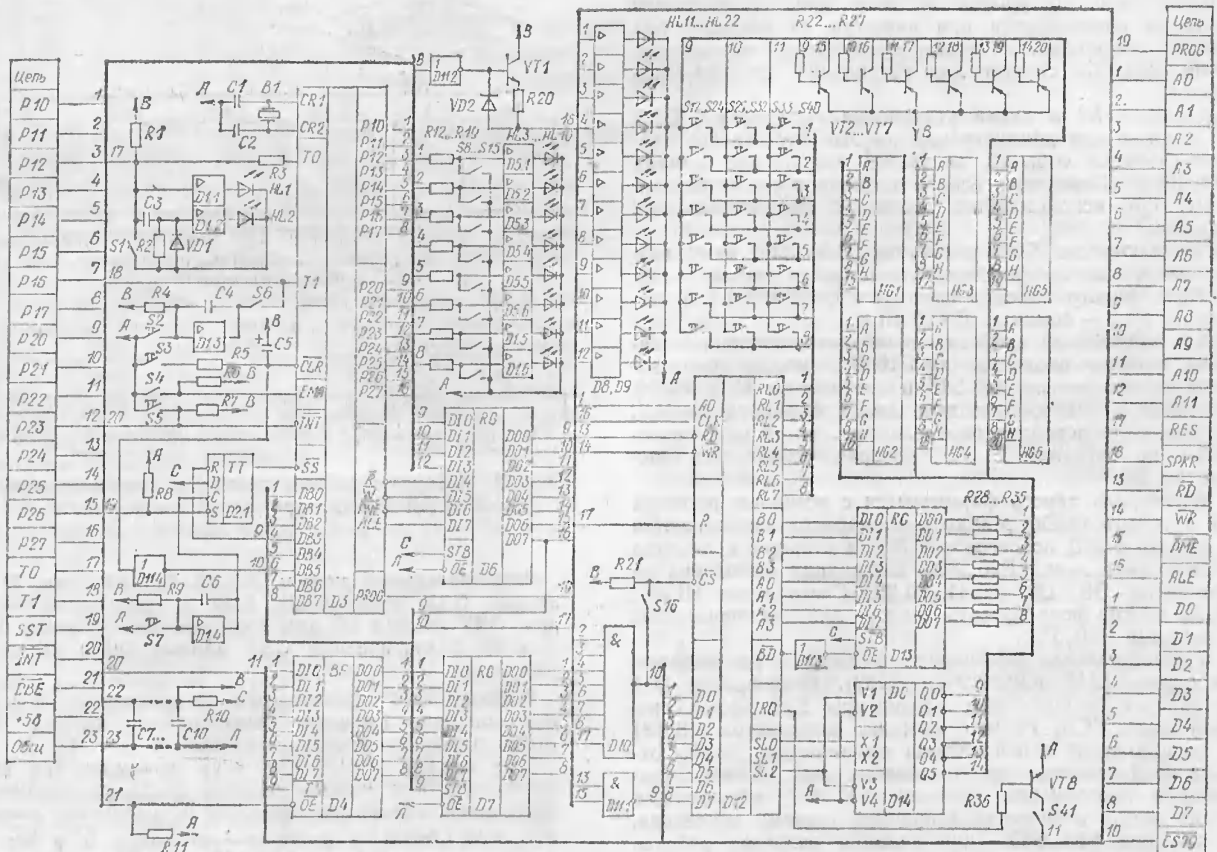
Учебная микроЭВМ УМПК-48 на основе однокристалльной БИС КМ1816ВЕ43 [1] предназначена для изучения внутренних элементов, их программирования и исследования различных режимов работы, методов разработки и отладки программного и схемотехнического обеспечения, а также методов отыскания неисправностей.

МикроЭВМ имеет открытую конструкцию и построена на основе трех функциональных модулей: базового УМПК-48/ВМ (240×240 мм) и двух расширительных УМПК-48/МР1 и УМПК-48/МР2 (120×240 мм).

ЭВМ в составе базового модуля (БМ) и расширительного модуля УМПК-48/МР1 выполняет цикл, рассчитанный на изучение структуры ОЭВМ и функционирования ее элементов (линий ввода-вывода, таймер-счетчика событий, системы прерываний и т. п.); системы команд и ориентированных на все алгоритмов об-

работки информации, временных диаграмм работы, организации различных интерфейсных функций (параллельного и последовательного ввода-вывода данных, временных интервалов и т. п.); расширения возможностей (за счет использования интерфейсных БИС серии КР580, внешних ЗУ и т. п.); организации отладочных режимов работы (пошаговое выполнение программ, эмуляция памяти команд); структуры и программного обеспечения микроконтроллеров конкретных объектов и методов отыскания неисправностей в них.

Для практического освоения основ разработки схемотехнического и программного обеспечения микроконтроллеров используются модули УМПК-48/ВМ, УМПК-48/МР1, УМПК-48/МР2, а для изучения кросс-средств при разработке и отладке программ для таких систем дополнительно требуется микроЭВМ более высокого уровня (ДВК, СМ1800).



С1, С2 - 20 нФ; С3 - 0,1 мкФ; С4, С6, С8, С10 - 0,047 мкФ; С5, С7 - 100 нФ; R1, R2, R4, R6, R9, R11 - 10 кОм; R3, R5, R7, R10, R12, R19, R24, R36 - 1 кОм; R20, R22, R33 - 270 Ом; D1, D5, D8, D9 - К561П14; D2 - К557Т12; D3 - КМ1816ВЕ43; D4 - КР580ВА86; D6, D7 - КР580МР2; D10 - К555А2; D11 - К555А3; D12 - КР580БВ79; D13 - КР580МР3; D14 - К555А4; D15...D17 - К555А2; D18...D20 - К555А2; D21...D23 - К555А2; D24...D26 - К555А2; D27...D29 - К555А2; D30...D32 - К555А2; D33...D35 - К555А2; D36...D38 - К555А2; D39...D41 - К555А2; D42...D44 - К555А2; D45...D47 - К555А2; D48...D50 - К555А2; D51...D53 - К555А2; D54...D56 - К555А2; D57...D59 - К555А2; D60...D62 - К555А2; D63...D65 - К555А2; D66...D68 - К555А2; D69...D71 - К555А2; D72...D74 - К555А2; D75...D77 - К555А2; D78...D80 - К555А2; D81...D83 - К555А2; D84...D86 - К555А2; D87...D89 - К555А2; D90...D92 - К555А2; D93...D95 - К555А2; D96...D98 - К555А2; D99...D100 - К555А2; VT1 - КТ315А; VT2...VT7 - КТ350А; VT8 - КТ315А; S41 - КТ315А; S70 - КТ315А.

Рис. 1. Принципиальная схема модуля УМПК-48/ВМ

Базовый модуль УМПК-48/ВМ включает в себя: ОЭВМ КМ1816ВЕ48, схему пошагового выполнения программ, элементы коммутации и индикации для демонстрации работы линий ввода-вывода ОЭВМ и ее внутренних узлов, связанных с ними, буферы магистралей данных (БМД) и управления (БМУ), схему формирования магистралей адреса (ФМА) ОЭВМ для подключения внешних ЗУ на других модулях и ее индикации, БИС интерфейса клавиатуры и дисплея КР580ВВ79, клавиатуру и шестизначный светодиодный семисегментный дисплей, дешифратор адреса, позволяющий отключить БИС КР580ВВ79 для демонстрации автономной работы ОЭВМ, схему выдачи звукового сигнала.

Все сигналы ОЭВМ, включая МД, МУ и МА, выведены на разъемы для подключения других модулей. Ряд выводов ОЭВМ и БИС КР580ВВ79 подключен к специальным контрольным точкам, которые используются для исследования временных диаграмм и при изучении методов отыскания неисправностей.

Основным элементом электрической принципиальной схемы БМ (рис. 1) является ОЭВМ КМ1816ВЕ48 (D3). Для организации работы внутреннего тактового генератора ОЭВМ служат емкости С1, С2 и кварцевый резонатор В1 (6 МГц). Начальная установка ОЭВМ осуществляется нажатием на клавишу S3, необходимая длительность импульса начальной установки определяется емкостью С5. Переключатель S4 дает возможность отключать внутреннее ПЗУ ОЭВМ при работе БМ с модулями расширения.

При наличии низкого уровня на входе триггера D2.1 ОЭВМ работает в реальном масштабе времени. После подачи высокого уровня на этот вход (выполнение команды производится при нажатии на клавишу S7) ОЭВМ переходит в режим пошагового выполнения. Цепь D1.4, С6 служит для устранения дребезга контактов.

Клавиша S2 и схема устранения ее дребезга D1.3, С4 служат для демонстрации работы внутреннего таймера-счетчика событий, а также выполнения команд JТ, JNT1. Переключатель S6 позволяет отключать эту схему при использовании линии T1 дополнительными схемами.

Переключатель S1 и элементы D1.1, НL1 дают возможность демонстрировать выполнение команд JТ0, JNT0, а фильтр высокой частоты С3, VD2, D1.2 и светодиод НL2 — команды ENT0 CLK.

Для проведения экспериментов с квазидвуправленными линиями ввода-вывода P10...P17 можно использовать переключатели S8...S15 и светодиоды НL3...НL10.

Усилитель D4 предназначен для буферизации МД. Направление передачи определяется схемой управления D11.1 по сигналам с МУ, которая усиливается элементом D6.

Магистраль адреса формируется с помощью регистра D7 и усилителя D6. Младшие 8 разрядов записываются в регистр с МД по сигналу с МУ, а старшие 4 разряда снимаются с линий P20...P23. Индикация выполнена на элементах D8, D9, НL11...НL22. Светодиоды НL19...НL22 можно использовать также для экспериментов с линиями P20...P23.

Функциональная клавиатура и дисплей реализованы на основе БИС КР580ВВ79 (D12). Дешифратор D14 организует сканирование клавиатуры S17...S40 и дисплея НG1...НG6. Размер матрицы клавиатуры (3×8) и расположение в ней клавиш выбраны так, чтобы отпала необходимость преобразования кода нажатой клавиши в программах. Клавиши S17...S32 служат для ввода чисел в шестнадцатиричной системе счисления, а клавиши S33...S40 — для задания режимов работы. Клавиша S5, не входящая в матрицу, служит для подачи сигнала на вход запроса прерывания ОЭВМ при изучении ее работы. С помощью переключателя S41 сигнал запроса прерывания БИС D12 может также подключаться к этому входу ОЭВМ. Дешифратор адреса

для БИС КР580ВВ79 реализован на элементе D10. Переключатель S16 дает возможность отключать БИС для демонстрации автономной работы ОЭВМ. Схема выдачи звукового сигнала построена на элементах D11.2, VT1, VD2, R20.

Программное обеспечение БМ (рис. 2) используется только при автономной работе для иллюстрации возможностей ОЭВМ, исследования ее временных диаграмм работы и системы прерываний, проведения экспериментов по отысканию неисправностей. Все демонстрационные программы хранятся во внутреннем ПЗУ БИС КМ1816ВЕ48, выборка команд из которого разрешается только при замкнутом положении переключателя S4.

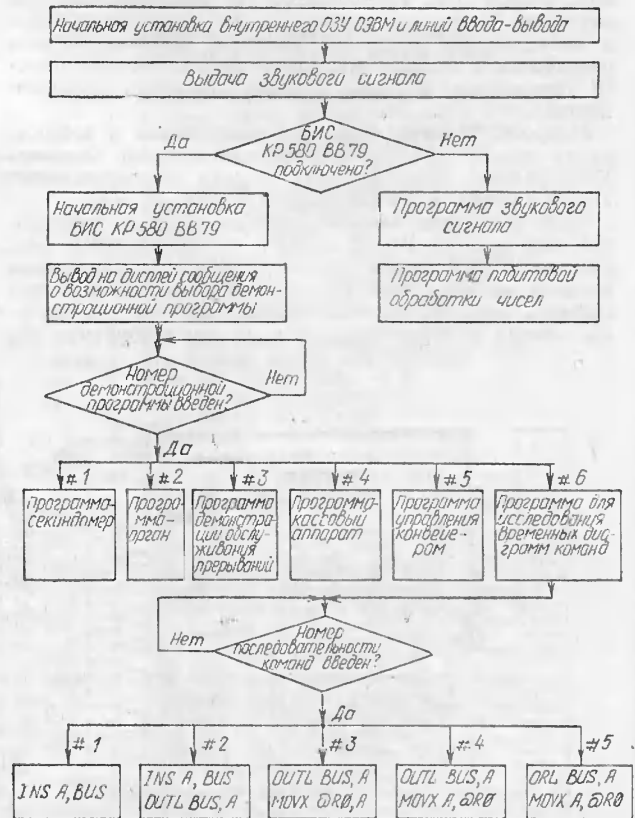


Рис. 2. Алгоритм работы программ демонстрации возможностей ОЭВМ при автономной работе БМ

Расширительный модуль УМПК-48/МР1 (рис. 3) содержит ПЗУ объемом 2К байт с программой-монитором, ОЗУ команд D7 для программ пользователя объемом 2К байт, внешнее ОЗУ данных D9 и схему управления.

Выборка ПЗУ производится по сигналу РМЕ в первой половине (нулевом банке памяти команд (ПК) МВО) адресного пространства ОЭВМ (рис. 4).

ОЗУ команд работает в двух режимах: при вводе программы как память данных (ПД) со страничной адресацией, где номер страницы определяется линиями P24...P26 ОЭВМ, а выборка — сигналами R и W; при выполнении программы ОЗУ D7 включается во вторую половину адресного пространства (первый банк МВ1) и выбирается сигналом РМЕ. Переключение режимов работы производится монитором с помощью линии P27 и мультиплексора D5.

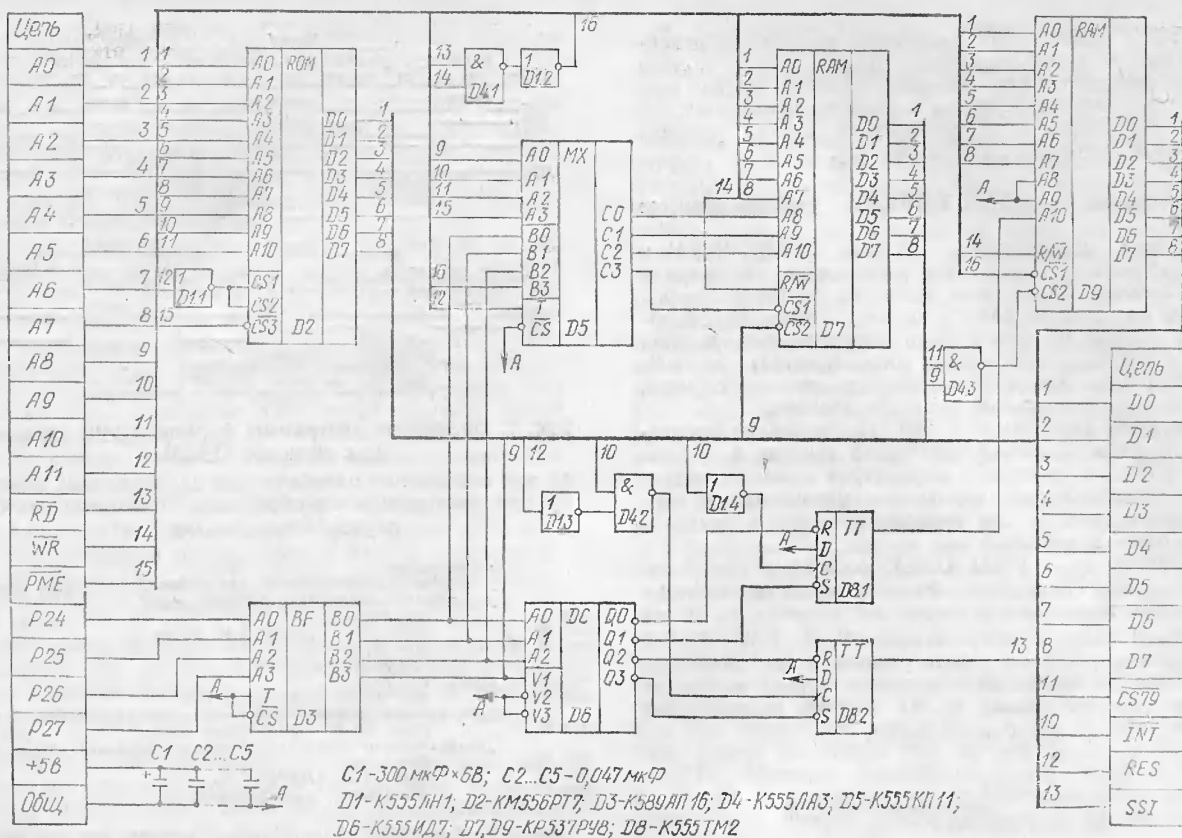


Рис. 3. Принципиальная схема модуля УМПК-48/MP1

Внешняя память данных D9 имеет две области: память данных монитора (256 байт) для хранения содержимого внутренней ПД ОЭВМ на момент останова при выходе из программы пользователя и память данных пользователя (256 байт).

Схема управления построена на элементах D6 и D8. С помощью триггера D8.1 производится переключение областей ОЗУ данных D9, а D8.2 — включение пошагового режима. Управление осуществляется дешифратором D6 по сигналам с линий P24...P27. Триггеры D8.1 и D8.2 сбрасываются при подаче запроса прерывания на ОЭВМ от клавиши S5 ВМ, а триггер D8.2 — сигналом начальной установки.

Программа-монитор модуля УМПК-48/MP1 (1,3К байта) предназначена для работы с ВМ (при отключенной внутренней памяти команд ОЭВМ) и ис-

пользует в качестве системных устройств ввода-вывода его клавиатуру и дисплей.

Монитор начинает работу с тестирования ОЗУ и начальной установки БИС KP580BB79. При обнаружении ошибки записи-считывания ОЗУ на дисплей выдается соответствующее сообщение и подается звуковой сигнал.

При успешном выполнении теста на дисплей выдается сообщение «начало» и ОЭВМ переходит в режим ожидания команды. После ввода команды осуществляется выполнение соответствующей программы (рис. 5).

Расширительный модуль УМПК-48/MP2 позволяет эмулировать внешнее ПЗУ команд ОЭВМ с помощью

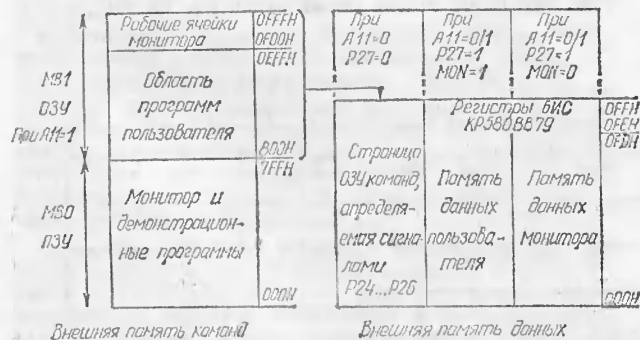


Рис. 4. Карта внешней памяти модуля УМПК-48/MP1

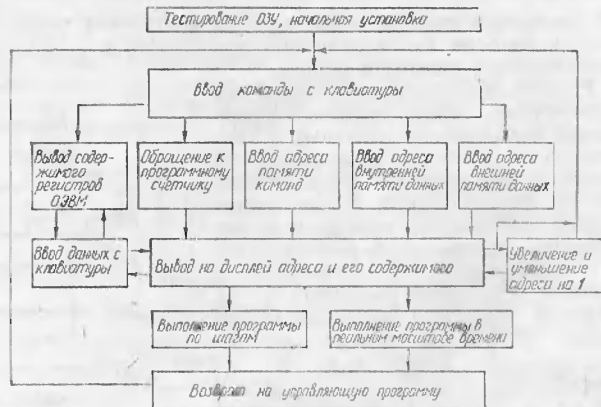


Рис. 5. Схема режимов работы монитора

микроЭВМ, обладающих развитыми средствами редактирования и кросс-средствами. Кроме этого модуль можно применять для изучения взаимодействия контроллеров различного уровня в децентрализованных системах управления с использованием других учебных микроЭВМ (например, УМПК-80) [2].

Использование микроЭВМ УМПК-48 в учебном процессе

При этом используются модули УМПК-48/ВМ и УМПК-48/МР1. Программы записываются в память и запускаются с помощью монитора расширительного модуля. Внутренняя память команд ОЭВМ, за исключением работы с портом P0 и прерываниями, должна быть отключена с помощью переключателя S4 БМ. Все программы, кроме специально огороженных случаев, выполняются в реальном масштабе времени.

Временные диаграммы ОЭВМ. Исследование временных диаграмм ОЭВМ КМ1816ВЕ48 состоит в наблюдении формы и временных параметров основных сигналов, вырабатываемых устройством управления и синхронизации (рис. 6) при выполнении одно- и двухбайтовых, одно- и двухцикловых команд.

Машинный цикл (МЦ) ОЭВМ состоит из пяти состояний (рис. 7), равных по длительности периоду внутренних тактовых импульсов, выдаваемых на вывод T0 (3/FCR). Длительность сигналов R, W, PME, PROG фиксирована и не может быть увеличена как, например, у МП БИС КР580ИК80 с помощью тактов ожидания (время выборки данных из ЗУ и УВВ должно быть не более 500 нс при FCR=6 МГц).

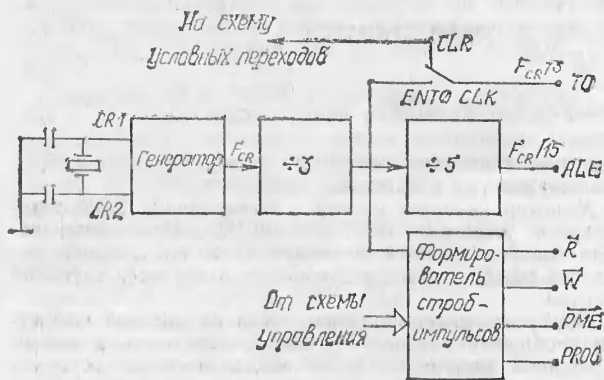


Рис. 6. Формирование сигналов управления и синхронизации в ОЭВМ

Длительность сигнала равна периоду внутренних тактовых импульсов или кратна ему при работе в режиме пошагового выполнения команд.

Управляющие сигналы исследуются двухлучевым осциллографом. Устойчивые изображения получают с помощью специально организованного цикла с программным формированием на выходах порта P1 синхронных импульсов для запуска развертки осциллографа (программа I, рис. 8). Вход У1 осциллографа подключается к контрольной точке выхода T0, на которой выводится внутренняя тактовая частота. Временные диаграммы сигналов PME, ALE, R, W, PROG при выполнении команд MOVX @R, A; MOVX A, @R; MOV @R, #0; ORLD P4, A; INS A, BUS; MOVP3 A, @A; NOP

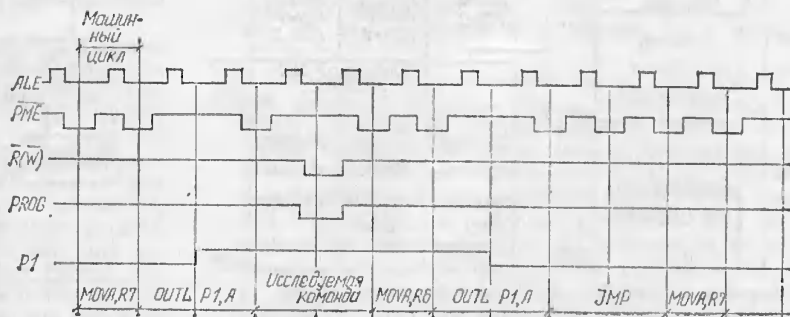


Рис. 8. Временные диаграммы работы программы I

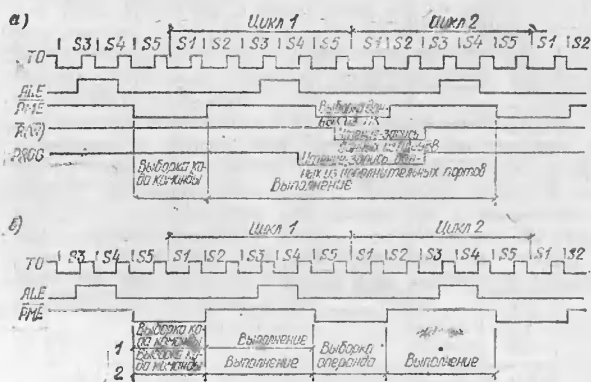


Рис. 7. Временные диаграммы формирования управляющих сигналов ОЭВМ:

- а) при выполнении однобайтовых двухцикловых команд;
- б) при выполнении однобайтовых однокциловых 1 и двухбайтовых команд 2

```

PROGRAMMA 1.
PROGRAMMA ПРЕНАЗНАЧЕНА ДЛЯ ИЗУЧЕНИЯ ВРЕМЕННЫХ ДИАГРАММ
ИССЛЕДУЕМОЙ КОМАНДЫ С ПОМОЩЬЮ ЦИКЛА.
880 75 DR6 880H
881 8E 00 ENT0 CLK ;ВЫВОД Т.Ч. НА T0
882 BF 00 MOV R6, #0
883 BF FF MOV R7, #0FFH
884 FF LOOP: MOV A, R7
885 39 OUTL P1, A ;УСТАНОВКА С.И
;*****
887 90 MOVX BR0, A ;ИССЛЕДУЕМАЯ КОМАНДА
;*****
888 FE MOV A, R6
889 39 OUTL P1, A ;СБРОС С.И.
89A 04 05 JMP LOOP ;ЗАЦИКЛИВАНИЕ
    
```

можно наблюдать, записывая их поочередно на место исследуемой команды в программе I и подключая вход У2 к соответствующим контрольным точкам. Это позволит проследить взаимное положение управляющих сигналов и сигнала T0, а также особенности их выдачи при выполнении различных типов команд.

Сигнал Условие формирования

Сигнал	Условие формирования
ALE	Каждый машинный цикл
PME	Во время чтения внешней памяти команд (кода команды, данных или адреса)
R	Во время выполнения команд MOVX A, @R или INS A, BUS
W	Во время выполнения команд MOVX @R, A или OUTL BUS, A
PROG	Во время выполнения команд MOVD A, P; ANLD P, A; MOVD P, A; ORLD P, A

Организация обмена информацией по линиям ввода-вывода. В ОЭВМ КМ1816ВЕ48 функции ввода-вывода выполняют двунаправленные линии DB0...DB7 (порт P0), квазидвунаправленные линии P10...P17 (порт P1) и P20...P27 (порт P2), а также линии T0, T1, INT.

Состояние линий T0, T1 и INT используется командами условных переходов. Кроме этого, линия T0 служит для выдачи внутренней тактовой частоты ($F_{CR}/3$), линия T1 является входом счетчика событий, а линия INT предназначена для запросов внешней прерывания.

Алгоритм обмена информацией по портам P0, P1, P2 зависит от режима работы и используемых команд.

Порт P0. Двухнаправленный порт P0 может использоваться в нескольких режимах:

двухнаправленной мультиплексной магистрали адреса данных с третьим состоянием при выборке из внешней ПК, а также обращении к внешним ПД или УВВ с помощью команд MOVX (адрес фиксируется сигналом ALE, а данные — сигналами R, W, PМЕ);

двухнаправленной магистрали данных при обмене с УВВ по командам стробуемого ввода-вывода INS и OUTL (для адресации УВВ могут служить линии портов P1 и P2);

8-разрядный входной порт с вводом данных по команде INS;

8-разрядный выходной порт с выводом данных по командам OUTL. ORL и ANL (строб записи \bar{W} в случае использования команд ORL и ANL не формируется). Команды INS, OUTL, MOVX могут чередоваться. Выполнение команды MOVX приводит к переводу линий порта P0 в третье состояние. Кроме этого, при использовании внешней ПК применение команды OUTL запрещено, так как запись в порт P0 информации приведет к ошибке чтения очередной команды из внешней ПК (если до этого не будет выполнена команда MOVX).

Применение команд стробуемого вывода возможно только при отсутствии внешней ПК, поэтому для проведения экспериментов используется демонстрационная программа для исследования временных диаграмм команд, записанная во внутренней ПК ОЭВМ. Программа представляет собой варианты программы 1. В зависимости от целей экспериментов, в программе вместо исследуемой команды ставятся команды, приведенные на рис. 2. Для их запуска необходимо замкнуть переключатель S4, провести начальную установку ОЭВМ и выбрать соответствующую демонстрационную программу.

Подключение входа У2 к контрольным точкам сигналов PМЕ, R, \bar{W} , DBO позволяет проследить следующие особенности формирования сигналов управления ОЭВМ:

при работе с внутренней ПК сигнал PМЕ не формируется;

после выполнения команд OUTL, ORL, ANL на линиях порта устанавливается выходная информация, а выполнение команд MOVX переводит их в третье состояние;

строб \bar{R} формируется во время выполнения команд INS A, BUS и MOVX A. @ R, а строб \bar{W} — OUTL BUS, A и MOVX @ R, A;

выдача информации при выполнении команды OUTL по сравнению с ORL и ANL производится раньше.

Квазидвухнаправленные порты P1 и P2 состоят из линий с аналогичной структурой. Каждая из них может быть использована независимо для ввода или вывода информации. Кроме этого, линии P21...P23 могут быть использованы как линии выдачи старших разрядов адреса внешней ПК и/или линии для подключения дополнительных портов.

Во время начальной установки по сигналу CLR все линии портов P1 и P2 устанавливаются в состояние ввода (т. е. в них записаны «Лог. 1»). Перед сменой направления передачи информации с вывода на ввод соответствующих разрядов портов нужно также установить в «Лог. 1» командой OUTL или ORL. Это не относится к линиям P20...P23 в случае их использования

для расширения. Так, при подключении внешней ПК вывод данных, записанных в разряды P20...P23, автоматически прерывается на время выдачи адреса. Использование этих линий для ввода практически невозможно, однако выходная информация с них может быть записана во внешний регистр по переднему фронту сигнала ALE. В случае же подключения к линиям P20...P23 дополнительных портов выходная информация не сохраняется (после выполнения команды ввода MOV D A, P линии P20...P23 автоматически устанавливаются в «Лог. 1», а после выполнения команд MOV D P, A; ORL P, A; ANL P, A в них остаются данные, переданные в дополнительный порт или из него).

Для проведения исследования работы порта P1 используется программа 1. Наблюдаются моменты выдачи информации на линии порта P1 при выполнении команд OUTL P1, A, ORL P1, # FFH и ANL P1, # 0. При этом вход У2 осциллографа подключается к контрольной точке P10, а моменты выдачи информации в порт фиксируются по заднему фронту синхронного импульса.

Запись в программе 1 первой команды OUTL P1, A по команде ORL P1, # OFFH, а второй — на ANL P1, # 0 позволяет убедиться, что информация будет выдаваться в порт P1 позже, чем в предыдущем случае.

Для проведения исследования порта P2 вход У2 осциллографа подключается к контрольной точке P23 и используется программа 1 с подстановкой, на место исследуемой, различных последовательностей команд. Раскрываются особенности работы линий P20...P23 при использовании внешней ПК и дополнительных портов. Так как программа работает в банке МВ1, при выдаче адреса на линию P23 на ней будет появляться «Лог. 1». Поэтому последовательность MOV A, # 0, OUTL P2, A демонстрирует присутствие выходной информации в момент выдачи переднего фронта сигнала ALE.

Команда MOV D P4, A позволяет исследовать временные диаграммы обмена информацией с дополнительным портом и проверить возможность ее использования для записи во внешний регистр по переднему фронту сигнала PROG 4-разрядного числа.

Последовательности MOV A, # 0, OUTL P2, A, MOV D A, P4 и MOV A, # 0, OUTL P2, A, CPL A, MOV D P4, A демонстрируют изменение состояния линий P20...P23 при выполнении команд обмена информацией с дополнительными портами.

Особенности работы таймера-счетчика событий ОЭВМ КМ1816ВЕ48. Его основу составляет 8-разрядный счетчик, работающий на сложение. Режим работы таймера-счетчика задается командами STRT, T, STRT. CNT, STOP TCNT, которые определяют источник тактовых импульсов для счетчика (рис. 9). При начальной установке ОЭВМ запрещаются счет (содержимое счетчика остается неизменным) и прерывания по его переполнению. Загрузка и чтение содержимого счетчика выполняется командами MOV T, A и MOV A, T. В случае, когда таймер-счетчик не используется по прямому назначению, его можно применять как регистр общего назначения. Если при работе таймера-счетчика возникнет переполнение из OFFH в 0, счет не прекращается

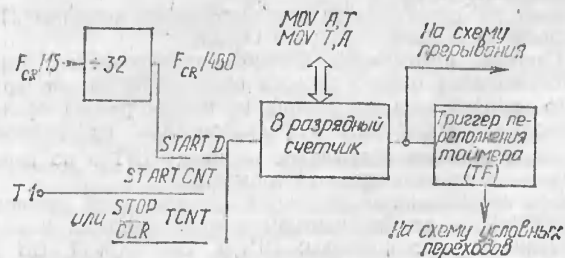


Рис. 9. Организация различных режимов таймера-счетчика событий

и устанавливается триггер флага переполнения таймера TF. Сброс флага осуществляется при подаче сигнала CLR или при выполнении команды его проверки JTF.

Особенности работы в режиме счетчика событий: инкремент счетчика выполняется при поступлении заднего фронта импульса на вход T1, максимальная скорость счета — один импульс за три машинных цикла, длительность импульса на входе T1 должна быть не менее 500 нс, счет не прекращается при останове во время пошагового выполнения команд. Этот режим удобно использовать для подсчета импульсов от различных датчиков.

Режим таймера используется для генерации временных интервалов. При работе в этом режиме на вход счетчика подаются импульсы с частотой $F_{CR}/480$ от предварительного делителя, который обнуляется во время выполнения команды STRT T. В отличие от предыдущего режима, счет не ведется во время останова при пошаговом выполнении команд, что очень удобно при отладке системы. Так как период импульсов, поступающих на вход счетчика, равен 80 мкс (для $F_{CR}=6$ МГц), то при работе таймера можно получать задержки длительностью от 80 мкс до 20 мс. Интервалы, большие чем 20 мс, можно получать с помощью специальной подпрограммы обработки прерываний от таймера, т. е. счетом переполнений в каком-либо регистре.

Для генерирования задержек с большей точностью необходимо использовать режим счетчика событий и внешнюю тактовую последовательность, например, деленную на три или более последовательностей с выхода ALE. При этом счет также не будет вестись во время останова при пошаговом выполнении программ. Задержки меньшей длительности можно также получать программно, организовав цикл.

Для организации управления счетчиком и индикации его состояния на светодиодах порта P1 используется программа 2. Импульсы на входе T1 получают с помощью клавиши S2, при этом переключатель S6 должен быть замкнут.

```

      *ПРОГРАММА 2.
      *ПРОГРАММА ПРЕДНАЗНАЧЕНА ДЛЯ ИЗУЧЕНИЯ РАБОТЫ
      *ТАЙМЕРА-СЧЕТЧИКА СОБЫТИЙ В РЕЖИМЕ СЧЕТЧИКА СОБЫТИЙ.
800      ORG 800H
800 23 F0 PROGR2: MOV A, #0FH ;НАЧ. СОСТОЯНИЕ ТАЙМЕРА
802 62      MOV T,A
803 45      STRI CNT ;ЗАПУСК СЧЕТЧИКА СОБЫТИЯ
804 42      DISP: MOV A,T
805 39      OUTL P1,A ;ВЫВОД СОСТОЯНИЯ СЧЕТЧИКА
806 16 0A   JTF EXIT ;ВХОД ПО ПЕРЕПОЛНЕНИЮ
808 04 04   JNP DISP ;ЗАЩИЩЕНИЕ
80A 16 11 EXIT: JTF ERR ;ПРОВЕРКА СБРОСА ФЛАГА TF
80C 23 55   MOV A, #55H
80E 39      OUTL P1,A ;ВХОД '55'-ИНДИКАЦИЯ ПЕРЕПОЛНЕНИЯ
80F 04 0B   JNP PROGR2 ;ПЕРЕХОД К НАЧАЛУ
811 23 EE ERR: MOV A, #EEH
813 39      OUTL P1,A ;ВЫВОД 'EE'- TF НЕ СБРОСИЛСЯ
814 04 0B   JNP PROGR2
  
```

Выполнение программы в пошаговом режиме и в реальном масштабе времени позволяет проследить следующие особенности таймера-счетчика:

- инкремент счетчика происходит при поступлении заднего фронта сигнала на вход T1;
- счет не прекращается в пошаговом режиме;
- флаг TF устанавливается при изменении состояния таймера из OFFH в 0;
- флаг TF сбрасывается при выполнении команды JTF и после начальной установки ОЭВМ.

Система прерываний ОЭВМ — одноуровневая (при обслуживании одного запроса прерывания запрос другого игнорируется до выхода из подпрограммы обслуживания прерывания), с фиксированным приоритетом и имеет внешнее прерывание по входу INT и по переполнению таймера-счетчика событий.

Оба прерывания запрещаются при начальной установке ОЭВМ, а для их программного разрешения и разрешения служат команды: DIS I, DIS TCNTI, EN I, EN TCNTI. Разрешение прерываний происходит сразу после выполнения команд EN I и EN TCNTI.

Если внешнее прерывание было разрешено, то подача уровня «Лог. 0» на вход INT вызовет переход к подпрограмме обработки внешнего прерывания с начальным адресом 03. Состояние входа INT анализируется по заднему фронту сигнала ALE в последнем цикле команды. Запрос прерываний от таймера-счетчика формируется при его переполнении, причем признак переполнения фиксируется во внутреннем триггере, который сбрасывается только после обслуживания прерывания таймера или начальной установки ОЭВМ. Флаг переполнения TF не сбрасывается при удовлетворении запроса прерывания таймера-счетчика. Подпрограмма обслуживания прерывания начинается с адреса 07.

Выполнение команд вызова осуществляется после выполнения последнего цикла текущей команды. При этом прерывания автоматически запрещаются. Разрешение прерываний происходит только после выполнения команды RETR, которой должна заканчиваться подпрограмма обслуживания прерывания. Команда RET, хотя и вызовет возврат в основную программу, но прерывания не разрешит и PSW не восстановит.

Оба вектора прерывания находятся в банке МВО. Там же должны находиться и подпрограммы обслуживания прерываний, причем они не могут обращаться к подпрограммам или фрагментам программ, находящимся в банке МВ1.

При переходе к подпрограмме обработки прерывания в стеке запоминается 12-разрядный адрес возврата и PSW. Поскольку в PSW входят только флаги CY, AC, FO, BS, то для сохранения аккумулятора, регистров и флага F1 (в случае если это необходимо) нужно принимать специальные меры. Удобно при входе в подпрограмму обслуживания прерывания производить переключение банка регистров и не пользоваться флагом F1 (за исключением случаев, когда флаг является выходным параметром подпрограммы).

Ограниченная величина стека ОЭВМ вызывает необходимость учитывать возможность его переполнения в программах с большой сложностью подпрограмм. Для предотвращения переполнения рекомендуется запрещать прерывания на время выполнения такого участка программы.

Так как состояние входа INT может быть определено командой условного перехода JNI, то возможно комбинированное использование этого входа. Это может быть полезным, например, при организации приемника последовательного канала связи. В этом случае старт-бит, пришедший на вход INT, вызывает переход к подпрограмме присема слова, остальные биты которого анализируются с помощью команды JNI.

Если в системе необходимо иметь два внешних прерывания или прерывание с запуском по фронту, можно воспользоваться прерыванием от таймера-счетчика. При этом счетчик загружается числом OFFH и запускается в режиме счетчика событий. Задний фронт сигнала на входе T1 вызовет переход к подпрограмме с адресом 07, если разрешено соответствующее прерывание. В противном случае, подпрограмма обработки прерывания будет вызвана сразу же после выполнения команды его разрешения. Использование входа счетчика событий в качестве входа запроса внешнего прерывания удобно также и тем, что в случае длительного запрета на его обслуживание имеется возможность фиксировать число поступающих запросов. При необходимости увеличить число внешних прерываний до 8, совместно с ОЭВМ удобно использовать схему приоритетных прерываний K589ИК14.

В случае работы с внешним запросом прерывания по входу INT может возникнуть проблема подтверждения его обслуживания (например, при использовании ИМС K589ИК14), которую можно решить использованием одной из линий ввода-вывода для программного формирования сигнала подтверждения прерыва-

ния. Другим решением этой проблемы может быть применение УВВ, имеющих необходимую для этого встроенную логику (например, интерфейсных БИС серии КР580).

Исследования с помощью программы демонстрации обслуживания прерываний в ПК ОЭВМ (см. рис. 2 и программу 3) иллюстрируют порядок обслуживания запросов прерываний (как внешнего, так и от таймера-счетчика), действие команд разрешения и запрещения прерываний.

```

ПРОГРАММА 3.
ПРОГРАММА 3 ПРЕДНАЗНАЧЕНА ДЛЯ ИЗУЧЕНИЯ СИСТЕМЫ ПРЕРЫВАНИЙ
139 ORG 130H
139 34 50 DEM7: CALL LDT ;ЗАГРУЗИТЬ ТАЙМЕР-СЧЕТЧИК
13C 05 EN I
13D 25 EN TCNTI
13E 16 42 LBL: JTF OUT ;УСТАНОВЛЕН ФЛАГ ПЕРЕПОЛНЕНИЯ - ПЕРЕХОД
140 24 3E JMP LBL ;ЗАКЛЮЧЕНИЕ И ОКОНЧАНИЕ ПРЕРЫВАНИЯ
142 23 CC OUT: MOV A,#00H;ВЫБОР '00',ЕСЛИ TF УСТАНОВЛЕН
144 39 OUTL P1,A
145 24 3E JMP LBL

;
147 00 EXSUB: NOP ;ПОДПРОГРАММА ОБСЛУЖИВАНИЯ ВНЕШНЕГО ЗАПРОСА
148 23 33 MOV A,#33H
149 39 OUTL P1,A ;ВЫВОД '33'-ПРИЗНАК ВНЕШНЕГО ПРЕРЫВАНИЯ
14B 93 RETR ;ВОЗВРАТ В ТОЧКУ ВЫЗОВА

;
14C 00 TCSub: NOP ;ПОДПРОГРАММА ОБСЛУЖИВАНИЯ ЗАПРОСА ТАЙМЕРА
14D 23 77 MOV A,#77H
14E 39 OUTL P1,A ;ВЫВОД '77'-ПРИЗНАК ПРЕРЫВАНИЯ ТАЙМЕРА
150 23 FE LDT: MOV A,#0F0H;ЗАГРУЗКА ТАЙМЕРА-СЧЕТЧИКА
152 62 MOV T,A
153 45 STRT CNT
154 93 RETR ;ВОЗВРАТ В ТОЧКУ ВЫЗОВА

;
003 ORG 3
003 24 47 JMP EXSUB ;ВЕКТОР ВНЕШНЕГО ПРЕРЫВАНИЯ

;
007 ORG 7
007 24 4C JMP TCSub ;ВЕКТОР ПРЕРЫВАНИЯ ТАЙМЕРА

```

Изменения адреса фиксируются при пошаговом выполнении демонстрационной программы. Запрос прерывания на вход INT подаются, нажимая на клавишу S5, а импульсы на вход счетчика событий — на клавишу S2. Для запуска программы нужно замкнуть переключатель S4, выполнить начальную установку ОЭВМ и выбрать необходимую программу.

Программа позволяет проследить следующие особенности системы прерывания ОЭВМ:

при подаче запроса на вход INT передача управления происходит по адресу 03, причем на повторные запросы ОЭВМ не реагирует до выполнения команды RETR;

запрос прерывания от таймера-счетчика формируется при изменении его состояния от OFFH в «0» и вызывает вызов подпрограммы по адресу 07;

если до выполнения команды RETR подпрограммы обработки прерывания (внешнего или таймера-счетчика) поступит и снимается запрос внешнего прерывания, то он не запоминается. Переполнение таймера фиксируется и сразу же, после выполнения команды RETR, начинается обработка прерывания по таймеру;

при одновременном поступлении запросов прерывания от таймера и внешнего, первым обслуживается внешний запрос;

прерывания после начальной установки ОЭВМ запрещены;

переход к подпрограмме обработки прерываний от таймера-счетчика и ее выполнение не сбрасывают флаг TF (команда JTF выполняется как в теле подпрограммы обслуживания запроса, так и после выхода из нее).

Использование памяти команд. Специфика организации памяти команд ОЭВМ КМ1816ВЕ48 состоит в том, что адресное пространство (4К байт) разбито на два одинаковых банка МВ0 и МВ1 и 16 страниц по 256 байт. Передача управления командами условных и косвенного переходов осуществляется в пределах одной страницы, заданием адреса в ней (а не смещения, как у других МП БИС). Команды JMP и CALL вы-

зывают переход по любому адресу, но в пределах текущего банка ПК. При начальной установке ОЭВМ происходит автоматическое включение банка МВ0. В дальнейшем выбор банка осуществляется командами SEL МВ0 и SEL МВ1. Однако выполнение этих команд лишь изменяет содержимое внутреннего триггера выбора банка МВ. Изменение состояния 12-го разряда адреса происходит лишь при выполнении команд JMP или CALL.

Содержимое триггера МВ невозможно считать, что создает определенные неудобства при программировании, особенно в случае вложенного вызова подпрограмм, расположенных в различных банках ПК. Это обусловлено тем, что при возврате из подпрограммы полный 12-разрядный адрес возврата восстанавливается из стека, но состояние триггера МВ при этом не изменяется. Для предотвращения ошибок в этом случае, после команд CALL нужно выполнить команду SEL МВ0.

При обслуживании запросов прерываний для устранения неопределенностей 12-й разряд адреса принудительно устанавливается в «0» (без изменения МВ). Выполнение команд SEL МВ в теле подпрограммы обслуживания прерывания (до команды RETR) вызывает только переключение внутреннего триггера МВ. Переход в другой банк будет осуществлен лишь первой, после возврата, командой JMP или CALL. Поэтому пользоваться командами выбора банка ПК в подпрограмме обслуживания прерывания не рекомендуется.

Исследования возможности выбора банка ПК командами SEL МВ проводятся с помощью программы 4 в пошаговом режиме для фиксации изменения адреса. Первая часть программы 4 расположена в банке МВ0 в ПЗУ монитора.

```

ПРОГРАММА 4.
ПРОГРАММА 4 ПРЕДНАЗНАЧЕНА ДЛЯ ИЗУЧЕНИЯ ПЕРЕКЛЮЧЕНИЯ
БАНКОВ ПАМЯТИ КОМАНД.
7F0 ORG 7F0H
7F0 00 M44: NOP
7F1 F5 SEL MB1
7F2 00 NOP
7F3 00 NOP
7F4 00 NOP
7F5 04 00 JMP M4

;
800 DR- 800H
800 E5 M4: SEL MB0
801 00 NOP
802 00 NOP
803 00 NOP
804 E4 F0 JMP M44

```

Выполнение программы показывает, что переход в другой банк после его переключения происходит только после выполнения команды JMP.

Использование памяти данных. Наряду с ПЗУ команд, ОЭВМ содержит ОЗУ данных, различающиеся по объему и способам адресации. ОЭВМ КМ1816ВЕ48 имеет внутреннюю ПД объемом 64 байта. Возможности ОЭВМ могут быть расширены за счет подключения внешней ПД объемом 256 байт. Во внутренней ПД определены 8-уровневый стек и два банка регистров общего назначения RB0 и RB1, обращение к которым может производиться с помощью прямого регистрового способа адресации. Косвенный регистровый способ адресации используется при обращении ко всем ячейкам как внутренней ПД (команды MOV), так и внешней ПД (команды MOVX), причем в качестве индексных могут использоваться регистры R0 и R1.

Переключение банков регистров осуществляется командами SEL RB0 и SEL RB1 и производится сразу после их выполнения. При начальной установке автоматически включается банк RB0.

Переключение банков удобно использовать в подпрограммах для сохранения значений регистров основной программы. Так как признак банка регистров BS входит в PSW, то после возврата из подпрограммы по

В. П. Арсенин, А. О. Воробьев, В. Н. Герасимович

ПРИМЕНЕНИЕ МИКРОСХЕМЫ K1802BB1 ДЛЯ УПРАВЛЕНИЯ ПАМЯТЬЮ

Схема обмена информацией БИС ОИ K1802BB1 выполнена по ТТЛШ-технологии в 42-выводном металло-керамическом корпусе и может эксплуатироваться в диапазоне температур от $-60...+125^{\circ}\text{C}$.

БИС ОИ (см. рис. 1) состоит из 4-разрядных регистров RG1...RG3, реализованных на D-триггерах типа «Защелка», 4-разрядного двоичного счетчика с возможностью параллельной загрузки RG0, реализованного на D-триггерах типа «M—S» с записью информации по фронту сигнала, дешифраторов DCX, DCC, DCB, DCA выбора необходимого регистра соответственно по каналам X, C, B, A, узлов управления, мультиплексоров MS0...MS3 выбора информации для записи в регистры, схемы сравнения, вырабатывающей признак равенства содержимого регистра RG0 и содержимого регистра RG3 и блока приема-выдачи данных на каналы A, B, C, X. Регистр RG0 может работать в режиме счетчика, с прибавлением к содержимому +1 вне режимов работы каналов.

Разрешением обмена информацией с каналами A, B, C, X управляют входы ECA, ECB, ECC, ECX. При «Лог. 0» на указанных входах разрешен обмен информацией выбранного регистра с соответствующим каналом.

Режим считывания или записи может быть задан только при разрешении обмена информацией с выбранным каналом. Это определяется входами ECA, ECB, ECC, ECX. Входы RA, RB, RC, RX определяют режим считывания информации из регистров БИС ОИ на каналы A, B, C, X (активный уровень — «Лог. 0»). Входы WA, WB, WC, WX определяют режим записи с каналов A, B, C, X в выбранные регистры БИС ОИ (активный уровень — «Лог. 0»).

Необходимый регистр БИС ОИ в режимах записи и считывания выбирается четырьмя дешифраторами DCA, DCB, DCC, DCX заданием двоичного кода на входы адреса AA0...AA1, AB0...AB1, AC0...AC1, AX0...AX1 для каналов A, B, C, X соответственно.

Регистры RG1...RG3 из одного состояния в другое переходят при подаче на входы записи «Лог. 0», регистр RG0 из одного состояния в другое переходит при перепаде логического уровня на одном из входов записи из «Лог. 0» в «Лог. 1».

Схема сравнения непрерывно сравнивает содержимое RG0 и RG3. В случае их совпадения на выходе F будет «Лог. 1». Содержимое регистра RG0 увеличивается при подаче на вход CI положительного перепада напряжения.

Можно непосредственно передавать информацию с одной магистрали на другую через любой регистр, кроме регистра RG0. Магистрали DA, DB, DC предназначены для работы на короткие линии связи, а магистраль DX на длинные согласованные.

Время задержки сигналов по линиям DCO...DC3, DBO...DB3, DAO...DA3, DX0...DX3 при передаче из одного канала в другой не превышает 80 нс. Задержка данных в канале по отношению к сигналам RA...RX не превышает 40 нс.

Основные электрические параметры (рис. 2) БИС ОИ K1802BB1 приведены в таблице, а временная диаграмма при передаче информации с канала DA в канал DB показана на рис. 3.

команде RETR банк регистров восстанавливается. При возврате по команде RET PSW и, следовательно, номер банка не изменяются. Поскольку подпрограммы обслуживания прерываний оканчиваются командой RETR, то после возврата из них всегда происходит восстановление выбранного в основной программе банка регистров. Состояние признака банка может быть считано командой MOV A, PSW, а затем проанализировано командой условного перехода JB4.

Для иллюстрации действия команд переключения банков регистров используется программа 5, позволяющая с помощью изменения состояния входа T0 (переключатель S1) выбрать банк RBO или RB1 и вывести на светодиоды порта P1 содержимое регистра R0 выбранного банка.

```

ПРОГРАММА 5.
ПРОГРАММА 5 ПРЕНАЗНАЧЕНА ДЛЯ ИЗУЧЕНИЯ ПЕРЕКЛЮЧЕНИЯ
БАНКОВ РЕГИСТРОВ. СОДЕРЖИМОЕ РЕГИСТРА R0 ВЫБРАННОГО
БАНКА ВЫВОДИТСЯ НА СВЕТОДИОДЫ P1. БАНК ВЫБИРАЕТСЯ
ПЕРЕКЛЮЧАТЕЛЕМ T0.
800      ORG 800H
800 88 01  MOV R0, #0 ;R0 --> R0 R00
802 05      SEL RB1
803 88 FF  MOV RB, #OFFH ;FF --> R0 RB1
805 05  M5:  SEL R00 ;КАКОЙ БАНК ВКЛЮЧИТЬ ?
806 26 09  JNT0 M50 ;T0=0 --> R00
808 05      SEL RB1 ;T0=1 --> RB1
809 05  M50: NOP
80A 14 10  CALL SUB ;ВЫЗВАТЬ ПОДПРОГРАММУ
80C F0      MOV A, R0
80D 39      OUTL P1, A ;ВЫВОД СОДЕРЖИМОГО R0
80E 04 05  JMP M5 ;ЗАЦИКЛИВАНИЕ
; ПОДПРОГРАММА РАБОТАЮЩАЯ С АРУЖНИМИ БАНКАМИ
810 05  SUB: SEL R00 ;КАКОЙ БАНК ВКЛЮЧИТЬ В ПОДПРОГРАММЕ ?
811 36 14  JNT1 MSUB ;T1=0 --> R00
813 05      SEL RB1 ;T1=1 --> RB1
814 33  MSUB: RETR

```

Выполнение программы показывает, что:

при начальной установке ОЭВМ автоматически включается банк RBO;

переключение банков регистров происходит непосредственно после выполнения команд SEL RB;

при возврате из подпрограммы по команде RETR восстанавливается прежний банк регистров.

Модульная конструкция учебной микроЭВМ УМПК-48 позволяет легко переходить от исследования самой ОЭВМ к изучению схемотехнического и программного обеспечения микроконтроллеров различных объектов на ее основе. В этом случае, кроме перечисленных модулей, пользователь может применять самостоятельно разработанные расширительные модули для реализации необходимых функций ввода-вывода (например, ввода сигналов от датчиков, управления шаговыми двигателями и т. п.). Описанные модули можно использовать также для моделирования микроконтроллеров на этапах проработки вариантов схемотехнического обеспечения и отладки программ.

Адрес для запроса дополнительной информации:
103498, Москва, МИЭТ.

ЛИТЕРАТУРА

- Кобылинский А. В., Липовецкий Г. П. Однокристалльные микроЭВМ серии K1816 // Микропроцессорные средства и системы, 1986, № 1, с. 10—19.
- Преспухин Л. Н., Панфилов Д. И., Романенко О. А., Шаронин С. Г. Микропроцессорная лаборатория по изучению микропроцессорных комплектов с фиксированным набором команд // Микропроцессорные средства и системы, 1985, № 1, с. 77—81.

Статья поступила 8 января 1986 г.

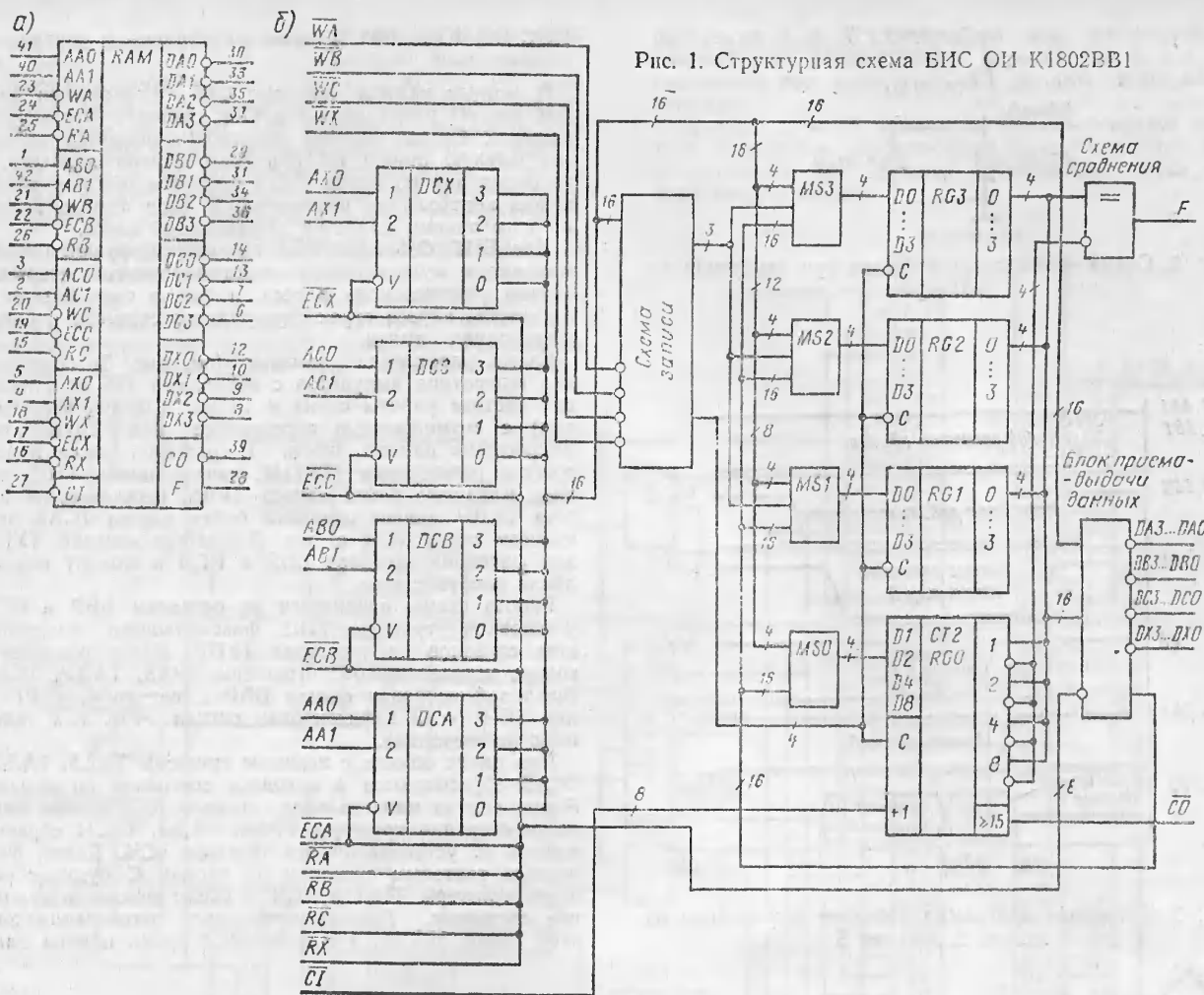


Рис. 1. Структурная схема БИС ОИ К1802ВВ1

Электрические параметры (при температуре $+25^{\circ}\text{C}$)

Таблица

Наименование параметра	Обозначение	Режим измерения	Норма		Единица измерения
			min	max	
Выходное напряжение «Лог. 0» для выходов канала X для остальных выходов	U_{OL}	$U_{SS} = -4,5 \text{ В}$ $I_H = 60 \text{ мА}$ $I_H = 15 \text{ мА}$		0,8 0,5	В
Выходное напряжение «Лог. 1»	U_{OH}	$U_{SS} = -4,5 \text{ В}$ $I_H = 1 \text{ мА}$	2,4		В
Ток потребления	I_{SS}	$U_{SS} = -5,5 \text{ В}$		280	мА
Входной ток логического нуля по входам R, W и входам каналов A, B, C, X по входам EC каналов A, B, C, X по входам A0, A1 каналов A, B, C, X по входу CI	I_{IL}	$U_{SS} = -5,5 \text{ В}$	-0,25 -0,5 -0,75 -2,0		мА
Входной ток «Лог. 1» по входной RW каналов A, B, C, X по входам EC каналов A, B, C, X по входам A0, A1 каналов A, B, C, X по входу CI	I_{IH}	$U_{SS} = U_I = -5,5 \text{ В}$		40 80 120 160 100	мкА
Выходной ток «Лог. 1» в состоянии «выключено» по входам-выходам каналов..., B, C	I_{OZH}	$U_{S_1} = U_0 = -5,5 \text{ В}$			мкА
Выходной ток «Лог. 1» по входам-выходам канала X, по выходу F	I_{OH}	$U_{SS} = U_0 = -5,5 \text{ В}$		100	мкА

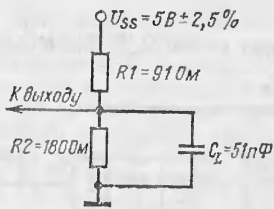


Рис. 2. Схема эквивалента нагрузки при измерении параметров

БИС ОИ К1802ВВ1 в схеме регенерации и управления динамической памятью

В модуле памяти емкостью 64К 18-разрядных слов (см. рис. 4) разряды 16-й и 17-й, считая с 0-го, — контрольные. Сигнал ошибки нечетности выдается на магистральную линию КА17 в фазе обмена данными, в то время как по линии КА16 выдается признак разрешения контроля на нечетность. В фазе вывода адреса по этим линиям выдается 18-разрядный адрес.

Две БИС ОИ К1802ВВ1 выполняют функции запоминания и мультиплексирования младшего и старшего байтов принимаемого адреса, а также счет и выдачу в качестве адреса строк 8-разрядного счетчика в цикле регенерации памяти.

Схема управления памятью (см. рис. 5) содержит два генератора импульсов с периодами 125 нс (тактовая частота работы схем) и 15 мкс (период регенерации) с возможностью перестройки под другой тип микросхемы памяти, шесть D-триггеров типа M-S: триггер регенерации (ТРГН), начала цикла ТНЦ, записи младшего байта адреса ТРАС, переключения адреса ТАДР, записи старшего байта адреса ТСАС, импульсов счета ТСЧ и два D-триггера-защелки ТУПР для фиксации сигналов ВБР и РГН в момент начала цикла работы схемы.

Работа схемы начинается по сигналам ВБР и РГН установкой триггера ТНЦ, фиксирующего состояние этих сигналов на триггерах ТУПР. Далее последовательно устанавливаются триггеры ТРАС, ТАДР, ТСАС (если зафиксирован сигнал ВБР и нет сигнала РГН) или ТСЧ (если зафиксирован сигнал РГН, т. е. идет цикл регенерации).

При цикле обмена с памятью триггеры ТРАС, ТАДР, ТСАС сбрасываются в исходное состояние по входам Р при снятии магистрального сигнала КСИА. При цикле регенерации триггеры ТРАС, ТАДР, ТРГН сбрасываются от установившегося триггера ТСЧ. Далее, очередным тактовым сигналом по входам С будут сброшены триггеры ТНЦ и ТСЧ и схема придет в исходное состояние. Длительность цикла регенерации этой схемы 500 нс, а минимальное время обмена дан-

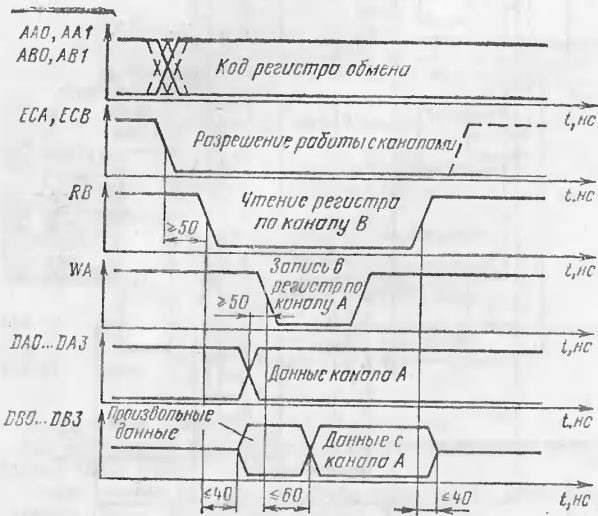


Рис. 3. Временная диаграмма передачи информации из канала А в канал В

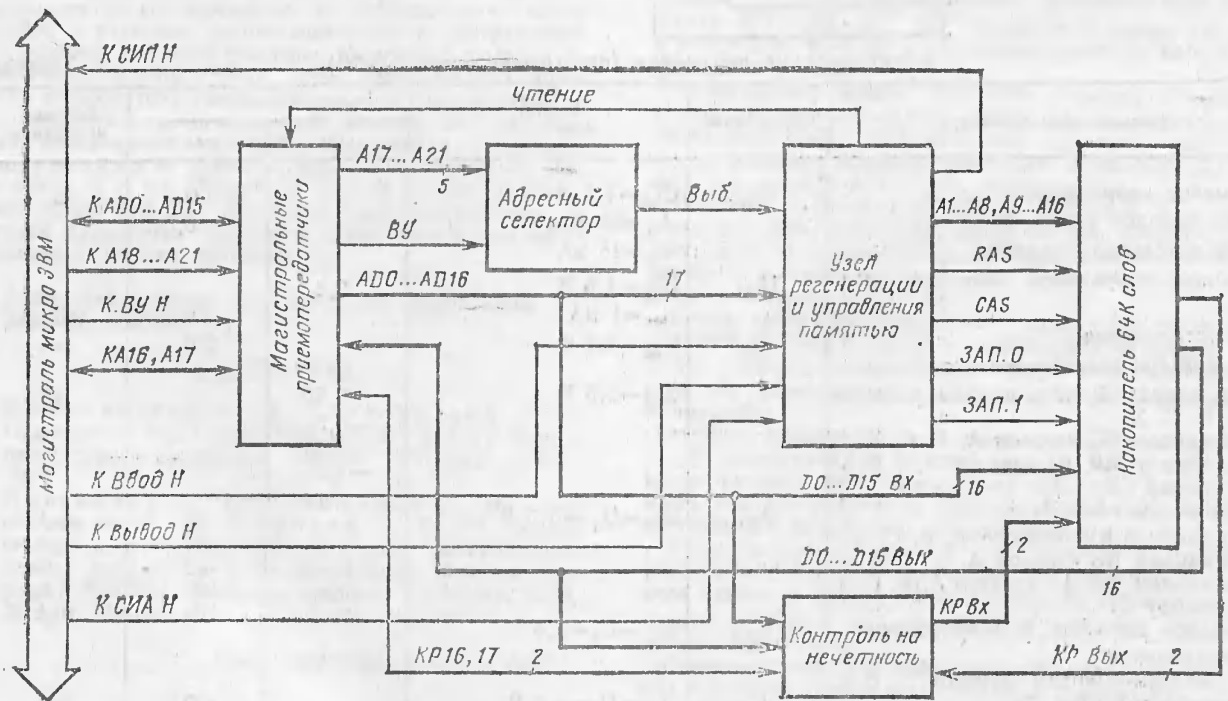


Рис. 4. Структурная схема модуля памяти

ными (до выставления сигнала КСИП на магистраль) 0,7 мкс.

Размеры платы модуля памяти — 135×240 мм (одинарная плата типа «Электроника 60»). Модуль содержит, кроме двух БИС ОИ К1802ВВ1 и 18-ти БИС К566РУ5, еще 25 микросхем серий К155, К533, К559 и К589. Схема выбора банка, устанавливаемого с помощью микропереключателей, позволяет модулю памяти работать в 22-разрядном адресном пространстве.

БИС ОИ К1802ВВ1 в схеме двухпортового ОЗУ на ИМС статического типа

В модуле двухпортового ОЗУ (64К 16-разрядных слов), на микросхемах памяти К537РУ9, использовано четыре микросхемы К1802ВВ1 (рис. 6). Модуль выполнен на двойной плате типа «Электроника 60» с

разъемами А и Б, отведенными под магистраль К1 (первый порт ввода-вывода), и разъемами В и Г, отведенными под магистраль К2 (второй порт ввода-вывода).

Простой схемный арбитр на плате позволяет работать с модулем одновременно двум микроЭВМ типа «Электроника 60» (каждая подключена через свой порт ввода-вывода).

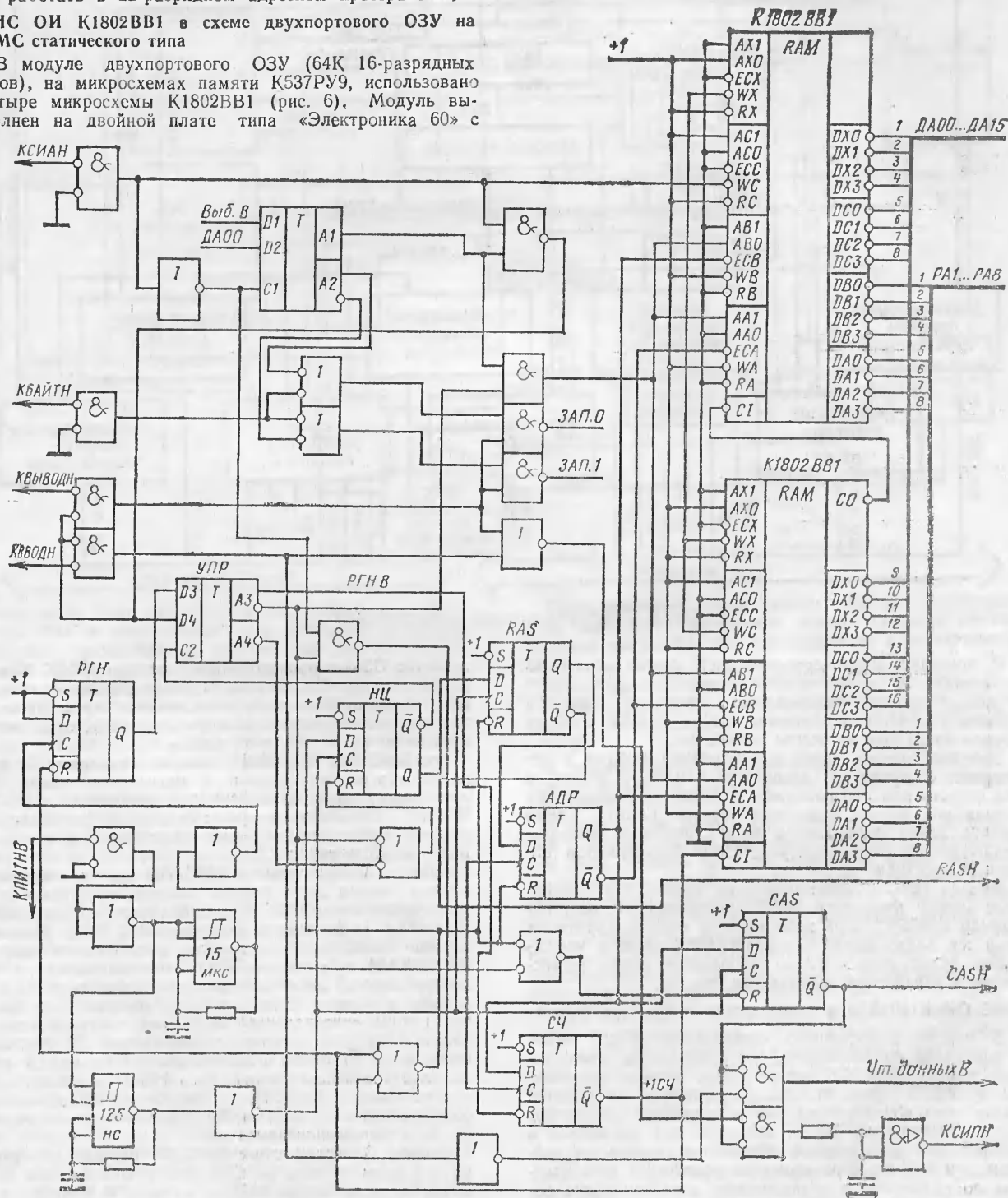


Рис. 5. Узел регенерации и управления памятью

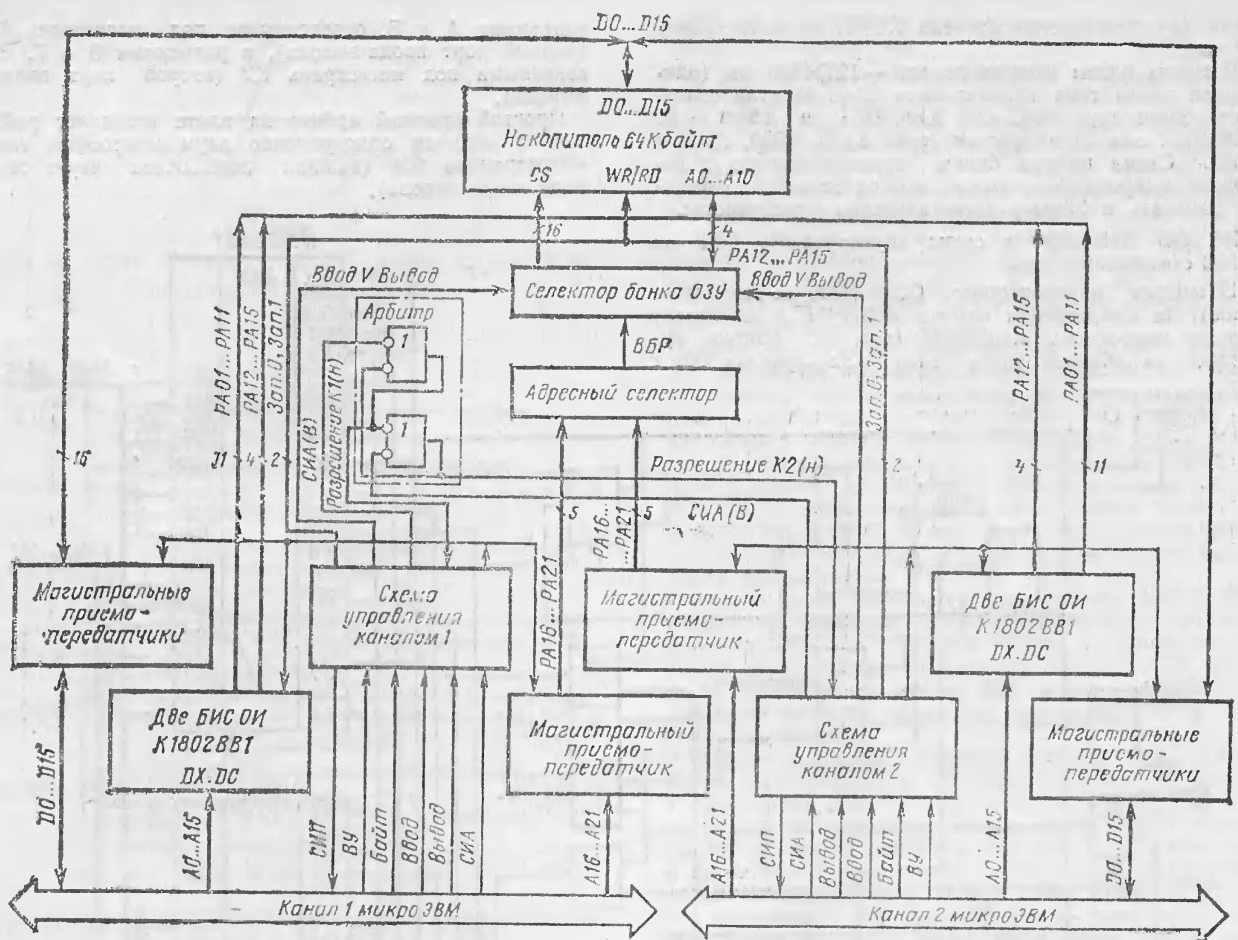


Рис. 6. Структурная схема двухпортового ОЗУ.

С помощью микропереключателей можно отключать магистрали K1 и K2, а также запрещать для записи с разных портов ввода-вывода отдельные страницы памяти (по 4К слов). Чтение же всего ОЗУ с обоих портов ввода-вывода всегда разрешено.

Это необходимо при использовании модуля в небольших системах без диспетчера памяти, а значит, и без средств для программного управления разрешением чтения или записи отдельных страниц памяти. Схема выбора банка, устанавливаемого с помощью микропереключателей, позволяет работать в 22-разрядном адресном пространстве.

Вместо ИМС статического типа K537PY9 в накопителе можно применить ИМС динамического типа, например K565PY5. При этом емкость модуля памяти на этой же плате может достигать 128К слов, а усложнение будет незначительно (добавится схема регенерации и управления памятью, см. рис. 5).

БИС ОИ K1802BB1 в схеме электронного «квиздиска»

В модуле электронного «квиздиска» (512К байт) с контролем на четность, построенного на микросхемах памяти K565PY5, использовано четыре микросхемы K1802BB1 (рис. 7). Модуль выполнен на двойной плате типа «Электроника 60» и имитирует накопитель типа «Электроника 7012», имеющий два механизма и работающий с одинарной плотностью записи на гибкий диск 203 мм. При этом быстродействие «квиздиска» по сравнению с механическим аналогом выше более чем в 1000 раз. Переключателем выбора режима можно устанавливать режим работы модуля также в

качестве ОЗУ с произвольным доступом (256К 18-разрядных слов). Такой модуль памяти может работать в системах с 22-разрядным адресным пространством, так как для произвольной адресации 256К слов требуется не менее 19 разрядов адреса.

Все БИС ОИ K1802BB1 своими каналами X подключаются непосредственно к магистрали микроЭВМ для приема-передачи информации по линиям KDA00...KDA15. Остальные разряды адреса и управляющие магистральные сигналы поступают через магистральные усилители серии K569.

Каналы А всех четырех БИС ОИ всегда открыты и служат только для вывода адреса, принимаемого с магистрали микроЭВМ и запоминаемого во внутреннем составе 16-разрядном регистре БИС ОИ. Разряды адреса KA00...KA15 совместно с разрядами адреса KA16...KA21 и сигналом KBY, поступающими на селектор адреса, дают возможность работать с банком памяти в режиме произвольного доступа или (если это режим «квиздиска») выбирают регистр команд (адрес 177170) и регистр данных (адрес 177172). При этом линии старшего и младшего байтов адреса объединены с помощью монтажного ИЛИ на входах микросхем памяти K565PY5 и выдаются поочередно в соответствии с сигналами PA и SA (от узлов регенерации и управления памятью).

Каналы В первых двух БИС ОИ служат для обмена младшим байтом данных между накопителем ОЗУ и магистралью микроЭВМ, а каналы В вторых двух БИС ОИ для обмена старшим байтом данных.

Отметим, что при работе в режиме «квиздиска»

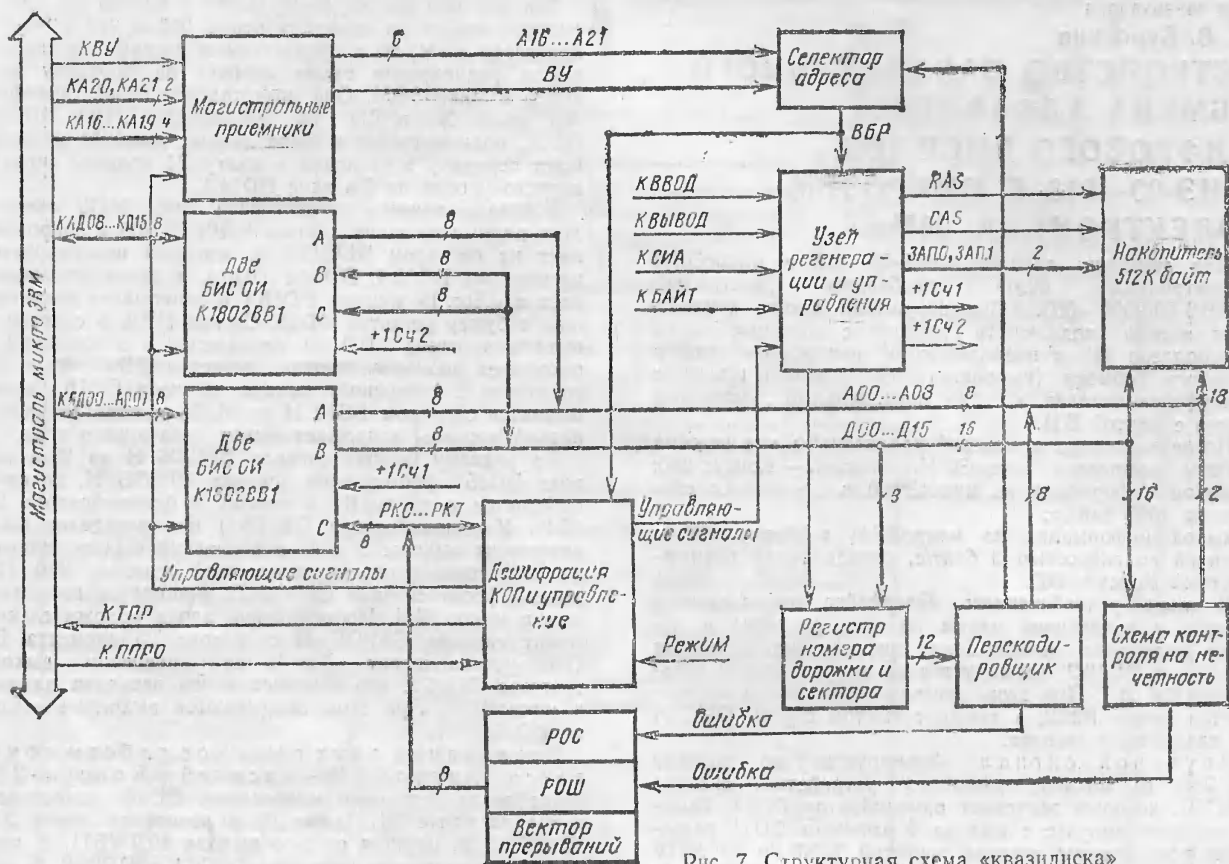


Рис. 7. Структурная схема «квазидиска»

первые 256К байт информации записываются в накопитель ОЗУ и считываются как младшие, а вторые 256К байт записываются как старшие байты 18-разрядного слова. В режиме произвольного доступа к памяти записываются и считываются параллельно младший и старший байты (обмен словами). Можно запоминать младший и старший байты данных во внутреннем 16-разрядном составном регистре БИС ОИ (буферном регистре считываемых или записываемых в ОЗУ данных).

Каналы С первых двух БИС ОИ служат для вывода регистра команд в режиме электронного «квазидиска» (внутренний составной 8-разрядный регистр БИС ОИ) и для ввода разрядов регистров РОШ, РОС и адреса вектора прерывания при выдаче их на магистраль микроЭВМ.

Каналы С вторых двух БИС ОИ объединены монтажными ИЛИ с каналами В первых двух БИС ОИ. Это необходимо для мультиплексирования старшего байта данных на линии младшего байта при выдаче на магистраль в режиме «квазидиска». В этом режиме обмен с магистралью идет только младшим байтом, как и при приеме данных из магистрали.

Составной 8-разрядный счетчик (внутри первых двух БИС ОИ) служит для регенерации ОЗУ, а такой же счетчик 2 (внутри вторых двух БИС ОИ) — для адресации буфера сектора на 128 байт, который организуется в том же накопителе. Это возможно потому, что действительная емкость механического накопителя, который имитируется, немного меньше объема в 512К байт, и составляет $128 \times 26 \times 77 \times 2 = 512\,512$ байт, т. е. остается свободным массив размером 11 776 байт. Здесь 26 — число секторов, 77 — число дорожек, 128 — емкость одного сектора в байтах и 2 — число дисков для двух механизмов.

Промежуточный буфер данных сектора необходим ради совпадения протоколов обмена данными по магистрали микроЭВМ для механического и электронного дисков.

Буфер данных сектора заполняется по команде в РК «Записать буфер», а перезаписывается по другим адресам в накопителе по команде в РК «Записать сектор».

В режиме чтения сектора буфер не организуется, а данные выдаются из накопителя прямо на магистраль микроЭВМ.

В отличие от ИГМД «Электроника 7012» информация с микроЭВМ обменивается побайтно в параллельном коде. Номера дорожки, сектора и номер механизма образуют первичный адрес. Адрес поступает на перекодирующий, построенный на ПЗУ, а с выхода ПЗУ вторичный адрес совместно со счетчиком байтов сектора подается на адресные входы микросхем памяти. Благодаря этому организуется непрерывное (без пропусков) адресное пространство для произвольного доступа к ячейкам памяти «квазидиска».

Хотя для более эффективной работы с эмулятором диска и упрощения аппаратуры в применяемой операционной системе необходим специальный драйвер, авторы воспользовались стандартным драйвером ДХ и эмулировали накопитель ГМД-7012 для иллюстрации возможности БИС К1802ВВ1.

Адрес для справок — 252039, Киев, проспект 40 лет Октября, 7, кв. 162. Телефон — 65-70-02(д.), Арсенину В. П.

Статья поступила 6 мая 1985 г.

И. В. Бурочкин

УСТРОЙСТВО ПАРАЛЛЕЛЬНОГО ОБМЕНА АЛФАВИТНО- ЦИФРОВОГО ДИСПЛЕЯ 15ИЭ-00—013 С МИКРОЭВМ «ЭЛЕКТРОНИКА 60М»

Для стыковки дисплея 15ИЭ-00—013 с микроЭВМ «Электроника 60М» необходим контроллер 15ВВВ-60/9600—003. При отсутствии такого контроллера можно подключить дисплей с помощью платы контроллера В1 и предлагаемого интерфейса параллельного обмена (выполняет все функции штатного интерфейса дисплея, а также параллельный дуплексный обмен с платой В1).

После внесенных изменений реализованы два режима работы комплекса микроЭВМ—дисплей—Консул-260: вывод информации из микроЭВМ в дисплей со скоростью 1500 байт/с;

вывод информации из микроЭВМ в Консул-260 и дисплей со скоростью 8 байт/с, определяемой возможностями Консула-260.

В схеме разработанного интерфейса параллельного обмена и измененной платы В1 (см. рисунок) в момент включения напряжения питания на элементах DD10.1 и DD10.2 формируется сигнал начальной установки ИУ Н*. При этом длительность сигнала определяется цепью R2C2, а также сигналом сброса ОЧС Н из клавиатуры дисплея.

Звуковой сигнал формируется по сигналу ВВВ10 из микропрограммного устройства дисплея (МПУ), который запускает мультивибратор DD11. Положительный импульс с выхода 4 элемента DD11 разрешает прохождение сигнала частотой 2400 Гц из МПУ через элементы DD6.2 и DD6.3 на базу транзистора VT1. С коллектора VT1 сигнал ДИН1 идет на один из выводов динамика, другой вывод которого подключен к +5 В через резистор R3. Длительностью управляет сигнал Фл2 В (его низкий уровень запрещает подачу звукового сигнала).

Флаги готовности клавиатуры, передатчика, приемника и режимов работы клавиатуры опрашиваются сигналами из МПУ (не прерываются во времени), инвертируемыми соответственно элементами DD2 и DD3. Эти сигналы разрешают прохождение сигналов флагов через соответствующие элементы DD4 и DD5 на вход 4 передатчика DD7. Выход 3 передатчика подключен к 8-разрядной шине данных дисплея КД00...КД07, а именно к КД07. Выходной вентилю DD7 открывается сигналом низкого уровня, формируемым микросхемами DD1 и DD6.

Приемник данных с клавиатуры реализован на передатчиках DD8 и DD9, входы которых подсоединены к шине данных клавиатуры ВК1...ВК7+ПАРИТЕТ, а выходы—к шине данных дисплея КД00...КД07.

Передачу данных в дисплей разрешает сигнал ВВВ1 Н, вырабатываемый в МПУ после опроса флага клавиатуры ФК Н-сигналом ВВ13 Н из МПУ.

Передатчик данных в микроЭВМ состоит из регистра DD15, триггера флага передатчика DD16.1 и схемы запрета. Входы регистра DD15 подключены к шине данных дисплея КД00...КД07, а выходы—к регистру данных клавиатуры (РДКЛ) D6 (В1) платы В1 (здесь и далее (В1) обозначает принадлежность элементов контроллеру В1) ЭВМ.

* Здесь и далее буквы «Н» и «В» после сокращенного названия сигнала обозначают соответственно низкий (не более 0,4 В) и высокий (не менее 2,4 В) активные уровни сигнала.

Так как при работе Консула 260 с платой В1 предусмотрен запрет на передачу кодов 016 и 017 в микропроцессор ЭВМ, то в предлагаемом интерфейсе дисплея также реализована схема запрета на передачу этих кодов в микроЭВМ. Она представляет собой дешифратор кодов 016 и 017 на элементах DD12.1...DD12.3, DD13, подключенный к шине данных дисплея, и запрещает передачу этих кодов в плату В1 подачей сигнала низкого уровня на 2-й вход DD14.1.

Передача данных организована так: МПУ выставляет данные на шину данных КД07...КД00 и сопровождает их сигналом ВЫВОД Н, который инвертируется элементами DD12.4, DD14.1 DD6.4, а затем устанавливает в «Лог. 1» триггер DD16.1 и записывает информацию в буфер регистра DD15. Сигнал ПРД В с прямого выхода триггера DD16.1 передается в клавиатуру и открывает выходные вентили регистра DD15. Флаг передатчика с инверсного выхода триггера DD16.1 опрашивается сигналом ВВ10 Н из МПУ и, пока сохраняет низкий уровень, запрещает вывод следующего кода.

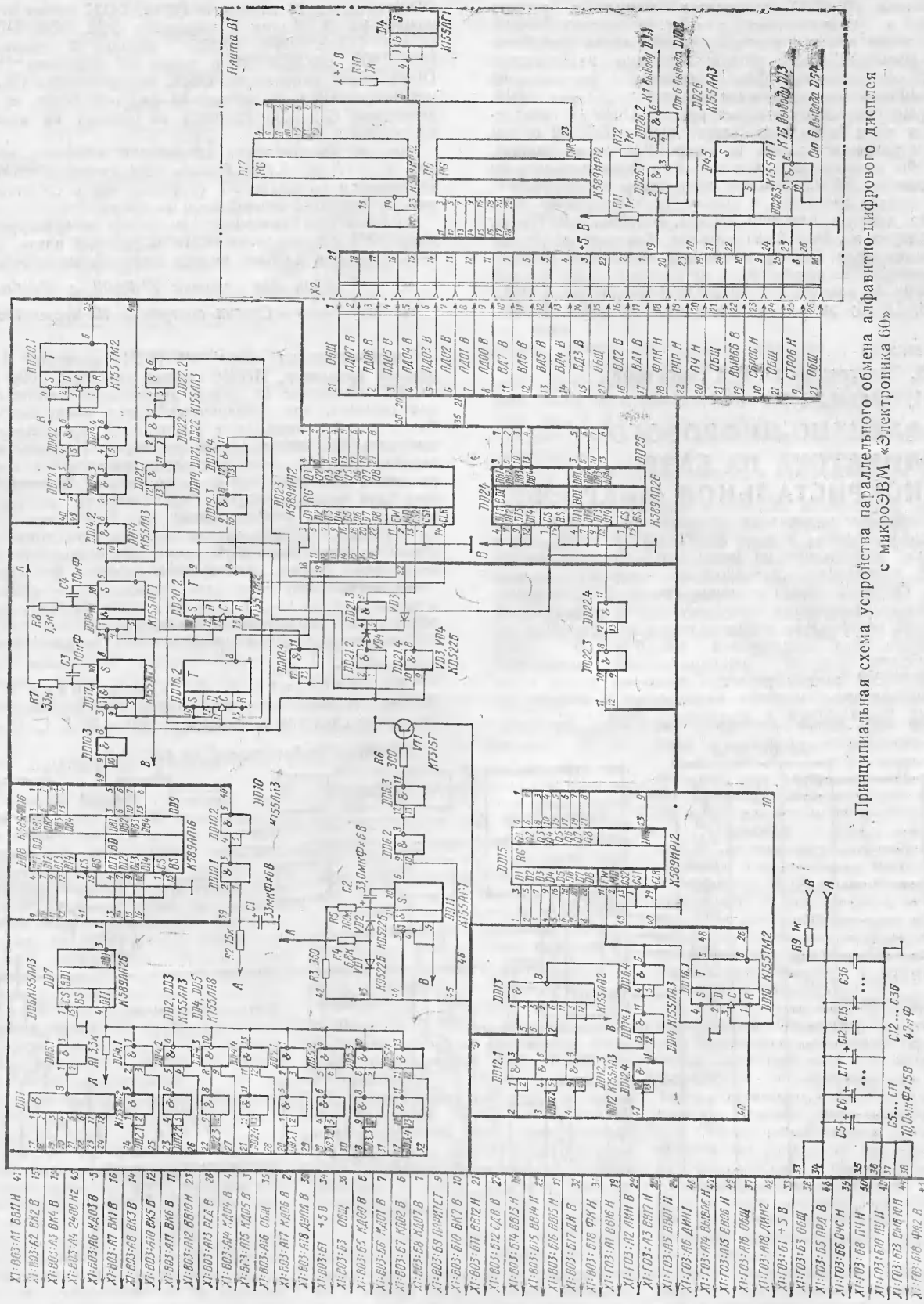
По заднему фронту сигнала ВВВ06 Н на 23-м выводе DD15 формируется сигнал СТРОБ Н, который передается в плату В1 к выводу 4 мультивибратора D4 (В1). У мультивибратора D4 (В1) при переделке были разорваны выводы 5 и 4, и последний был подключен к +5 В через вновь установленный резистор R10. Передний фронт сигнала СТРОБ Н инициирует ввод данных в микроЭВМ. По окончании ввода положительный фронт сигнала СБРОС Н с вывода 23 элемента D6 (В1) устанавливает «Лог. 1» на инверсном выходе триггера DD16.1, что означает конец передачи данных в микроЭВМ. При этом закрываются выходные вентили DD15.

Реализация двух режимов работы комплекса микроЭВМ—дисплей—Консул-260 потребовала установки микросхемы DD26 непосредственно на плате В1. Также была разорвана линия ЗАПРЕТ 66 В, идущая от 6-го вывода 40,2 (В1). В разрыв были включены элементы DD26.1 и DD26.2. К выводу 2 элемента DD26.1 были подключены линия сигнала ПЧ Н с клавиатуры дисплея для управления режимами работы комплекса и 5-й вывод D45 (В1), предварительно отсоединенный от R77 (В1). Была отсоединена цепь от 1-го вывода D34 (В1) и подсоединена к 8-му выводу DD26.3, вход 9 которой подключен к 6-му выводу D34 (В1). Работой элемента DD26.3 также управляет сигнал ПЧ Н.

Резисторы R11 и R12 сохраняют работоспособность платы В1 с Консулом-260 при отключенном дисплее.

Приемник данных из микроЭВМ состоит из регистра DD23, триггера флага приемника DD20.1, триггера флага запрета DD16.2 и устройства выбора набора знаков МПУ. Последнее включает пужный набор знаков в дисплей для верного отображения русских и латинских букв посылкой соответственно кодов 016 и 017 перед каждым видимым кодом из ЭВМ.

Данные из микроЭВМ поступают с выходов регистра данных печати (РДПЧ) D17 (В1) на входы регистра DD23 и сопровождаются сигналом ВВВ66 В с 13-го вывода D17 (В1), который устанавливает в «Лог. 1» триггер флага приемника DD20.1 и триггер флага запрета DD16.2, а также сбрасывает в «Лог. 0» триггер DD20.2. Сигнал ФПК Н с инверсного выхода триггера DD16.2 поступает на вход 5 элемента DD26.2 и запрещает ввод новых данных в дисплей. Флаг приемника с инверсного выхода DD20.1 опрашивается сигналом ВВ01 Н из МПУ. После этого вырабатывается сигнал ВВ06 Н, открывающий передатчики DD24 и DD25. На их выходах формируется код 016 или 017 в зависимости от дешифровки элементом DD22.3 поступающих данных из ЭВМ. Выходы DD24 и DD25 соединены с шиной данных дисплея. Сигнал ВВС1 Н сбрасывает в «Лог. 0» флаг приемника (DD20.1) и запускает последовательную цепь мультивибраторов DD17 и DD18. Импульс с выхода последнего вновь устанавливает флаг



Принципиальная схема устройства параллельного обмена адресно-цифрового сигнала с микроЭВМ «Электроника 60»

приемника (DD20.1), записывает данные в регистр DD23 и устанавливает в «Лог.1» триггер DD20.2. MPU после анализа флага приемника вновь вырабатывает сигнал ВВ06 Н, вводящий данные из регистра DD23, сбрасывающий флаг приемника и запускающий последовательную цепь одновибраторов DD17 и DD18. Импульс последнего очищает регистр DD23, устанавливает в «Лог.0» триггер флага запрета DD16.2 и тем самым разрешает вывод из микроЭВМ новых данных.

Чтобы клавиатура дисплея функционально не зависела от MPU, на плате клавиатуры переключатель SA1 следует поставить в положение «Настройка». При работе дисплея клавиши «РЕД», «ЛИН», «ДУП» на клавиатуре дисплея были нажаты. Клавиша «ПЧ» нажимается, если необходим обмен данными только между микроЭВМ и дисплеем.

Вывод 14 элементов DD1...DD6, DD10...DD14, DD16...DD22, DD26; вывод 16 элементов DD7...DD9, DD24...

DD25 и вывод 24 элементов DD15, DD23 имеют напряжение +5 В. Вывод 7 элементов DD1...DD6, DD10...DD14, DD16...DD22, DD26; вывод 8 элементов DD7...DD9, DD24...DD25 и вывод 12 элементов DD15, DD23 имеют напряжение ОБЩ. Конденсаторы С5...С11 устанавливаются по одному на ряд микросхем, а конденсаторы С12...С36 ставятся по одному на каждую микросхему.

Монтаж электрических соединений выполнен проводом ПЭВТЛ-2/Ø 0,19. Разъем Х2 типа ГРПМ1-45 укрепляется на плате В1 (ответная часть соединяется витыми парами с интерфейсом дисплея).

Предлагаемый интерфейс выполнен в конструктиве микроЭВМ «Электроника 60М» на нулевой плате, устанавливаемой в дисплей вместо штатного интерфейса.

Телефон для справок: 37-20-00, г. Куйбышев.

Статья поступила 15 марта 1986 г.

УДК 681.32

А. П. Дмитренко, О. В. Старостенко

КОНТРОЛЛЕР

АЛФАВИТНО-ЦИФРОВОГО

ИНДИКАТОРА НА БАЗЕ

ОДНОКРИСТАЛЬНОЙ МИКРОЭВМ

Контроллер индикатора предназначен для последовательного приема в коде КОИ7 (ГОСТ 13052-74) и вывода информации на индикатор, представляющий собой шестнадцать светодиодных точечных матриц 5×7. Обмен данными с управляющей ЭВМ происходит по старт-стопному протоколу обмена информацией. Питание контроллера осуществляется от источника постоянного тока напряжением $5В \pm 5\%$. В состав контроллера кроме однокристальной микроЭВМ КМ1816ВЕ48 входят регистры хранения кодов строк с дешифратором, матрица индикаторов с формирователями вытекающего и втекающего тока (рис. 1).

Однокристальная микроЭВМ (DD1) содержит 8-разрядный процессор, ППЗУ команд емкостью 1024 байта, ОЗУ емкостью 64 байта, 8-разрядный таймер-счетчик событий, три 8-разрядных порта ввода-вывода*. Такие ресурсы позволяют реализовать контроллер с минимальными аппаратными затратами. Информация о строках отображаемых символов хранится в восьми регистрах DD3...DD10. Формирователи вытекающего тока для столбцов DD11...DD21 собраны на транзисторных ключах и обеспечивают ток для свечения точки матрицы. Формирователи втекающего тока для строк представляют собой ключи, обеспечивающие свечение всей строки. Электрическая схема формирователей втекающего тока для первых восьми факторов представлена на рис. 2. Для остальных восьми индикаторов схема формирователей аналогична.

Индикаторное табло организовано в две строки

* Кобылинский А. В., Липовецкий Г. П. Однокристальные микроЭВМ серии К1816// Микропроцессорные средства и системы.— 1986.— № 1, С. 10—19.

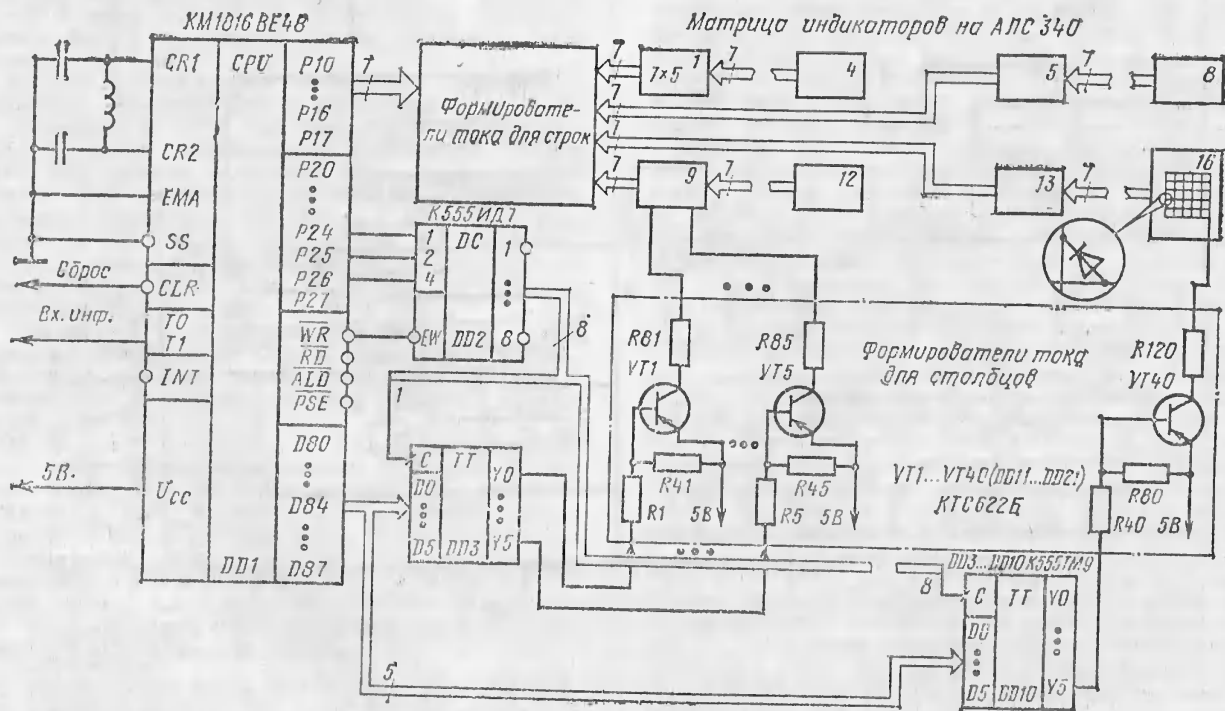


Рис. 1. Функциональная схема контроллера индикатора

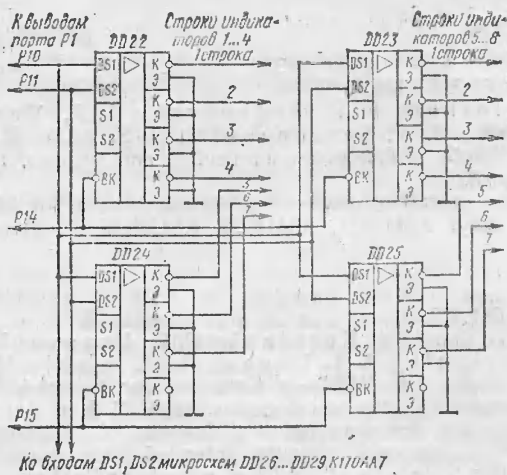


Рис. 2. Электрическая схема формирователей втекающего тока для строк

по восемь знакомест в каждой, а микроЭВМ осуществляет сканирование матрицы в 14 строк и 40 столбцов. Работа контроллера заключается в приеме байта информации, обработке принятой информации и формировании сигналов для активирования строк матрицы индикаторов. После инициализации микроЭВМ в ОЗУ DD1 формируется специальный буфер объемом 16 байт. В каждом байте буфера хранится адрес первой строки выводимых на соответствующее знакомство индикатора символов.

Первоначально индикатор погашен. После окончания очередного цикла сканирования матрицы индикаторов микроЭВМ выдает сигнал ГОТ ПР (готовность приема), после чего программа контроллера перехо-

дит на опрос входа таймера-счетчика Т1 и ожидает прихода стартового бита последовательной передачи информационного байта. Если стартбит не пришел, снимается сигнал ГОТ ПР и происходит очередной цикл регенерации изображения. МикроЭВМ принимает информационный символ и преобразует его в адрес первой строки символа в знакогенераторе. Знакогенератор реализован на внутреннем ППЗУ DD1. Символ закодирован семью байтами (строки символа), которые расположены в памяти последовательно.

Сформированный адрес очередного символа помещается в ячейку буфера ОЗУ, соответствующую младшему знакоместу индикатора. Одновременно вся информация буфера сдвигается влево на один разряд с потерей старшего знакоместа. Затем происходит очередной цикл регенерации изображения. Таким образом информация выводится на индикатор с младшего знакоместа матрицы со сдвигом остальных символов справа налево.

При обмене источником информации с контроллера кодами символов можно последовательно просматривать информацию любой длины в виде «бегущей строки». Если организовать протокол обмена таким образом, что в КИ будут передаваться не только коды символов, но и адрес знакоместа, то можно выводить информацию в любой отдельный разряд индикатора. Скорость приема информации в контроллере составляет 1,4К байт/с.

Объем управляющей программы контроллера составляет 0,3К байта. В остальном объеме ППЗУ DD1 размещается не менее 100 различных кодов символов. Описанный в статье контроллер индикации можно использовать при разработке алфавитно-цифровых индикаторных панелей цифровых измерительных приборов, различных микроЭВМ и отладочных систем.

Адрес для переписки: 290646, г. Львов, ул. Мира, 12, Политехнический институт.

Статья поступила 10 марта 1986 г.

(Ответ на письмо, с. 16).

ВАШЕ МНЕНИЕ?

Редакцию радует, что почта журнала несет не только многочисленные статьи, предлагаемые для публикации, но и множество читательских откликов. Многие письма затрагивают тему, ставшую предметом бурных дискуссий еще до выхода в свет первого номера журнала. Должны ли публикуемые статьи содержать подробные схемы устройств, полные тексты программ, описания практических «тонкостей» работы с микропроцессорной элементной базой или достаточно принятой во многих изданиях системы прямоугольников — «кирпичей» блоксхем? Публикуя письмо А. Ю. Командного, мы приглашаем читателей в своих письмах продолжить разговор на эту тему.

Понятно, что редакция «МП» не имеет возможности экспериментально проверить все публикуемые технические решения, и их работоспособность остается, как правило, на совести авторов статей. Рецензенты могут иногда предсказать, какими недостатками будут обладать публикуемые устройства, и предлагают авторам доработать статьи. Однако мы не считаем, что журнальные пуб-

ликации могут предопределить итоги макетирования конструкции.

Журнал пытается публиковать материалы, показывающие все богатое разнообразие вариантов применения МП-элементной базы и техники, поскольку имеющихся сегодня справочных материалов все еще часто оказывается недостаточно для получения ясного представления о работе описанных там устройств и элементов. Это заметно и например по публикуемому письму: при соединении входов CS и OE время доступа к ППЗУ K573PФ2 не увеличивается и определяется временем доступа по сигналу CS (≤ 450 нс, типично 300 нс). В ПЭВМ «Ириша» применено стандартное включение БИС KP580BH59. Кроме того, эта БИС вообще не формирует «трех сигналов», о которых упоминает А. Ю. Командный, — она формирует лишь единственный сигнал запроса прерывания INT, уровень которого запоминается в процессоре в конце каждого машинного цикла. Поэтому для чтения из этой БИС трех байт команды CALL тремя сигналами INTA процессора запрос INT сбрасывается по окончании второго INTA. Имеющиеся в нашем распоряжении материалы по БИС 8259 не дают рекомендаций по задержке сигналов, предла-

гая соединять выход INT 8259 и вход INT 8080А непосредственно.

И в будущем мы планируем публикацию трех основных типов статей по микропроцессорной схемотехнике и программному обеспечению.

1. Подробные описания конструкций и программ, таких, как ПЭВМ «Ириша», универсальные интерфейсы к серийным ЭВМ, существенно расширяющие их возможности, их программное обеспечение, — все это необходимо массовому пользователю и предназначено для самостоятельного повторения. Поэтому и все схемы приводятся полностью.

2. В статьях, иллюстрирующих применение МП-средств и БИС, будут публиковаться только существенно оригинальные или полезные фрагменты принципиальных схем. Такие статьи предназначены в основном для разработчиков аппаратуры.

3. Статьи, описывающие специализированные устройства или программное обеспечение, а также серийную аппаратуру, найдут отражение в журнале в виде короткой информации или рефератов, по которым можно получить дополнительную информацию у авторов.

Такая систематизация публикаций кажется нам наиболее целесообразной и возможна только на основе анализа мнений читателей.

УДК 681.325—181.48

Калошкин Э. П., Сухопаров А. И., Верниковский Е. А., Серженкович Д. С. **Оперативные запоминающие устройства серии К541** // Микропроцессорные средства и системы.—1986.—№ 6.—С. 3.

Рассмотрены режимы работы и основные характеристики микросхем статического ОЗУ с произвольной выборкой: К541РУ1, К541РУ2, К541РУ3.

UDC 681.325—181.48

Kaloshkin E. P., Suchoparov A. I., Vernikowsky E. A., Serzhenovich D. S. **RAMs of the K541 family.** / Microprocessor devices and systems, 1986, N 6, p. 3.

Main operation modes and technical data for static RAM chips K541RU1, K541RU2, K541RU3 are given.

УДК 681.322.1

Полосин А. Н., Карпинский Н. Г., Лозовой И. О., Половянюк А. И., Ургант О. В., Дябин М. И. **Учебный компьютер «Электроника УК НЦ»** // Микропроцессорные средства и системы, 1986, № 6, с. 14.

Рассмотрены состав, технические характеристики и основные режимы работы персональной ЭВМ «Электроника УК НЦ», предназначенной для оборудования кабинетов вычислительной техники всех типов учебных заведений.

UDC 681.322.1

Polosin A. N., Karpinsky N. G., Lozovoi I. O., Polovianiyuk A. I., Urgant O. V., Diabin M. I. **"Electronica UK NTS" — a Computer for School** // Microprocessor Devices and Systems, 1986, N 6, p. 14.

The paper describes technical features of "Electronica UK NTS" personal computer intended for studies of computer science in all kinds of schools and colleges.

УДК 681.3.06

Липаев В. В., Потапов А. И. **Длительность разработки сложных программных средств** // Микропроцессорные средства и системы.—1986.—№ 6.—С. 17.

В статье рассматриваются вопросы определения длительности разработки программных средств (ПС) микроЭВМ по накопленной базе данных реальных разработок. Вводятся и обосновываются понятия «невозможных» и «нерациональных» длительностей, зависящих в основном от объема ПС и других факторов.

UDC 681.3.06

Lipayev V. V., Potapov A. I. **On the duration of development of complex program tools** / Microprocessor devices and systems, 1986, N 6, p. 17.

The problems of effort required for development of complex software products and its estimation based on authors database on real projects are discussed. The concept of "impossible" and "irrational" effort is introduced, which is determined by program size and some other factors.

УДК 681.3.06—181.4

Комаров И. Е., Туманов А. А. **Использование ассемблерных фрагментов при программировании на языках Бейсик-01 и Бейсик-02 ПЭВМ «Искра 226»** // Микропроцессорные средства и системы.—1986.—№ 6.—С. 20.

Рассмотрены методы повышения быстродействия программ на языках «Бейсик-01» и «Бейсик-02» при работе со сложными структурами данных. Описаны метод определения и использования адресов нескольких массивов данных в одном ассемблерном фрагменте и способ ускоренного выполнения ассемблерных подпрограмм предварительной загрузкой ассемблерного фрагмента.

UDC 681.326—181.4

Komarov I. E., Tumanov A. A. **Some methods of effective use of assembly language fragments when programming ISKRA226 in BASIC-01 and BASIC-02.** / Microprocessor devices and systems, 1986, N 6, p. 20.

Some methods of performance improvement for programs written in BASIC and performing manipulations with complex data structures are described. The authors give a method for receiving and use of the address pointers at several data arrays in the same assembly language fragment and propose means for speeding up assembly code by preliminary load of such fragments.

УДК 681.3.06

Грудинин М. М., Сенченкова А. Ю. **Операционная система в ПЗУ** // Микропроцессорные средства и системы.—1986.—№ 6.—С. 22.

Рассматривается операционная система, способная работать в постоянном запоминающем устройстве (ОС ПЗУ), разработанная в СКТБ информационных систем Института проблем материаловедения АН УССР для микроЭВМ семейства «Электроника 60».

UDC 681.3.06

Grudin M. M., Senchenkova A. Yu. **ROM-resident operating system for "Electronica-60" microcomputer.** / Microprocessor devices and systems, 1986, N 6, p. 22.

A brief review of the ROM-resident expandable operating system and its applications for real-time tasks support is given.

УДК 681.325

Щелкунов Н. Н., Дианов А. П. **Техника программирования 8-разрядных микроконтроллеров** // Микропроцессорные средства и системы.—1986.—№ 6.—С. 23.

Приведены сведения о программировании одноплатного микроконтроллера реального времени, выполненного на базе серии КР580. Рассмотрена двухуровневая система ввода-вывода и средства поддержки режима реального времени, образующие независимую от аппаратуры программную среду управления общесистемными функциями микроконтроллера.

UDC 681.325

Sheikunov N. N., Dianov A. P. **8-bit microcontroller programming technique.** // Microprocessor devices and systems, 1986, N 6, p. 23.

The article reviews software support for a single-board real-time microcontroller with КР580 family central processor unit. The system software has two-level I/O subsystem and real-time support, creating a hardware-independent environment which performs system control operations.

УДК 681.325

Герштейн Ю. С. Си-реализация языка нисходящего разбора // Микропроцессорные средства и системы.— 1986.— № 6.— С. 36.

Описана Си-реализация языка нисходящего разбора (ЯНР), предназначенного для разработки диалоговых систем, интерпретаторов (компиляторов) с языков программирования и текстовых преобразователей.

УДК 681.3

Елинер Э. И., Клименко А. Д., Костылев Д. А. Локальная сеть на базе сетевого программного обеспечения АЛИСА // Микропроцессорные средства и системы.— 1986.— № 5.— С. 41.

Приводятся результаты разработки локальной иерархической сети ВЦКПСМ. Описывается работа сетевого программного обеспечения АЛИСА и технических средств ЛВС. Излагаются принципы дальнейшего развития ЛВС ВЦКПСМ.

УДК 681.325.5

Макаров А. И. Микропроцессорные системы с ограниченной программируемостью архитектуры // Микропроцессорные средства и системы.— 1986.— № 6.— С. 52.

Предлагается вариант построения многопроцессорной системы на базе процессоров с системой команд микроЭВМ «Электроника 60», позволяющий легко расширить систему добавлением однотипных модулей. Рассмотрены вопросы изменения архитектуры системы, аппаратной ее реализации и программного обеспечения. Уделено внимание повышению надежности и живучести системы в целом.

УДК 681.325.5—681.327

Баранов В. Г., Калягин С. Н., Бажанов Ю. С., Корсакова Т. А. Применение БИС К1801ВП1—035 в интерфейсных платах малых локальных вычислительных сетей // Микропроцессорные средства и системы.— 1986.— № 6.— С. 55.

Даны функциональные и технические характеристики интерфейсной платы на базе БИС К1801ВП1—035, с помощью которой за три месяца построена малая локальная вычислительная сеть типа «звезда» с центральным комплексом на основе мини-ЭВМ «Электроника 100/25» и шестью периферийными комплексами на основе «Электроника 60М».

УДК 681.325.5—181.4

Солонин В. Ю. Селекция импульсов полезного сигнала процессором КР580ИК80А // Микропроцессорные средства и системы.— 1986.— № 6.— С. 69.

Описана селекция импульсов полезного сигнала микропроцессором КР580ИК80А с подавлением помех, имеющих амплитуду как меньше, так и больше полезного сигнала.

УДК 681.323:621.391.266

Байков В. Д., Кабанов В. В., Попов А. М.—Микропроцессорная организация цифровых фильтров на базе МПК БИС КР580 // Микропроцессорные средства и системы.— 1986.— № 6.— С. 70.

Рассматривается организация мультипроцессорной системы на основе комплекта БИС КР580 и предлагается эффективный алгоритм вычисления свертки в данной системе без операций над словами двойной длины.

УДК 681.325

Gershtein Yu. S. C-language implementation of the downward lexic analyser. / Microprocessor devices and systems, 1986, N 6, p. 36.

Downward lexic analyser program written in C is described. The analyser may be useful for building dialogue systems, programming language compilers and interpreters, and text processors.

UDC 681.3

Eliner E. I., Kilimenko A. D., Kostylev D. A. Local area network in multiuser SM computer center supported by ALICE software. / Microprocessor devices and systems, 1986, N 6, p. 41.

The results of development of multilevel local area network for SM computers are discussed. The operation of LAN software and hardware is described. Some prospects of future system extension are also shown.

UDC 681.325.5

Макаров А. И. Микропроцессорные системы с ограниченной программируемостью архитектуры. / Microprocessor devices and systems, 1986, N 6, p. 52.

A variant of multiprocessor system incorporating LSI-11 compatible processors is proposed. The approach enable easy system expansion by adding new similar processor modules. The possibilities of architecture transformation of the whole system, its hardware implementation and software support is discussed. Attention to the problems of fail-free operation is also paid.

UDC 681.325.5-681.327

Баранов В. Г., Калягин С. Н., Бажанов Ю. С., Корсакова Т. А. Serial interface chip К1801РВ1-035 in communication interface cards of small computer LANs. : Microprocessor devices and systems, 1986, N 6, p. 55.

Authors describe operation and technical features of the communication interface card using К1801РВ1-035 LSIs. It took only three months to build a small star-type LAN with "Electronica-100/25" as a host and six "Electronica-60" as remote computers.

UDC 681.325.5-181.4

Солонин В. Ю. Noise-immune pulse selector with КУ580ИК80 CPU. / Microprocessor devices and systems, 1986, N 6, p. 69.

Adaptive noise-immune amplitude selector which suppresses noise that may be lower or even higher than signal is described.

UDC 681.323:621.391.266

Байков В. Д., Кабанов В. В., Попов А. М.—Microprocessor digital filters using КР580 CPU / Microprocessor devices and systems, 1986, N 6, p. 70.

The authors examine the design of multiprocessor digital filters implemented with КР580 CPU and propose high-performance convolution algorithm, which needs no operations with two-byte words.

Карась И. З. — Вопросы правового обеспечения информатики 1

МИКРОПРОЦЕССОРНАЯ ТЕХНИКА

Быстродействующая векторная вычислительная система ПС-320 5

Бартеньев А. А., Беляев Е. Г. — Программируемый контроллер для накопителей на гибких магнитных мини-дисках 4

Барышнев А. Б., Малашевич Б. М., Маликов А. Г., Натопта Е. Е., Хохлов М. М., Шабалин А. И. — Совмещение операционных систем персональных компьютеров с различной системой команд в одном издании 4

Белоус А. И., Горовой В. В., Прибыльский А. В., Силин А. В. — Интерфейсные интегральные микросхемы серии К583 2

Белоус А. И., Вайнилович О. С., Журба В. М., Сорочкина Н. Ю., Шалимо В. В. — БИС накапливающего 24-разрядного сумматора с интерфейсом 2

Барыков А. В. — Модификация устройств параллельного обмена микроЭВМ «Электроника 60» 5

Бобков В. А., Бурмистров Ю. П., Кособрухов В. А., Уткин Ю. В., Чернуха Б. Н. — Однокристалльные 4-разрядные микроЭВМ серии КР1820 1

Бобков В. А., Сухопаров А. И., Гузик В. Ф., Каляев А. В., Попов Ю. П., Станишевский О. Б. — Микропроцессор К1815ВФЗ для цифровой обработки сигналов 2

Волков Р. И., Горский В. П., Дыхуня В. Л., Коваленко С. С., Машевич П. Р. — Однокристалльный микропроцессор КМ1801ВМ3 4

Геник Б. Л., Муренко Л. Л., Иванов Е. А., Красовский С. Я. — Аппаратно-программные адаптеры операционных систем персональных ЭВМ 4

Говорун В. Н., Горбунов Н. В., Мамаков П. В., Рыбаков В. Г., Сытин А. Н., Холоденко Г. М. — МикроЭВМ и управляющие модули на базе МПК БИС К1810 5

Горовой В. В., Евдокимов В. А., Медведев В. И., Сахаров А. М. — БИС специализированного АЛУ К1815МА1 3

Давыдовская В. В., Малинин А. В., Таратын И. А., Усов Г. И. — Быстродействующие асинхронные статические ОЗУ К537РУ14 4

Делкарт В. М., Степанов В. Н., Крамфус И. Р. — Одноплатая микроЭВМ на базе МПК БИС К1810 5

Калошкин Э. П., Белоус А. И., Палненко В. В., Подрубный О. В., Журба В. М. — БИС универсального процессорного элемента К1815ВФ1 2

Калошкин Э. П., Васильев Л. В. — Микропроцессорный комплект БИС для конвейерных систем цифровой обработки сигналов 2

Калошкин Э. П., Сухопаров А. И., Верниковский Е. А., Сержанович Д. С. — Оперативные запоминающие устройства серии К541 6

Кобылинский А. В., Москалевский А. И., Темченко В. А. — Однокристалльный высокопроизводительный 16-разрядный микропроцессор КМ1810ВМ86 1

Кобылинский А. В., Липовецкий Г. П. — Однокристалльные микроЭВМ серии К1816 1

Кобылинский А. В., Береза А. В., Сабадаш Н. Г., Тесленко А. К., Темченко В. А. — Система команд микропроцессоров КМ1810ВМ86 2

Коваленко В. А., Олейник А. В., Пархоменко Л. П., Солдатенко Л. М. — БИС контроллера КР1818ВГ93 для накопителя на гибком диске 3

Кулешов В. И., Прибыльский А. В., Сякерский В. С., Яковлев Ю. В. — БИС ортогональной матрицы регистров сдвига К1815ИР1 3

Милованов А. И., Силин А. В., Сахаров А. М., Яковлев Ю. В. — Быстродействующий сумматор последовательных чисел К1815ИМ1 2

Панфилов Д. И., Романенко О. А., Сафанок В. С., Шаронин С. Г., Шаронин Ю. Г. — Учебная микроЭВМ на основе микропроцессора КМ1810ВМ1 5

Попов Ю. П., Милованов А. И., Медведев В. И., Васильев Л. В. — Преобразователь последовательных параллельных кодов К1815ПР1 3

Тарутин О. Б. — Интерфейс четырех внешних устройств стандарта ИРПР к микроЭВМ «Электроника 60» 5

Шкамарда А. Н. — Шестнадцатиразрядные микроЭВМ семейства СМ 1800 5

ПЕРСОНАЛЬНЫЕ КОМПЬЮТЕРЫ

Велихов Е. П., Персанцев И. Г., Рахимов А. Т., Рой Н. Н., Сирухин А. В., Щербаков О. А. — Персональный компьютер в системе автоматизации физического эксперимента 1

Гиглавиц А. В., Котляров В. П. — Возможен ли идеальный персональный компьютер? 4

Глушкова Г. Г., Иванов Е. А. — МикроЭВМ семейства «Электроника» 4

Кокорин В. С., Криднер Л. С., Попов А. А., Хохлов М. М. — Тенденция развития диалоговых вычислительных комплексов 4

Муренко Л. Л., Иванов Е. А., Красовский С. Я., Кушир В. Д. — Персональные ЭВМ «Электроника Т3-29МК» 4

Погорель С. Д., Слободянюк А. И., Суворов А. Е., Юрасов А. А. — Персональная ЭВМ «Нейрон И9.66» 4

Пыхтин В. Я. — ЕС 1840 — базовая персональная ЭВМ единой системы 4

Тиллини Д. А. — Персональная ЭВМ «Океан 240» 2

Ярошевская М. Б. — Персональная ЭВМ «Искра 1030.11» 4

Периферийные устройства персональных компьютеров

Безродный М. С. — Классификация и характеристики дисплеев для микроЭВМ 4

Малашевич Б. М., Романов Ф. И. — Конструкция и компоновка диалоговых вычислительных комплексов 4

Сорока С. И., Зябченко И. А., Измайков Р. И., Кузнецов К. Д., Кацнельсон З. Г. — Видеомониторы для персональных ЭВМ 4

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Александров А. Л., Башмаков Е. С., Гуткин М. Л., Либеров А. Б. — О стандарте языка Бейсик 5

Алексеенко А. Г., Гапоненко А. В., Иванников А. Д., Курилов И. Д. — Разработка и отладка микропрограммного обеспечения цифровых систем на основе секционированных микропроцессоров 1

Антонов Б. В., Глазер С. Ф., Маликов А. Г., Шабалин А. И. — Система автоматизации проектирования программного обеспечения для однокристалльной микроЭВМ К1816ВЕ48 3

Волков С. В., Дудкин М. В., Казмин А. Н., Мени А. А., Шерстюк А. В., Кузнецов М. Н., Целяпин А. Н., Ушкевич В. В. — Программное обеспечение коммуникационных процессоров в распределенных вычислительных комплексах 6

Баран Е. Д., Кошелева Е. И., Салмина Е. М. — Адаплизатор микропроцессорных систем	5	Шелкунов Н. Н., Дианов А. П. — Техника программирования 8-разрядных микроконтроллеров	6
Безобразов В. С., Ларин Б. И., Сохранов В. Ю., Шипкович А. А. — Отладочный комплекс для 32-разрядной микроЭВМ на основе ДВК-2	5	Как учить программированию	
Белов А. М., Иванов Е. А., Муренко Л. Л. — Комплексы кросс-программ «Электроника Микросек»	3	Программа курса «Основы информатики и вычислительной техники»	2
Большинский С. М., Полтава А. Н. — Драйвер НГМД удвоенной плотности для ОС РАФОС	6	Митрофанов С. П. — Пути решения компьютерной грамотности учащихся северной сельской опорной общеобразовательной средней школы	5
Брябрин В. М., Ландау И. Я., Неменман М. Е. — О системе кодирования для персональных ЭВМ	4	Штильман З. М., Штильман Б. М. — Метод программирования на Бейсике и Фокале на основе алгоритмического языка	3
Брябрин В. М., Чижов А. А. — Архитектура операционной системы АЛЬФА-ДОС/ДОС-16	4	ПРИМЕНЕНИЕ МИКРОПРОЦЕССОРНЫХ СРЕДСТВ	
Глазов А. Б., Костарев С. А., Суханова Е. В. — Эффективные программы умножения для микропроцессора КР580ИК80А	5	Акуней Ю. А., Антонов Б. В., Маликов А. Г., Марусин Т. В., Тер-Арутюнов В. Н. — Универсальный микроконтроллер «Электроника МК-48» на основе однокристалльной микроЭВМ К1816ВЕ48	3
Герштейн Ю. С. — СИ-реализация языка нисходящего разбора	6	Байков В. Д., Кабанов В. В., Попов А. М. — Многопроцессорная организация цифровых фильтров на базе МПК БИС КР580	6
Гнездилова Г. Г. — Резидентная сервисная система для персонального компьютера	4	Баранов В. Г., Калягин С. Н., Бажанов Ю. С., Корсакова Т. А. — Применение БИС К1801ВП1-035 в интерфейсных платах малых локальных вычислительных сетей	6
Грудинин М. М., Сенченкова А. Ю. — Операционная система в ПЗУ	6	Баронец В. Д., Бойцов Р. Н., Воробьев А. Г., Забегалова Г. И. — Устройство связи микроЭВМ с объектами управления по волоконно-оптическим линиям	2
Евлампиев Р. А., Галузо Е. В., Голованов В. П. — Отладочная система для однокристалльной микроЭВМ КМ1816ВЕ48	3	Безобразов В. С., Димов В. А., Мякотин А. В., Сохранов В. Ю., Шишкович А. А. — Контроллер графического дисплея	3
Илюкович А. А. — Информационно-поисковая система «Кадры» на базе микроЭВМ «Электроника 60»	6	Боложанин В. А., Скворцов В. А., Слизень Н. Е. — Комплекс микропроцессорных средств для информационно-измерительных систем	6
Жихарев В. И. — Программатор на базе микроЭВМ «Электроника ДЗ-28»	6	Гинзбург В. Д. — Микропроцессорный измеритель периода сигнала	6
Зобин Г. Я., Кривопапальцев Е. С., Минкович А. Б., Огнев А. И., Эпштейн Г. Ф. — Кросс-система для однокристалльной микроЭВМ КМ1816ВЕ48	3	Гусев В. Н., Купцов В. В., Пантелеев С. В., Роговцев А. А. — Интеллектуальный терминал на базе ДВК-1	1
Кобылинский А. В., Сабадах Н. Г., Тесленко А. К. — Система автоматизации программирования однокристалльной микроЭВМ	3	Дудолодов А. К. — Система автоматизации подготовки шахматистов «Дебют»	2
Комаров И. Е., Туманов А. А. — Использование ассемблерных фрагментов при программировании на языках Бейсик-01 и Бейсик-02 ПЭВМ «Искра 226»	6	Елинер Э. И., Клименко А. Д., Костылев Д. А. — Локальная сеть на базе сетевого программного обеспечения АЛИСА	6
Корнилов А. Р., Костин А. Е. — Планировщик параллельных процессов для ОС ДВК	4	Кормин Е. Г. — Повышение отказоустойчивости систем автоматизации на основе микроЭВМ «Электроника 60М»	3
Корчак А. Е. — Бейсик/GS — язык программирования школьных и бытовых компьютеров	5	Куклин В. В., Калинин Н. М., Бобров Ю. А. — Отладочный модуль на базе однокристалльного микропроцессора К1801ВМ1	2
Лавров С. С. — Представление и использование знаний в автоматизированных системах	3	Курдюмов Г. Л. — Системы ручного управления непрерывно меняющимися сигналами	2
Лебедев Г. В. — Разработка интерактивных программ на основе принципа непосредственного редактирования информации	1	Ляпунов М. М., Беляев Б. И. — Персональная графическая станция ПЕГАС	3
Липаев В. В., Потапов А. И. — Длительность разработки сложных программных средств	6	Макаров А. И. — Многопроцессорные системы с ограниченной программируемостью архитектуры	6
Лобанов В. И. — Архитектура отладочных средств для микроконтроллера	3	Матвеев А. А., Пономарев Ю. П. — Микропроцессорный модем-2400 для каналов тональной частоты	6
Мозговой Г. П., Менова С. С., Семин Е. И., Трещалин О. В. — Мета-ассемблер МЕАСС для микропроцессорных систем с наращиваемой разрядностью	3	Мирская Г. Я. — Встроенная микропроцессорная система — основа современного средства измерения	1
Найденев А. В., Романенков В. А. — Программатор ППЗУ на базе микроЭВМ «Электроника К1-20»	5	Насруллаев Н. Н., Нусратов О. К., Ситков С. Б., Симонян Р. К., Дворянкина Е. Д. — Многотерминальная система отображения информации	3
Половинкин И. К. — Кросс-система для микроЭВМ «Электроника ДЗ-28» на базе ЭВМ «Искра 226»	5	Перельмутер В. М. — Микропроцессорные системы управления электроприводами	5
Поом К. Э., Моор А. Э., Ребане Р. В., Арулаане Т. Э. — Операционная система для ПЭВМ «Искра 226»	5	Преснухин Л. И., Куеств В. А., Зубарев П. В., Вильсон А. Л. — Класс ДВК для изучения дисциплин «Радиоэлектроника» и «Импульсная техника»	4
Попов А. Л. — Информационно-поисковая система «Библиотека» и ее реализация на персональной ЭВМ	6	Преснухин Л. Н., Соловьев А. Н., Кузнецов Н. Н., Семичастинов О. Л., Разумовский К. П. — Контроллеры индикации и клавиатуры на основе некоммутированных вентиляционных матриц	5
Попурин А. В. — Система программирования Диалог для микропроцессора КР580ИК80	2		
Тарасов В. В. — Несложная система отладки	5		
Тарков М. С. — Организация удаленного исполнения команд монитора РАФОС в вычислительной системе МИКРОС	6		
Телух В. П., Лебедева Г. В., Никитина Н. Л., Пухальская Г. И. — Размещаемый в ПЗУ пакет стандартных программ для микроЭВМ «Электроника 60» и ДВК	2		

Преснухин Л. Н., Белильников В. И., Волков Ю. И., Уртайт О. В., Шапки В. Г. — Адаптер локальной вычислительной сети на базе БИС К1801ВП1-065	6	Жулай С. Г., Конько В. В. — Микроконтроллер вывода информации на газоразрядную индикаторную панель	3
Раденко М. Е., Сеньков В. К. — Реализация волоконно-оптического интерфейса для микропроцессорных систем	2	Книгин С. Б. — Многоголосый электромузыкальный синтезатор	5
Сасов А. Ю. — Микротомография и цифровая обработка изображений на микроЭВМ «Искра 226»	1	Кушир В. Е., Паффилов Д. И., Шаронин С. Г. — Учебная микроЭВМ на основе однокристалльной ЭВМ КМ1816ВЕ48	6
Соловьев В. Ю. — Селекция импульсов полезного сигнала микропроцессором КР5801К80А	6	Лукьянов Д. А. — «Электроника 256К» — эмулятор диска для комплексов на основе микроЭВМ «Электроника 60» и ДВК	2
Фельдман В. М. — Проверка и ремонт микроЭВМ в условиях эксплуатации	2	Панфилов Д. И., Шаронин С. Г., Яковлев С. Е. — Организация обмена информацией с микроЭВМ в параллельном коде	5
Фролов Г. И., Косенков С. М., Шахнов В. А., Зайцев В. В., Куроедов А. В. — Комплексный класс технических средств на базе микроЭВМ «Электроника БК-0010Ш» и ДВК-2МШ	4	Попов С. Н. — Спецпроцессор на базе микропроцессора КР5801К80А в комплексе с мини-ЭВМ	3
Чернов В. Н., Шестаков М. Г., Устьян Е. В. — Использование БИС серии КР1802 для создания адаптера магистралей СМ ЭВМ и «Электроника 60»	3	Романов В. Ю., Барышников В. Н., Воронов М. А., Паначев Ф. И. — Графические возможности персональной ЭВМ «Ириша»	1
Цвелодуб О. В., Щелкунов Н. Н. — Контроллер локальной вычислительной сети	6	Романов В. Ю., Барышников В. Н., Воронов М. А., Паначев Ф. И. — Персональная ЭВМ «Ириша»: периферийные устройства, источник питания	3
Чмиль В. М., Ющенко Б. И. — Мультипроцессорная система контроля и управления приемными устройствами спутниковой связи	6	Селицкий Сем. С., Селицкий Ст. С., Сидоров В. Н. — Простой пультовой дисплей	3
Шевченко А. А. — Одноплатный многоцелевой микроконтроллер на базе БИС серии КР580	5	Тилинин Д. А., Глазачев Н. К., Айсанов Р. Б. — Персональная ЭВМ «Океан 240.2»	4
Щелкунов Н. Н., Дианов А. П. — Универсальный одноплатный микроконтроллер	5	Томчук А. А., Музалевский И. В., Вдовыко Т. В., Невзоров В. Н. — Модуль счетчика-таймера в стандарте микроЭВМ «Электроника 60»	2
Юрочкин А. Г. — Система автоматизации диагностики и контроля МСВТ на базе микроЭВМ «Электроника МС 1212»	5	Щелкунов Н. Н., Дианов А. П. — Процедуры программирования логических матриц	2
Юрочкин А. Г. — Принципы построения программного обеспечения комплексов диагностики и контроля МСВТ	5	Щербаков О. А. — Особенности применения ПЛМ в микропроцессорных системах	2
УЧЕБНЫЙ ЦЕНТР		ПЗУ — разнообразие применений	
Арсенин В. П., Воробьев А. С., Герасимович В. Н. — Применение микросхемы К1802ВВ1 для управления памятью	6	Лукьянов Д. А. — ПЗУ — универсальный элемент цифровой техники	1
Барышников В. Н., Быстров В. П., Воронов М. А., Паначев Ф. И., Романов В. Ю. — Модуль процессора персональной ЭВМ «Ириша»	2	Щелкунов Н. Н., Дианов А. П. — ПЗУ вместо произвольной логики	1
Барышников В. Н., Воронов М. А., Кулаков В. Б., Романов В. Ю. — Программа вывода символьной и графической информации персональной ЭВМ «Ириша»	4	Квазидиск	
Барышников В. Н., Воронов М. А., Галутин Ю. В., Романов В. Ю., Рушайло-Арно А. Л. — Программное обеспечение ПЭВМ «Ириша»	3	Злотник Е. М., Стежко И. К., Анищенко В. В., Киркоров С. И. — «Электронный диск» для вычислительных комплексов на базе микроЭВМ семейства «Электроника 60»	5
Бурочкин И. В. — Устройство параллельного обмена алфавитно-цифрового дисплея 15ИЭ-00-013 с микроЭВМ «Электроника 60»	6	Сорокин Ю. Ю., Лаврентьев В. В., Максимьяк С. П., Суба В. В. — «Электронный диск» для микроЭВМ «Электроника 60М»	5
Васильев С. И., Леонов В. Б. — Сопряжение микроЭВМ «Электроника ДЗ-28» с интерфейсом БИС КР580	5	«Круглый стол» журнала	
Гойхман П. А., Добровинский М. Е., Удовченко А. Б., Фланчик Б. С. — Устройство приоритетного прерывания для микроЭВМ «Электроника 60»	1	Громов Г. Р. — Микропроцессоры в Политехническом	2
Горбачев С. Ф., Демин А. П. — Оперативное запоминающее устройство с внешним скоростным каналом ввода-вывода информации в микроЭВМ «Электроника 60»	3	Громов Г. Р. — Автоформализация профессиональных знаний	3
Дианов А. П., Щелкунов Н. Н. — Технические средства программирования логических схем	2	Ершов А. П. — Как учить программированию	1
Дмитренко А. П., Накалов Е. Ф. — Контроллер клавишного пульта	5	Корнейчук А. А., Расторгуев А. А. — «Круглый стол» с острыми углами	2
Дмитренко А. П., Старостенко О. В. — Контроллер алфавитноцифрового индикатора на базе однокристалльной микроЭВМ	6	Консультационный пункт	
Дубровский И. И., Финякин Л. Н., Кафаров В. В. — Устройство параллельного ввода аналоговой информации в микроЭВМ	5	Арешев Т. А. — Патентная охрана микропроцессорной техники	3
		Письма читателей	
		Кетков Ю. Л. — Ошибки перевода и оригинала	5
		Сенин Г. В.	
		Справочная информация	
		Микросхема динамического ОЗУ КР565РУ6	1
		Микросхема статического ОЗУ КМ132РУ5	2
		Микросхемы статического ОЗУ КР132РУ6	3
		Микросхемы статического ОЗУ К537РУ6 и К537РУ9	4
		Репрограммируемое ПЗУ КМ558РР3	5

МИКРОПРОЦЕССОРНАЯ ТЕМАТИКА В КНИГАХ ИЗДАТЕЛЬСТВА „ФИНАНСЫ И СТАТИСТИКА”

Книги по данной тематике в планах издательства занимают значительное место.

Большой интерес читателей вызовет справочное издание под редакцией генерального конструктора СМ ЭВМ Н. Л. Прохорова «СМ ЭВМ: Комплексирование и применение» (2-е перераб. изд., 1986). Авторы — разработчики данных систем — стремились представить читателям достаточно полную информацию по номенклатуре технических и программных средств, отразить принципиальные особенности построения архитектуры основных моделей СМ ЭВМ. В книге представлены системы реального времени, инструментальные, многопользовательские и мобильные операционные системы, пакеты прикладных программ, системы сетевой телеобработки. Важное практическое значение имеют сформулированные принципы комплексирования СМ ЭВМ. При подготовке книги были учтены пожелания и замечания ряда проектно-конструкторских организаций, заводов и научно-исследовательских институтов.

Для специалистов, работающих с СМ ЭВМ, в 1986 г. издана книга Л. Н. Столяра и В. А. Шапошникова «Средства проверки работоспособности оборудования СМ ЭВМ». В ней описываются два пакета прикладных программ, предназначенные для тестирования оборудования СМ ЭВМ и позволяющие повысить надежность вычислительных систем, достоверность решения задач, уменьшить частоту сбоев, сократить время восстановления работоспособности ЭВМ при возникновении неисправностей.

В этом году вышла книга Л. К. Гребенникова, Л. А. Летника «Программирование микропроцессорных систем на языке ПЛ/М». Книга рассчитана на разработчиков микропроцессорных систем, знакомых с их архитектурой и основами программирования. Ее основная задача — помочь быстрее освоить язык ПЛ/М, предназначенный для программирования микропроцессорных систем, построенных на базе отечественного микропроцессора серии КР580.

В плане выпуска литературы широко представлены практические руководства для широкого круга пользователей. Так, для специалистов по автоматизации учета полезна книга И. Н. Ладычука «Учет труда и заработной платы с использованием ЭВМ «Искра 555» (1986), в которой опи-

сан пакет прикладных программ автоматизированной обработки информации по учету труда и заработной платы.

В 1987 г. выйдет второе переработанное издание книги Г. В. Вигдорчика, А. Ю. Воробьева, В. Д. Праченко «Основы программирования на ассемблере для СМ ЭВМ», которое предназначено для лиц, не имеющих навыков работы на СМ ЭВМ.

Для системных и проблемных программистов, а также для инженеров и научных работников, занятых разработкой различных систем управления на базе СМ ЭВМ, будет полезно справочное издание «Операционная система СМ ЭВМ ОСРВ» (1987), посвященное наиболее распространенной системе для ЭВМ данного класса. Авторы особое внимание уделяют структуре, составу, функциям операционной системы реального времени, организации файловой системы и ввода-вывода, пакетам прикладных программ, выполняемых под управлением ОСРВ.

Книга М. А. Аппака «Автоматизированные места на базе микроЭВМ «Искра 226» знакомит экономистов, проектировщиков, конструкторов, программистов и работников других профессий с организацией автоматизированных рабочих мест на «Искре 226», созданием баз данных, языками пользователей для различных АРМов. Изложение иллюстрируется конкретными примерами реализации автоматизированных рабочих мест.

Непрофессиональным пользователям адресуются практические руководства В. Э. Баласяна, С. В. Богдюкевича, В. А. Шахвердова «Программирование на микроЭВМ «Искра 226». В нем приводятся основные сведения о данной машине, необходимые для работы на ней, излагаются основы программирования на Бейсике. Книга содержит большое количество примеров программ.

В связи с планируемым в текущей пятилетке массовым выпуском персональных ЭВМ Единой системы издательство предлагает читателям книгу «Персональные компьютеры Единой системы ЭВМ», подготовленную разработчиками этих ЭВМ — А. П. Запольским, В. Б. Шклярком, А. Н. Чистяковым. В ней приведены необходимые сведения об аппаратных особенностях и наборе программных средств персональных ЭВМ. Читатель, имеющий к тому же до-

ступ к ЭВМ, может последовательно проработать все этапы прохождения задач, проиграть на ЭВМ приведенные в книге многочисленные примеры и тем самым овладеть общими навыками работы с персональной ЭВМ. Книга ориентирована в первую очередь на лиц, не имеющих специальной подготовки в области вычислительной техники.

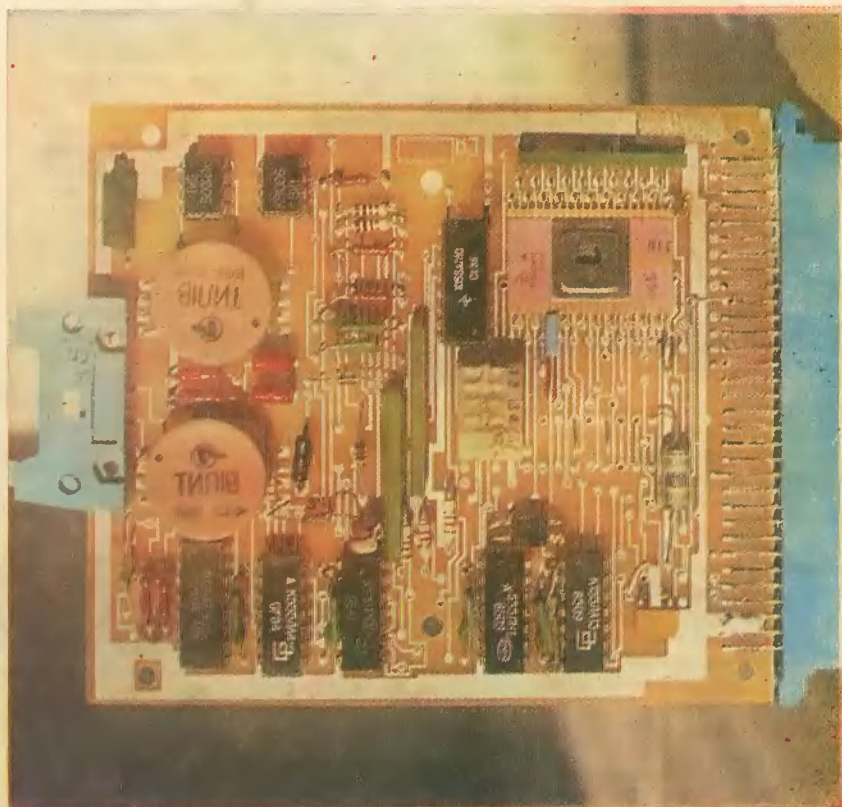
Проблемы организации производства и использования персональных ЭВМ, разрабатываемых в социалистических странах, находят свое отражение в сборниках статей «Вычислительная техника социалистических стран» (вып. 20, 1986 г., вып. 21, 22, 1987 г.), подготовленных в рамках Межправительственной комиссии по сотрудничеству социалистических стран в области вычислительной техники (МПК по ВТ). В них описываются советские персональные профессиональные ЭВМ Единой системы, приводятся их характеристики, состав базового программного обеспечения, рассматриваются особенности их комплексного централизованного обслуживания. Статьи сборника знакомят также с персональными ЭВМ производства ГДР, ПНР, НРБ, ЧССР, ВНР и их применением.

Внедрение персональных ЭВМ отечественного производства, совместимых с ПК фирмы ИБМ, сделало весьма актуальным издание ряда переводных работ по тематике «персональные ЭВМ».

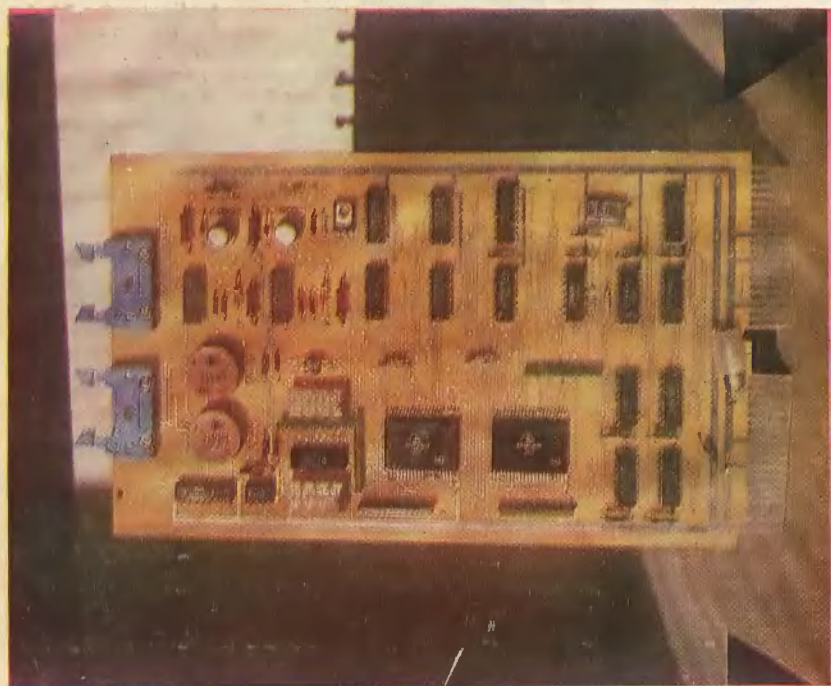
Перспективным направлением в массовом профессиональном использовании персональных компьютеров является создание систем коллективного пользования на основе архитектуры локальных сетей микроЭВМ или сетей, связывающих ПК с мини- и большими ЭВМ. В 1986 г. выпущена монография Д. Флинта «Локальные сети ЭВМ: архитектура, принципы построения и реализации».

Планируется издание работ: Д. Хазер, Дж. Хирт, Б. Хоукис «Операционная система MS-DOS: популярное руководство»; Г. Моррил «Бейсик для ПК ИЕМ»; У. Дойл «Табличный процессор Суперкалк для персонального компьютера». Все три книги представляют собой практические руководства, ориентированные на любого пользователя, и охватывают основные компоненты работы с ПК.

АДАПТЕР ЛОКАЛЬНОЙ ВЫЧИСЛИТЕЛЬНОЙ СЕТИ НА ОСНОВЕ БИС К1801ВП1-065



(К ст. Преснухина Л. Н. и др.)



Сетевой адаптер для интерфейсного модуля УК НЦ

Сетевой адаптер с двумя каналами: адаптер кольцевой сети и стандартное устройство последовательного обмена, работающего на линию «20мА — токовая петля».