

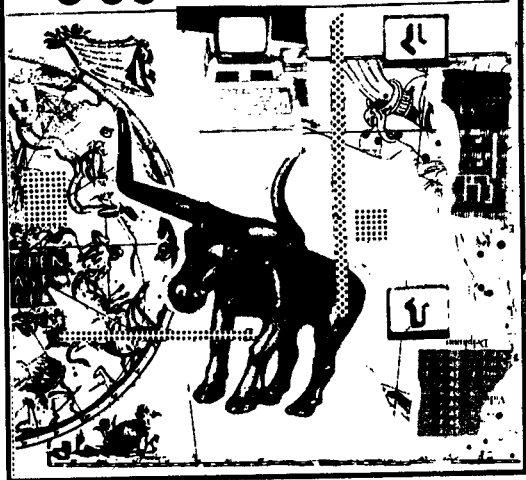
## Планируемое содержание выпуска 5'95:

- Практикум на ассемблере (продолжение)
- Руководство системного программиста БК-0011(М) (продолжение)
- Мышь и джойстик
- Электронный диск
- Нестандартные шрифты на БЕЙСИКе и ассемблере
- Стереоскопические изображения на БК
- Начинающему пользователю: что такое машинные коды драйверы принтеров
- Обмен опытом

## ПЕРСОНАЛЬНЫЙ КОМПЬЮТЕР

БК-0010  
Компьютика БК-0011м

5'95



## Подписка 1996 года Первое полугодие

Название журнала	Подписной индекс по каталогу «Роспечать»	Периодичность	Стоимость подписки на полугодие
«Персональный компьютер БК-0010—БК-0011М»	73177 для индивидуальных подписчиков	1 раз в 2 мес.	19 500 руб.
	73092 для предприятий и организаций		36 000 руб.
Название журнала	Подписка по каталогу «Книга-сервис»		Периодичность
«Персональный компьютер БК-0010—БК-0011М»	для индивидуальных подписчиков		1 раз в 2 мес.
	для предприятий и организаций		
«Компьютер УКНЦ»	для индивидуальных подписчиков		1 раз в 3 мес.
	для предприятий и организаций		
«Компьютика»	для индивидуальных подписчиков		1 раз в 2 мес.
Книга «Компьютер УКНЦ: справочник практика»	для индивидуальных подписчиков		—

Международная Академия Информатизации

# ПЕРСОНАЛЬНЫЙ КОМПЬЮТЕР



**Компьютика**

**4'95 (10)**

Издается с 1993 г.

**БК-0010**

**БК-0011м**

**В НОМЕРЕ**



Практикум на ассемблере

Руководство программиста  
БК-0011(М)

Математический сопроцессор для БК

Обзор баз данных для БК

Трехканальный музыкальный  
синтезатор для БК-0010(.01)

Доработка принтеров серии  
МС-6312 под печатающие  
головки нового типа

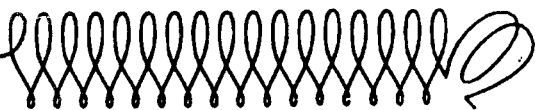
**НАЧИНАЮЩИМ:** подключение  
дискового к БК



Издательство **Компьютика**

Москва 1995

# Авторы выпуска



Аскеров Р.	Кузнецов А. И.
Бриндеев А. В.	Насыров Д. Р.
Животовский П. П.	Неробеев С. М.
Зальцман Ю. А.	Румянцев С. К.
Зыков В. В.	Сергеев Б. Н.
Кондратович М. И.	Сорокин А. В.
Кузнецов А. В.	Шаклеин В. Г.

---

**РЕДАКТОРЫ:** *ВАСИЛЬЕВ Б. М.*  
*УСЕНКОВ Д. Ю.*

Свидетельство о регистрации средства массовой информации № 013550  
от 26 апреля 1995 г.

**ПЕРЕПЕЧАТКА МАТЕРИАЛОВ ТОЛЬКО С РАЗРЕШЕНИЯ  
РЕДАКЦИИ ЖУРНАЛА**

E-Mail: [mail@infoobr.msk.su](mailto:mail@infoobr.msk.su)  
Телефон: (095) 151-19-40

© Издательство «Компьютика», 1995 г.



## От редакции

В наше хлопотное, далеко не мирное и, случается, безрадостное время трудно найти достойный повод для торжеств. Однако у нас — сотрудников редакции журнала «Персональный компьютер БК-0010 — БК-0011М», да, надеемся, и у наших читателей, — целых три торжественных события. Во-первых, это десятилетие компьютера БК, о чем уже говорилось во втором номере журнала за 1995 г. Во-вторых, сам выпуск, который вы сейчас держите в руках, можно считать юбилейным: с начала издания он десятый по счету (согласно титулу, где в скобках указана «сквозная» нумерация). И наконец, в-третьих, своего рода «День независимости»: впервые наши читатели держат в руках не приложение к журналу «Информатика и образование», а один из журналов нового независимого издательства «Компьютика».

Впрочем, «чур на стареньком»: новое название лишь отражает тот факт, что коллектив редакции бывших приложений «Персональный компьютер БК-0010 — БК-0011М» и «Персональный компьютер УКНЦ» выделился из состава издательства «Информатика и образование» в самостоятельную издательскую фирму. Конечно, произошло это вовсе не из желания «поучаствовать в всеобщей погоне за прибылью» (приложения по БК и УКНЦ и раньше держались скорее на энтузиазме редакторов) и даже не из стремления к «независимости от центра». Просто, перефразируя известную цитату из курса марксизма-ленинизма — «верхи не захотели далее заниматься отечественной вычислительной техникой, а низы (т. е. редакция приложений и читатели) не могли от них отказаться» — банальнейший конфликт с руководством «ИНФО». При этом коллектив редакции бывших приложений не изменился, как и основные «стратегические» направления в характере публикуемых материалов.

Что же касается названия издательства, думаем, оно всем понятно: «математика», «информатика»... «компьютика»! Наука о компьютерах: о самой ПЭВМ, ее периферии, программном обеспечении, об освоении работы с ней и практических вопросах пользования. Как ни странно, несмотря на «компьютерный бум», буквально захлестнувший в последнее время нашу страну, до сих пор эта наука (пока еще не получившая официально-энциклопедического статуса, но, надо думать, только «пока») названия не имеет. Рассматривается лишь некий круг вопросов практического характера с описательно-длинным общим наименованием: «вычислительная техника и ее применение». (У кого-то из читателей, наверное, уже вертится на языке ехидный вопрос: как же так, господа издатели! А «информатика» — это не «научно-компьютерный» термин? Действительно, долгое время за неимением лучшего им пользуются во всех «окомпьютеризованных» случаях. Но, строго говоря, это неправильно: «информатика» означает науку об информации и ее обработке, компьютер же в ее рамках должен рассматриваться не сам по себе, а только как инструмент для работы с информацией.)

Воздав должное третьему из названных вначале поводов для «торжественных речений», перейдем ко второму. Кажется, давно ли вышел в свет самый первый номер (со львом на обложке)? Неполных два года, отделяющих нас от этого события, могут

кому-то показаться слишком маленьким интервалом, чтобы обращать на него внимание. Но позволим себе хоть раз последовать логике Удава из всем известного мультика: «38 попугаев», да еще «с крылышком», звучит гораздо солиднее, чем «два слоненка». И воздать должное цифре «десять»: пусть это еще не десятилетие (и даже не пяти-), но, будем надеяться, такие праздничные даты у нас еще впереди.

Однако хватит «самохвалки», пора перейти, наконец, к поздравлениям настоящего «юбиляра» — компьютера БК (точнее, его старейшей модели БК-0010 с «пленочной» клавиатурой и единственным встроенным языком программирования ФОКАЛ). Десять лет назад первая машина с этим всем нам хорошо знакомым названием сошла с заводского конвейера. Может быть, до сих пор у кого-то из счастливых — БКманов стоит на столе «та самая» машина с заводским номером «1». (Реально их должно быть несколько, так как выпускались они на семи заводах только в СССР, а кроме того, еще в Польше, Чехословакии, Венгрии и даже в Китае.)

Десять лет для человека — срок не очень большой, но все же достаточный, чтобы из малыша-дошколенка воспитать дядю-старшеклассника. Для электронной же техники, а вычислительной в особенности, десятилетие иной раз сравнимо с веком: так быстро сменяются их поколения. Вот, к примеру, IBM-совместимые: через два-три года появляется новая модификация — РС, XT, 286, 386, 486, «Пентиум», а на горизонте — уже «686» (интересно, как его назовут в отечественной популярной литературе). А что же БК? Конечно же, и она, благодаря усилиям энтузиастов-БКшников, получила существенное развитие. Правда, наиболее бурно это происходит последние 3—5 лет: при социализме не было возможности организовывать собственные коммерческие предприятия, официальные же структуры чаще всего не уделяли БК ни малейшего внимания — так что не подстегиваемый «меркантильными соображениями» интерес разработчиков HARDa и SOFTa угасал довольно быстро (гордость за успешно написанные программы, конечно тоже неплохой стимул, но ведь и семью кормить надо!) Распространение же программ и схем аппаратных доработок и периферии старым методом «слухом земля полнится» (по цепочке от одного приятеля к другому) было, увы, слишком медленным даже в столице, а тем более «в глубинке». Отчасти спасали положение «клубы любителей БК», но оглянь-таки только в крупных городах. «Провинциальным» же БКшникам оставалось только уповать на публикации в журналах «Наука и жизнь», «Информатика и образование», «Вычислительная техника и ее применение» и др.

«Августовская революция» принесла россиянам пусть только «формальную», но все же свободу: хотя бы в организации собственного бизнеса. (Впрочем, в нашем случае это название не очень подходит, в сознании большинства сограждан оно, чаще всего вполне справедливо, является лишь «импортным» аналогом старого советского термина «спекуляция». С небогатых же в большинстве своем пользователей БК сверхприбылей не получишь, здесь дело затевается скорее «не юрсты ради», а из того же энтузиазма.)

Давайте посмотрим, с чем БК встречает свой десятилетний юбилей, что могут предложить своим клиентам многочисленные клубы, кооперативы и «частные лица», специализирующиеся на сфере БК.

Во-первых, сам компьютер. Еще «при рождении» он был не так уж и плох, как многие думают — даже по сравнению с общепризнанным «кумиром» IBM: при практически одинаковых прочих параметрах (память 64 кб, бытовой магнитофон в качестве накопителя и т. д.) 16-разрядный процессор БК обеспечивал не худшее быстродействие, чем первые модели 8-разрядных 8086-х, даже с учетом их более высокой тактовой частоты. А БКшные три цвета при 256×256 графических точках были реализованы на IBM только с появлением адаптера CGA гораздо позже начала выпуска самой машины (самые первые дисплеи MDA были монохромными и не имели графики). Теперь же

БК-0011(М) имеет 128 кб ОЗУ (причем очень немногие программы используют ее полностью, обычно ОЗУ задействуется для организации «электронного диска»), а на БК-0010(.01) дополнительное ОЗУ 16 кб уже стало общепринятым стандартом при подключении дисководов. В перспективе — появление ДОЗУ 128 кб для «десятой» модели, расширение памяти БК до 512 или 640 кб (это хотя и не серийная, но существующая не только на бумаге разработка) и даже «теоретические» два гигабайта (!) памяти с прямым, а не постраничным доступом. Поищите-ка другую такую же машину индивидуального, а не коллективного (как Сгау) пользования!

Упомянутые три цвета на экране также уже не предел. На БК-0011(М) можно выбирать триаду цветов из 16-цветной палитры, а изготовив не слишком сложную приставку (в свое время описанную в «ИНФО»), можно получить на экране БК-0010(.01) восемь цветов одновременно. (Впрочем, те же цвета с ненамного меньшим качеством можно получить и чисто программно безо всяких доработок схемы.) Решена проблема и с подключением к бытовому телевизору, когда-то волновавшая всех начинающих пользователей БК. Почти любой современный телеприемник уже снабжен видеовходом, а если нет — нетрудно купить видеоадаптер для подключения через антенное гнездо. При желании же вы можете и вовсе приобрести отдельный дисплей (черно-белый или цветной), причем сравнительно недорого.

Что остается еще? Почти все десять лет БК ругали на все лады за вечно «дребезжащую» клавиатуру. Теперь же всего за 30—40 тысяч рублей вы можете заказать установку клавиатуры повышенной надежности с пленочными контактами под кнопками. (Кстати, по удобству работы с русским алфавитом не всякая IBM может сравниться с БКшкой.)

Теперь посмотрим на периферию. Самое главное достижение в этой области, конечно же, **дисковод**. Если года три назад дисковод на БК был скорее «экзотикой», ныне в качестве средства хранения информации он уверенно вытесняет магнитофон. Вместе с дисководом пришли уже упомянутое дополнительное ОЗУ для БК-0010(.01), «дисковый» БЕЙСИК (возможность работать на встроенном вильнюсском БЕЙСИКе с диском), кнопка RESET и мало пока еще используемый прошитый в ПЗУ контроллера дисководов драйвер плавающей арифметики для ассемблера. Однако дисководы (в том числе и трехдюймовые) недолго были абсолютными фаворитами: сегодня начинается активное использование **жестких дисков** (винчестеров) емкостью от 20 до 80 Мб и более, оставляющих дисководам лишь роль средства переноса информации с одной машины на другую.

**Джойстики и мышь** давно уже стали привычным атрибутом на БК. Реже используются **световое перо, дигитайзер, чувствительный к надавливанию пера** (карандаша или шариковой ручки) **планшет**. И наконец, совсем недавно к БК адаптирован **фотопистолет** от «Денди» и созданы соответствующие игровые программы.

Совместно с БК успешно работают **принтеры** самых разных типов — от старых электронных пишущих машинок «Консул» или списанных телетайпных аппаратов до матричных EPSON и даже лазерных. А от струйных принтеров, являющихся на IBM «последним писком моды», многие БКшники «воротят нос» из-за проблем с заправкой краски.

Несколько лет назад в продаже имелись чехословацкие **графопостроители**, выпускаемые специально для БК. (Те же, у кого есть четвертый выпуск нашего журнала за 1994 г. и кто умеет работать с паяльником и слесарными инструментами, могут изготовить самодельный плоттер.)

Для БК имеется и две разновидности **сканеров**: на основе телекамеры и фотосчитывающая головка для плоттера или принтера (см. №3 за 1995 г.). Брянские же пользователи разрабатывают телеприставку типа «Videoblaster», которая позволит перегнать в БК любой понравившийся кадр из телепередачи или видеофильма.

Кроме всего этого, существуют и более «специфические» устройства: **программаторы ПЗУ, внешние блоки памяти для электронных дисков** и т. п. Перечислить все, не рискуя утомить читателя долгим повествованием, вряд ли удастся.

Спектр программного обеспечения для БК еще более широк: сложно упомянуть даже все его типы, не говоря уже о самих программах. Весь «джентльменский набор» SOFTWARE, который, по общепринятому мнению, должен быть на «истинно профессиональном» компьютере, на БК имеется.

**Дисковые операционные системы.** Если первые из них содержали лишь набор основных команд обращения к диску, то современная ДОС благодаря наличию файлового интерфейса по типу IBM-овского Norton Commander делает работу с БК практически неотличимой от пользования IBM-совместимой машиной. Последние версии наиболее популярных систем ANDOS, MK-DOS и NORD обеспечивают возможность работы с подкаталогами, а ANDOS, помимо того, использует IBM-овский формат записи файлов на диск, так что можно без труда переносить с IBM на БК и обратно тексты и графику. В комплект каждой ОС входит набор утилит, облегчающих работу с дисками, и драйверов для обеспечения совместимости разных ДОС между собой.

**Языки программирования.** Если в самой первой модификации БК-0010 был только ФОКАЛ, теперь к вашим услугам встроенный БЕЙСИК, а также загружаемые трансляторы с таких мощных языков, как ФОРТ, ЛОГО, ПАСКАЛЬ, СИ, ПРОЛОГ и др. Да и ассемблер, хотя он и считается в общепринятой классификации языком «низкого уровня», нельзя «сбрасывать со счетов». Работать с ним умеют уже многие пользователи БК, и к тому же, по свидетельству программистов, имевших дело с ассемблером IBM, БКшный ассемблер наиболее удобен из всех возможных. (Вообще же «любой недостаток имеет свои достоинства»: дефицит ОЗУ и низкое быстродействие заставляют программистов-БКманов совершенствовать свои алгоритмы, а после перехода на IBM бывшие пользователи БК становятся более чем высококлассными специалистами.)

**Текстовые редакторы.** В этой области «новинкой сезона» является VortEX 4.0, отличающийся от всем известного предшественника неограниченной длиной обрабатываемого текста (лишь бы уместился на дискете) и встроенный перекодировщик форматов БК-IBM. По мощности этот текстовый редактор вполне сравним с IBM-овскими Лексиконом и Wordom.

**Графические средства.** Простой в освоении и удобный БКPAINT давно уже снискал себе множество поклонников, как и векторные редакторы серии GRAF. Для БК-0011(М) разработана программа TesnoArt, обеспечивающая практически все мыслимые графические возможности, в том числе произвольную деформацию фрагментов изображения.

Отдельное «подсемейство» в роду машинной графики составляют **нестандартные шрифты**. Современные редакторы шрифтов и средства их печати позволяют создавать любые символы произвольного начертания и выводить ими тексты на принтер. А на БК-0011(М) нестандартный шрифт может быть загружен в «десяточный» монитор (содержащийся в дисковых ОС, например в ANDOSe) для постоянного использования при выводе текстов на экран взамен стандартного.

**Звуковые, речевые и музыкальные средства.** Один из лучших современных музыкальных редакторов — MAESTRO обеспечивает ввод, исполнение и запись в виде автономных модулей четырехголосных мелодий с сопровождением ударных инструментов по всем правилам музыкальной нотации. Все большую популярность приобретает и аппаратная доработка — музыкальный процессор, позволяющий получать на БК любые звуковые эффекты и высококачественную стереофоническую музыку. Что же касается синтезаторов речи, всем

известна программа «Говорун», позволяющая «озвучивать» любой русский текст. (На IBM подобные речевые синтезаторы появились лишь сравнительно недавно.)

**Базы данных.** Система управления BASIS, работающая в среде ANDOS, позволяет организовывать базы данных объемом до 800 кб (емкость стандартной дискеты) со всеми возможностями поиска требуемых записей. Теперь же, с подключением винчестера, вероятно появление баз данных еще большей длины, исчисляемой в мегабайтах.

**Электронные таблицы.** Имеющаяся на БК программа CALC имеет фактически все возможности, предоставляемые IBM-овским SuperCalcom, и пригодна не только для первичного обучения, но и для многих серьезных приложений.

**Деловые интегрированные системы.** На IBM такие программы, объединяющие в себе текстовый редактор, электронную таблицу и записную книжку, сегодня получили широкое распространение. Имеются подобные разработки и на БК, например «Блокнот», работающий в ОС ANDOS и МК-DOS.

**Системы цифрового моделирования.** На БК (в том числе и для десятой модели) имеются программы, позволяющие моделировать работу не очень сложных радиоэлектронных схем и выполнять в автоматическом режиме разводку двухсторонних печатных плат.

**Обучающие программы.** На БК они могут сравниться по количеству разве что с играми. Есть среди них средства для первоначального освоения компьютера детьми и обучения программированию, «автоматизированные опросники» и демонстрационные ролики по самым разным предметам. Более того, благодаря легкости работы с БЕЙСИКом любой преподаватель может самостоятельно разрабатывать учебные программы или адаптировать готовые. Что же касается учебных классов, локальные сети фирмы «ЛинТех» позволяют использовать в качестве рабочего места преподавателя вместо привычной ДВКшки высоконадежный IBM-совместимый компьютер (один на весь класс, ученические же машины — это дешевые, но не менее надежные БК) и даже обеспечивать работу на периферийных БКшках некоторых IBM-овских программ (правда, пока только монохромных и без графики).

**И конечно, игры.** Ими БКшка никогда не была обижена, но за последние 2—3 года качество игровых программ для БК выросло как минимум на порядок. Появились аналоги многих игр с IBM: «Объемный Тетрис», «Формула-1», «King's Bounty», «WarLords», «Eye of Be Hold»... По имеющейся информации, для БК пишется даже аналог DOOMa, нашумевшего среди пользователей IBM-совместимых ПК. Имеется и огромное количество более простых игр — как «по мотивам» реализованных на IBM и Spectrum, так и не имеющих аналогов на других компьютерах.

На этом мы прервем перечисление, так как спектр наработанного на БК программного обеспечения слишком обширен, чтобы даже пытаться перечислить все в одной короткой статье.

Итак, БК из простенькой машины, пригодной (по мнению некоторых) разве что для игр, за десять лет превратился в солидный профессиональный компьютер с возможностями, вполне достаточными для домашнего применения, учебных и большинства производственных целей, сравнимый по мощности с IBM PC/XT и во многом приближающийся к AT-286. Стоимость же стандартного комплекта остается невысокой (БК-0010.01 с новой клавиатурой — около 100 тыс. рублей, БК-0011М — приблизительно 160 тыс. рублей). Даже полный комплект с цветным монитором, принтером и двумя дисководом вряд ли обойдется дороже 500 тыс., тогда как самая слабенькая IBMка с монохромным дисплеем и одним дисководом на 360 кб (вдвое меньше, чем на БК!) — это минимум 300 долларов. Так что, надо думать, до «заката» компьютеров семейства БК еще очень далеко.





Продолжаем уроки нашей заочной школы программирования на ассемблере. В отличие от публиковавшихся в предыдущих выпусках журнала материалов Ю. А. Зальцмана, автор предлагаемой вниманию читателей серии статей основывает свою методику на практической работе с компьютером.

**Б. Н. Сергеев,**  
Москва

## Практикум на ассемблере

### Что нужно иметь для работы

Читать эту статью лучше всего сидя за готовым к работе БК-0010(.01) или БК-0011М, в который загружена особая СИСТЕМНАЯ ПРОГРАММА — АССЕМБЛЕР, а позднее — и ОТЛАДЧИК. Подготовьте дисковод (можно использовать и магнитофон, но с ним работать не так удобно). На диске (кассете) должны быть записаны следующие системные программы:

- **АССЕМБЛЕР-ТРАНСЛЯТОР** — позволяет создавать программы на языке ассемблера. Таких трансляторов сегодня разработано несколько, но дальнейшее изложение будет вестись применительно к ассемблер-транслятору Turbo4 (или его новой версии Turbo6). Можно воспользоваться и более старыми версиями, но они имеют несколько худшие возможности. Если же вместо Turbo у вас есть ассемблер-транслятор под другим названием, то возможны трудности при выполнении предлагаемых упражнений, так как команды другого варианта транслятора могут отличаться от принятых в Turbo. Тем не менее выполнение упражнений принципиально возможно в любом варианте ассемблер-транслятора\*;
- **ОТЛАДЧИК** — позволяет отлаживать программы, написанные на ассемблере и

оттранслированные в коды, писать небольшие программы на ассемблере и в кодах, просматривать содержимое ячеек памяти БК, проследить, как работает написанная вами (или чужая) программа, и многое другое.

Отладчиков тоже существует много, но ниже разговор пойдет о программе OTL16. Подойдут также версии OTL9 и OTL12, а чуть позже мы будем пользоваться и отладчиком MIRAGE;

- **ДИЗАССЕМБЛЕР** — с его помощью можно подготовить любую кодовую программу для последующей доработки в ассемблере. В нашем случае наиболее подходящим будет Dizas.WS.

Ко всем перечисленным системным программам желательно иметь документацию (описания и руководства пользователя). Необходимы также следующие справочные материалы:

- таблица кодов символов БК;
- список команд (инструкций) ассемблера, расположенных по алфавиту и по возрастанию кодов;
- описание ячеек системной области БК;
- описание прерываний ЕМТ.

Эти справочные данные приведены в Приложении.

\* Существует единый стандарт ассемблера для компьютеров типа БК, ДБК, СМ и других, построенных на основе процессоров серии 1801. Все имеющиеся версии ассемблер-трансляторов для БК обязаны полностью поддерживать данный стандарт (с точностью до сокращений названий команд, например RESET чаще всего для краткости заменяется на RES). Но в отдельных случаях предлагается использовать особые обозначения для некоторых команд и регистров процессора: CALL @#адрес вместо JSR R7, @#адрес, RET вместо RTS R7, кроме того, регистр R7 часто обозначается как PC, а R6 как SP. Главные же отличия между трансляторами заключаются в наборе так называемых псевдокоманд, служащих для включения в состав генерируемой исполняемой машинной программы необходимых данных — чисел, кодов символов, текстовых строк, адресов вызова подпрограмм и т. д., а также прочих вспомогательных целей. Возможны также отличия в допустимой длине имен меток и в ограничениях на сами эти имена, а также в используемых командах управления самим транслятором (чтение/запись ассемблерного текста, его редактирование, генерация и запись исполняемой программы и др.). Однако, располагая документацией к имеющемуся транслятору, не очень сложно учесть все эти отличия и без особых трудностей выполнить все предлагаемые в статье примеры. — *Прим. ред.*

## Ассемблер

Перед началом работы БК должен находиться в режиме пускового монитора, о чем сообщает «вопросительный знак» в левом верхнем углу экрана. Загрузим и запустим ассемблер Turbo4:

```
? M26000 "ВВОД"
ИМЯ? Turbo4 "ВВОД"
? S "ВВОД"
```

Здесь и далее обычным шрифтом показаны символы, выводимые компьютером на экран дисплея, а жирным — вводимые пользователем. Обозначение «ВВОД» (или «БК») — это клавиша, обозначенная на клавиатуре БК в виде изогнутой стрелки. Нажимая ее, мы сообщаем компьютеру, что набранная команда полностью введена и ее можно принять к исполнению.

Дисплей желательно иметь черно-белый, так как на нем четче видны буквы и знаки и, соответственно, меньше устают глаза. (Чтобы уберечь глаза от перенапряжения, не смотрите долго и непрерывно на экран (особенно на цветной), разглядывая мелкие детали изображения. Вообще же просиживать перед дисплеем более 1.5 — 2 часов в день не рекомендуется.)

Если при наборе команды вы ошиблись и обнаружили это до нажатия на «ВВОД», ошибку можно исправить с помощью клавиши «ЗАБОЙ». (Она находится под красной клавишей с надписью «СТОП».) При нажатии на нее курсор сдвинется на один шаг влево и сотрет последний напечатанный вами (ошибочный) символ. Теперь можно ввести новый, уже правильный знак\*.

Рассмотрим более подробно назначение каждой из введенных нами команд (показанных жирным шрифтом в вышеприведенном примере):

- **M26000** — указание компьютеру загрузить программу с адреса 26000 (для другой версии транслятора адрес может быть иным). Если же адрес не указывать вообще (сразу после набора символа «M» нажать «ВВОД»), программа будет загружена с адреса, с которого она ранее была записана на кассету или дискету;

- **Turbo4** — загрузка в память БК программы с именем Turbo4. (Затем нужно включить магнитофон на воспроизведение и ждать, пока не закончится загрузка. Если она прошла успешно, то на экране появится знак «?», иначе — сообщение «ОШИБКА»);
- **S** — запуск загруженной программы.

При работе с дисководом следует действовать согласно документации к используемой дисковой операционной системе (автор рекомендует пользователям применять систему ANDOS как наиболее удобную для работы с ассемблером).

После загрузки программы Turbo4 в левом верхнем углу экрана появляется знак «>». Это говорит о том, что ассемблер-транслятор находится в режиме диалога с пользователем, — другими словами, готов принять к исполнению любую команду из списка, приведенного в его описании.

Даем команду SC (последовательно нажав соответствующие две клавиши в регистре ЛАТ ЗАГЛ). Внешний вид экрана изменился — ассемблер перешел в режим редактирования. Теперь вы можете набирать свою программу.

Изначально курсор находится в левом верхнем углу экрана. Проверьте, что БК находится в режиме ЛАТ ЗАГЛ, если это не так, нажмите клавиши «ЛАТ» и «ЗАГЛ». Строчные, а тем более русские буквы ассемблер не понимает\*\*.

Наберем следующие команды:

```
MOV #270,R0
EMT 16
.END
```

После каждой команды нажимаем «ВВОД» — курсор перепрыгивает в начало следующей строки. На экране это будет выглядеть так:

```
MOV #270,R0
EMT 16
.END
```

\* Если ошибочный символ не последний введенный, нужно несколькими нажатиями клавиши «ЗАБОЙ» стереть все набранные после него знаки (вместе с ошибочным), а затем заново набрать всю стертую часть команды. Кроме того, такой способ можно применять только при вводе имени файла (на запрос монитора «ИМЯ?»). Если же вы неверно ввели число, например в команде M, лучше всего нажать «ПРОБЕЛ» и затем набрать все число заново. Напомним, что все эти рекомендации относятся к работе с пусковым монитором, в отладчиках, ассемблер-трансляторах и других программах редактирующие действия могут быть иными, о чем можно узнать из документации к каждой конкретной программе. — Прим. ред.

\*\* Это ограничение относится лишь к именам меток и самим командам ассемблера, а в комментариях же и во вставляемых в программу для последующего вывода на экран текстах допустимы любые символы. — Прим. ред.

Наша самая первая программа набрана. Посмотрите внимательно на дело своих рук: все введенное вами должно точно совпадать с приведенным здесь текстом до мельчайших подробностей — буквы, пробелы, запятые, цифры... Если вы все-таки ошиблись, то «не уходя» от неверно написанного символа, его тут же можно стереть с помощью клавиши «ЗАБОЙ». Другой способ — «наложить» курсор на ошибочный символ с помощью клавиш управления курсором (с короткими стрелками «←», «↑», «→» и «↓») и нажать клавишу с нужным символом. При этом текст «раздвигается», сохраняя на экране все ранее написанное. А чтобы убрать лишний символ, используйте клавишу «СДВИЖКА В СТРОКЕ».

Исправив обнаруженные ошибки, нажмите клавиши «ВС» и «ТАБ» — на экране снова появится признак диалогового режима, который мы видели после запуска ассемблер-транслятора.

Даем команду СО (в режиме ЛАТ-ЗАГЛ, но некоторые версии Turbo терпеть не могут и к русским буквам). Появляются сообщения «Ждите», «Ошибка нет» (или «Ошибка 000000»), «Длина ...». Для нас сейчас главное — увидеть сообщение «Ошибка нет» (или «Ошибка 000000»). Если это так, то можете себя поздравить с первой маленькой победой — ассемблер не нашел в тексте вашей программы ЯВНЫХ ГАУПОСТЕЙ!

Если же транслятор сообщает об ошибках, не отчаивайтесь! Не ошибается тот, кто ничего не делает. Посмотрите, что это за ошибки, вернитесь в редактор по команде SC, исправьте их и снова нажмите «ВС», «ТАБ», введите команду СО и т. д., пока не добьетесь заветного сообщения «Ошибка нет». Теперь можно запустить нашу программу на исполнение, для чего дайте команду RU.

Если набранная вами программа действительно написана правильно, то на экране появится значок «♥», а вы можете поздравить себя с окончательной победой — ваша первая в жизни программа работоспособна! Хотя вы, возможно, и не поняли, что она делает. А она выполнила МОЕ (т. е. автора данной статьи) задание напечатать на экране символ «бубна».

Кто-то из читателей наверняка спросит: «Ну и что? И зачем это нужно?» Ответ будет таким: это нужно вам так же, как первокласснику писать «палочки», чтобы в будущем из них составлять буквы, из букв — слова, из слов ... сочинение в любом объеме. Так что самое интересное еще впереди!

А теперь вы выходите в первый «самостоятельный полет». Положите перед собой таблицу кодов символов БК (которая приведена в Приложении). В ней для каждого символа сверху надписано число — его восьмисричный код. В нашей программе число 270 — это код символа «бубна». Заменяйте его кодами других символов и запускайте программу. Получается? Если ДА, то вы на верном пути. Если

НЕТ — забудьте это дело и играйте в компьютерные игры! (Последняя фраза конечно же шутка. Прочтите этот раздел еще раз и попробуйте сделать это снова. И рано или поздно у вас все получится: «терпение и труд все перетрут»!)

## Рекомендуемая литература об ассемблере

Если вы решились не отступать, то этой статье вам наверняка окажется недостаточно и придется заняться поисками информации в других источниках.

К большому сожалению, учебника для само-самого начинающего изучать самостоятельно программирование на ассемблере не существует (кроме разве лишь цикла статей Ю. А. Зальцмана, опубликованных в журналах «Персональный компьютер БК-0010 — БК-0011М», с №1 за 1994 г. по №3 за 1995 г.). А книги, список которых приводится ниже, для начинающего или непонятны, или недоступны. Но именно в них можно найти ответы на многие вопросы. Перечислим эти книги в порядке возрастания трудности понимания излагаемого материала. (т. е. первая книга из этого списка является наиболее подходящей для начального обучения.)

1. Митрюхин В. и др. Программирование на БК-0010.01. Чебоксары, 1993.
2. Фрэнк Т.С. PDP-11. Архитектура и программирование. 1986.
3. Сичановский Дж. У. Программирование на МАКРО-11 и организация PDP-11. М.: Радио и связь, 1985.

Позже, уже основательно «поднатюрен» в ассемблере БК, можно читать описания ЭВМ, использующих аналогичный язык ассемблера: PDP-11, «Электроника-60», «ДБК-2» и др. Много полезно можно найти и в журналах «Информатика и образование», «Вычислительная техника и ее применение», «Наука и жизнь» и «Байтик»\*.

В будущем окажутся полезными и такие книги:

4. Виргорчик Г. и др. Основы программирования на ассемблере для СМ ЭВМ. 1982.
5. Сингер М. Мини-ЭВМ PDP-11. Программирование на языке ассемблера и организация машины. 1984.
6. Лин В. PDP-11 и VAX-11. Архитектура ЭВМ и программирование на языке ассемблера. М.: Радио и связь, 1989.

Очень хорошо, если вы сперва поработаете на ФОКАЛе или БЕЙСИКЕ, так как принципы программирования на всех языках одинаковы, а на БЕЙСИКЕ и ФОКАЛе они усваиваются намного проще. Полученные же при этом знания будут полезны вам при изучении ассемблера.

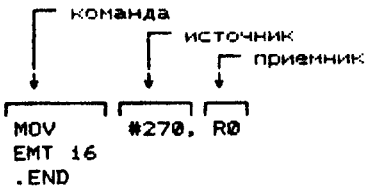
\* Во втором номере журнала «Персональный компьютер БК-0010 — БК-0011М» имеется список различных материалов о БК, ранее опубликованных в перечисленных изданиях. — Прим. ред.

Многие пользователи задают вопрос: что лучше, ассемблер или БЕЙСИК? Ответить на него можно так: БЕЙСИК, как и любой другой язык программирования **ВЫСОКОГО** уровня, можно сравнить с детским конструктором — набором готовых кубиков, из которых можно сложить домик, но отдельные блоки последнего не могут быть меньше, чем самый маленький кубик. Тогда ассемблер можно сравнить с пластилином, из которого можно слепить домик любой формы, даже с изогнутыми стенами любой кривизны. Но лепить из пластилина ту же стену домика труднее и дольше, чем сложить ее из готовых кубиков. Поэтому даже самые «крутые» программисты не отказываются иногда от БЕЙСИКа или ФОКАЛА, хотя часто и отзываются о них довольно пренебрежительно.

### Команда EMT 16

Разберем теперь «устройство» написанной нами программы (см. предыдущий пример) и то, как она выполняется. БК, а точнее, его «мозг», называемый центральным процессором (сокращенно ЦП), получив приказ пользователя RU, начинает выполнять нашу программу по одной строке.

На первом шаге он «смотрит» на первую строку программы — команду MOV.



Центральный процессор заглядывает в **ИСТОЧНИК**, видит там число 270, снимает с него копию (читает) и помещает эту копию в **ПРИЕМНИК**. Знак «#» перед числом 270 указывает процессору, что эти цифры надо понимать как числовую константу (в других случаях это может оказаться не константой, а, скажем, адресом содержащейся требуемое число ячейки памяти). Запятая отделяет приемник от источника. **ПРИЕМНИКОМ** в данном случае является «внутренний» регистр процессора с именем R0. (Всего таких регистров у ЦП восемь: R0, R1, R2, R3, R4, R5, R6 (или SP) и R7 (или PC), но R6 и R7 имеют специальное назначение и использовать их нужно с осторожностью.) Пока можно считать регистры своего рода аналогами переменных в БЕЙСИКе — когда мы указываем имя Rn (где n=0...5) в качестве источника или приемника, в операции используется содержащееся в нем числовое значение.

Итак, число (в данном случае 270) оказывается в регистре R0. Если это число понадобится, то можно в любое время снять с него копию (прочитать) и использовать для своей цели, оставив само число неизменным. Можно это число оттуда «выбросить» (очистить

приемник) или занести в регистр другое число, при этом старое теряется безвозвратно. Следует отметить, что в данном примере нужно поместить число именно в R0, а не, скажем, в R1, иначе программа не сможет выполнить требуемые действия.

На втором шаге ЦП обращается ко второй строке и «приказывает» команде EMT16 выполнить то, что она умеет. А она умеет делать только одно — печатать на экран изображение символа, код которого должен находиться **ИМЕННО** в регистре R0. Поэтому-то в предыдущей строке мы и поместили число 270 в R0. Функция EMT16, «увидев» код символа в регистре R0, «заглядывает» в таблицу кодов БК (которая в особом виде содержится в ПЗУ), находит нужный символ и печатает его на экране.

ЦП, убедившись, что EMT16 выполнила свою часть работы, а именно напечатала символ на экране, переходит к следующей строке программы. На третьем шаге он «смотрит» на третью строку программы: .END, понимает, что дальше делать нечего, останавливает работу и сообщает нам об этом, выдавая на экран знак «».

\* \* \*

То, что EMT16 при выполнении своих обязанностей только читает число в R0, но не «портит» его, показывает следующая программа, которая является развитием предыдущей. После запуска по команде RU на экране печатается столько символов «бубна», сколько команд EMT16 мы написали в программе. Значит, каждая EMT16, «заглядывая» в R0, видит там код символа, который предыдущая EMT16 оставила на месте, не «испортив» и не «выбросив».

```

MOV #270, R0
EMT 16
EMT 16
EMT 16
.END
  
```

В качестве самостоятельного задания усовершенствуйте программу: вместо числа 270 впишите число 7, а команду EMT 16 повторите 10 или более раз. Объясните, что делает новая программа.

Набранную программу можно сохранить для использования в будущем, записав ее на диск (кассету) по команде ST. Программа, записанная на диск, называется английским словом «файл».

Подав команду ST, вы увидите запрос транслятора: «Имя?». Введите какое-нибудь имя, под которым ваша программа будет записана на диск. Это может быть слово длиной от 1 до 8 символов из букв, цифр, знаков в регистрах РУС, ЛАТ, СТР или ЗАГЛ. (Здесь термин «регистр» следует понимать как «режим работы клавиатуры», а не ячейка памяти. Такое двойное толкование слова происходит от недостатка изобретательности сочинителей терминов.)

Чтобы получить самостоятельный файл, который можно запускать без загрузки в ассемблер-транслятор (загружаемый модуль), следует записать его на диск по команде SA. Однако для редактирования в трансляторе загружаемая программа уже не годится. Наилучшим будет такое решение: записывать два файла — один по команде ST (его потом можно загружать в транслятор и дорабатывать), а другой — по команде SA.

```

1) MOV #43, R0      2) MOV #365, R0      3) MOV #252, R0
   EMT 16           EMT 16           EMT 16
   MOV #36, R0      MOV #322, R0      MOV #243, R0
   EMT 16           EMT 16           EMT 16
   MOV #44, R0      MOV #301, R0      MOV #12, R0
   EMT 16           EMT 16           EMT 16
   MOV #35, R0      MOV #48, R0       MOV #246, R0
   EMT 16           EMT 16           EMT 16
   MOV #277, R0     MOV #41, R0       MOV #271, R0
   EMT 16           EMT 16           EMT 16
   .END             .END             .END

```

4) Напишите программу, которая нарисует такую картинку (используйте букву «o» и полуграфику):



## Команда EMT 24

Команда EMT16, как говорилось ранее, выводит символ на экран по коду, записанному в R0. Есть EMT и с другими номерами для выполнения других задач.

Попробуем поработать с EMT24. Добавьте к началу любой приведенной ранее программы следующий фрагмент из трех строк, вставляя вместо X и Y числа по своему выбору из предлагаемого диапазона.

```

MOV #X, R1 ; X - число от 0 до 77
MOV #Y, R2 ; Y - число от 0 до 26
EMT 24
.END

```

(Участки строк текста, входящие в состав программы на ассемблере и начинающиеся с символа «;», называются КОММЕНТАРИЯМИ. Их роль та же, что и у операторов REM в БЕЙСИКе: при трансляции они игнорируются и служат только для пояснений выполняемых в программе действий. — Прим. ред.)

Запустите программу. Что получится? Измените значения X и Y. Как это влияет на результат? Что будет, если вставляемые числа будут больше предлагаемых? А что, если вставить отрицательное число? Попробуйте предсказать, хотя бы приблизительно, в какую точку экрана попадет курсор, если задать X=48, а Y=11. Запустите программу и подсчитайте, на сколько позиций вы ошиблись.

После столь основательного экспериментального изучения вряд ли требуется подробно объяснить, что делает команда EMT24. Она устанавливает курсор по координатам X и Y, записанным, соответственно, в регистры R1 и R2.

*Рекомендация к последнему заданию:*

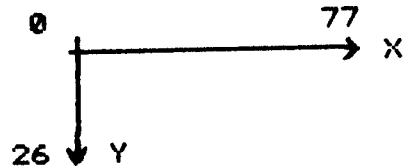
Загрузите с диска по команде LA сохраненную ранее программу (если ее нет, напишите

Сохраненный по команде ST на диске или на ленте файл можно загрузить в транслятор по команде LO, перейти в редактор по SC, внести изменения, опробовать в работе и опять записать на диск. Так ваша программа будет «расти» и вам не придется каждый раз набирать заново уже «сочиненный» фрагмент!

А вот примеры, работу которых вы должны разобрать сами. Запишите их на диск по команде ST. Они нам еще понадобятся.

заново). Перейдите в редактор по SC — на экране должен появиться текст загруженной программы. Подведите курсор в левый верхний угол и нажмите «ВВОД». Текст опустится на одну строку, освобождая место для добавления новых строк в начало.

Как уже было сказано, число X, записанное в R1, — это координата X, а число Y в R2 — координата Y на плоскости экрана, куда устанавливается курсор по EMT24 перед выводом очередного символа по EMT16.



Координатная сетка экрана немного отличается от принятой в математике, но к этому вы быстро привыкнете. Отличие это в том, что координата Y увеличивается сверху вниз (а в математике принято снизу вверх). При «широком» курсоре X изменяется от 0 до 37.

При работе программы иногда мешает введенная ассемблером служебная информация («Ждите...», «Длина ...» и др.). Ее можно убрать, если в самое начало текста добавить такие две строки:

```

MOV #14, R0
EMT 16

```

Код 14 (см. таблицу кодов) — это команда «СБРОС», она-то и очищает экран.

Приложения

1. Таблица кодов клавиш БК.

8-ричные			Писк														
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
			КТ				↓	←	ТАБ	←			СЕР	←			
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
10-чные			СУГ														
40	41	42	43	44	45	46	47	50	51	52	53	54	55	56	57		
-		°	•	я	z	&	'	(	)	*	+	.	-	.	/		
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47		
100	101	102	103	104	105	106	107	110	111	112	113	114	115	116	117		
•	А	В	С	Д	Е	Ф	В	Н	И	Ј	К	Л	М	Н	О		
64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79		
140	141	142	143	144	145	146	147	150	151	152	153	154	155	156	157		
.	а	б	с	д	е	ф	г	h	і	Ј	к	l	м	н	о		
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111		
200	201	202	203	204	205	206	207	210	211	212	213	214	215	216	217		
		ИИД		БЛОК								РП					
		СУ		РЕД								ар					
												СЕР					
128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143		
240	241	242	243	244	245	246	247	250	251	252	253	254	255	256	257		
Ѕ	┘	Ф	┘	┘	┘	┘	=	┘	┘	Г	Т	┘	┘	┘	┘		
160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175		
300	301	302	303	304	305	306	307	310	311	312	313	314	315	316	317		
я	а	б	ц	д	е	ф	г	х	и	я	к	л	м	н	о		
192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207		
340	341	342	343	344	345	346	347	350	351	352	353	354	355	356	357		
п	а	б	ц	д	е	ф	г	х	и	я	к	л	м	н	о		
224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239		

УСТ ТАБ = СУ М  
 СЕР ТАБ = СУ П  
 СЕР СС = СУ АР2 Ч

АР2↑ = 230/152  
 АР2↑ = 231/153  
 АР2↑ = 232/154  
 АР2↓ = 233/155

СУ = латЗАГЛ - 100/В  
 АР2 = латЗАГЛ + 140/В  
 НР = -20

20	21	22	23	24	25	26	27	30	31	32	33	34	35	36	37
		к	BC	ГТ		К—	I→	<→	→	↑	↓	↖	↗	↘	↙
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
		сyR		сyT								сyЭ	сyК	сyЧ	сyЪ
60	61	62	63	64	65	66	67	70	71	72	73	74	75	76	77
0	1	2	3	4	5	6	7	8	9	:	:	<	=	>	?
48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
120	121	122	123	124	125	126	127	130	131	132	133	134	135	136	137
P	Q	R	B	T	U	V	W	X	Y	Z	[	\	]	^	_
80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
160	161	162	163	164	165	166	167	170	171	172	173	174	175	176	177
p	q	r	s	t	u	v	w	x	y	z	(	)	-	.	■
112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
220	221	222	223	224	225	226	227	230	231	232	233	234	235	236	237
ШАГ	нр	нр	нр	нр	су	су	су	ар	→	е/н	у/ш	Сим	Экр	уст	под
144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
										ар:	ар:	ар,	ар-	ар.	ар/
260	261	262	263	264	265	266	267	270	271	272	273	274	275	276	277
↓	+	↑	↑	+	-	↓		◆	↓	+	↑	+	↑	+	■
176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
320	321	322	323	324	325	326	327	330	331	332	333	334	335	336	337
п	я	р	с	т	у	ж	в	ь	ы	е	ш	э	ц	ч	ъ
208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
360	361	362	363	364	365	366	367	370	371	372	373	374	375	376	377
П	Я	Р	С	Т	У	Ж	В	Ь	Ы	Э	Ш	Э	Ц	Ч	.
240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Курсор: е/н = есть/нет

у/ш = узкий/широкий = 64/32 симв/стр

РЕД = ар &lt;→

РП = ар СБР

ар = АР2 нр = нижний регистр = шифт

ГРАФ = су ар Е

ЗАП = су ар F

СТИР = су ар W

## 2. Инструкции ассемблера (по возрастанию их кодов)\*.

Код	Мнемоника	Условия	Назначение
000000	HALT		останов
1	WAIT		ожидание
2	RTI		возврат из прерывания с установкой трассировки
3	BPT		прерывание по вектору 14
4	IOT		прерывание по вектору 20
5	RESET		сброс шины внешних устройств
6	RTT		возврат из прерывания без установки трассировки
1пп	JMP		безусловный переход
20г	RTS		возврат из подпрограммы
240	NOP		нет операции
241	CLC	---0	очистка C
242	CLV	--0-	очистка V
244	CLZ	-0--	очистка Z
250	CLN	0---	очистка N
257	CCC	0000	очистка всех битов
261	SEC	---1	установка C
262	SEV	--1-	установка V
264	SEZ	-1--	установка Z
270	SEN	1---	установка N
277	SCC	1111	установка всех битов
3пп	SWAB	++00	перестановка байтов
400/777	BR		ветвление безусловное
1000/1377	BNE		ветвление, если не равно 0
1400/1777	BEQ		ветвление, если равно 0
2000/2377	BGE		ветвление, если больше или равно 0 со знаком
2400/2777	BLT		ветвление, если меньше 0 со знаком
3000/3377	BGT		ветвление, если больше 0 со знаком
3400/3777	BLE		ветвление, если меньше или равно 0 со знаком
4Rпп	JSR		обращение к подпрограмме
* 50пп	CLR <sup>~</sup>	0100	очистка
* 51пп	COM <sup>~</sup>	++01	инверсия
* 52пп	INC <sup>~</sup>	+++-	увеличение на 1
* 53пп	DEC <sup>~</sup>	+++-	уменьшение на 1
* 54пп	NEG <sup>~</sup>	++++	смена знака
* 55пп	ADC <sup>~</sup>	++++	прибавление бита C
* 56пп	SBC <sup>~</sup>	++++	вычитание бита C
* 57пп	TST <sup>~</sup>	++00	проверка и установка битов условий
* 60пп	ROR <sup>~</sup>	++++	вращение вправо
* 61пп	ROL <sup>~</sup>	++++	вращение влево
* 62пп	ASR <sup>~</sup>	++++	сдвиг вправо
* 63пп	ASL <sup>~</sup>	++++	сдвиг влево
64сс	MARK	----	возврат из подпрограммы с очисткой стека
67пп	SXT	++0-	расширение знака
*1иппп	MOV <sup>~</sup>	++0-	пересылка

\* Более подробные сведения о командах ассемблера, их кодах и соответствующих им битах условий можно найти в книге: *Осетинский Л. Г., Осетинский М. Г., Писаревский А. Н. ФОКАЛ — диалоговый язык для микро- и мини-ЭВМ. Л., Машиностроение, 1988. — Прим. ред.*



Код	Мнемоника	Условия	Назначение
*2иипп	СМР <sup>~</sup> И,П	+++	сравнение (П условно вычитается из И)
*3иипп	ВIT <sup>~</sup>	++0-	проверка бита
*4иипп	ВIC <sup>~</sup>	++0-	очистка бита
*5иипп	ВIS <sup>~</sup>	++0-	установка бита
6иипп	ADD	+++	сложение
74гпп	XOR	++0-	исключающее ИЛИ
77гсс	SOB R,M		цикл (R-1 и, если не 0, идти к метке M)
100000,100377	BPL		ветвление, если плюс
100400,100777	BMI		ветвление, если минус
101000,101377	BHI		ветвление, если больше 0 без знака
101400,101777	BLO		ветвление, если меньше или равно 0 без знака
102000,102377	BVC		ветвление, если переполнения нет
102400,102777	BVS		ветвление, если переполнение есть
103000,103377	BCC		ветвление, если переноса нет
	BHIS		ветвление, если больше или равно 0 без знака
103400,103777	BCS		ветвление, если перенос есть
	BLO		ветвление, если меньше 0 без знака
104000—104377	EMT		прерывания для системных программ
104400—104777	TRAP		командные прерывания
16иипп	SUB И,П	+++	вычитание П - И → П

Примечание. Условные обозначения: ии,И — источник, пп,П — приемник, г,R — регистр (РОН), сс — смещение, ~ — в байтовых командах на этом месте ставится символ «В», \* — в байтовых командах на этом месте ставится «1», + — бит условия изменяется в зависимости от результата операции, - — бит условия не меняется, 0 или 1 в графе «Условия» означает обнуление или установку в «1» соответствующего бита.

В кодах команд ведущие (незначащие) нули опущены.

### 3. Инструкции ассемблера (по алфавиту).

Команда	Назначение
ADC(В) А	прибавление бита С к А
ADD И,П	сложение (И + П → П)
ASL(В) А	сдвигка А влево (разряд 0 обнуляется)
ASR(В) А	сдвигка А вправо
BCC	ветвление, если C=0 (переноса нет)
BCS	ветвление, если C=1 (перенос есть)
BEQ	ветвление, если Z=1 (результат равен 0)
BGE	ветвление, если больше или равно 0 со знаком
BGT	ветвление, если больше 0 со знаком
BHI	ветвление, если больше 0 без знака
BHIS	ветвление, если больше или равно 0 без знака (аналог BCC)
ВIC(В) И,П	сброс битов П по установленным битам И
ВIS(В) И,П	установка битов П по установленным битам И
ВIT(В) #И,П	проверка битов П по числу И и установка битов условий
BLE	ветвление, если меньше или равно 0 со знаком
BLO	ветвление, если меньше 0 без знака (аналог BCS)
BLOS	ветвление, если меньше или равно 0 без знака
BLT	ветвление, если меньше 0 со знаком
BMI	ветвление, если N=1 («МИНУС»)
BNE	ветвление, если Z=0 (результат не равен 0)

Команда	Назначение
/BPL	ветвление, если N=0 («ПЛЮС»)
BPT	прерывание по вектору 14
BR	безусловный переход
BVC	ветвление, если V=0 (переполнения нет)
BVS	ветвление, если V=1 (переполнение есть)
CCC / SCC	установка всех битов условий в 0 / в 1
CLC / SEC	установка бита C в 0 / в 1
CLN / SEN	установка бита N в 0 / в 1
CLR(B) A	обнуление A
CLV / SEV	установка бита V в 0 / в 1
CLZ / SEZ	установка бита Z в 0 / в 1
CMP(B) И,П	сравнение И с П
COM(B) A	замена битов A на противоположные
DEC(B) A	уменьшение A на 1
EMT	прерывание по вектору 30
HALT	останов ЦП
INC(B) A	увеличение A на 1
IOT	прерывание ввода-вывода по вектору 20
JMP M	переход к M
JSR	переход к ПП (CALL M аналогично JSR PC,M)
MARK	пометка стека
MTPS	перенос байта из регистра в ССП (кроме бита 4)
MFPS	перенос байта из ССП в регистр
MOV(B) И,П	копирование И в П
NEG(B) A	смена знака A
NOP	нет операции
RESET	сброс шины в исходное состояние
ROL(B) A	вращение влево на один разряд через бит C
ROR(B) A	вращение вправо A на один разряд через бит C
RTI	возврат из прерывания с ловушкой трассировки
RTS	возврат из ПП (RET аналогично RTS PC)
RTT	возврат из прерывания без ловушки трассировки
SBC(B) A	вычитание бита C из A
SOB R,M	цикл (вычитание R-1 и, если не равно 0, переход к метке M с меньшим адресом)
SUB И,П	вычитание П - И → П
SWAB A	обмен байтов местами
SXT A	распространение знакового бита
TRAP	переход к ловушке по вектору 34
TST(B) A	проверка A
WAIT	ожидание прерывания
XOR R,Б	исключающее ИЛИ

#### 4. Системные регистры БК.

Регистр состояния клавиатуры (177660).

Используются 2 бита:

6 — разрешение/запрет прерывания от клавиатуры, доступен по записи и чтению («0» — разрешено, «1» — запрещено);

7 — флаг состояния клавиатуры, доступен только по чтению («1» — поступил новый код клавиши, «0» — код прочитан).

Если прерывание от клавиатуры разрешено (в бите 6 записан «0»), то при установке бита 7 в «1» (поступление нового кода клавиши) производится прерывание от клавиатуры, выдается звуковой сигнал и выполняются действия, соответствующие нажатой клавише.  
При чтении регистра данных бит 7 сбрасывается в 0.

#### Регистр данных клавиатуры (177662).

Доступен только по чтению. При нажатии клавиши ее 7-разрядный код записывается в биты 0—6, если предыдущий уже прочитан. Биты 7—15 не используются.

#### Регистр смещения (177664).

Используется для указания начала экранного ОЗУ и организации рулонного смещения экрана. Доступен по записи и чтению.

Биты 0—7 — байт, указывающий смещение адреса начала экрана по сравнению с исходным 40000, причем каждая единица его содержимого соответствует 100 байтам экранного ОЗУ (или одной TV строке). При начальной установке экрана данный байт равен 330, тогда экран начинается с адреса 40000. Изменение этого значения на 1 приводит к сдвигу изображения по вертикали на одну точечную (TV) строку.

Бит 9 — задание режима РП («1» — используется весь экран (исходная установка), «0» — при включенном режиме РП).

Биты 8, 9—15 не используются.

#### Регистр порта ввода/вывода (177714).

Доступен по записи и чтению. Служит для работы с внешними устройствами, подключенными к разъему УП.

При ЗАПИСИ по указанному адресу данные поступают в порт ВЫВОДА. Прочитать записанные в порт вывода данные невозможно, это следует учитывать при работе с регистром. При ЧТЕНИИ данные извлекаются из порта ВВОДА.

Следует учитывать, что порт ввода/вывода ИНВЕРТИРУЕТ сигнал, т. е. при записи в регистр слова 000000 на выходе порта будет 177777 и наоборот. Исходное состояние регистра — 000000, порта — 177777.

Аппаратная часть порта состоит из четырех микросхем K589IP12.

Соответствие разрядов порта контактам УП

Разряды порта		0	1	2	3	4	5	6	7
Контакты разъема УП	Порт ввода	B24	A24	B23	B17	B20	A20	B22	A23
	Порт вывода	A16	A13	B12	B10	B5	B7	B6	A7
Разряды порта		8	9	10	11	12	13	14	15
Контакты разъема УП	Порт ввода	B31	A31	B32	A32	B30	A29	B29	A30
	Порт вывода	A28	B28	A27	B27	A26	B26	A25	B25

Нагрузочная способность порта вывода ограничена возможностями микросхем K589IP12.

Дополнительно на УП выведены сигналы:

**СБРОС (выходной)** — для начальной установки внешнего устройства командой RESET (контакт A1), нагрузочная способность ограничена (сигнал снимается с вывода INIT ЦП);

**IRQ2 (входной)** — требование прерывания от внешнего устройства (контакт B1), вектор прерывания 100;

общий — контакты A11, B11, A18, B18, A19, B19;

+5 В — A8, B8, A9, B9.

Устройства, подключаемые к УП, должны иметь уровни сигналов: «1» =  $0v < U_1 < 0.5v$ , «0» =  $2.4v < U_0 < 5.25v$ .

#### Регистр управления системными внешними устройствами (177716).

Старший байт (биты 8—15) служит для указания адреса начальной загрузки (всегда равен 100000), с которого запускается ЦП при включении питания. Доступен только по чтению.

Младший байт при этом принимается равным 0.  
Биты 0—3 служат для отображения (задания) режимов работы ЦП и содержат флаги NZVC\*.

Доступны только по чтению.

Биты 4—7 используются для управления системными внешними устройствами и имеют выход на внутренний 4-разрядный порт ввода/вывода (аналогично порту УП), причем запись производится только в порт вывода, а чтение — только из порта ввода.

\* Данный факт редакцией не проверялся. — Прим. ред.

**Порт ВЫВОДА** (биты доступны только по ЗАПИСИ):

- 4 — для передачи данных на линию, исходное состояние = 1;
- 5 — для передачи данных на магнитофон или сигнала готовности на линию, исходное состояние = 0;
- 6 — для передачи данных на магнитофон и выдачи звуковых сигналов, исходное состояние = 0;
- 7 — для управления двигателем магнитофона («1» — СТОП, «0» — ПУСК).

**Порт ВВОДА** (биты доступны только по ЧТЕНИЮ):

- 4 — для чтения данных с линии;
- 5 — для чтения данных с магнитофона;
- 6 — для индикации нажатия клавиш («0» — нажата, «1» — все клавиши отпущены), для реализации режима ПОВТОР;
- 7 — для чтения сигнала готовности с линии.

### 5. Системная область БК (служебные ячейки).

**Векторы прерываний.** Первая из ячеек содержит адрес подпрограммы обработки прерывания, вторая — значение ССП на время обработки:

- 4, 6 — прерывание по СТОП, HALT и по ошибке передачи данных;
- 10, 12 — прерывание по резервной команде;
- 14, 16 — прерывание по T-биту (трассировка);
- 20, 22 — прерывание по команде IOT;
- 24, 26 — прерывание по сбою питания;
- 30, 32 — прерывание по EMT;
- 34, 36 — прерывание по команде TRAP.

**Флаги состояния дисплея.** Используются отдельные байты ячеек, 377 означает включенный режим, 0 — выключенный:

- 40 — 32 символа в строке;
- 41 — инверсия экрана;
- 42 — расширенная память (РП);
- 43 — «ЛАТ»/«РУС» («ЛАТ» — 0, «ЛАТ ЗАГЛ» — 40, «РУС» — 200, «РУС ЗАГЛ» — 240, блокировка клавиатуры — 377);
- 44 — подчеркивание;
- 45 — инверсия символов;
- 46 — ИНД СУ;
- 47 — БЛОК РЕД;
- 50 — ГРАФ;
- 51 — ЗАП;
- 52 — СТИР;
- 53 — 32 символа в служебной строке;
- 54 — подчеркивание в служебной строке;
- 55 — инверсия символов в служебной строке;
- 56 — курсор.

**Ячейки драйвера клавиатуры:**

- 60, 62 — прерывание от клавиатуры (кроме «СТОП») по нижнему регистру;
- 64—76 — не используются;
- 100, 102 — прерывание от таймера пользователя (сигнал IRQ2);
- 104 (байт) — буфер клавиатуры, код последнего введенного символа;
- 105 (байт) — признак записи кода в буфер по EMT 6 (0 — символ не введен);

- 106 — слово задержки при ожидании отпущения клавиши «ПОВТ»: величина паузы, формируемой при опросе состояния регистра признаков клавиатуры, когда содержимое ячейки 110 не равно 0 (буфер константы повтора, константа повтора T-паузы), равно 20000 в символьном режиме, 2000 в графическом;
- 110 (байт) — флаг нажатия клавиши «ПОВТ»: признак опроса регистра состояния клавиатуры и ввода символа из регистра данных клавиатуры, если клавиша нажата. После опроса значение в этой ячейке ОБНУЛЯЕТСЯ (признак повтора кода);
- 111 (байт) — счетчик табуляции: флаг и количество пробелов, выдаваемых EMT 6 (после ввода каждого из пробелов их количество уменьшается на 1);
- 112—120 (8 байт) — положение меток табуляторов (побитно);
- 122 — количество выдаваемых символов строки ключа (при выдаче очередного символа уменьшается на 1);
- 124 — адрес очередного символа строки ключа;
- 126—150 — адреса упакованных строк ключей (первый байт — длина строки, далее сами символы).

**Ячейки драйвера дисплея:**

- 152 (байт) — флаг нормализации положения курсора, вызывает содержимое ячейки 156 к 160;
- 153 (байт) — тип операции в режиме ГРАФ (0 — СТИР, 1 — ЗАП);
- 154 (байт) — маска позиции графического курсора на горизонтальной оси в пределах байта (вертикальная координата определяется по ячейке 174) и маска точек, рисуемых курсором в графическом режиме;
- 155 (байт) — начальная маска позиции графического курсора (графической точки);

- 156 — координата символьного курсора (мл. байт = X (0...77), ст. байт = Y), в режиме 32 символа в строке — позиция курсора в строке (целая часть от деления его точной координаты на 2), в графическом режиме — номер «рабочего» знакомиеста в пределах символьного курсора;
- 160 — абсолютный адрес верхнего левого угла изображения курсора в ОЗУ экрана, отсюда начинается вывод очередного символа;
- 162 — константа смещения одного алфавитно-цифрового символа относительно другого (ширина символа, интервал между символами);
- 164 — величина ОЗУ экрана в символах;
- 166 — адрес графического курсора относительно начала информационной части ОЗУ экрана (в байтах);
- 170 — абсолютный адрес графического курсора в ОЗУ экрана;
- 172 — длина генерируемого графического вектора;
- 174 — счетчик ТВ-строк в графическом режиме (вертикальная координата графического курсора);
- 176 — координата X последней выведенной по ЕМТ 30, ЕМТ 32 графической точки;
- 200 — то же для координаты Y;
- 202 — начальный адрес ОЗУ экрана (40000 или 70000), адрес начала поля служебной строки;
- 204 — ширина поля служебной строки (при равенстве 0 служебная строка не отображается);
- 206 — размер области видеоОЗУ (в байтах);
- 210 — объем рабочей области экрана без служебной строки;
- 212 — код цвета фона на экране;
- 214 — код цвета символов;
- 216 — код цвета фона служебной строки;
- 220 — код цвета символов в служебной строке;
- 222 — счетчик кодов, содержит количество выводимых символов при однократном нажатии клавиши;
- 224 — условный номер текста, выводимого в служебную строку в графическом режиме в позиции 20—23 (десятичн.), считая с правого края служебной строки: 0 — пусто, 2 — ГРАФ, 3 — ЗАП, 4 — СТИР. При нажатии любой клавиши происходит прерывание от клавиатуры, а затем вызывается подпрограмма подзвучки по адресу 102032, которая в зависимости от флага 224 вызывает подпрог-

рамму нормализации, устанавливающую индикаторы в служебной строке.

#### Ячейки драйвера магнитофона:

- 226—252 — не используются, зарезервированы для работы с локальной сетью;
- 250 — длина блока параметров магнитофона;
- 252 (байт) — код «РУС»/«ЛАТ» при передаче по ТАГ-каналу;
- 253 (байт) — то же при приеме;
- 254 — константа скорости передачи по ТАГ-каналу;
- 256 — зарезервирована для сохранения копии содержимого порта вывода;
- 260 — адрес подпрограммы, выполняемой при прерывании по вектору 60 (от клавиатуры). Если дополнительная обработка не нужна, то этот адрес должен быть равен 0. Адрес подпрограммы, выполняемой при вызове ЕМТ 60;
- 262 — определяет передаваемый в подпрограмму код клавиши «ВВОД» (0 — 12, иначе 15), признак наличия языка Focal;
- 264 — начальный адрес программы, загруженной с магнитофона или ТАГ-канала;
- 266 — длина загруженной программы;
- 270—272 — не используются;
- 274, 276 — прерывание по нижнему регистру («AP2») клавиатуры;
- 300 (байт) — индикатор полярности импульсов в файле, читаемом с магнитофона (0 — прямой сигнал, иначе инверсный);
- 301 (байт) — байт ответа драйвера магнитофона;
- 302 — флаг фиктивного чтения (0 — контрольная сумма подсчитывается, иначе ее подсчет не производится);
- 304 — шаг смещения байтов при чтении;
- 306 — адрес блока параметров драйвера магнитофона;
- 310 — значение SP после входа в драйвер магнитофона (буфер стека);
- 312 — значение контрольной суммы массива, вычисленной перед его записью;
- 314 — средняя длина сигнала, определенная, по настроечной последовательности;
- 316 — не используется;
- 320 (байт) — команда драйверу магнитофона (0 — останов мотора, 1 — пуск мотора, 2 — запись, 3 — чтение, 4 — фиктивное чтение);
- 321 (байт) — ответ драйвера магнитофона (0 — ошибок нет, 1 — имя найденного файла не совпадает с заданным, 2 — ошибка контрольной суммы, 4 — останов по команде оператора);

- 322 — адрес записываемого массива или адрес загрузки;
- 324 — длина читаемого/записываемого массива;
- 326 (16 байт) — имя файла;

#### 6. Полезные подпрограммы монитора БК.

- 102052 — звук с возрастающей частотой: R2 — длительность (не более #300), R3 — частота.
- 100472 — ввод восьмеричного числа (результат в R5).
- 100460 — печать упакованной строки (адрес в R3).
- 100644 — печать текста «ОШИБКА».
- 102566 — прокрутка экрана на одну строку вниз.

#### 7. Прерывания EMT

Система прерываний EMT предназначена для вызова служебных подпрограмм, имеющих в мониторе БК. Программа, обрабатывающая прерывания (EMT-диспетчер), использует младший байт EMT как номер подпрограммы в системной таблице.

##### Клавиатура:

- EMT 4 — инициализация векторов прерывания клавиатуры. Содержимое R0 не сохраняется;
- EMT 6 — чтение кода символа с клавиатуры. Выход: код нажатой клавиши в младшем байте R0;
- EMT 10 — чтение строки с клавиатуры. Выход: R1 — адрес буфера, куда вводить строку, R2 — младший байт — максимальная длина строки, старший — символ-ограничитель.
- После записи в память очередного введенного символа содержимое R1 увели-

- 346 — адрес текущего файла;
- 350 — длина текущего файла;
- 352 (16 байт) — имя текущего файла;
- 366 — адрес блока паритета.

- 103600 — сдвигка в строке.
- 103700 — раздвижка в строке.
- 103760 — удаление одной строки.
- 104020 — вставка пустой строки.
- 105232 — очистка строки справа от курсора.
- 110346 и 110362 — сохранение/восстановление регистров R0—R3 в стеке (вызов по команде JSR R4, <адрес>).
- 110376 — очистка экрана.

##### Дисплей:

- EMT 14 — установка исходных режимов отображения на экране, установка всех векторов прерывания, очистка экрана. R0—R4 не сохраняются. Стск не изменяется. Сброс рабочих ячеек драйверов в исходное состояние;
- EMT 16 — вывод символа по коду в младшем байте R0;
- EMT 20 — вывод строки символов. Вход: R1 — адрес строки, R2 — младший байт — длина строки, старший — символ-ограничитель. Если вывод заканчивается по символу-ограничителю, то его код передается последним. После окончания вывода в R1 будет записан адрес байта после последнего выведенного, в R2 — разность между заданной и реальной длинами строки;
- EMT 22 — вывод символа в служебную строку. Вход: R0 — код символа (0 —

чивается на 1, а после окончания ввода в R2 будет записана разность между заданной длиной и реальной. Ввод строки прекращается, если длина строки станет равна заданной или введен символ, указанный в старшем байте R2. Если ввод строки окончен по символу-ограничителю, последний записывается в конце строки.

- EMT 12 — установка ключей K1—K10 клавиатуры. Вход: R0 — номер ключа от 1 до 10, R1 — адрес текста ключа, заканчивающегося нулевым байтом.

- очистка строки), R1 — номер позиции в служебной строке, начиная с 0. Следует помнить, что часть строки справа используется для вывода служебных сообщений;
- EMT 24 — установка курсора по координатам X=R1, Y=R2;
- EMT 26 — получение координат курсора: R1=X, R2=Y;
- EMT 30 — рисование точки по координатам X=R1, Y=R2. R0=1 — запись точки, R0=0 — стирание;
- EMT 32 — рисование вектора, входные данные — как для EMT 30;
- EMT 34 — получение в R0 слова состояния дисплея, в котором каждый разряд есть индикатор включения соответствующего режима («1» — включено; «0» — отключено).

## Формат слова состояния дисплея

Разряд	Режим
0	32 символа в строке
1	инверсия экрана
2	РП
3	РУС
4	подчеркивание символа
5	инверсия символа
6	индикация СУ
7	блокировка редактирования

Разряд	Режим
8	режим текстовой графики ГРАФ
9	запись в режиме ГРАФ
10	стирание в режиме ГРАФ
11	32 символа в служебной строке
12	подчеркивание символа в служебной строке
13	инверсия символа в служебной строке
14	гашение курсора
15	не используется

## Магнитофон:

- EMT 36 — работа с магнитофоном. В R1 задается адрес блока параметров (обычно с адреса 320).

## Формат блока параметров для работы с магнитофоном

Номер байта	Адрес (стандартный)	Содержание
0	320	команда (0 — останов двигателя, 1 — пуск, 2 — запись массива на ленту, 3 — чтение массива с ленты, 4 — фиктивное чтение)
1	321	код завершения операции (0 — без ошибок, 1 — неправильное имя файла, 2 — ошибка контрольной суммы, 3 — останов по клавише СТОП)
2	322	адрес массива в ОЗУ
4	324	длина массива на запись
6	326	имя массива (16 байт)
22	346	адрес обнаруженного на ленте массива
24	350	длина обнаруженного на ленте массива
26	352	имя обнаруженного на ленте массива (16 байт)

Если при загрузке задать адрес равным 0, то файл будет загружаться с адреса, указанного на ленте. После успешной загрузки файла адрес его загрузки и длина записываются в ячейки 264 и 266 соответственно.

## Компьютеры, периферия и новейшие программы по почте:

- БК-0010(01), БК-0011(М), ZX-SPECTRUM 48, 128, 256, 512, УКНЦ (МС-0511), IBM-совместимые («ПОИСК»...) до AT 486.
- Контроллеры, дисководы, магнитофоны, стримеры, блоки питания, мониторы, принтеры, модемы, модуляторы, джойстики, диски CD-ROM.
- Измерительная аппаратура: тестеры, осциллографы и др.
- Можем выслать телефон-секретарь с определителем номера.

**НА ВСЮ АППАРАТУРУ САМЫЕ НИЗКИЕ ЦЕНЫ.**

**Наш адрес: 189510, Санкт-Петербург, Ломоносов, а/я 649, «КИ-ЧПМ».**

*Для ответа присылайте подписанный конверт с марками.*



Данное руководство рассчитано на пользователей микроЭВМ «Электроника МС 0513» (БК-0011(М)), программирующихся на ассемблере. В руководстве описаны состав и функции базовой операционной системы ЭВМ (БОС) и способы обращения к ним из пользовательских программ, а также команды монитора БОС, позволяющие работать с ЭВМ в интерактивном режиме, вводить, отлаживать и запускать программы в кодах, считывать и записывать файлы на магнитную ленту и т. д.

# Руководство системного программиста БК-0011(М)

## Базовая операционная система

Краткое описание микроЭВМ БК-0011(М)

### Архитектура ЭВМ

Компьютер БК-0011(М) построен на основе микропроцессорного комплекта серии К1801 и включает в себя:

- однокристалльный 16-разрядный микропроцессор К1801ВМ1;
- оперативное запоминающее устройство (ОЗУ) емкостью 128 кб со страничной организацией, выполненное на микросхемах К565РУ5;
- системное ПЗУ емкостью 16 кб;
- контроллер клавиатуры на базе БИС К1801ВП1-014;
- контроллер ОЗУ и телевизионного графического дисплея на базе БИС К1801ВП1-037;
- контроллер накопителя на бытовом кассетном магнитофоне;
- 16-разрядный параллельный программируемый интерфейс.

Отдельные функциональные узлы ЭВМ соединены с помощью межмодульного параллельного интерфейса МПИ ОСТ 11.305.903-80.

Обслуживание перечисленных устройств, а также внешних дополнительных блоков контроллеров локальной сети и НГМД осуществляется соответствующими программами — драйверами, расположенными в системном ПЗУ ЭВМ.

Распределение адресного пространства ЭВМ. Виртуальное адресное пространство ЭВМ определяется возможностями процессора К1801ВМ1 и составляет 64 кб. Распределение адресного пространства показано на рис.1.

Переключение страниц ОЗУ/ПЗУ осуществляется программно с использованием системного регистра ЭВМ.

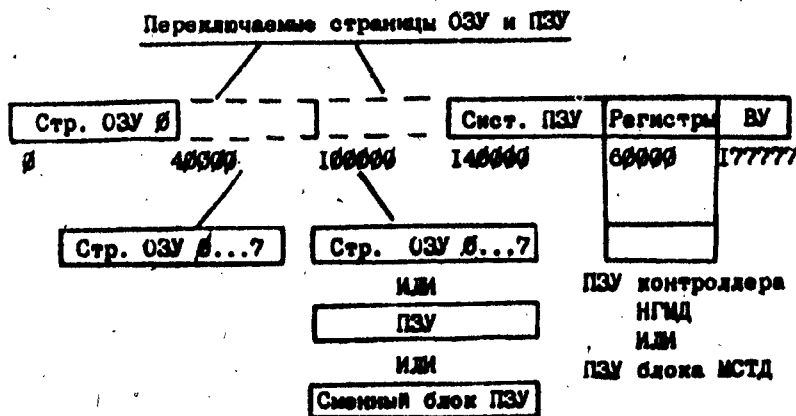


Рис. 1. Распределение адресного пространства ЭВМ



Следует обратить внимание, что каждая страница ОЗУ может быть одновременно подключена к двум окнам виртуального адресного пространства, а статическая страница (адреса 0—40000) даже к трем. Переключение страниц из пользовательской программы может осуществляться только с помощью запросов базовой операционной системы.

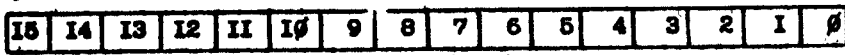
Две страницы ОЗУ являются буферами экрана графического дисплея. Один из буферов постоянно отображается на экране телевизионного приемника. Управление отображением того или иного буфера экрана также возложено на базовую операционную систему, причем имеется возможность «скрытого» вывода на экран, т. е. формирование изображения из программы пользователя в одном буфере, в то время как на экран выводится другой.

Одна из переключаемых страниц ОЗУ является служебной страницей базовой операционной системы ЭВМ и недоступна пользователям для подключения. Однако часть ее, не используемая базовой операционной системой, доступна для чтения-записи с помощью запросов драйвера ОЗУ/ПЗУ.

### Регистры внешних устройств ЭВМ

Работа ЭВМ с периферийными (внешними) устройствами производится через регистры внешних устройств, для которых выделен диапазон адресного пространства 170000—177777. Ниже описаны регистры стандартных периферийных устройств. Остальные адреса регистров зарезервированы для подключения дополнительной периферии.

Регистр состояния клавиатуры (адрес 177660).



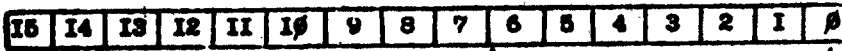
↑                    ↑  
готовность        маскирование прерываний

- Разряд 7: «1» означает, что в регистре данных клавиатуры находится код нажатой клавиши. Устанавливается при нажатии на клавишу, сбрасывается при чтении регистра данных клавиатуры. При нулевом состоянии 6-го разряда установка в «1» вызывает прерывание. Доступен только по чтению.
- Разряд 6: «0» разрешает выработку сигнала требования прерывания по единичному значению разряда 7.

По сигналу сброса магистрали разряд 6 устанавливается в «1», разряд 7 обнуляется. Остальные разряды регистра не используются, читаются как «0».

Примечание. Смотрите также информацию о бите 6 системного регистра ЭВМ.

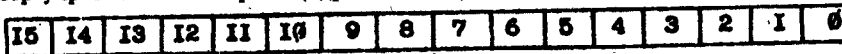
Регистр данных клавиатуры (адрес 177662).



} код символа

Разряды 0—6 содержат код нажатой клавиши. Разряды 7—15 не используются, читаются как «0». Регистр доступен только по чтению. Данные сохраняются в регистре до нажатия следующей клавиши клавиатуры. При установленном в «1» седьмом разряде регистра состояния запись в регистр данных блокируется и нажатия клавиш игнорируются.

Регистр управления палитрами (адрес 177662).



} код палитры

↑ управление таймером

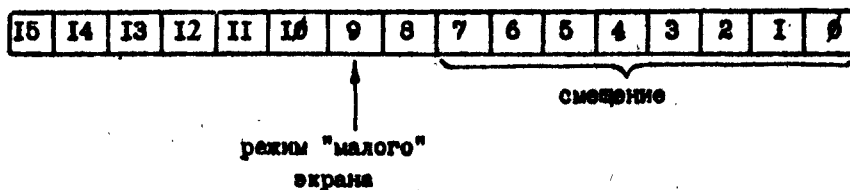
↑ управление буфером экрана

Регистр доступен только по записи.

- Разряд 15: управляет отображением буферов экрана. При значении «0» отображается буфер 0 (страница 5), при «1» — буфер 1 (страница 6).
- Разряд 14: управляет включением системного таймера. При значении «0» таймер включен, при «1» — выключен.
- Разряды 11—8: код палитры, которая задает кодировку цветов на экране в режиме 256 точек на строку.

Текущее состояние регистра можно определить с помощью запросов БОС.

**Регистр смещения** (адрес 177664). Предназначен для управления «рулонным» сдвигом информации на экране по вертикали путем задания адреса в буфере экрана, с которого должна начинаться регенерация изображения.



- Разряд 9: при записи «1» на экране отображается весь буфер экрана (256 телевизионных строк). При нулевом значении в верхней части раstra отображается 1/4 часть (старшие адреса) экранного ОЗУ, нижняя часть экрана не отображается. Данный режим не используется базовой операционной системой.

• Разряды 7—0: адрес строки экранного ОЗУ, с которой начинается отображение экрана. ОЗУ буфера экрана считается «замкнутым в кольцо». Экран состоит из 256 строк по 64 байта в каждой. Строки начинаются с адресов, кратных 1008. Первый байт каждой строки расположен в ее левом конце, байты отображаются начиная с младшего бита. Смещение задается в строках, т. е. изменение кода смещения на 1 приводит к кольцевому сдвигу информации на экране на одну строку раstra. Исходному состоянию (когда первый байт буфера экрана отображается в самой верхней строке раstra) соответствует значение смещения 3308. Увеличение значения соответствует сдвигу информации вверх, уменьшение — вниз.

Остальные разряды не используются. Регистр доступен по записи и чтению.

**Регистр параллельного программируемого интерфейса** (адрес 177714). Предназначен для записи информации в выходной порт и чтения информации (16 разрядов) со входного порта. Возможность прочесть состояние выходного порта отсутствует. Фактически в ЭВМ реализованы два независимых регистра, имеющие один и тот же адрес:

- доступный только по записи регистр выходного порта,
- доступный только по чтению регистр входного порта.

**Системный регистр** (адрес 177716). Используется при начальном запуске процессора, для переключения страниц ОЗУ/ПЗУ, управления внешними устройствами ЭВМ.

Фактически по указанному адресу расположены три регистра:

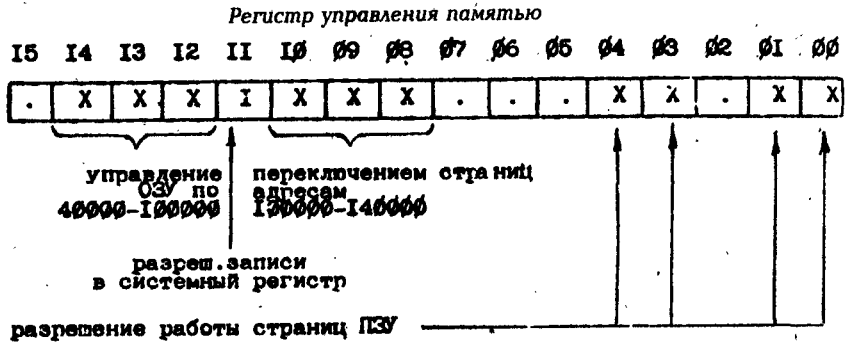
- регистр, доступный только по чтению;
- два регистра, доступных только по записи (регистр управления памятью и регистр управления магнитофоном).



#### Регистр чтения

- Разряды 15—8: определяют адрес начального пуска ЭВМ по включении питания (140000).
- Разряд 7: должен быть равен «1».
- Разряд 6: устанавливается в «0» при нажатии любой клавиши ЭВМ и в «1» — после отпущения клавиши.
- Разряд 5: разряд данных контроллера магнитофона.
- Разряд 2: устанавливается в «1» при любой операции записи в системный регистр и сбрасывается в «0» по окончании операции чтения системного регистра. Об использовании этого разряда см. в разделе, посвященном особенностям выполнения некоторых команд ЭВМ. Остальные разряды не используются и читаются как «0».

ЕУ 20 32 50



- Разряды 14—12: определяют номер страницы ОЗУ, подключаемой на адреса 40000—100000 (см. ниже).
- Разряд 11: должен всегда содержать «1».
- Разряды 10—8: определяют номер страницы ОЗУ, подключаемой на адреса 100000—140000.
- Разряд 4: разрешает подключение к адресному пространству 100000—140000 страницы ПЗУ №3, отключая с соответствующих адресов страницу ОЗУ.
- Разряд 3: разрешает подключение к адресному пространству 100000—140000 страницы ПЗУ №2, отключая страницу ОЗУ.
- Разряд 1: разрешает подключение к адресному пространству 100000—140000 страницы ПЗУ №1, отключая страницу ОЗУ.
- Разряд 0: разрешает подключение к адресному пространству 100000—140000 страницы ПЗУ №0, отключая страницу ОЗУ.

Кодировка переключаемых страниц ОЗУ

Бит			Номер страницы
14 (10)	13 (9)	12 (8)	
1	1	0	страница 0 (постоянно отображена 0..40000)
0	0	0	страница 1
0	1	0	страница 2
0	1	1	страница 3
1	0	0	страница 4
0	0	1	страница 5 (буфер экрана 0)
1	1	1	страница 6 (буфер экрана 1)
1	0	1	страница 7 (системная страница)

**Примечание.** Запись в регистр возможна только при установленном 11-м бите и только по командам, работающим с регистром как с целым словом.

Страницы 0, 1 и 2 ПЗУ зарезервированы. Страница 0 и адресное пространство 100000—117777 страницы 2 предназначены для ПЗУ языка программирования BASIC, а в адресах 120000—137777 страницы 1 расположены программы базовой операционной системы. Адреса 100000—117777 страницы 1 зарезервированы для расширения функций БОС. Адреса 120000—137777 страницы 2 используется ПЗУ драйвера локальной сети. Страница 3 ПЗУ предназначена для подключения внешних блоков ПЗУ к магистрали ЭВМ.



Регистр доступен только по записи при значении разряда 11, равном «0».

**Регистры дополнительных внешних устройств.** Регистры блоков контроллера локальной сети (КАС), ИРПС, КНГМА, устройства управления цветовой палитрой описаны в соответствующей технической документации.

### *Особенности выполнения команд*

Система команд микроЭВМ БК-0011(М) соответствует принятой в ЭВМ «Электроника-60» (стандарт DEC) и определяется процессором 1801ВМ1. Однако, в связи с рядом особенностей архитектуры, существует важное отличие от ЭВМ семейств СМ, «Электроника-60», «Электроника-85» и ДВК. Команда HALT (код 000000) вызывает прерывание выполнения программы и переход по вектору прерывания 4, при этом устанавливается бит — признак записи в системный регистр. БОС различает обычное прерывание по вектору 4 и останов. Программа пользователя имеет возможность перехватить только обычное прерывание по вектору 4. При останове произойдет переход в монитор БОС с выводом на экран содержимого счетчика команд ЭВМ.

Следует заметить, что правильная обработка клавиши «СТОП» возможна только при корректном значении указателя стека. Кроме этого, из-за особенностей аппаратуры иногда может выдаваться неправильное значение счетчика команд. В некоторых случаях (нажатие на «СТОП» во время выполнения запроса БОС) невозможно продолжение программы по команде «Р». Исходя из всего сказанного, клавишей «СТОП» следует пользоваться очень осторожно.

### *Векторы прерываний и зарезервированные ячейки БОС*

Обработка прерываний в ЭВМ производится по векторам, размещенным в области 0—376. Некоторые ячейки этой области используются базовой операционной системой для внутренних целей.

- 4, 6: вектор прерывания по ошибке обращения к каналу, команде HALT и клавише «СТОП».
- 10, 12: вектор прерывания по несуществующей команде (в БОС не используется).
- 14, 16: вектор прерывания по команде ВРТ и Т-биту (используется при нажатии клавиши «ШАГ»).
- 20, 22: вектор прерывания по команде IOT (не используется).
- 24, 26: вектор прерывания по аварии питания (не используется).
- 30, 32: вектор прерывания по команде EMT.
- 34, 36: вектор прерывания по команде TRAP (не используется).
- 40: адрес пуска по команде монитора «G» (используется при операциях с магнитофоном).
- 52 (младший байт): код возврата системных вызовов.
- 60, 62: вектор прерывания клавиатуры (алфавитно-цифровые и управляющие коды).
- 100, 102: вектор прерывания таймера.
- 110, 112: программный вектор клавиатуры (адрес пользовательской программы завершения).
- 114: копия по записи системного регистра (управление памятью).
- 116: копия по записи системного регистра (управление магнитофоном).
- 150—156, 164, 250, 252: используются для внутренних нужд БОС.
- 166: используется в качестве псевдовектора EMT для перехвата пользовательской программой команд EMT, не обработанных БОС.
- 270, 272: вектор прерывания параллельного порта.
- 274, 276: вектор прерывания клавиатуры (регистр AP2 и функциональные клавиши).

### **Функции базовой операционной системы**

Базовая операционная система включает в себя следующие программные модули:

- программа начального пуска;
- клавиатурный монитор;
- таблица системных вызовов;
- обработчик команд emt;
- драйвер ОЗУ/ПЗУ;
- драйвер клавиатуры;
- драйвер экрана;
- драйвер кассетного магнитофона;
- драйвер принтера;
- драйвер накопителя на ГМД.

### *Программа начального пуска*

Программа начального пуска предназначена для инициализации базовой операционной системы при включении ЭВМ, определения конфигурации компьютера и запуска других программ.

При включении ЭВМ программа определяет наличие дополнительных блоков ПЗУ, устанавливая соответствующие флаги. Если в момент пуска нажата и удерживается какая-либо клавиша на клавиатуре, программа начального пуска передает управление клавиатурному монитору. В противном случае производятся следующие действия: если к ЭВМ подключен

контроллер НГМД, программа начального пуска вызывает дисковый загрузчик, расположенный в контроллере НГМД по адресу 160000. Дисковый загрузчик поочередно пытается загрузить операционную систему (ОС) с приводов 0 и 1 (для БК-0011 — также 2 и 3). Если все попытки оказались неудачными («карман» дисковода пуст, дискета не системная или запарчена и т. д.), загрузчик возвращает управление программе начального пуска. Затем программа поочередно пытается запустить блоки ПЗУ, начиная с блока с наибольшим номером. Первое слово программы в ПЗУ должно быть командой NOP (код 240). Вызов ПЗУ производится командой JSR PC, поэтому программа в ПЗУ может вернуть управление программе начального пуска командой RTS PC для продолжения загрузки. Если программ в ПЗУ нет или они вернули управление, производится переход в клавиатурный монитор.

### Клавиатурный монитор

Клавиатурный монитор является программой, работающей в диалоге с пользователем ЭВМ и позволяющей ему загружать программы с магнитной ленты, запускать их на выполнение, просматривать и корректировать содержимое ячеек ОЗУ, регистров процессора и внешних устройств, записывать участки ОЗУ на магнитную ленту, загружать операционную систему с диска и т. д.

Все эти функции выполняются пользователем при помощи команд монитора. Язык команд монитора максимально приближен ко входному языку режима пульта терминала ЭВМ «Электроника-60» и ДВК.

**Команды монитора.** Признаком готовности монитора к приему очередной команды оператора является символ «@» в первой позиции строки экрана. Числа вводятся в восьмеричной системе счисления. Набирать незначащие нули необязательно.

- XXXXXX/ — открыть ячейку с восьмеричным адресом XXXXXX.

По этой команде на экран выводится восьмеричное содержимое ячейки ОЗУ, ПЗУ или регистра внешнего устройства с адресом XXXXXX. Если такой регистр отсутствует или страница ОЗУ не подключена к заданному адресу, выводится сообщение об ошибке — символ «?» и команда игнорируется.

- XXXXXX\ — открыть байт с адресом XXXXXX.

Содержимое байта или слова можно изменить, введя восьмеричное число (новое содержимое) и подав одну из команд: «ВВОД» (клавиша «↵»), «открыть следующую ячейку», «открыть предыдущую ячейку», «открыть косвенно», «открыть по смещению».

- RX/ — открыть регистр X (X=0..7 или «S»). Открывает регистр общего назначения процессора (R0—R7) или регистр состояния процессора (PS).

- Команды «/» и «\» без аргумента — открывают «текущую», т. е. последнюю открывавшуюся ранее ячейку.

- «ВВОД» («↵») — закрыть слово, байт или регистр. Если пользователем было введено новое содержимое, оно будет записано.

- «Стрелка вниз» («↓») — закрыть текущее слово, байт или регистр (если надо, модифицировав содержимое как по команде «ВВОД») и открыть следующее слово, байт или регистр (при этом печатается адрес открываемой ячейки). После ячейки с адресом 177777 открывается ячейка 000000, после R7 — R0, после PS данная команда работает как «ВВОД».

- «Стрелка вверх» («↑») — закрыть (с модификацией) текущее и открыть предыдущее слово, байт или регистр.

- «Стрелка влево» («←») — закрыть (с модификацией) текущую ячейку, вычислить адрес по формуле: Адрес = <Адрес текущей ячейки> + <ее новое содержимое>+2 и открыть как слово ячейку с соответствующим адресом.

Эта команда используется для открытия аргумента команды ЭВМ, использующей метод адресации 67.

- «Стрелка вправо» («→») — закрыть (с модификацией) текущую ячейку, вычислить адрес по формуле: Адрес = <Адрес ячейки> + <содержимое младшего байта>\*2+2 и открыть как слово ячейку с соответствующим адресом.

Содержимое байта рассматривается как число со знаком. Команда используется для открытия ячейки, на которую выполняется переход по команде типа «BR».

- @ — закрыть (с модификацией) текущую ячейку и открыть ячейку по адресу, равному новому содержимому закрытой ячейки.

- N;MC — отобразить страницу с номером N с адреса, задаваемого аргументом M. Страницы с восьмеричными номерами 10, 11, 12 13 относятся к страницам ПЗУ 0, 1, 2 и 3 соответственно. Значение M, равное нулю, означает адреса 40000—100000, «1» — адреса 100000—140000.

- XXXB — загрузить операционную систему с НГМД с номером XXX и запустить ее. Если номер не задан, производится загрузка с автоматическим поиском системного диска.

- XXXXXXL — загрузить с адреса XXXXXX двоичный файл с магнитной ленты. Если адрес не задан, загрузка осуществляется с адреса, указанного в заголовке файла.

Базовая операционная система поддерживает формат записи на магнитной ленте, совместимый с ЭВМ БК-0010(.01). После подачи этой команды на экран выдается вопрос «Имя файла?». Оператор должен ввести имя файла на магнитной ленте длиной до 16 символов. Если необходимо просмотреть каталог ленты, нужно ввести вместо имени файла символ «?». Если нажать клавишу «ВВОД» без ввода имени, будет загружен первый встретившийся файл.

- «Область»: U — вывести в файл информацию из ОЗУ. «Область» — конструкция вида XXXXXX—YYYYYY, где XXXXXX и YYYYYY — начальный и конечный (включительно) адреса записываемой области ОЗУ, или вида XXXXXX+YYYYYY, где XXXXXX — начальный адрес записываемой области ОЗУ, а YYYYYY — ее длина в байтах. После выдачи команды задается вопрос «Имя?», при этом оператор должен ввести имя файла, который он хочет записать на ленту.
- XXXXXXM — включить/выключить двигатель магнитофона (используется при перемотке). Если XXXXXX не равно 0, двигатель включается, иначе выключается. Отсутствие аргумента соответствует нулевому.
- XXXXXXG — запустить программу с адреса XXXXXX. При отсутствии аргумента запуск производится со стартового адреса программы, который должен содержаться в ячейке USERPC (адрес 000040).
- P — продолжить выполнение программы с адреса, на котором оно было прервано командой HALT или клавишей «СТОП».
- «ШАГ» — выполнить одну команду программы и вернуться в монитор.
- «ЗАБОЙ» («←») — аннулировать последнюю введенную цифру. Другие символы не стираются. Команда не выполняется, если вводимое число равно нулю.
- «СУ/Е» — переход в непосредственный режим. Используется для настройки драйвера экрана. В него передается символ ESC (код 33), после чего монитор перенаправляет драйверу экрана все вводимые символы. Монитор остается в непосредственном режиме до ввода комбинации «СУ/Е». ESC-последовательности, воспринимаемые драйвером экрана, указаны в приложении 2.

#### Примеры:

- СУ/Е 0 СУ/@ — установка цветного режима;  
 СУ/Е ; 2 1 СУ/@ — установка цвета фона 2 и цвета символов 1.

#### Таблица системных вызовов

Таблица системных вызовов располагается в ПЗУ начиная с адреса 140010 и является массивом адресов программ-обработчиков системных вызовов. Вызов функций БОС производится командой JSR PC,@ADDR, где ADDR — адрес одного из слов таблицы. Например: JSR PC,@140010.

Для передачи параметров используется регистр процессора R0, а в некоторых случаях и R1. При возврате из БОС содержимое регистров процессора сохраняется, за исключением особо оговоренных случаев. Содержимое таблицы рассматривается ниже.

#### Диспетчер EMT

Диспетчер EMT позволяет наряду со способом вызова по JSR использовать команды EMT. Способ передачи параметров совпадает с вызовами JSR. При обработке команды EMT с номером, превышающим допустимый, диспетчер проверяет содержимое ячейки 166. Если оно не равно нулю, управление передается по адресу, находящемуся в этой ячейке. Регистр PS процессора устанавливается в соответствии с содержимым ячейки 32. Возврат из пользовательского обработчика производится непосредственно в вызвавшую программу командой RTI.

Достоинства использования EMT — это позиционная независимость и меньший объем кодовой программы. Недостатки: пользовательская программа может использовать EMT с младшими номерами для своих нужд; вызовы по EMT выполняются дольше, чем по JSR; при некоторых условиях, возникающих при выполнении вызовов и вызывающих переход в монитор (клавиша «СТОП», неверный вызов макрокоманд .BJMP и .BJSR), выводится адрес, принадлежащий не программе, а диспетчеру EMT.

#### Драйвер ОЗУ/ПЗУ

Драйвер ОЗУ/ПЗУ позволяет подключать по запросу пользовательской программы заданные страницы ОЗУ/ПЗУ, определять состояние памяти и страниц, читать и записывать слова, байты и массивы слов в скрытые страницы, обращаться к подпрограммам, находящимся в скрытых страницах, а также обеспечивает защиту системной страницы.

#### Драйвер клавиатуры

Драйвер клавиатуры обеспечивает прием символов с клавиатуры, их перекодировку в соответствии с таблицами КОИ-8, КОИ-7Н0, КОИ-7Н1, КОИ-7Н2 и запись во внутренний буфер, выдачу символов по запросу и по прерыванию в программу пользователя с ожиданием.

и без, программирование и подстановку функциональных клавиш, автоповтор последнего символа, редактирование вводимой и существующей строки, а также акустическую обратную связь при нажатии на клавиши (щелчок пьезодинамика). Драйвер может работать в двух режимах: прямом, когда коды функциональных клавиш передаются в программу без изменений, и с перекодировкой, когда некоторые функциональные клавиши выдают другие коды (см. приложение 1).

### *Драйвер экрана*

Драйвер экрана обеспечивает вывод на экран символьной, псевдографической и графической информации, очистку экрана, сдвиг вверх и вниз, управление цветами и палитрами, переключение буферов экрана, управление курсором.

Вывод символьной и графической информации возможен в трех режимах:

- 256 точек в строке, 4 цвета, 32 символа в строке;
- 512 точек в строке, 2 цвета, 64 символа в строке;
- 512 точек в строке, 2 цвета, 80 символов в строке.

Вывод алфавитно-цифровой информации производится в соответствии с одной из таблиц: КОИ-8, КОИ-7Н0, КОИ-7Н1, КОИ-7Н2. Знакогенератор символов располагается в системной странице ОЗУ начиная со смещения 10000. Хотя системная страница не может быть подключена, область знакогенератора доступна по чтению и записи с помощью запросов драйвера ОЗУ/ПЗУ. Описание каждого символа состоит из 10 байт, соответствующим телевизионным строкам. Байт с младшим адресом соответствует верхней строке символа, младший бит байта — левой точке в строке. Таблица начинается с изображения символа с восьмеричным кодом 40. Далее подряд идут изображения символов с кодами от 41 до 377. При начальном пуске ЭВМ в ОЗУ переписывается стандартный знакогенератор из ПЗУ. Программа пользователя может заменить любую часть знакогенератора.

В режиме 80 символов в строке на экран выводятся первые 6 битов каждого байта.

На экране может быть отображен алфавитно-цифровой курсор, указывающий позицию текущего символьного вывода. Возможны две формы курсора: в виде прямоугольника, инвертирующего изображение текущего символа, и в виде подчеркивания этого символа.

При выводе в последней строке символа «ПС» (код 12) или «ПФ» (код 14) сдвиг экрана вверх может производиться как скачком, так и плавно (по желанию пользователя).

Возможно задание атрибутов выводимых символов: двойная ширина, подчеркивание, инверсия. Допускается также вывод символов двойной высоты (в два приема: сначала выводится верхняя часть буквы, затем курсор смещается на строку ниже в той же позиции и выводится нижняя часть). Двойная высота может сочетаться с любой шириной, но подчеркивание символов двойной высоты не производится.

При выводе на экран графической информации возможно два режима работы: с замещением текущим цветом или с инверсией существующего цвета, а также задание прямоугольного рабочего окна (изменения изображения на экране доступны при этом только в пределах границ данного окна).

### *Драйвер кассетного магнитофона*

Драйвер кассетного магнитофона обеспечивает поиск, запись и считывание файлов с бытового кассетного магнитофона в формате, совместимом с БК-0010.

### *Драйвер принтера*

Драйвер принтера обеспечивает вывод на печатающее устройство алфавитно-цифровой информации с перекодировкой по таблицам КОИ-8, КОИ-7Н0, КОИ-7Н1 и КОИ-7Н2, определение готовности ПУ, отслеживание позиции печатающей головки.

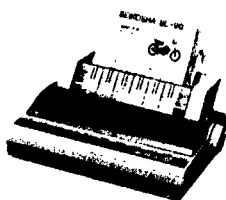
### *Драйвер НГМД*

Драйвер НГМД обеспечивает запись и считывание информации с гибких магнитных дисков в формате, совместимом с контроллером КМД от ДВК-3, а также форматирование дисков.

*(Продолжение следует.)*



# HARD & SOFT



Струйные принтеры серии MC-6312 — одни из наиболее популярных среди пользователей БК. Но вот незадача: принтеры и печатающие головки нового типа, выпускаемые заводом «Радий» в настоящее время, рассчитаны (для совместимости со стандартом Hewlett Packard) на большее рабочее напряжение, чем выпускавшиеся ранее. И при использовании новых головок на старых моделях с недостаточным напряжением возможна частичная непечатака символов (обычно их вертикальных линий).

В данной статье рассказывается, как доработать старую модель принтера для нормального использования новых печатающих головок.

А. И. Кузнецов,

Москва

## Доработка принтеров серии MC-6312

Модели MS(MP)-6312 и MS-6312M

В принтерах серии MS(MP)-6312 и MS-6312M при использовании термоструйных печатающих головок (ТСПГ) старого типа («Электроника MC-6902», «MC-6902.01» или «MC-6902.02») рабочее напряжение устанавливается с помощью подстроечного резистора R8\* (см. схему платы стабилизатора принтера на рис. 1) равным 16,2—17,0 В. Для работы с новыми головками «Электроника MC-6902.03», а также с выпускаемыми фирмой Hewlett Packard головками N 51604A, 51605R(G,B), 51616A, 92261A и др. (в том числе для трехцветной печати) необходимо рабочее напряжение 22,2—23,0 В. Чтобы сохранить при этом и возможность работы с ТСПГ старого типа, лучше всего обеспечить выработку двух рабочих напряжений с выбором нужного переключением DIP-выключателя.

Доработка схемы сводится к установке в разрыв соединения между точками 1 и 2 (рис. 1) стабилизатора KC-162A2. Затем точка 1 на плате стабилизатора (размещение элементов на плате показано на рис. 2) соединяется монтажным проводом с 8-й позицией DIP-переключателя. (Он должен быть установлен в принтере на плате контроллера с левой стороны в предусмотренном для этого месте. При отсутствии DIP-переключателя его необходимо установить.) Для надежности можно параллельно соединить 7-й и 8-й ползунки DIP-переключателя параллельно. Выбор нужного рабочего напряжения производится включением-выключением DIP-переключателя, который при этом соединяет или отключает точку 1 на плате стабилизатора от контакта «общий» (рис. 3).

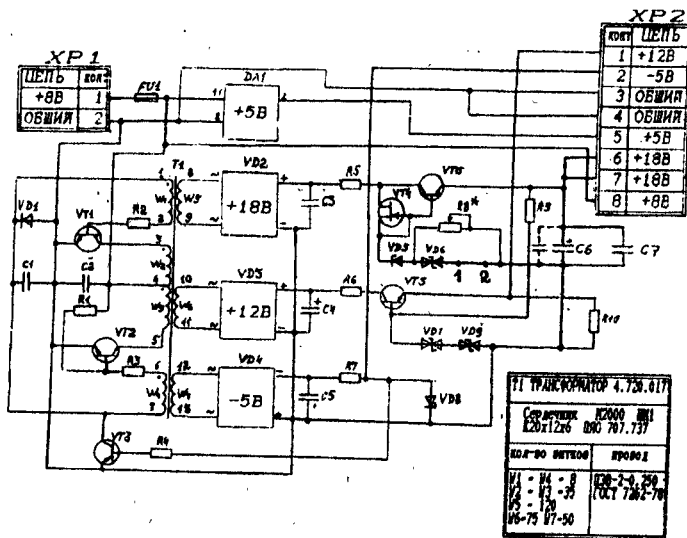
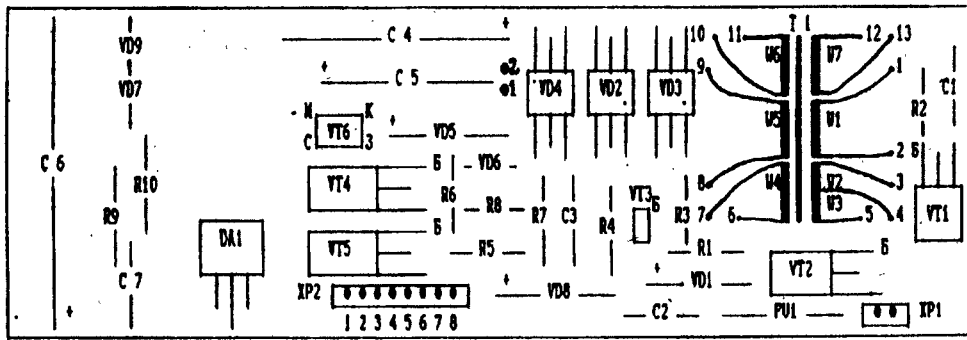


Рис. 1. Плата стабилизатора (A5). Принципиальная схема





РЕЗИСТОРЫ			ДИОДЫ			КОНДЕНСАТОРЫ		
R1	МЛТ-0,25	10 КОМ	VD1	Д-15225	C1	ДМ-88-100-0,22	МКФ	
R2	МЛТ-0,25	10 КОМ	VD2, VD3, VD4	Д-1307А	C2, C3, C7	ДВ-68-100-2,100	МКФ	
R3	МЛТ-0,25	10 КОМ	VD5	Д-1715А	C4	ДВ-68-100-2,100	МКФ	
R4	МЛТ-0,25	10 КОМ	VD6, VD7, VD9	Д-162А2	C5	ДВ-68-100-2,100	МКФ	
R5	МЛТ-0,25	10 КОМ	VD8	КС147А	C6	КС0-29 258-470	МКФ	
R6	МЛТ-0,25	390 Ом	ТРАНЗИСТОРЫ			МИКРОСХЕМЫ		
R7	МЛТ-0,25	390 Ом	VT1, VT2	КТ817В	DA1	КР142ЕР5А		
R8	МЛТ-0,25	390 Ом	VT3, VT6	КТ817В	ВСТАВКА ПЛАВКАЯ			
R9	МЛТ-0,25	10 КОМ	VT4	КТ303Е	P01		-2,0 А	
R10	МЛТ-0,25	10 КОМ						

Рис. 2. Расположение элементов на плате стабилизатора

### Модель МС-6312

В принтерах модели МС-6312 (будьте внимательны: различные модели струйных принтеров отличаются только русскими или латинскими буквами в названии) при использовании печатающих головок «Электроника МС-6902», «МС-6902.01» и «МС-6902.02» рабочее напряжение устанавливается равным 17,2—18,0 В. Новые же головки «МС-6902.03» (и выпускаемые Hewlett Packard) рассчитаны на напряжение 23,2—24,0 В.

Доработка схемы принтера для возможности выработки двух рабочих напряжений делается так же, как это описано выше для моделей MS(MP)-6312 и MS-6312M. Но в МС-6312 DIP-переключатель изначально отсутствует и его необходимо установить на свободном месте слева на плате контроллера. Точка «1» соединяется монтажным проводом с ползунком 2 DIP-переключателя, другой контакт которого должен быть подсоединен к «общей» шине на плате. Переключение рабочего напряжения производится, как и для моделей MS-6312M и MS(MP)-6312, соединением или отключением точки «1» с «общей» шиной путем включения-выключения DIP-переключателя (рис. 4). Для большей надежности можно включить ползунки 1 и 2 DIP-переключателя параллельно.

DIP-переключатель (тип ВДМ1-2)

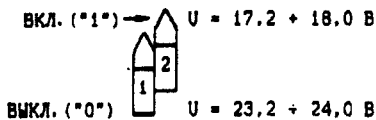


Рис. 3

DIP-переключатель (тип ВДМ1-8)

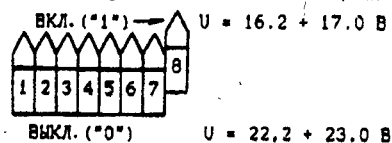
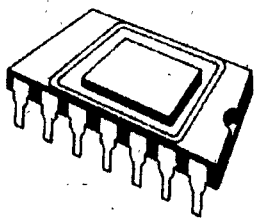


Рис. 4

**Примечание.** Включение рассчитанных на пониженное рабочее напряжение печатающих головок старого типа на напряжение 22,2—23,0 В приводит к перегоранию терморезисторов (которые производят выброс капелек чернил из сопел головки), т. е. к выходу ТСПГ из строя. Головки с перегоревшими терморезисторами восстановлению и гарантийному обслуживанию не подлежат.



На мощных IBM-совместимых компьютерах кроме центрального процессора (основного) обычно имеется так называемый математический сопроцессор. Последний ориентирован на выполнение расчетных операций (умножение, деление, вычисления с плавающей точкой, математические функции) и освобождает центральный процессор, умеющий работать только с целыми числами, от обработки сложных эмулирующих программ.

На БК установка математического сопроцессора, к сожалению, не предусмотрена схемотехнически. Но почему бы не реализовать его в виде периферийного устройства? Автор данной статьи делится с читателями опытом подключения к БК микрокалькулятора БЗ-23 (или аналогичного), который как раз и исполняет роль математического сопроцессора.

**В. В. Зыков,**

*г. Елизово, Камчатская обл.*

## Математический сопроцессор для БК

При всем многообразии имеющихся сегодня в России персональных компьютеров, ПЭВМ БК-0010(.01) и БК-0011(М) занимают особое место, не в последнюю очередь благодаря своей относительной дешевизне и наличию обширного набора программного обеспечения, способного удовлетворить практически любые капризы пользователя. Подключение дисководов и дополнительного ОЗУ (а теперь — и винчестера) превращает БК в профессиональную машину. Но, имея огромное количество достоинств, БК не свободен от недостатков. Одним из самых ощутимых среди них остается ограниченный объем пользовательского ОЗУ. И чем более сложные задачи ставятся перед БК, тем это ограничение становится заметнее. Особенно оно проявляется при решении математических задач, требующих многократных расчетов. Конечно, разработаны всевозможные библиотеки вычислительных подпрограмм, но они, как правило, занимают в памяти очень много места (в основном из-за того, что «сердце» компьютера — процессор 1801ВМ1А, к сожалению, «умеет» выполнять только самые простые логические и арифметические действия).

Конечно, можно экономить место в ОЗУ за счет использования при написании программ модульной (оверлейной) структуры, когда необходимые блоки подгружаются с внешнего носителя в процессе работы. Можно использовать и защитные подпрограммы ФОКАЛа и БЕЙСИКа, но это не всегда удобно. (Например, если ДОЗУ занято операционной системой или вообще отсутствует.) На IBMках все проще: достаточно установить математический сопроцессор, берущий на себя все сложные расчеты. Но почему бы не реализовать тот же механизм на БК?

Предлагаемая вниманию читателей система позволяет объединить в единый комплекс компьютер БК-0010.01 и калькулятор БЗ-23. В принципе, все сказанное ниже справедливо практически для любого компьютера и калькулятора, работающего с ТТЛ-логикой. Если же использовать программируемый микрокалькулятор, появляется возможность хранить на дисках или кассетах БК и загружать при необходимости в калькулятор написанные для него программы. Единственным недостатком данной системы является относительно низкое по сравнению с БК быстродействие калькулятора, что, однако, вполне компенсируется предоставляемыми им новыми возможностями.

### Принципы работы комплекса БК-0010.01/БЗ-23

Основной принцип работы адаптера неоднократно использовался разными авторами при разработке всевозможных приставок к микрокалькуляторам [1, 2, 3, 4] и заключается в имитации нажатия клавиш калькулятора путем срабатывания транзисторных ключей. (Изготовление некоторых из этих устройств, в частности [2], показало низкую надежность срабатывания таких ключей, поэтому в данной разработке коммутация выводов калькулятора производится электронными ключами на микросхемах К143КТ1 или им подобных. Необходимость наличия отрицательного напряжения источника питания для этих элементов неудобств не

создаёт, так как оно имеется в большинстве микрокалькуляторов на выходе преобразователя напряжения, в БЗ-23 это вывод 2 элемента Я-154.)

Каждой клавише микрокалькулятора соответствует определенный двоичный код. Для его передачи задействованы разряды 9—15 регистра вывода (&177714), что не мешает одновременной работе принтера или другого устройства, подключенного к младшим разрядам. Напомним, что клавиатуру любого счетного устройства можно представить в виде матрицы, в которой пересечение (реально — замыкание) вертикального и горизонтального столбцов соответствует одной из клавиш. На рис. 1 показана такая матрица для микрокалькулятора «Электроника БЗ-23», выполненного на м/с К145ИП11. Цифры соответствуют выводам микросхемы. (Для другого калькулятора соответствующую информацию можно найти в прилагаемой к его инструкции схеме.)

	37	36	35	34	33	32	31	30	29	
$\%$	9	6	3	.	$\sqrt{\quad}$	$\times 0$	$\sqrt{\quad}$	$+0$		11
СБР	7	4	1	0	M=0	M=0	X→M-	/-/		13
Q←→X	8	5	2	.	M=0	M=0	X→M-	/-/		14
=	:	*	-	+	M→X	X→M+	X→M-	X→M+		15

Рис. 1

Обозначения в таблице:

- СБР — очистка содержимого регистра,
- Q←→X — обмен содержимого регистров,
- $\sqrt{\quad}$  — извлечение квадратного корня,
- M=0 — обнуление регистра памяти,
- M→X — вывод содержимого регистра памяти на индикатор,
- X→M+ — прибавление числа на индикаторе к содержимому регистра памяти,
- X→M- — вычитание числа на индикаторе из содержимого регистра памяти,
- /-/ — смена знака числа на индикаторе,
- \*Q — аналогично паре действий «\*» и «=»,
- +Q — аналогично паре действий «+» и «=».

Из приведенной таблицы видно, что далеко не все возможности м/с К145ИП11 реализованы в БЗ-23: это извлечение квадратного корня, смена знака содержимого регистров, обмен содержимого регистров между собой, работа с памятью. Теперь же с помощью БК мы сможем задействовать эти функции.

Двоичное число, подаваемое на регистр ввода-вывода, имеет следующую структуру: на разряды 9—11 подается код, соответствующий горизонтальным линиям матрицы (их четыре), на разряды 12—15 — вертикальным (их девять). Пока регистр обнулен, на выводах УП БК действуют высокие уровни (см. рис. 5). Эти же уровни присутствуют на входах дешифраторов DD1—DD2, а на управляющих входах коммутаторов DA1—DA7, которые в данном случае находятся в «разомкнутом» состоянии, содержится «логический 0». При записи «логической единицы» в один (или несколько) разрядов порта на выходе УП появляется «логический 0» (напомним, что в УП используется инверсный код). Микросхемы DD1 и DD2 дешифрируют поданные на их входы число. На одном из их выходов появляется «логический 0», который через инвертор поступает на управляющий вход соответствующего коммутатора. Происходит замыкание необходимых выводов микросхем К145ИП11. Таким образом, подавая через порт ввода-вывода цепочку кодов, можно производить на калькуляторе различные вычисления. (Кстати, при этом необходимо после ввода каждого кода производить очистку регистра порта командой CLR @#177714, чтобы разомкнуть коммутирующие ключи.) Восьмеричные и двоичные коды клавиш микрокалькулятора (МК) сведены в следующую таблицу.

Состояние регистра вывода		Восьм. код	Выход DD1	Вход МК	Выход DD2	Вход МК	Клавиатура МК
биты 9—11	биты 12—15						
111	1111	177 000	1	11	0	37	%
011	1111	176 000	3	13	0	37	СВР
101	1111	175 000	5	14	0	37	Q←→X
001	1111	174 000	7	15	0	37	=
111	0111	167 000	1	11	1	36	9
011	0111	166 000	3	13	1	36	7
101	0111	165 000	5	14	1	36	8
001	0111	164 000	7	15	1	36	:
111	1011	157 000	1	11	2	35	6
011	1011	156 000	3	13	2	35	4
101	1011	155 000	5	14	2	35	5
001	1011	154 000	7	15	2	35	*
111	0011	147 000	1	11	3	34	3
011	0011	146 000	3	13	3	34	1
101	0011	145 000	5	14	3	34	2
001	0011	144 000	7	15	3	34	-
101	1101	135 000	5	14	4	33	.
011	1101	136 000	3	13	4	33	0
001	1101	134 000	7	15	4	33	+
111	0101	127 000	1	11	5	32	√
011	0101	126 000	3	13	5	32	M=0
001	0101	124 000	7	15	5	32	M→X
111	1001	117 000	1	11	6	31	·Q
001	1001	114 000	7	15	6	31	X→M+
011	0001	106 000	3	13	7	30	X→M-
111	1110	77 000	1	11	8	29	+Q
011	1110	76 000	3	13	8	29	/-/

Снятие результатов вычислений производится с выводов индикатора микрокалькулятора. Для различия цифры достаточно использовать элементы «а», «б», «в», «г», «д» и «и» каждого сегмента светодиодного индикатора. На рис. 6 показан один из вариантов сопряжения входа УП БК-0010.01 с индикатором АЛС 318.

Если в калькуляторе применены люминесцентные приборы отображения информации, имеющие напряжения питания, превышающие уровни ТТЛ, то можно воспользоваться одним из приведенных ниже вариантов доработки схемы на рис. 2 и 3. В принципе, доработка сводится к ограничению напряжения на входах УП БК до уровня ТТЛ, так что могут быть и любые другие варианты. Данные фрагменты схем взяты из [3] и [4].

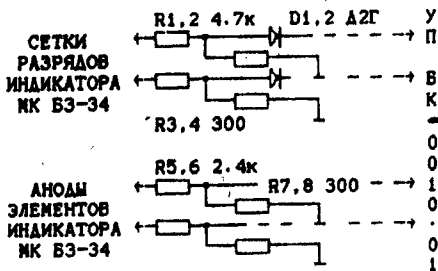


Рис. 2

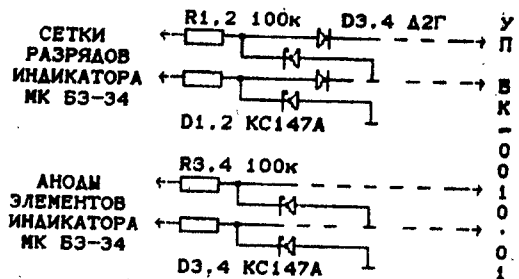


Рис. 3

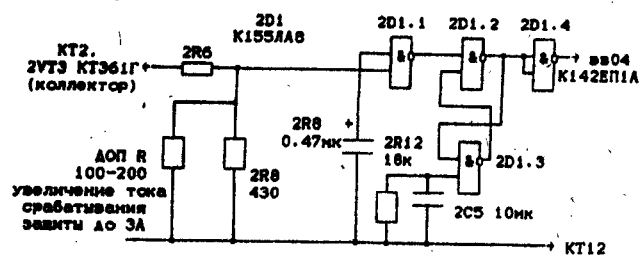


Рис. 4

Питание приставки и калькулятора осуществляется от БК. Чтобы предотвратить срабатывание защиты в блоке питания БК, необходимо параллельно резистору 2R8 (по схеме БП) подключить резистор сопротивлением 100 Ом (если от этого же блока питается и дисковод) или 200 Ом (только для приставки). Фрагмент доработанной схемы БП приведен на рис. 4.

В приложениях 1—3 даны основные подпрограммы для поддержки работы вышеописанной конструкции с подробными комментариями, а также простейшая демонстрационная программа.

#### Литература

1. *Бронштейн М.* Приставка-автомат к микрокалькулятору БЗ-23 //Радио. 1989. №6.
2. *Баранов В.* Приставка-автомат к микрокалькулятору БЗ-23 (дополнения) //Радио. 1990. №12.
3. *Волков Ф.* Приставка-программатор к микрокалькулятору //Радио. 1990. №12.
4. *Караваев А.* Микрокалькулятор... управляет моделью //Радио. 1988. №4.

#### Приложение 1. Подпрограмма передачи кода операции с БК на калькулятор

```

COD: CLR #177714 ;очистка регистра
      MOV #КОД,#177714 ;передача кода (КОД)
      MOV #10000,R4 ;задержки для согласования
      MOV R4,R5 ; работы процессора БК
A: SOB R4,A ; и м/с К145ИП11
     CLR #177714
B: SOB R5,B
     RTS PC
  
```

Это действие не представляет трудности, поэтому рассмотрим один из простейших вариантов передачи кода. Задержки после каждой операции необходимы, так как скорости работы процессора БК и микросхемы К145ИП11 сильно отличаются. Задержка, равная #10000, получена экспериментально при отладке демонстрационной программы (приложение 3) и может быть иной в зависимости от конкретного алгоритма.

#### Приложение 2. Подпрограмма считывания данных с индикатора АЛС318 микрокалькулятора

В индикаторах микрокалькуляторов применяется динамическая система индикации, т. е. цифры загораются последовательно с довольно большой частотой, что создаёт эффект непрерывного горения. Поэтому при получении данных с регистра ввода-вывода каждая цифра фактически будет считана 11—12 раз, причем не обязательно с разряда 9 (начального). Задача подпрограммы — определение начала числа, удаление «лишних» цифр и вывод результата на дисплей. Для нормальной обработки состояние регистра ввода-вывода считывается 600 (шестью) раз.

**Примечание.** Обе приведенные здесь подпрограммы (передача кода операции и прием результатов) перемещаемы, т. е. могут располагаться в любом месте адресного пространства БК.

```

IND: MOV #37000,R2 ;600-кратное считывание данных
      MOV #600,R1 ; регистра ввода-вывода
C: MOV #177714,(R2)+
     SOB R1,C
      MOV #37000,R3 ;определение характера числа на
      MOV #200,R1 ; индикаторе. Оно начинается с:
D: MOV (R3)+,R2
     CMP #1000,R2 ;точки разряда 9?
     BEQ B1
     CMP #40000,R2 ;минуса разряда 9?
  
```

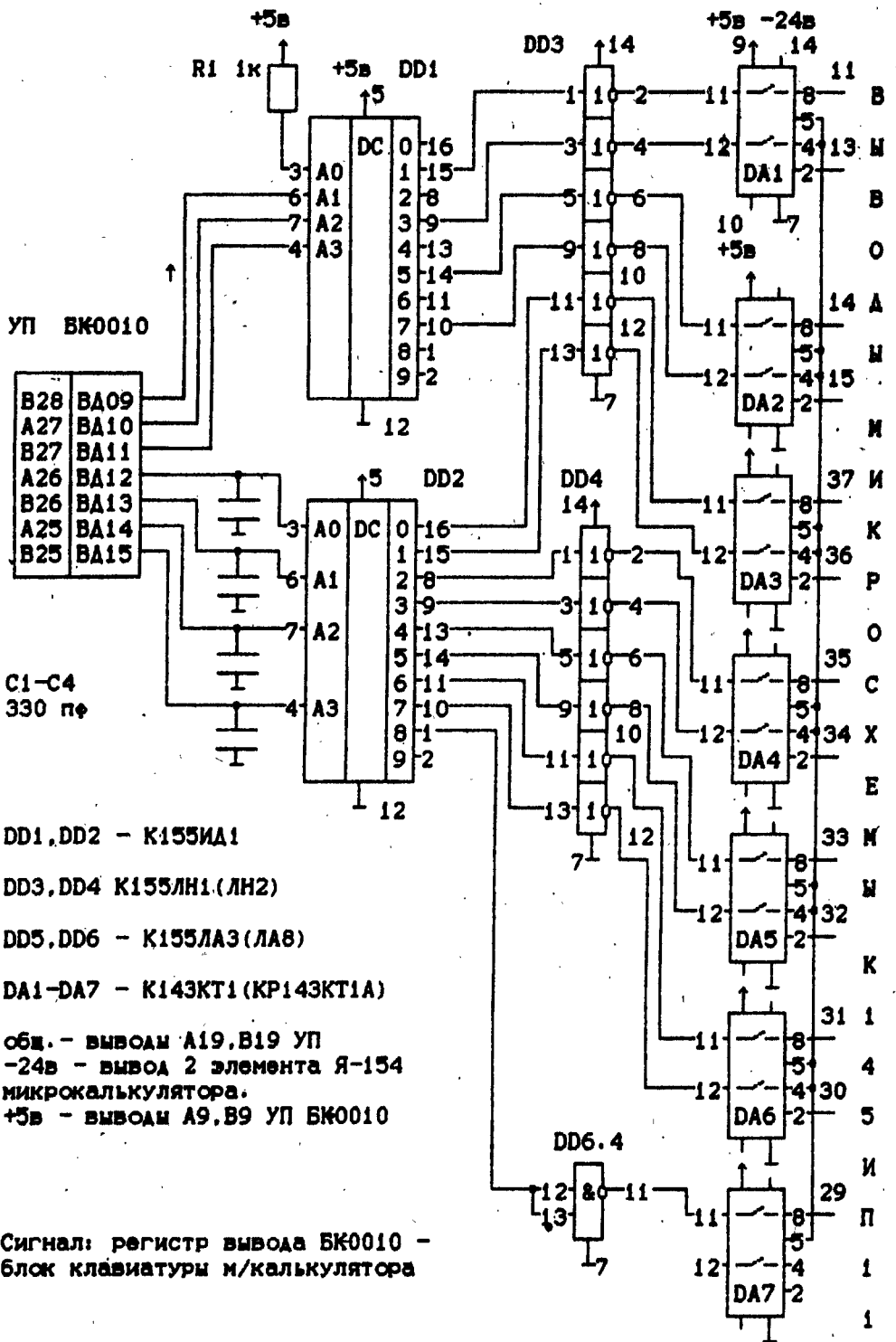


Рис. 5

УП БК-0010

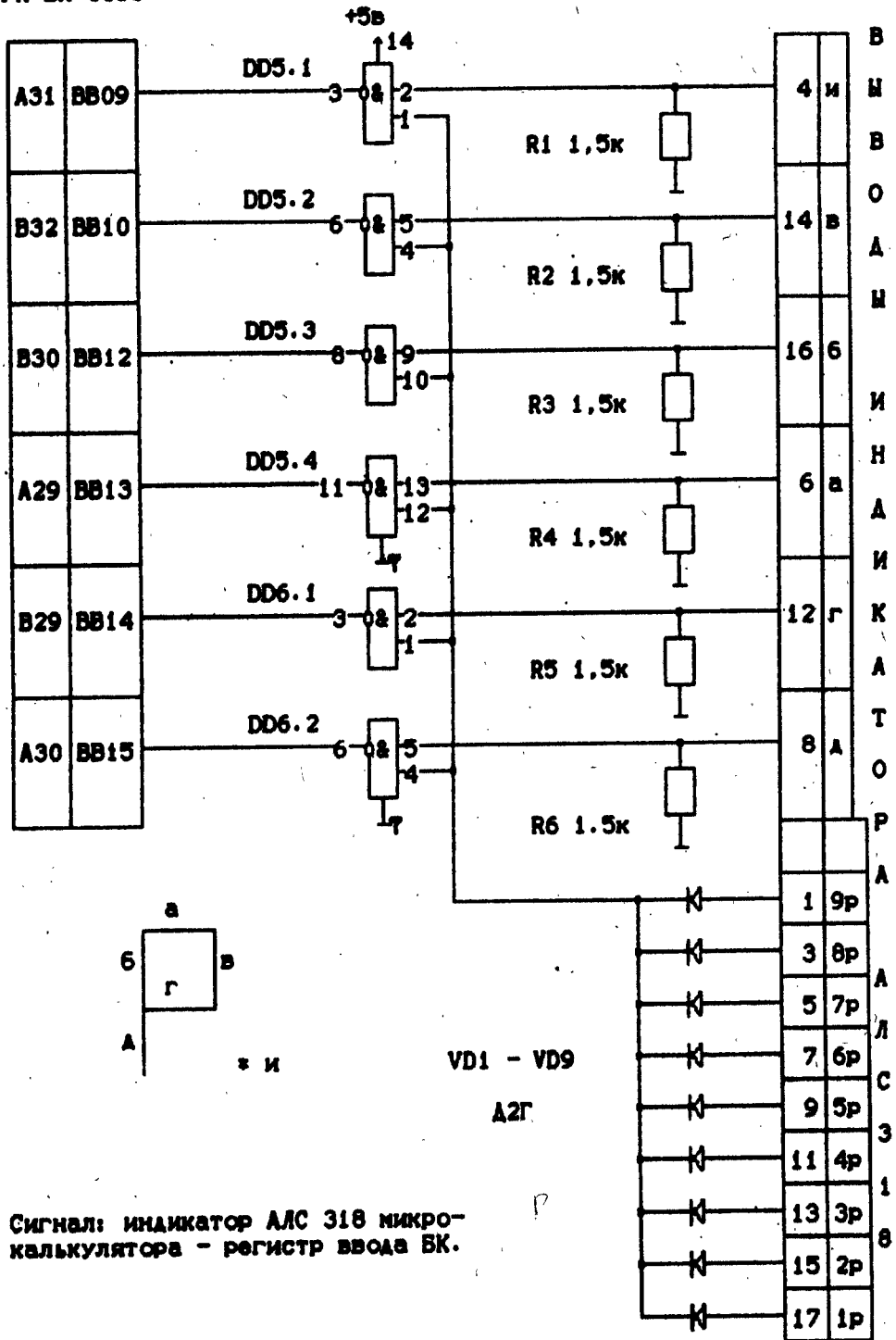


Рис. 6

```

BEQ B1
CMP #41000,R2           ; точки и минуса разряда 9?
BEQ B1
SOB R1,D
CLR R5
BR F1                   ; разряд 9 не светится...
B1:  MOV R2,R5
B2:  JSR PC,E
     BR F
E:   MOV (R3),R1         ; подпрограмма перекодировки
     MOV PC,R4           ; данных в коды микроЭВМ
     ADD (PC)+,R4        ; и вывод числа на дисплей
     .COD1+2
W:   CMP R1,(R4)+
     BNE W
     ADD #56,R4
     MOV R4,R1
     MOV #2,R2
     EMT 20
R:   RTS PC
M:   CMP #0,(R3)+       ; поиск начала цифры, отсеивание
     BNE M               ; "лишних" цифр, определение конца
     ADD #6,R3           ; числа (разряд 1 индикатора)
     RTS PC
F:   JSR PC,M
     CMP #0,(R3)
     BEQ F
     CMP R5,(R3)
     BEQ R
     BR B2
F1:  JSR PC,M
     CMP R5,(R3)
     BNE F1
F2:  CMP R5,(R3)+
     BEQ F2
F3:  JSR PC,E
     JSR PC,M
     CMP R5,(R3)
     BNE F3
     BR R
COD1: .#13000           0 ;коды (показания регистра ввода-вывода),
      .#2000            1 ; соответствующие цифрам
      .#162000          2 ; на индикаторе
      .#62000           3
      .#52000           4
      .#70000           5
      .#170000          6
      .#22000           7
      .#172000          8
      .#72000           9
      .#3000            1.
      .#163000          2.
      .#63000           3.
      .#53000           4.
      .#71000           5.
      .#171000          6.
      .#23000           7.
      .#173000          8.
      .#73000           9.
      .#133000          0.
      .#40000           -
      .#41000           -
    
```



```

.#1000      " ."
.E
.#003460    0 ;таблица перекодировки
.#003461    1
.#003462    2
.#003463    3
.#003464    4
.#003465    5
.#003466    6
.#003467    7
.#003470    8
.#003471    9
.#027061    1.
.#027062    2.
.#027063    3.
.#027064    4.
.#027065    5.
.#027066    6.
.#027067    7.
.#027070    8.
.#027071    9.
.#027060    0.
.#003455    -
.#027055    -
.#003456    " ."
END

```

### Приложение 3. Демонстрационная программа

Данная программа фактически дублирует обычную работу с микрокалькулятором. Все действия выполняются на клавиатуре БК, а полученный результат выводится после знаков «=», «%», «^» и «R» на дисплей.

```

PROG:  EMT 6                ;очистка экрана
        MOV #14,R0
        EMT 16
        CLR @#177714
Y:      MOV #12,R0
        EMT 16
A3:     MOV PC,R1           ;определение адреса таблицы кодов
        ADD (PC)+,R1
        .@COD3+2
        MOV #34,R2
        EMT 6              ;принятие команды и
        MOV R0,R3          ;ее вывод на дисплей
        EMT 16
B3:     CMP R3,(R1)+       ;команда допустима?
        BEQ C3
        SOB R2,B3
        MOV #30,R0
        EMT 16
        BR A3              ;если нет, то повторить ввод
C3:     ADD #77,R1         ;перекодировка
        MOV (R1),@#177714 ;передача кода на регистр ввода-вывода
        MOV #10000,R4     ;программная задержка
        MOV R4,R5
D3:     SOB R4,D3
        CLR @#177714
N3:     SOB R5,N3
        CMP #75,R0        ;знак "="?
        BEQ IN           ;если да, то вывод на экран
        CMP #45,R0       ;знак "%"?

```

```

BEQ IN          ;если да, то вывод на экран
CMP #136,R0     ;знак "√" (квадратный корень)?
BEQ IN          ;если да, то вывод на экран
CMP #122,R0     ;вывод числа с индикатора ("R")?
BEQ IN          ;да - просто вывод числа
                ; без математических операций
BR A3           ;продолжение ввода
    
```

```

IN:  CMP #0,@#177714 ;программная задержка
     BEQ IN
     MOV #12,R0
     EMT 16
     JSR PC,IND
     JMP Y
     .E
    
```

COD3:	; операция	; клавиша БК
.#45	; %	%
.#14	; СБР	СБР
.#121	; Q←→X	Q
.#75	; =	=
.#71	; 9	9
.#67	; 7	7
.#70	; 8	8
.#57	; /	/
.#66	; 6	6
.#64	; 4	4
.#65	; 5	5
.#52	; *	*
.#63	; 3	3
.#61	; 1	1
.#62	; 2	2
.#55	; -	-
.#56	; ". "	". "
.#60	; 0	0
.#53	; +	+
.#136	; √	√
.#115	; M=0	M
.#130	; M→X	X
.#43	; +Q	#
.#120	; X→M+	P
.#125	; X→M-	U
.#42	; +Q	"
.#111	; /- /	I
.#122	; чтение числа	R
	; на индикаторе	

.E.E.E.E

```

.#177000      %
.#176000      СБР
.#175000      Q←→X
.#174000      =
.#167000      9
.#166000      7
.#165000      8
.#164000      /
.#157000      6
.#156000      4
.#155000      5
.#154000      *
.#147000      3
.#146000      2
.#145000      1
    
```

.#144000	-
.#135000	.
.#136000	0
.#134000	+
.#127000	$\sqrt{\quad}$
.#126000	M=6
.#124000	M→X
.#117000	*0
.#114000	X→M+
.#106000	X→M-
.#77000	+Q
.#76000	/--/
.#0	нет операции

IND: ; сюда нужно скопировать текст  
; подпрограммы из Приложения 2

END

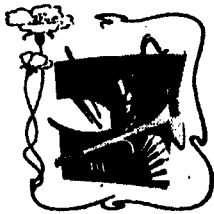
## От редакции

Идея применения калькулятора (в том числе и старого, уже не используемого «по прямому назначению») в качестве внешнего математического сопроцессора очень интересна. Действительно, обычно для реализации сложных вычислений (в том числе с плавающей точкой) в ассемблерных программах приходится использовать объемистые наборы подпрограмм (а иногда — и писать их самому). Даже если к БК подключен КНГМД с ПЗУ «326 прошивки», где содержится готовая библиотека таких подпрограмм (см.: Персональный компьютер БК-0010 — БК-0011М. 1994. №5), их отработка отнимает значительное время, что особенно неприятно для игр. Автор же данной статьи предлагает использовать готовую «математическую микросхему» калькулятора, передавать ей код и аргументы требуемой операции и принимать результат. (Кстати, именно так работают процессор и математический сопроцессор на IBM-совместимых ЭВМ.) А если вместо БЗ-23 подключить к БК более мощный калькулятор (например, программируемый БЗ-34), можно с его помощью обрабатывать тригонометрические функции и даже поручать калькулятору вычисления по несложным формулам, «загоняя» в него соответствующие «микропрограммы». (К сожалению, для сложных расчетов такой способ мало применим — слишком низко быстродействие калькулятора.)

Специалистам по аппаратному обеспечению для БК имеет смысл подумать и о таком варианте, как установка непосредственно на плату БК одной только «математической микросхемы», извлеченной из старого калькулятора (может быть даже с не работающей клавиатурой и индикатором), подобно музыкальному процессору. Это несколько усложнит схемы сопряжения, но зато сделает всю конструкцию более компактной.

И наконец, следует обсудить проблему написания программного обеспечения для БК с учетом возможного наличия «внешнего сопроцессора». Чтобы программы были универсальными (пригодными и для БК без калькулятора), в качестве одного из вариантов можно взять аналогичную стратегию работы с математическим сопроцессором на IBM. Транслятор с ассемблера (или языка программирования высокого уровня) должен определять (либо запрашивать у пользователя) наличие калькуляторного устройства и, соответственно, подстыковывать к создаваемой программе библиотеку подпрограмм поддержки калькулятора или эмуляции вычислений средствами самого процессора. (Разумеется, по-разному будут транслироваться и вычислительные операторы обрабатываемой программы.) А еще лучше добавлять подобный «определяющий» блок и обе библиотеки подпрограмм в каждую исполняемую программу: есть «внешний сопроцессор» — обращаться к нему, нет — к его программной эмуляции. Правда, при этом усугубляется проблема дефицита ОЗУ, так что желательно выполнить «определятель» и обе библиотеки в виде оверлеев и подгружать из них нужные. Например, так: «определятель» (тестирует конфигурацию и устанавливает в зависимости от нее специально предусмотренную ячейку — флаг) — основная программа — одна из библиотек (в зависимости от флага) и драйвер для преобразования универсальных команд математических операций программы в конкретные обращения к подгружаемой библиотеке.

Завершая обсуждение статьи, отметим следующий момент. Бит 7 регистра @#177716 (по крайней мере в БК-0011(М)) служит указателем наличия (вернее, отсутствия, что обозначено «лог. 1») в системе «настоящего» математического сопроцессора. Зачем это было сделано в стандартной модели, неизвестно; теперь же можно использовать данный разряд по прямому назначению, подведя к соответствующему выводу микросхемы сигнал наличия внешнего калькулятора.



Публикация в №1 нашего журнала за 1995 г. материала о подключении к БК музыкального процессора и создании с его помощью мелодий заинтересовала многих любителей БК. Но вот задача: подключить музыкальный процессор можно только к одиннадцатой модели. А как же тогда быть тысячам владельцев БК-0010?

Специально для пользователей «десятки» авторами данной статьи разработан музыкальный синтезатор, реализующий практически все функции музыкального процессора для БК-0011М.

**С. М. Неробеев, А. В. Сорокин,**

*Москва*

## **Трехканальный музыкальный синтезатор для БК-0010(01)**

За десять лет, прошедших с начала выпуска, для БК-0010 появилось множество программных и аппаратных разработок, которые сделали его одним из лучших бытовых компьютеров в нашей стране. Повысилось качество игровых программ, но по-прежнему для их музыкального оформления приходится довольствоваться весьма скромными возможностями, которые предоставляет штатный пьезоэлектрический динамик. Между тем высококачественные звуковые эффекты и мелодии, наряду с профессиональной графикой, являются неотъемлемой частью игровых программ на других персональных ЭВМ, в частности на игровом компьютере ZX-Spectrum.

Попытки программным либо аппаратным путем улучшить музыкальное оформление предпринимались неоднократно. Существует множество музыкальных редакторов (TVM, MELOWOMAN, MAESTRO, MUSDRUM), позволяющих создавать трех-, четырех- и даже пятиголосные мелодии, но качество их звучания ограничено плохими частотными характеристиками пьезодинамика, а также отсутствием обертонов в выходном сигнале. Поэтому даже подключение специального усилителя с низкоомным динамиком вместо пьезоэлектрического не избавляет от последнего недостатка, так как он обусловлен аппаратными особенностями ЭВМ.

Цифро-аналоговый преобразователь «Совок» (его схема была опубликована в №1 за 1995 г., в статье В. П. Юрова) в сочетании с внешним усилителем звуковой частоты позволяет получить наиболее качественное монофоническое звучание оцифрованной мелодии. Однако закодированная музыка в этом случае занимает в памяти недопустимо много места, что очень актуально для БК-0010. Кроме того, воспроизведение такой мелодии происходит под непрерывным управлением центрального процессора, следовательно, параллельное ее исполнение с основной программой значительно влияет на скорость последней. Сам же цифро-аналоговый преобразователь целиком занимает порт ввода-вывода и не допускает подключения каких-либо периферийных устройств одновременно с ним, так как резисторы ЦАП шунтируют разряды порта на «землю».

Безусловно, на сегодня наиболее качественное стереофоническое воспроизведение мелодий обеспечивает музыкальный процессор, очень хорошо зарекомендовавший себя на БК-0011М. К сожалению, подключение этого процессора к БК-0010 сопряжено с определенными трудностями, о которых мы упоминали в статье «Подключение музыкального процессора к БК-0011М» (№1 за 1995 г.), кроме того, практически все использующее этот процессор программное обеспечение будет работать только на БК-0011М.

Предлагаемый нами музыкальный синтезатор является аналогом музыкального процессора и обладает всеми его достоинствами: компактным хранением текста мелодии при высоком качестве ее исполнения, возможностью работы по прерыванию, простотой схемы. Музыкальный синтезатор может работать без всяких изменений как на БК-0010, так и на БК-0011. Он содержит микросхему отечественного производства К580ВИ53, делитель частоты и стереофонический усилитель. Дополнительно предусмотрена возможность подключения внешнего усилителя звуковой частоты. Музыкальный синтезатор может быть подключен к разъему порта как внешнее устройство или установлен внутри корпуса вычислительной машины. При этом он, в отличие от ЦАП, не шунтирует разряды порта ввода-вывода и, следовательно, позволяет подключать одновременно с ним любые периферийные устройства.

У предлагаемой разработки есть только один существенный недостаток по сравнению с музыкальным процессором: микросхема К580ВИ53 не позволяет управлять громкостью звука.

Избавиться от этого недостатка можно установкой еще одной такой же микросхемы, однако это приведет к неоправданному усложнению схемы.

Программирование музыкального синтезатора осуществляется через регистр с адресом 177714 посредством записи в него информации о тексте мелодии. Попытка чтения из этого регистра дает неопределенный результат, не несущий никакой информации о состоянии синтезатора. Музыкальный синтезатор имеет три независимых канала, каждый из которых может генерировать выходное колебание в виде меандра с частотой  $F_i = FT/p_i$ , где  $FT = 1.5$  МГц,  $F_i$  — частота на выходе канала с номером  $i$  ( $i = 0, 1, 2$ ),  $p_i$  — число, записываемое в счетчик канала с номером  $i$  и определяющее его коэффициент деления.

Выходы всех трех каналов определяются таким образом, что имеется возможность работы как на два отдельных громкоговорителя (режим «псевдостерео»), так и на один (режим «моно»). В режиме «псевдостерео» на один громкоговоритель работает нулевой канал, на другой — второй канал, а первый канал работает одновременно на оба громкоговорителя. В режиме «моно» все каналы работают на один громкоговоритель.

Все три канала музыкального синтезатора программируются независимо. Для работы таймера один раз в начале программы необходимо записать в регистр 177714 три слова инициализации (по одному на каждый канал). Формат слова инициализации приведен на рис. 1.

177714:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	1	1	1	1	1	-	-	1	1	1	1	1	0

└──────────┘  
номер канала

Рис. 1

Биты 7 и 8 определяют номер инициализируемого канала и приведены в табл. 1.

Таблица 1

Номер канала	Бит 7 или 9	Бит 6 или 8
0	0	0
1	0	1
2	1	1

После этого в каждый канал музыкального синтезатора можно записывать числа  $p_i$ , определяющие частоту выходного колебания  $F_i$ . Число записывается в два этапа путем пересылки по адресу 177714 двух слов, формат которых приведен на рис. 2. Первым пересылается младший байт числа  $p_i$ , затем его старший байт (число  $p_i$  — 16-разрядное, состоит из двух байтов).

177714:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	1	0	0	0	-	-	D7	D6	D5	D4	D3	D2	D1	D0

└──────────┘  
номер канала

Рис. 2

На рис. 2 биты 9 и 8 определяют номер канала, в который пересылаются байты числа  $p_i$  (соответствие также приведено в табл. 1), а разряды 0 — 7 определяют содержимое пересылаемого байта.

После того как запрограммированы все три канала (в каждый канал занесено число  $p_i$ , определяющее частоту  $F_i$ ), музыкальный синтезатор готов к работе. Каждый из его каналов можно включить или выключить, устанавливая соответствующий бит регистра 177714. Соответствие битов и режимов коммутации приведено в табл. 2.

Таблица 2

	Коммутация тактовой частоты $F_T$	Коммутация канала №2	Коммутация канала №1	Коммутация канала №0
Номер бита	13	12	11	10
Включение	0	1	1	1
Выключение	1	0	0	0

При включении первого канала он будет работать непрерывно, генерируя частоту  $F_1$  до тех пор, пока его не выключат.

Для правильного функционирования микросхемы необходимо особое внимание обратить на то, что вся информация в регистре порта 177714 представляется в инверсном виде, т. е. запрограммированный «0» соответствует подаче на микросхему «1» и наоборот. Поэтому прежде чем пересылать в порт слово, сформированное с соответствии с вышеприведенным описанием, его необходимо инвертировать.

Перед тем как приступить к самостоятельному программированию музыкального синтезатора, советуем тщательно проанализировать следующий пример листинга.

; инициализация всех каналов

```

MOV #37476, R0 ; в R0 записывается число,
COM R0 ; инициализирующее канал N0
MOV R0, @#177714 ; перед записью в регистр
MOV #37576, R0 ; информация инвертируется
COM R0 ; инициализация канала N0
MOV R0, @#177714 ; в R0 - число, инициализирующее
MOV #37676, R0 ; канал N1
COM R0 ; инициализация канала N1
MOV R0, @#177714 ; в R0 - число, инициализирующее
COM R0 ; канал N2
MOV R0, @#177714 ; инициализация канала N2

```

; программирование каналов: запись коэффициентов деления  
;  $N_0 = 200$ ,  $N_1 = 555$ ,  $N_2 = 1200$  (числа восьмеричные)

; канал N0

```

MOV #20200, R0 ; число 200 помещается
COM R0 ; в первый байт
MOV R0, @#177714 ; пересылка первого байта
COM R0 ; в регистр 177714
MOV #20000, R0 ; старший байт - нулевой
COM R0 ;
MOV R0, @#177714 ; пересылка второго байта

```

; канал N1

```

MOV #20555, R0 ; число 555 выходит за пределы
COM R0 ; одного байта. В первом байте
MOV R0, @#177714 ; размещается число 155,
MOV #20401, R0 ; во втором байте размещается
COM R0 ; число 1
MOV R0, @#177714 ;

```

; канал N2

```

MOV #21200, R0 ; число 1200 выходит за пределы
COM R0 ; одного байта. В первом
MOV R0, @#177714 ; байте - число 200,
MOV #21002, R0 ; во втором байте - число 2
COM R0 ;
MOV R0, @#177714 ;

```

; каналы запрограммированы, но не включены

BIS #20000, @#177714 ; включение тактовой частоты FT  
 VIC #20000, @#177714 ; включение канала N0 (звучит канал N0)  
 VIC #40000, @#177714 ; включение канала N1 (звучат каналы N0 и N1)  
 VIC #10000, @#177714 ; включение канала N2 (звучат все каналы: N0, N1 и N2)

; каналы включены

Выключение тактовой частоты производится командой VIC, каналов — командой BIS.

**Примечание.** Включать и выключать каналы можно одновременно с программированием, устанавливая соответствующие биты (табл. 2) при размещении байтов в регистре R0 (см. примеры). Но в этом случае необходимо внимательно следить за тем, чтобы очередное пересылаемое слово не затирало разрядов, установленных при предыдущей пересылке.

При подключении музыкального синтезатора следует воспользоваться незадействованными контактами порта в соответствии с табл. 3.

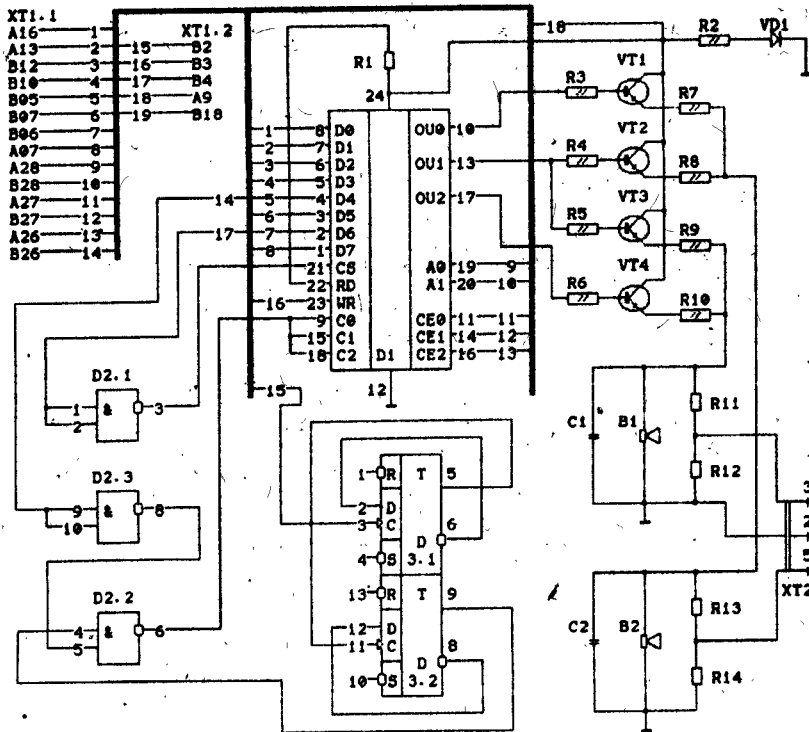
Таблица 3

Номер контакта	Сигнал	БК-0010(.01)	БК-0011M
B2	CLC	1 (D16)	6 (D5)
B3	CS1	13 (D8)	3 (D30)
B4	CS2	12 (D1)	1 (D30)

## Перечень элементов схемы

Обозначение	Наименование	Количество
C1, C2	Конденсатор КМ-5а-Н90-2.2пФ	2
<b>Микросхемы</b>		
D1	K580BI53	1
D2	K155AA3	1
D3	K155TM2	1
<b>Резисторы</b>		
R1	МАТ-0.125-1 кОм	1
R2	МАТ-0.125-300 Ом	1
R3—R6	МАТ-0.125-82 Ом	4
R7—R10	МАТ-0.125-51 Ом	4
R11, R13	МАТ-0.125-750 Ом	2
R12, R14	МАТ-0.125-220 Ом	2
VD1	Светодиод АЛ307Б	1
VT1—VT4	Транзистор КТ815Б	4
B1, B2	Головка динамическая 0.5-ГДШ-15, 8 Ом	2
XT1	Вилка СНП58-64/94х9В-23-2	1
XT2	Соединитель ОНЦ-КГ-4-5/16Р	1

## СХЕМА ТРЕХКАНАЛЬНОГО МУЗЫКАЛЬНОГО СИНТЕЗАТОРА

\* \* \* **Внимание! Опечатка** \* \* \*

В ассемблерном листинге программы UKOR (Канивец И. В. Укорочение БЕЙСИК-программ при трансляции. Персональный компьютер БК-0010 — БК-0011М. №2. 1995, с. 45) два оператора :#62700 и .@LEMT+2 расположены в одной и той же, четвертой по счету, строке. Хотя это, строго говоря, не является ошибкой, не все ассемблер-трансляторы для БК (в частности, популярная программа M18) правильно обрабатывают подобную запись. Поэтому при вводе листинга рекомендуется разместить указанные операторы по одному в разных строках. Ниже приведен исправленный фрагмент ассемблерного текста.

```

; *****
; Начальный загрузчик
MOV PC, R0
.#62700
.@LEMT+2
MOV R0, @#30
. . . . .

```

По этой же причине при генерации кодового листинга UKOR (с. 48) в его начале пропущен код смещения к ячейке LEMT+2. Начало первой строки кодов должно выглядеть так:

018700 062700 000014 010037 000030 и т. д.

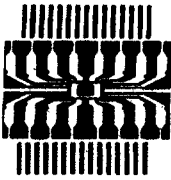
Соответственно, контрольная сумма программы будет равна 142325.

\* \* \*

В листинге ассемблерного дампа прошивки ПЗУ КИГМА (№5. 1994, с. 41) допущена ошибка. В строке с меткой 12\$ вместо команды BIT #400000,(R4) следует читать: BIT #40000,(R4).

Приносим читателям свои извинения.





Пожалуй, самым крупным недостатком БК-0010(.01) является малый объем памяти, и по этой причине разработка блоков дополнительного ОЗУ и ПЗУ привлекла значительное внимание читателей. В предыдущем выпуске журнала мы уже затрагивали эту тему, а сейчас БКманам предлагается еще один вариант подобного устройства.

А. В. Кузнецов,

г. Сокол

## ДОПОЛНИТЕЛЬНЫЙ БЛОК ДОЗУ И ПЗУ ДЛЯ БК-0010(.01)

Среди различных контроллеров дисководов, используемых на БК-0010(.01), сегодня наибольшую популярность получили КНГМД от БК-0011(М) и выпускаемый МП «Ширакаци». (Последний особенно широко распространен в Армении и Грузии.) Однако, к большому сожалению любителей БК, всегда мечтавших иметь возможность работы с дисководом, к «армянскому» контроллеру практически нет никакой программной поддержки. Правда, МП «Ширакаци» выпустило для своей разработки версию ДОС, но она настолько слаба, что о ней не хочется вспоминать (да и не у всех она есть). Чтобы облегчить работу с контроллером НГМД «Ширакаци» и использовать его наиболее эффективно, была разработана предлагаемая схема. (Опытные образцы блока ДОЗУ/ПЗУ прошли испытание в течение более чем годового срока.) Блок содержит в себе дополнительное оперативное запоминающее устройство (ДОЗУ) емкостью 8 кб и постоянное запоминающее устройство емкостью также на 8 кб с прошитой ДОС. ДОЗУ располагается по адресам 140000—160000, а ПЗУ — 120000—140000 (авторы разработки — А. Мартиросян и А. Айказян).

Описываемый блок ДОЗУ/ПЗУ, в принципе, совместим с любым контроллером. Его также можно использовать и для ЭВМ без дисковода — как дополнительный модуль ДОЗУ/ПЗУ. Кроме того, если заменить микросхемы К573РФ5 (ПЗУ) на К573РУ10 (ОЗУ), то получится просто блок ДОЗУ и можно будет загружать свои системные программы или трансляторы с языков высокого уровня на месте БЕЙСИКа и ФОКАЛа. Если же подключить к микросхемам К573РУ10 буферный источник питания, то получится энергонезависимое ОЗУ, и тогда вообще можно будет отказаться от программируемых ПЗУ К573РФ5.

Рассмотрим основные принципы построения и функционирования схемы блока. Микросхемы D1 и D2 составляют регистр адреса выбираемой ячейки памяти, а их выходы об-

разуют шину адреса ЗУ. Адрес обращения к ЗУ фиксируется в регистре по сигналу SYNC (СИА), проходящему через линию задержки, выполненную на резисторе и конденсаторе (вывод микросхем STB). Четыре старших разряда шины адреса на выходе микросхемы D2 использованы для адресации ДОЗУ и ПЗУ, они подведены к селектору адреса, роль которого выполняет микросхема D5. Самый младший разряд выхода D1 используется микросхемой D4.2 для переключения четности адреса. При наличии сигнала WTBT (БАЙТ) элемент D6.1 или D6.3 микросхемы выдает инверсный сигнал на D6.2 или D6.3 соответственно. И только после проверки микросхемой D12 условного присутствия на ее входах одного и того же сигнала D6.2 или D6.3 вырабатывается соответствующий сигнал для включения группы микросхем ЗУ младших или старших байтов.

В зависимости от наличия сигнала DIN (ВВОД) или DOUT (ВЫВОД) логические элементы микросхем D9 и D10 вырабатывают сигналы записи или чтения, используемые для выборки кристаллов С2 или W обеих групп микросхем ЗУ. Сигнал RPLY (СИП) генерируется элементами D4.1, D11, D4.3 и D3.4 при активном сигнале DIN или DOUT.

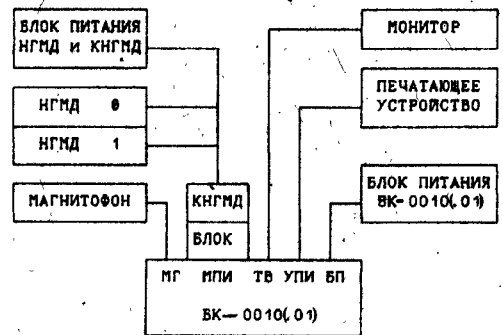
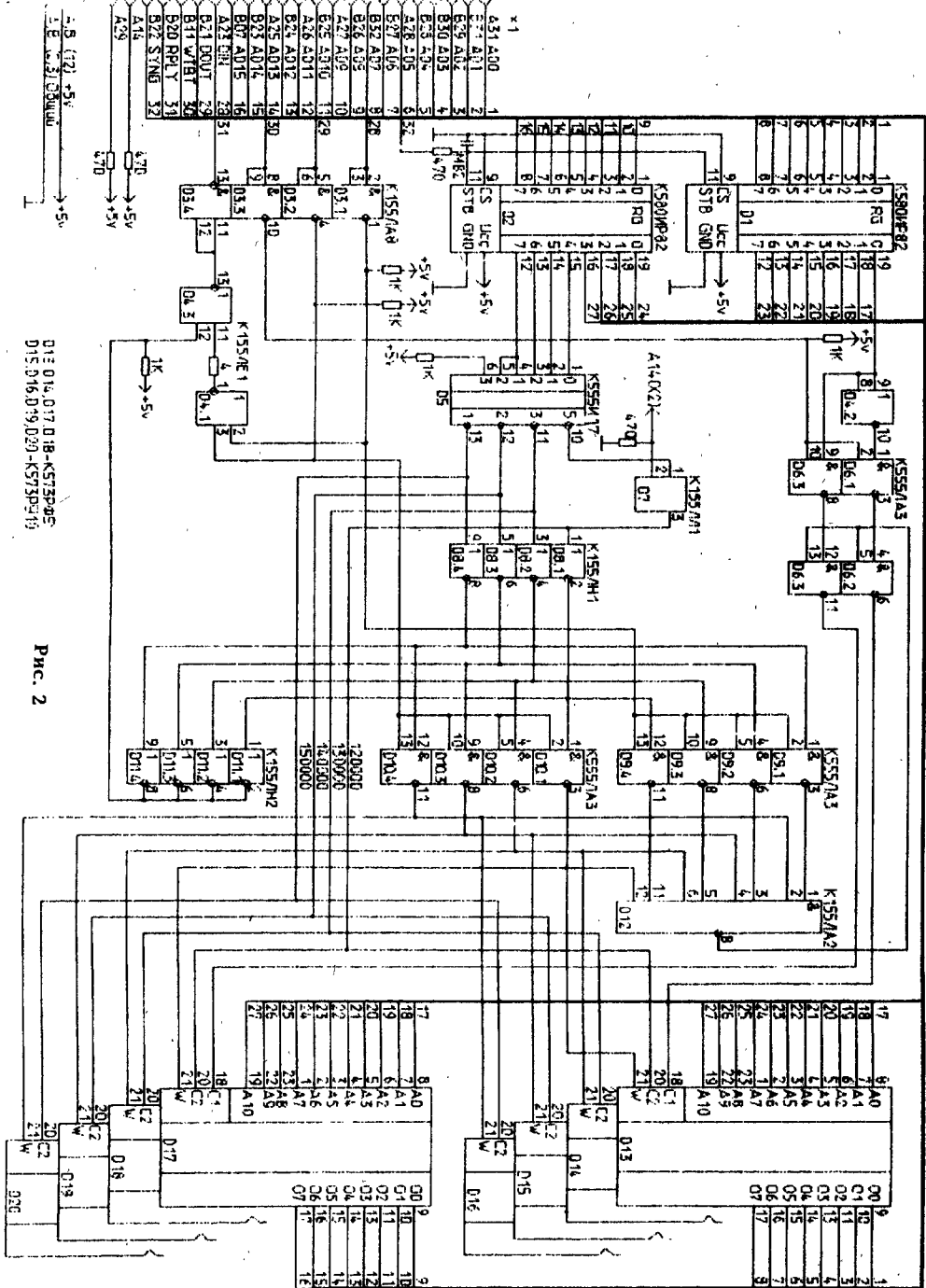


Рис.1. Структурная схема подключения блока ДОЗУ/ПЗУ



-5 (12) +5V  
 F1 0.25A 250V  
 L

D1E 016, D17, D18, K5732P45  
 D15, D16, D19, D20, K5732P19

Вход-выход А14 разъема X2 используется контроллером НГМД для включения-выключения части ПЗУ блока, расположенной в адресном пространстве 120000 — 130000.

Потребляемый блоком ДОЗУ/ПЗУ ток равен 250 мА при напряжении 5 В (источником является блок питания компьютера). Конструктивно он выполнен на двух двухсторонних печатных платах, расположенных друг над другом и размещенных в корпусе от модуля МСТА.

Главным преимуществом блока, на наш взгляд, является простота работы с ним. Любой пользователь, даже не имеющий опыта общения с компьютером, легко сможет манипулировать файлами, так как сразу же после включения машины на экран выводится оболочка дисковой операционной системы, прошитой в ПЗУ (автор ДОС — А. Н. Девятилов). Кроме того, по возможности вся информация выдается на русском языке (в отличие от таких систем, как ANDOS, NORD или HORTON).

На данном этапе в ДОС реализованы следующие команды:

- **КАТАЛОГ** — выводит перечень файлов в левое окно. При просмотре каталога можно использовать клавиши «ВВЕРХ» и «ВНИЗ». Для выбора дисководов задействована клавиша «КТ». Требуемые файлы можно пометить с помощью клавиши «ПРОБЕЛ», повторное ее нажатие отменяет выделение.
- **ЗАПУСК** — считывает программу, выбранную в списке файлов, и запускает ее. В случае ошибки считывание повторяется трижды. Возможен запуск «многосерийных» игр, таких, как XENON, RICK, ПЕРЕВАЛ, НЕПОБЕДИМЫЙ, FERRARI2 и др., что с «голым» контроллером (без ДОС в расширенном ОЗУ или ПЗУ) было бы невозможно;
- **ПЕЧАТЬ** — обеспечивает вывод файлов в текстовом виде на экран для просмотра;

• **УДАЛЕНИЕ** — удаляет помеченные файлы;

• **КОПИРОВАНИЕ** — позволяет копировать файлы с магнитофона на диск и обратно или с диска на диск. Формат диска распознается автоматически. С дискета копируются только помеченные файлы, с магнитофона — все подряд. Копировщик выполнен в виде дополнительной утилиты, загружается автоматически с диска (если он там есть) сразу же после обращения к этой команде и сохраняется до конца работы с ЭВМ;

• **СЖАТИЕ ДИСКА** — обеспечивает «сжатие» (сквизирование) информации на магнитном диске (удаление «мусора», уплотнение пустых участков между файлами);

• **ФОРМАТИРОВАНИЕ** — позволяет форматировать и инициализировать магнитные диски;

• **ПЕРЕИМЕНОВАНИЕ** — используется для переименования файлов;

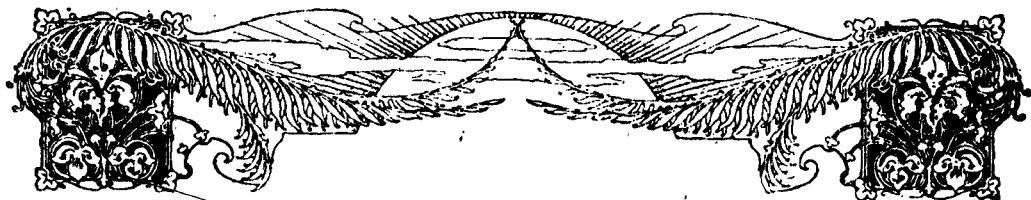
• **ПРОВЕРКА ДИСКА** — контролирует состояние информации на магнитном диске. При работе производит посекторное считывание с диска и вывод на экран монитора результатов по каждому сектору;

• **ВЫХОД** — обеспечивает выход в пусковой монитор.

Для использования цветного дисплея дополнительно предусмотрена команда **COLOR** (клавиша «С»).

В пакет ДОС также входит утилита **КАТАЛОГИЗАТОР**, которая позволяет вывести только нужную информацию о файлах в данном каталоге на экран или на принтер. Драйвер принтера — подгружаемый и сохраняется в программе, если ее заново записать на диск. **КАТАЛОГИЗАТОР** позволяет сохранить список прочитанных файлов и всю информацию о них в формате EDASP.

*Мы обращаемся ко всем обладателям контроллера НГМД МП «Ширакаци», особенно к энтузиастам-программистам, готовым принять участие в разработке программного обеспечения с использованием вышеуказанного блока. Письма и пожелания по дальнейшему развитию блока можно направлять по адресу: 162100, Вологодская обл., г. Сокол, ул. Рабочая, 11. Кузнецову А. В.*





Несколько месяцев назад почти одновременно появились новые версии двух наиболее известных баз данных для БК: NORD-BASE 2.1 (А. Г. Прудковский) и BASIS 2.1 (С. Камнев). Пользователи получили возможность выбирать, но, чтобы сделать правильный выбор, нужно сравнить затраты и получаемый результат для той и другой программы. В данной статье автор проводит краткий сравнительный анализ вышеназванных баз данных с точки зрения пользователя.

П. П. Животовский,

г. Снежногорск

## Базы данных для БК: сравнительный обзор

Начиная разговор о каких-либо программах, прежде всего необходимо выяснить, какие требования они предъявляют компьютеру и периферии, в какой программной среде работают.

Для функционирования базы данных NORD-BASE А. Г. Прудковским была специально создана ОС NORD 2.17 (позже возможность работы с этой базой данных была введена и в новую версию ОС NORD 3.1Б). Но операционная система NORD защищена от несанкционированного копирования, и хотя тем самым обеспечивается защита авторских прав, но резко сокращается круг пользователей. Даже зарегистрированным пользователям не очень удобно держать отдельную практически пустую дискету ради операционной системы и установщика. А что делать, если в один далеко не прекрасный день операционная система «не захочет» устанавливаться — посылать дискету в Москву? Правда, справедливости ради следует заметить, что NORD — одна из наиболее надежных из всех существующих на БК операционных систем, хотя и работает медленнее других.

В отличие от А. Г. Прудковского, разработчик базы данных BASIS 2.1 С. Камнев отказался от автономной ОС (как это было в предыдущих версиях) и использует ANDOS 3.1, что, по мнению многих пользователей, является более прогрессивным решением. Но автор данной статьи придерживается иной точки зрения: автономная ОС при всех ее неудобствах (тех же, что рассмотрены выше применительно к NORD) все же обеспечивала какую-то независимость базы данных от общих течений среди разработчиков и пользователей ОС. А теперь «жизнь» программы BASIS полностью подчинена воле создателя ANDOS А. М. Надежина: стоит лишь чуть-чуть нарушить совместимость очередной версии ОС с существующей (3.1) или, скажем, изменить адреса тех или иных внутренних подпрограмм, и пользователям придется либо приобретать новую версию BASIS (а С. Камневу нужно будет ее написать), либо ради работы с базой данных возвращаться к более старой версии ОС. (Учитывая длительное и плодотворное сотрудничество С. Камнева и А. М. Надежина, хотя бы в подготовке пакетов ANDOS, можно считать, что указанная автором статьи опасность попасть в зависимость друг от друга является скорее теоретической, нежели реальной. — *Прим. ред.*) Правда, нельзя не отметить и значительные «плюсы» отказа от собственной ОС: благодаря этому и программа, и создаваемые с ее помощью базы данных можно безо всяких проблем хранить на любой дискете в формате ANDOS и свободно переписывать средствами DiskMaster, создавая любое количество резервных копий. Это делает BASIS несколько более привлекательным для пользователей по сравнению с NORD-BASE.

Рассмотрев «отношение» той и другой базы данных к «окружающей среде», перейдем к сравнению их собственных характеристик. Главными же характеристиками любой базы являются объем одновременно обрабатываемой информации, скорость такой обработки и удобство задания модели поиска. И вот в этом рассматриваемые программы расходятся принципиально.

Любая универсальность, как правило, рассчитана на массового потребителя, но не позволяет использовать все возможности аппаратной части. Выжать же из «железа» максимальные мощности можно лишь при создании программ специально для данной модели компьютера. По этому пути пошел А. Г. Прудковский, четко разграничив БК-0010(01) и БК-0011(М), что позволило ему создать для одиннадцатой модели более мощную базу данных. В этом легко убедиться, сравнив несколько основных параметров NORD-BASE и BASIS.

1. База данных NORD-BASE обрабатывает сразу 15 кб информации, тогда как BASIS — только 4 кб. Это, в свою очередь, определяет количество обрабатываемых за один прием записей, что хорошо видно на примере прилагаемой в комплекте BASIS базы телефонных

номеров. При работе с BASIS за один раз прочитывается только 99 записей, так что для перегонки базы в текстовый файл требуется пять ее загрузок. (Причина этого в ограничении не только объема буфера, но и количества записей.) Значительную часть памяти «съедает» и горизонтальная разделительная черта под каждой строкой, хотя можно было бы прекрасно обойтись и без нее.

NORD-BASE же «проглатывает» упомянутую базу телефонных номеров целиком, а если судить по степени заполнения экранного ОЗУ, позволит добавить туда еще с полсотни записей. Это позволяет и произвести сортировку всех записей одновременно, что невозможно в предыдущем случае. И наконец, при создании текстового файла не требуется повторных загрузок базы данных. Получаемый эффект становится существенно более заметным при увеличении количества записей до нескольких тысяч (например, в крупном телефонном справочнике). В этом случае сортировка записей в BASIS практически невозможна, так как для этого потребовалось бы загрузить фрагмент объемом в несколько буферов, а сортировка производится только в пределах одного буфера.

2. На первый взгляд кажется, что BASIS работает быстрее, но не следует забывать, что она при этом обрабатывает лишь 99 записей, тогда как NORD-BASE — до 400.

3. Работу с моделью поиска рассмотрим на примере базы «междугородные телефонные коды» из комплекта BASIS. Запустите программу BASIS и задайте модель поиска «Н\*», что должно означать «все города, названия которых начинаются с буквы Н». Сделав это, вы с удивлением увидите в выведенном списке названия «Камень-на-Оби», «Ростов-на-Дону» и т. д. Или пусть у вас имеется база данных, содержащая слова на английском и русском языках (например, каталог файлов). При поиске BASIS выводит записи как с латинскими, так и с русскими названиями, начинающимися на одинаковые буквы («А», «О», «Р» и т. д.), что не позволяет отделить русские слова от английских. База же NORD-BASE не имеет указанных недостатков и выводит данные в строгом соответствии модели поиска.

4. Максимальный размер одной записи по вертикали в BASIS равен восьми строкам и ограничивается снизу горизонтальной чертой, что не позволяет создавать базу из элементов псевдографики, задевая большее число строк (скажем, условные изображения микросхем). В NORD-BASE же это вполне возможно, хотя и требует использования нескольких записей подряд, что тоже не совсем удобно. Правда, применение в BASIS дополнительной функции несколько расширяет ее возможности, позволяя дополнить основной файл базы данных вспомогательной графической картинкой (пример — база «транзисторы»).

Соответствующие возможности NORD-BASE можно наглядно продемонстрировать на примере созданной автором данной статьи базы данных «справочник микросхем». Всего в ней на сегодня 314 элементов, и работа над наполнением базы продолжается.

Номер	ИЗОБРАЖЕНИЕ/ФУНКЦИЯ	ТИП/ТАБЛИЦА
2669	4-раврядный сумматор	
2670	вычитатель	
2671		Н
2672	6-А S N1	выв. Назначение
2673	5-В	выв. Назначение
2674	13-Р S	1 Тантовый вход 13 Сброс
2675	1-С	2 Выход суммат. 14 Выход сумматор
2676	4-Н	3 Свободный 15 Свободный
2677	7-А S N2	4 Выбор режима 16 Выбор режима
2678	0-В	5 Информационный вход 17 Информационный вход
2679	0-В	6 Информационный вход 18 Информационный вход
2680	0-Р S	7 Информационный вход 19 Информационный вход
2681	0-С	8 Информационный вход 20 Информационный вход
2682	0-Н	9 Выбор режима 21 Выбор режима
2683	0-С	10 Свободный 22 Свободный
2684	18-А S N3	11 Выход суммат. 23 Выход сумматор
2685	17-В	12 Обши 24 Усс
2686	0-Р S	14 К555ИМ7
2687	0-С	24 SN74L3395
2688	16-Н	
2689		
2690	19-А S N4	
2691	20-В	
2692	0-Р S	23
2693	0-С	
2694	21-Н	
2695		

Приведенный на рисунке фрагмент занимает 26 текстовых строк (более одного экрана). Это может быть сделано только в NORD-BASE 2.1, так как в BASIS количество строк под одну запись жестко ограничено и принудительно отчеркивается горизонтальной линией. Хотя в NORD-BASE принято, казалось бы, еще более строгое ограничение — всего одна строка для каждой записи с собственным номером, в данной базе нет горизонтальных полос и можно «обмануть» ее, заняв под каждый элемент любую количество строк. Правда, разработчик NORD-BASE ограничил количество строк порядковым номером 9999, что определяет максимально возможное количество элементов в базе данных. В упомянутой выше базе микросхем последний использованный номер строки равен 5995, так что туда поместится еще около 300 элементов.

Однако NORD-BASE имеет и существенно затрудняющий работу недостаток — отсутствие поиска до первого встреченного значения, соответствующего модели. Ведь ясно, что в базе имеется только одна микросхема, скажем К555ИМ7, так что, найдя ее, нет смысла продолжать поиск до конца базы. Алгоритм про-

граммы NORD-BASE предполагает быстрый поиск по порядковым номерам в первой колонке (благодаря чему головка дисковода не «шаркает» по дискете, а сразу попадает в нужное место). Поэтому для ускорения поиска пришлось создать дополнительную базу — указатель, где для каждой микросхемы указаны порядковые номера строк в основной базе. При таком, казалось бы, не совсем рациональном построении справочника поиск нужного элемента занимает всего 40 секунд, тогда как при поиске сразу в основной базе (по типу микросхемы) — три минуты.

Справедливости ради следует заметить, что сравнение NORD-BASE и BASIS нельзя свести только к критике последней. Так, при загрузке BASIS сразу выходит в основное меню. Это более удобно, чем в NORD-BASE, где после запуска открывается новая (пустая) база, а для выхода в меню нужна дополнительная операция.

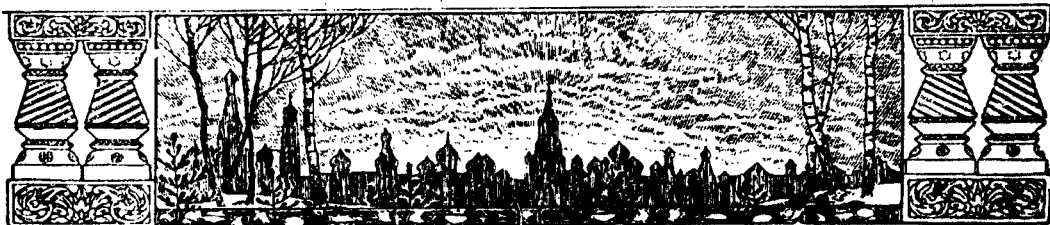
Обобщая рассмотренные преимущества и недостатки двух наиболее мощных баз данных для БК, можно дать разработчикам подобных программ следующие рекомендации.

- Четко разграничить версии для БК-0010(.01) и БК-0011(М) с как можно более полным использованием возможностей одиннадцатой модели. (Так, хотя объем буфера NORD-BASE для БК-0011(М) и так достаточно велик по сравнению с BASIS, можно сделать его еще большим.)
- Для расширения областей использования баз данных предусмотреть создание текстовых файлов — картотек, использование «карточек» (стандартных экранных форм) и работу с графическими образами.
- Предусмотреть возможность поиска до первого соответствующего модели элемента базы.
- Подсказка должна вызываться в любой момент работы с программой и, желательно, быть контекстной. (В NORD-BASE же, например, для вызова опции «Help» нужно выйти из текущего режима, а после вызова и просмотра подсказки снова войти обратно — быстрее просто пролистать распечатку документации.)
- В NORD-BASE сразу же после запуска открывается структура для новой базы данных. Но чаще приходится работать с уже созданными ранее базами, так что удобнее было бы после запуска открывать каталог структур (опция «Read Struct»).
- Нужно дать пользователю возможность самостоятельно указывать количество строк в каждой записи (например, при открытии новой базы). В NORD-BASE вместо этого можно увеличить максимальное число строк с 9999 до 99999.

Полезно также расширить возможности операционной системы или файловой оболочки, чтобы средствами встроенного редактора можно было корректировать файлы и структуры базы (сдвигка и раздвигка строк, перенос строк и их слияние). Особенно это касается ОС NORD, где простейший встроенный редактор уже имеется.

## От редакции

Перечень рекомендаций автора статьи разработчикам баз данных можно дополнить такой идеей, как реализация на БК языка программирования систем управления базами данных CLIPPER (хотя бы его «минимальной версии»). На первый взгляд это предложение может показаться чересчур смелым, учитывая дефицит свободной памяти на БК и объем транслятора CLIPPER на IBM-совместимых ПЭВМ, но сегодня даже для БК-0010(.01) существует достаточно мощный Паскаль (в том числе «Турбо-»), Пролог, текстовый редактор Votex с неограниченным размером файла и другие, не худшие, чем на IBM, программы. Так что можно надеяться, что среди БКманов найдется разработчик, который на вопрос: «CLIPPER на БК — неужели это возможно?» ответит: «А почему бы и нет!»





Когда у вас имеется всего две-три дискеты с программами, их содержимое нетрудно запомнить. Но если их накопилось два-три десятка, для поиска нужного файла приходится иной раз просматривать все диски подряд. Чтобы облегчить жизнь пользователям, минскими программистами создана специальная программа — каталогизатор дискет, которая и описывается в данной статье.

**М. И. Кондратович,**

г. Минск

## Суперкаталогизатор дискет для ANDOS

Попытки написания программ, облегчающих поиск файлов на дисках и магнитофонных кассетах, — так называемых каталогизаторов — далеко не нов. Так, в первом номере журнала «Персональный компьютер БК-0010 — БК-0011М» за 1994 г. была опубликована статья, в которой В. П. Юров описывает каталогизатор файлов ДИСКАТ2.МВИ. До появления ANDOS v3.1 он был действительно самым лучшим и универсальным на БК, но не лишен и некоторых недостатков:

- не универсальный драйвер принтера (например, D100E/PC автора «не хочет» работать с ним);
- не поддерживает работу с датой создания файла;
- подкаталоги индицируются как обычные файлы.

Разумеется, все это мелочи, но достаточно неприятные.

После выхода в свет СУБД BASIS v2.10 появилась другая разработка — CATALOG.M, маленькая утилита, заносящая каталог дискеты ANDOS в базу данных в формате BASIS. Но при этом проставляются лишь номер диска, имена файлов, даты их создания и примечания.

Когда выяснилось, что ни один из существующих каталогизаторов не удовлетворяет требованиям автора этой статьи, появилась мысль написать собственную программу. За образец был взят каталогизатор дискет NORD INFORMER для ОС NORD (автор В. Кобяков, Москва). Суть его алгоритма заключается в том, что каталоги дисков копируются в том виде, в каком они есть, в хранилище — специально подготовленный файл большого объема. С хранилищем можно работать как с базой данных, а также восстанавливать каталоги дисков при повреждении содержимого нулевой дорожки.

В соответствии с этими принципами и был написан суперкаталогизатор ANDOS INFORMER. Его первая версия для ANDOS v2.50\$ появилась еще в августе 1994 г. и уже тогда поддерживала работу с датой (автором этой статьи была разработана программа, обрабатывающая ANDOS v2.50\$ так, чтобы при записи файла в каталоге помечалась дата.)

Недостатками первой версии были необходимость наличия 16 кб дополнительного ОЗУ на БК-0010(.01) и невозможность работы на БК-0011(М).

В январе 1995 г. закончена вторая версия программы, полностью ориентированная на ANDOS v 3.1 и сейчас представляемая вниманию читателей. Автор не ставил целью обеспечить ее многофункциональность и универсальность, но попытался сделать каталогизацию максимально удобной для пользователя.

Рассмотрим программу ANDOS INFORMER подробнее. В ее основу лег тот же принцип, что и в NORD INFORMER: сохранение каталога и FAT в стандартном виде с возможностью обратного восстановления. Программа работает в ОС ANDOS v3.1 на БК-0011(М) и на БК-0010(.01) с ДОЗУ не менее 8 кб. После запуска на экран выводится заставка и INFORMER ждет нажатия на любую клавишу. Затем экран очищается, программа проверяет, архивный ли диск находится в дисковом, с которого она была загружена, и, если нет, требует установить его. Если всё в порядке, загружаются оверлей и выводится основное меню (рис. 1). Если оверлей на диске не обнаружены, система без какого-либо перезапроса выходит в ОС.

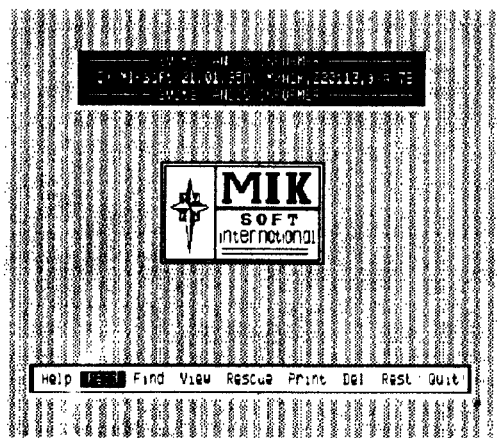


Рис. 1

На архивном диске должны находиться сама система ANDOS INFORMER, ее оверлеи и хранилище — файл объемом 385 кластеров (770 кб), в который можно записать 128 директорий. (Сомневаюсь, чтобы даже у самого отъявленного хакера дискет было больше.) Первые четыре блока этого файла — каталог содержимого хранилища, где отмечается порядковый номер сохраняемого директория, присваиваемое ему при сохранении имя и дата сохранения. В комплект поставки входит программа для создания архивного диска — MOUNT2.SYS.

Выбор режимов работы каталогизатора происходит в горизонтальном основном меню перемещением инверсного указателя. В программе предусмотрена развернутая подсказка по режимам и используемым в них клавишам. На BK-0010.01 оверлей HELP каждый раз подгружается с диска, а на BK-0011(M) он перебрасывается в шестую страницу ОЗУ, откуда вызывается при необходимости.

При считывании директория система поддерживает все возможные устройства ANDOS от A до Z, для выбора нужного устройства требуется нажать соответствующую буквенную клавишу «А» — «Z». После считывания можно ввести собственное имя дискеты (до 12 символов) и записать директорий в хранилище. Перед записью система всегда проверяет, архивный ли диск установлен в дисководе, и, если нет, требует установить его. Так что при работе на одном дисководе сменить диск никогда не поздно.

Поиск файла в хранилище начинается с выбора диска, с которого начнется поиск. В левой части экрана выводится каталог хранилища, перемещая по нему инверсную строку, вы выбираете имя диска. Это удобно, когда точно известно, что на первых по счету в списке дисках искомого файла нет. Затем вводится модель искомого файла (рис. 2). При редактировании модели с помощью стрелок вправо-влево выбирается редактируемый пункт. Искать файл можно по имени, расши-

рению, адресу, дате или длине. Пробел в имени и расширении соответствует любому символу, в адресе, дате и длине ту же роль исполняет ноль (т. е. модели адреса «1000» соответствуют файлы с адресом от 1000 до 1777, а если вас интересуют все файлы, записанные в 1995 году, установите дату 00-00-95).

При поиске содержащиеся в хранилище директории начиная с выбранного один за другим подгружаются в буфер. При обнаружении файлов, соответствующих заданной модели, их имена (до четырех) выводятся в окне оперативных сообщений. В это время поиск можно приостановить, нажав клавишу «ПРОБЕЛ» (рис. 3).

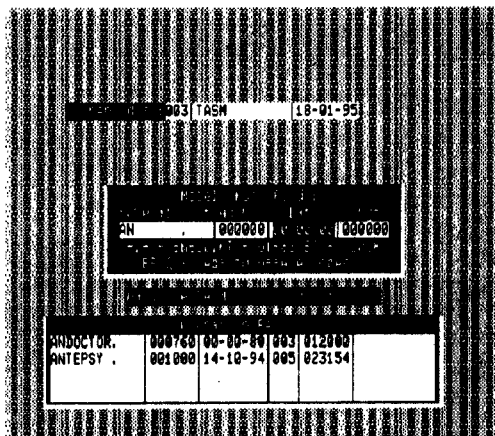


Рис. 3

При просмотре содержимого выбранного директория он выбирается из списка — каталога хранилища. Если перед этим производился поиск и были обнаружены файлы, соответствующие модели, то в правой части экрана выводится окно с номерами директориев, содержащих эти файлы (рис. 4). Выбранный из списка директорий загружается в буфер (если он уже загружен, повторное его считывание не производится).

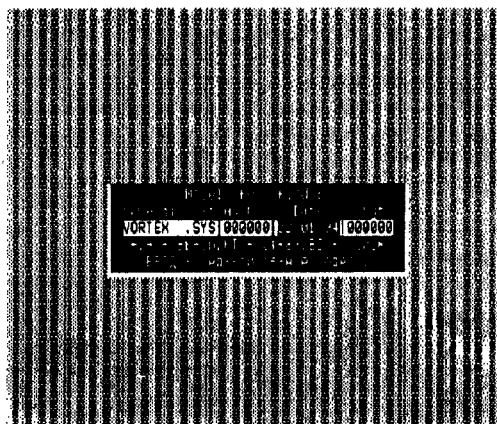


Рис. 2

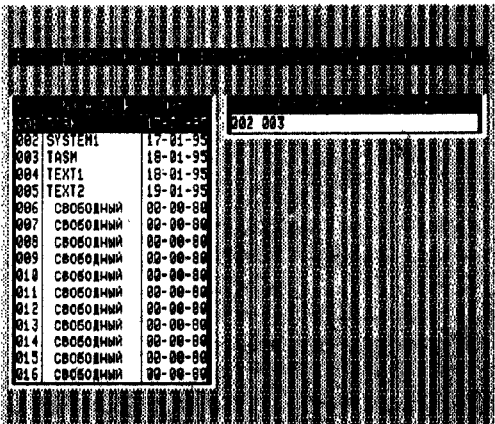


Рис. 4



EDML3EM.	001000	18-07-94	004	017150
EDFAT4	001000	00-00-80	003	013040
EDEPS1	001000	00-00-80	003	012474
EDEPS2	001000	00-00-80	003	012476

Рис. 5

Так же как и при поиске, можно ввести модель просмотра. (Для показа директория целиком нужно нажать клавишу «КТ».) Модель индицируется вверху экрана (рис. 5).

ANDOS	SYS	177777	12-08-94	016	077777
УТИЛИТЫ				DIR	
М-ФАЙЛЫ				DIR	
Д-ДИСК				DIR	
DESS				DIR	
ПРОГРАММЫ				DIR	
БАЗИС				DIR	
INFORMAT.	001000	12-08-94	003	011364	
ANSETUP	001000	12-08-94	001	003660	
COMTXT	001000	25-07-94	001	003160	
ANKY	001000	20-05-94	002	005134	
LANERASE	001000	31-07-94	001	002274	
HELP	030000	26-07-94	002	007273	
USER	030000	02-06-94	001	004000	
VIEW	030000	31-07-94	001	003232	

Рис. 6

Для каждого файла выводится имя, расширение, адрес, объем в кластерах, дата создания, длина и резервная область. (Индикация резервной области осталась от первой версии программы. Автор предполагал размещать там тексты комментариев и даже написал специальный редактор каталога — EDKAT.) Подкатало-

002	SYSTEM	17-01-95		
003	TASH	18-01-95		
004	TEXT1	18-01-95		
005	TEXT2	19-01-95		
006	СВОБОДНЫЙ	00-00-00		
007	СВОБОДНЫЙ	00-00-00		
008	СВОБОДНЫЙ	00-00-00		
009	СВОБОДНЫЙ	00-00-00		
010	СВОБОДНЫЙ	00-00-00		
011	СВОБОДНЫЙ	00-00-00		
012	СВОБОДНЫЙ	00-00-00		
013	СВОБОДНЫЙ	00-00-00		
014	СВОБОДНЫЙ	00-00-00		
015	СВОБОДНЫЙ	00-00-00		
016	СВОБОДНЫЙ	00-00-00		

Рис. 7

ги выделяются меткой «DIR» в графе «С1.» (рис. 6). Удаленные файлы отображаются жирными буквами. При просмотре можно листать содержимое директория постранично.

Особо следует отметить тот факт, что система может восстанавливать директории обратно на диски (рис. 7). Эта функция крайне полезна, если каталог или FAT на диске безвозвратно поврежден. (Собственно, ради нее и горюхился весь огородец.) Поддерживаются все устройства ANDOS. Перед записью директория на диск производится запрос подтверждения.

Чтобы восстановить каталог и FAT, вначале необходимо заново отформатировать нулевую дорожку, например, в каком-либо дисководчике. Так как загрузчик в хранилище не записывается, дискету нужно инициализировать, затем запустить каталогизатор и воспользоваться пунктом меню «Rescue».

Выбранный директорий можно распечатать посредством системного драйвера принтера ANDOS. Печать производится в одну колеску, в начале списка выводится имя директория.

Неужный директорий можно удалить из хранилища, при этом ему присваивается статус «удаленный». Если же вы по ошибке удалили не тот директорий, его можно тут же восстановить, выбрав его в списке и вновь введя имя.

Для приобретения каталогизатора ANDOS INFORMER, а также других разработок мшинских программистов можно обращаться в редакцию журнала по телефону: (095) 151-19-40.



# Нам пишут

**А. В. Бриндеев,**

Москва

## Доработка программ для дискового БЕЙСИКа

Эта статья предназначена для владельцев дисководов с контроллером, оснащенным дисковым БЕЙСИКом по стандарту «Альтек», или аналогичным, работающим с операционной системой ANDOS 3.1. В состав прилагаемого к ANDOS комплекта утилит входит программа DBASIC, переключающая триггер в контроллере дисковода и перехватывающая файловые обращения БЕЙСИКа. Основное правило, которое нужно соблюдать при работе с DBASIC, состоит в следующем: программа пользователя не должна переписывать вектор системных прерываний EMT (ячейка @#30).

В журнале «Информатика и образование», №4 за 1993 г., была опубликована статья Д. Ю. Усенкова «Реализация многоцветной закрашки на БЕЙСИКЕ», с ассемблерным листингом подпрограммы 16PAIN.BIN для расширения возможностей вильнюсского БЕЙСИКа. К сожалению, программа оказалась работоспособной только в магнитофонном варианте. После некоторых изысканий выяснилось, что 16PAIN.BIN после небольшой доработки может работать и с DBASIC. Для этого нужно изменить в листинге на ассемблере несколько строк с помощью транслятора M18.K. Ниже приводится исправленный фрагмент:

```
.....  
RTI  
A1: JMP @#1002  
ERR: JMP @#146404  
.....
```

После этого программу нужно оттранслировать и скомпоновать, как было рекомендовано в журнале «Информатика и образование», — с адреса #37300.

В том же четвертом выпуске «ИНФО» была напечатана и статья И. А. Сапегина «О приручении клавиши "СТОП"», которая также отказывалась работать в дисковой версии БЕЙСИКа. Немного покопавшись, удалось решить и эту задачу:

```
10 FOR I=PEEK(1044)+4 TO 16384 ST 6  
20 IF ABS(65536*(SGN(PEEK(I))-1))/2+PEEK(I)=<A>THP015876,PEEK(I-4)ELNEXT I  
30 POKE 15874,5572  
40 POKE 15878,92  
50 POKE 4,15874  
60 REM ...ВАША ПРОГРАММА...
```

Проблема заключалась в том, что заносимая в ОЗУ в строках 30 и 40 машинная подпрограмма портила коды DBASIC, следовательно, нормальная работа с дисководом была невозможной.



Р. Аскеров,

г. Брянск

## Условные переходы в ассемблере

Огромное количество ошибок в ассемблерных программах возникает из-за неверно написанных условных переходов. Эти ошибки, как правило, редко обнаруживаются при тестировании, поскольку далеко не всякую программу можно проверить по всем возможным сценариям работы.

Основной и наиболее распространенной причиной этих ошибок является неправильное понимание ассемблерных мнемоник. Обычно считается, что переход по команде BPL (Branch if Plus) происходит только при положительном значении проверяемого числа. И лишь при отладке одной из программ обнаружилось, что это далеко не так! Во избежание дальнейших заблуждений были составлены таблицы «на все случаи жизни», предлагаемые вашему вниманию в данной статье.

Ветвление	Условие	Примеры аргументов команды CMP и установленные биты CCF										Примеры аргументов команды TST			
		1,2 NC	2,2 Z	2,1	-2,-1 NC	-1,-1 Z	-1,-2	1,-1 C	-1,1 N	-77777,1 V	1,-77777 NVC	-1 (N)	0 (Z)	1	
BNE	$\neq X$	+	-	+	+	-	+	+	+	+	+	+	+	-	+
BEQ	$= X$	-	+	-	-	+	-	-	-	-	-	-	-	+	-
BPL	$\geq 3$	-	+	+	-	+	+	+	-	+	-	-	-	+	+
BMI	$< 3$	+	-	-	+	-	-	-	+	-	+	+	+	-	-
BCC	$\geq 6$	-	+	+	-	+	+	-	+	+	+	-	-	+	+
BCS	$< 6$	+	-	-	+	-	-	+	-	-	-	+	+	-	-
BVC		+	+	+	+	+	+	+	+	-	-	-	-	+	+
BVS		-	-	-	-	-	-	-	-	+	+	+	+	-	-
BGE	$\geq 3$	-	+	+	-	+	+	+	-	-	+	+	+	-	+
BLT	$< 3$	+	-	-	+	-	-	-	+	+	-	-	-	+	-
BGT	$> 3$	-	-	+	-	-	+	+	-	-	+	+	+	-	+
BLE	$\leq 3$	+	+	-	+	+	-	-	+	+	+	-	-	+	+
BHI	$> 6$	-	-	+	-	-	+	-	+	+	+	-	-	+	+
BLOS	$\leq 6$	+	+	-	+	+	-	+	-	-	+	+	+	-	-

### Примечания.

1. Команды BCC и BCS эквивалентны соответственно BHIS и BLO.
2. Знаковое число -1 соответствует беззнаковому 177777, -2 — 177776, ... -77777 — 100000.
3. Обозначения условий ветвления (вторая графа таблицы):  
6 — условие для беззнаковых величин,  
3 — условие для знаковых величин,  
X — условие не зависит от способа представления значений.
4. Бит V устанавливается в единицу, когда аргументы команды CMP имеют разные знаки, а результат проверки — тот же знак, что и второй аргумент.
5. Плюс на пересечении строки и столбца таблицы означает, что при указанных в столбце аргументах записанный в соответствующей строке оператор вызовет условный переход, минус — не вызовет.

(Данная информация редакцией не проверялась.)



## Виновник найден!

В опубликованном в №3 за 1994 г. обзоре дисковых операционных систем В. П. Юрсоватов на то, что какая-то дисковая программа (в качестве «подозреваемых» им названы операционные системы MDOS, NORTON, NORD, MKDOS и копировщики) подпортила дискеты, из-за чего перестали запускаться файлы с нечетной длиной. Установить «лично преступника» В. П. Юрсовату не удалось, поэтому вышеперечисленные программные средства так и остались в числе «обвиняемых».

На самом деле «виновник происшествия» многим известен точно: это NORTON v1. (ПЗУ-версия) или аналогичной ему. Причиной повреждения каталогов дискет является э. ментарная ошибка автора этой операционной системы, который не учел, что драйвер магнитофона может обращаться к ячейкам ОЗУ по адресам с точностью до байта, а драйвер НГМД только с точностью до двухбайтного слова. Поэтому при копировании средствами данной (файлов с нечетной длиной происходил их сдвиг на один байт. Так что с остальных «подозреваемых» (OS MDOS, MKDOS и NORD) обвинение можно снять.

Адрес	Было	Заменить на
122362	5828	3628
122438	148	3536
126166	48484	8
126178	152327	4767
126172	78157	174376
126174	35331	42737
126176	53448	1
126288	51581	372
126282	61448	287
126284	71157	4767
126286	67568	1174
126218	68562	42737
126212	64564	1
126214	67157	372
126216	23848	287

Для исправления этой ошибки в версии 1.05 нужно изменить в теле операционной системы несколько кодов (имеется в виду прошивка в ПЗУ, как это было сделано изначально, а отдельная программа — программа прошивки, загружаемая в расширенное ОЗУ, как это делается чаще всего).

Ошибка устранена простейшим способом введением дополнительных подпрограмм коррекции вычисляемого адреса начала файла до четного. Эти подпрограммы (причине дефицита свободного места) пришлось разместить взамен текстового сообщения об авторе операционной системы вывод которого на экран после доработки заблокирован. Автор, однако, сам виноват надо было внимательнее отлаживать со творение.

Расследование провел Ю. А. Зальцман.

В первом выпуске журнала «Персональный компьютер БК-0010 — БК-0011М» за 1995 г. была опубликована статья С. П. Чабана «Чтобы зазвучал БК-0011». Предложенный в ней способ модификации программ интересен, но, увы, применим далеко не всегда: слишком уж сложится генерируется звук во многих современных программных разработках. Автор данной статьи предлагает более радикальный метод — простейшую аппаратную доработку схемы генерации звука на БК-0011(М).

**В. Г. Шаклеин,**

г. Новгород

## Доработка канала звука в БК-0011

Известно, что программы для БК-0010 «молчат» на БК-0011. Это связано с тем, что в одиннадцатой модели при обращении к регистру 177716 для генерации звука обязательно должен быть установлен бит 11. Можно, конечно, переделать звуковые константы программ, но тогда они не будут звучать на БК-0011М, так как там при обращении к регистру 0#17771 бит 11 должен быть обязательно сброшен.

Предлагаемая вниманию читателей небольшая аппаратная доработка БК обеспечит нормальное звучание на ней всех программ, как устанавливающих бит 11, так и не делающих этого (т. е. звук будет формироваться точно так же, как на БК-0010). Вскройте корпус БК-0011(М), найдите в нем микросхему D13, освободите ее вывод 9 от старых связей и соедините с выводом 8 микросхемы D7.

Единственный недостаток такой доработки — щелчки реле при «стандартных» переключениях страниц в некоторых программах. (Для их устранения заносимые в регистр Ф#177716 константы надо увеличивать на 200, например вместо 16000 загружать туда код 16200 и т. п. На БК-0011М и тем более на БК-0010 это никаких отрицательных последствий не вызывает.) Однако, учитывая, что на БК-0011 пользоваться магнитофоном приходится редко, реле можно и отключить совсем.



**Д. Р. Насыров,**

*г. Казань-73*

## «Секретные» пароли в играх для БК

Для удобства отладки разработчики компьютерных игр часто предусматривают в своих творениях отладочные пароли, набрав которые можно получить вечную жизнь, оружие и т. д. Немного покопавшись в программах, автор данной заметки отыскал такие пароли для ряда БКшных игр.

**CHRONOS:** если набрать в режиме управления слово **MEGA+**, не придется постоянно держать нажатой кнопку «FIRE».

**EXPERT:** набрав в основном меню **DIM&FB**, можно стать бессмертным.

**SAVAGE:** воскреснуть три раза орду поможет набранный в режиме управления (кнопка «Ц») пароль **SNAP**.

**TRANTOR:** чтобы ваш боец смог три раза воскреснуть, в режиме управления (кнопка «Ц») нужно набрать пароль **ЛЕНИН**, а для получения защиты — **ЛЕНИП** (в том же режиме).

**BREAKING BALL:** для входа в редактор стен во время игры нужно набрать (в латинском регистре) **R** и затем **ASHOKA**.

**SUPERMAN:** чтобы сразу увидеть концовку игры, выйдите из нее в монитор и наберите **S36312**.

**XENON:** для входа в игру нужно набрать слово **MISLED**, а для получения вечной жизни — во время игры ввести **BROKEN**. В режиме управления предусмотрен также пароль **TECNICAL**, сразу же после которого нажимается одна из следующих букв — **A** (увеличение количества призов), **C** (очистка табло рекордов), **S** (смещение табло рекордов), **N** (нормальный режим игры) или **E** (выход).

**HORROR SHOW:** для входа в редактор в меню нужно выделить пункт **EDIT** и ввести пару цифр — **9** и **1**.

### \*\*\* Внимание! Опечатка \*\*\*

В конце листинга программы **DBAS** (Дисковый БЕЙСИК) в статье А. В. Милюкова (№2 за 1995 г., с. 71) ошибочно-продублирована строка с меткой **T**. Кроме того, четвертый, считая от конца листинга, оператор **MOV** оказался «разрезанным» на две части. Правильный фрагмент ассемблерного текста программы **DBAS** должен быть таким:

```
; отключить ПЗУ
MOV #148228,Ф#177716
; старый вектор
T:   MOV #188112,Ф#38
      EMT 36
      BR STOP
EMU: JSR R7,Ф#116876
; =EMT 36
```

```
STOP: MOV #1888,R6
      MOV #14,Ф#177138
; подключить ПЗУ
      MOV #48228,Ф#177716
SET:  MOV #1888,Ф#4
      BR S
FLAG: . #8
```

Приносим читателям свои извинения.



# ГЛЮКАДЕМИЯ

Я верю в интуицию и воображение... Воображение важнее знания, ибо знание ограничено, воображение же охватывает всё на свете...

А. Эйнштейн

## Доработка драйвера принтера в БК-РАИТ

Многим пользователям БК хорошо известен графический редактор БК-РАИТ (созданный в 1989 г. А. В. Бакериным). Эта программа очень удобна для рисования различных картинок, особенно для начинающего пользователя. Меню-пиктограммы операций и выпадающие меню делают процесс рисования наглядным и удобным.

Но эта программа имеет и значительные недостатки. Во-первых, ее клавиатурный вариант не очень хорошо работает на некоторых экземплярах БК-0010.01 (при нажатии и удержании клавиши-стрелки курсор вместо непрерывного движения, пока клавиша не будет отпущена, рывком сдвигается на некоторое расстояние и останавливается, так что для продолжения движения надо отпустить клавишу и нажать ее снова). Поэтому чаще пользуются модификациями программы с управлением от джойстика и от мыши. (Работать с мышью особенно удобно.) Во-вторых, БК-РАИТ работает только с магнитофоном и при записи «рубит» файл на пять частей. Правда, программа при этом упаковывает эти части в самом настоящем IBM-овском формате РСХ. Но для переноса полученной картинке в другие графические программы такой формат очень неудобен и к тому же непригоден для многих дисковых систем. И в-третьих, программа (в том числе распространяемый в комплекте КУВТ-86 вариант РАИТ.BIN, имеющий управление и от мыши, и от клавиатуры одновременно) содержит не очень удобный драйвер печати на принтер. Ради того чтобы не преобразовывать колонки по 8 точек в байты для вывода на принтер, как это делается обычно, картинка при печати поворачивается на 90°. В цикле с экрана считываются байты (начиная с верхнего правого края экрана до нижнего левого угла картинке, столбец за столбцом шириной в 8 точек). В результате такого огрубления мы выигрываем в объеме, занимаемом драйвером в памяти, но проигрываем в качестве изображения — оно вытягивается по вертикали по сравнению с экраном. Более того, драйвер печати изначально ориентирован на принтер типа robotron CM 6329 [1], а не на популярные среди БКманов МС6312 и МС6313. По этой причине (а может быть, и просто из-за ошибки разработчика) происходят дополнительные искажения при печати: когда принтер выводит очередной столбец в 8 точек и, выполнив перевод строки, начинает печатать следующий, две соседние полосы картинки «наезжают» друг на друга. При этом наклонные линии превращаются в ломанные, а символы минимального размера (стандартного для БК режима 64 символа в строке) искажаются в зависимости от месторасположения иногда до печатаемого состояния. (Вообще говоря, этот эффект словами описать довольно трудно, но те, кто пробовал печатать из БК-РАИТа на МС6312, наверное, уже хорошо поняли смысл сказанного.)

Вот об исправлении последней ошибки и пойдет речь\*. Как показало короткое расследование, «виновной» является выдаваемая на принтер в начале печати картинке команда установки интервала между строками. В исходном варианте БК-РАИТ содержатся байты #33 и #61 (ESC <1> — установка интервала в 7/96 дюйма). Возможно, для CM 6329 это то, что нужно, но для МС6312 да и для МС6313 этого явно мало. Вот и получается, что при печати очередной полосы картинке она своим верхним (по отношению к печатающей головке принтера) краем «наезжает» на нижний край предыдущей. Из-за этого и искажаются символы и наклонные линии.

Первое, что может прийти в голову, — замена этого интервала на другой. Для МС6312 и МС6313 больше всего подходит интервал в 6/72 дюйма [2]. (Кстати, для МС6312 тоже нужно устанавливать интервал 6/72, а не 8/72, как это рекомендовано в [2]!) Но такой команды в МС6312 нет. Тогда выбираем ESC <3> (п/288 дюйма) и задаем п=30 (восемь).

Коды команды установки интервала между строками принтера записаны в программе БК-РАИТ с адреса @#22040:

\* Версии БК-РАИТ, доработанные для обеспечения совместимости с любой дисковой ОС для БК-0010.01, а также программу распаковщик пятифрагментных рисунков БК-РАИТ в обычный формат копии экрана можно приобрести в редакции. — *Прим. ред.*

```

@#22040: 030433 ; <61><33> ; <ESC>1
          000000 ; ноль - признак окончания записи

```

Заменим коды (в любом отладчике, например в DEBU10) на другие:

```

@#22040: 031433 ; <63><33> ; <ESC>3
          000030 ; 30/288 = 0/72 дюйма

```

Вот это уже «то, что доктор прописал»: теперь картинка печатается без досадных искажений. Да и общая «сплюснутость» по горизонтали полученного на бумаге изображения по сравнению с экраном несколько уменьшилась.

Для интересующихся отметим также, что с адреса @#21776 содержится подпрограмма реализации команды «Печать» в меню команд: вывод текста с просьбой подключить принтер и нажать «ВВОД» (сам этот текст находится в памяти с адреса @#22142), проверка BIT #400,@#177714 (подключен ли принтер) и собственно печать копии экрана. С адреса @#22250 размещена подпрограмма, выводящая на принтер ряд управляющих кодов. Ее вызов осуществляется цепочкой команд ассемблера:

```

JSR R5,@#22250 ; вызов подпрограммы
<код1><код2> ; строка выводимых
               ; кодов...
<ноль>        ; нулевой байт - признак окончания записи
<.....>      ; следующий оператор ассемблера

```

Программа вывода на принтер отдельного байта, содержащегося в младшем байте R0, расположена с адреса @#22272.

С адреса @#22070 записана (в таком же формате, как и рассмотренная нами команда ESC<1>) команда перевода строки — два кода LF и CR (восьмеричные числа 12 и 15, соответственно) и команда включения графического режима: ESC L 144 2 (двойная плотность печати с низкой скоростью, 120 точек на дюйм). А по адресу @#22130 находится пара команд:

```

MOV #14,R0
JSR R7,@#22272.

```

которая обеспечивает перевод формата (выброс отпечатанного листа из принтера). Если вам надоел перевод формата после печати каждой копии (например, при работе с бумажной лентой), нужно заменить две указанные команды четырьмя операторами NOP (по адресам от @#22130 до @#22136).

## Литература

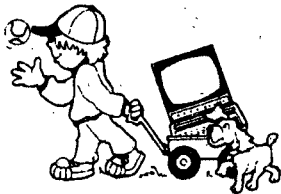
1. Вычислительная техника и ее применение. 1991. №1. С. 45.
2. Информатика и образование. 1992. №5—6. С. 62.

## Новые фактуры закраски для БК-PAINT

Как известно, графический редактор БК-PAINT (автор А. В. Бакерин) предназначен для работы с черно-белым монитором, поэтому для закраски замкнутых областей, в отличие от БЕЙСИКовского оператора PAINT, здесь используются не цвета, а фактуры (сетки из точек с различной плотностью расположения, наклонных, горизонтальных или вертикальных линий, типовых фрагментов («кирпичи», «яблоки»), а также сплошная заливка). На цветном же экране эти «псевдоцвета» смотрятся, прямо скажем, неважно. Хорошо было бы иметь в меню команду подгрузки внешних фактур (или изменения имеющихся), подобную загрузке дополнительного шрифта, но она, увы, не предусмотрена...

Впрочем, изменить фактуры все равно можно. Вся палитра (24 фактуры) располагается в теле программы по адресам 32724072 (восьм.). Каждая фактура представляет собой спрайт шириной 2 байта и высотой 8 графических строк. Зная об этом, можно заменить их на другие с помощью отладчика (например, DEBU10) или отредактировать в программе RS.V2, после чего нужно записать новую копию всей программы вместе с новыми фактурами.

# НАЧИНАЮЩИМ ПОЛЬЗОВАТЕЛЯМ



Как подключить к своей БКшке дисковод? Что для этого нужно приобрести? Какие программы требуются для работы с диском? Эти и другие подобные вопросы часто приходится слышать от многих пользователей БК, не только начинающих. Конечно, данная тема слишком обширна, чтобы надеяться сообщить всю необходимую информацию в одной журнальной статье. Ниже приводится ряд наиболее общих рекомендаций и необходимых теоретических сведений о дисковом HARDe и SOFTe. Надеемся, что они помогут читателям наконец-то покончить с вечными проблемами с магнитофоном и превратить БК в полноценную персональную ПЭВМ.

От редакции

## Подключение дисковода к БК

Преимущества, обеспечиваемые подключением к БК дисковода, вряд ли нуждаются в длительном обсуждении. Достаточно лишь кратко упомянуть наиболее существенные моменты, чтобы почти наверняка убедить читателя в полезности этого периферийного устройства.

*Надежность хранения и обмена информацией.* Многим пользователям БК (да и других домашних ЭВМ, использующих в качестве внешнего запоминающего устройства бытовой кассетный магнитофон) наверняка известно чувство досады, когда какая-либо полезная программа, важный текст или нужные данные не читаются с кассеты. Чтобы обезопасить себя от потери ценной информации, приходится записывать по две, три и более копий каждого файла, и все равно полной гарантии надежности нет. Магнитофоны нередко «зажевывают» ленту, сминая ее в гармошку или выкручивая вдоль. Особенно часто этот грех случается с советскими (а ныне российскими и СНГ'овыми) моделями магнитофонов. Импортные же не всем доступны, да и пятиштырьковые разъемы для подключения компьютера на них чаще всего отсутствуют. И наконец, кому не знакома ситуация, когда для чтения программ с чужой кассеты (купленной или взятой на время у приятеля) приходится отверткой подкручивать винт магнитной головки, на слух добиваясь наиболее четкого сигнала. А потом, чтобы записать только что считанное на свою кассету, снова браться за отвертку? Или «писать как пишется», а потом подкручивать винт для каждой программы? В результате считывание информации с магнитофонных кассет из вспомогательной операции превращается в своеобразное искусство.

(Примечание для тех, кто пока еще мучается с магнитофоном. Если несмотря на все манипуляции отверткой звук все равно остается глухим, а сигнал — слишком слабым для БК, внимательно осмотрите «рабочий» торец кассеты, где магнитная лента выведена наружу. В его середине установлена подпорная металлическая пластинка с закрепленным на ней кусочком войлока. Часто на новых кассетах эта пластинка совершенно плоская, а войлок даже не касается ленты. В таком случае нужно осторожно подогнуть пластинку вперед кончиком узкого ножа или динцетом, чтобы она при чтении прижимала ленту к поверхности магнитной головки. Делать это нужно очень аккуратно, при излишнем усилии вставленные в прорези корпуса загнутые «ушки» подпорной пластинки могут выскочить, и тогда кассету придется разбирать, чтобы вернуть пластинку на место. Однако «по трудам и награда»: после такой доработки информация с кассеты, как правило, читается намного лучше.)

Дискеты же, чтение и запись информации на которые обеспечивает дисковод, как носитель информации гораздо надежнее. Если дисковод в порядке, а дискета хранится в надлежащих условиях (нельзя сгибать ее, держать под прямыми солнечными лучами или рядом с магнитом, ставить на нее тяжелые предметы и хвататься за магнитный слой пальцами), потеря информации становится исключительно редким явлением. А поскольку магнитные головки дисковода надежно «спрятаны» под крышкой корпуса, их положение практически у всех пользователей стандартно, так что чужие дискеты читаются без «отверточной технологии».



*Скорость и удобство доступа к файлам.* При работе с магнитофоном, чтобы загрузить требуемый файл, нужно точно знать, в каком месте кассеты он записан (для чего магнитофон должен быть оборудован счетчиком расхода ленты), или искать его, «пролистывая» кассету с самого начала. И конечно, обязательно нужен каталог содержимого каждой кассеты, так как единственный способ узнать, что на ней записано, — это просмотр всех 60, а то и 90 минут магнитной ленты.

На дискете же вся информация записывается в виде концентрических круговых дорожек: покрытый магнитным слоем кружок из гибкой пластиковой пленки, заваренный в прямоугольный защитный конверт с прорезью для доступа к магнитному слою, быстро вращается, а две магнитные головки (с двух сторон дискеты) перемещаются шаговым двигателем по радиусу вдоль прорези конверта. Начало каждой дорожки компьютер (точнее, подключаемый к нему специальный блок сопряжения с дисководом — контроллер) определяет автоматически, а список всех хранящихся на дискете файлов находится в специально выделенной на ней области — каталоге. При записи или стирании файлов каталог автоматически обновляется, а при установке в дисковод новой дискеты ее каталог первым делом считывается компьютером и демонстрируется на экране. Так что, если вы забыли, где находится требуемый файл, не проблема быстро просмотреть имеющиеся дискеты и найти его в выданном на экран списке.

Сами операции чтения и записи выполняются (по сравнению с магнитофонным обменом) почти мгновенно: загрузка файла длиной 4000b (например, копии экранного ОЗУ) с диска занимает 2—3 секунды. Не намного дольше производится и запись такого же объема информации.

*Работа с многофайловыми программами.* Перечисленные выше возможности дискового обмена особенно удобны при использовании многофайловых программ, подгружающих в процессе исполнения те или иные рабочие фрагменты (оверлеи). Именно так работают большинство современных игр для БК и многие прикладные программы. Если при работе с магнитофоном приходилось вручную подматывать ленту к началу каждого оверлея и долго ждать, пока он загрузится (а при ошибке чтения — начинать все с самого первого файла), то с диска многофайловые программы загружаются так же удобно и быстро, как однофайловые. Кстати, практически никакой доработки имеющихся на кассетах программ при этом

не требуется, достаточно лишь переписать их на дискету с помощью копировщика.

*Возможность удаления файлов.* Представьте себе, что какая-то программа перестала быть нужной. Раньше такие файлы бесполезно занимали место на кассете, так как записать поверх них что-то другое без риска испортить предыдущий и последующий файлы не удаётся. На дискете же в любой момент можно удалить ненужный файл, при этом занимаемое им место освобождается для чего-то другого. Возможна и замена уже существующего файла новой версией (скажем, при сохранении временных копий текста в процессе набора), причем не требуется создавать промежуточные копии (это особенно оценят программисты, которым приходится многократно дорабатывать и отлаживать свои творения).

*Совместимость с другими ПЭВМ.* Многие пользователи БК, наверное, хоть раз да «пустили слюнки» при виде мощных IBM-совместимых компьютеров на выставке или в витрине магазина. Как было бы приятно иметь такую машину дома! Удобный и простой в освоении файловый интерфейс Norton Commander, возможность работы с текстами без ограничения их длины, отсутствие мучений с магнитофоном... Однако, не умаляя всех достоинств «профессионала», следует заметить, что возможности БК с подключаемым дисководом (а тем более винчестером) не намного слабее. О надежности дискового обмена уже сказано выше, а специальные программы — дисковые операционные системы, обеспечивающие этот обмен, снабжены интерфейсом, почти неотличимым от Nortonовского: те же файловые панели, встроенные функции копирования, переименования и удаления файлов, операции листания и сортировки каталогов, просмотра текстов. А если у вас на работе имеется IBM, можно переносить с нее на БК понравившиеся тексты (для БК с дисководом уже создан ряд программ для просмотра и редактирования текстов любой длины), а иногда и графику без особой потери качества. Или, наоборот, можно набирать тексты или листинги программ, рисовать картинки на домашнем компьютере, а затем переносить их на IBM.

После всего сказанного большинство читателей наверняка решили установить на БК дисковод. Но тут же возникает множество вопросов: что для этого нужно купить, как выбрать конкретную модель дисковода и подключить его к БК? Постараемся в этой статье хотя бы кратко ответить на них.

Для начала перечислим все, что нужно иметь для работы на БК с дисководом (кроме, разумеется, самого БК):

- дискковод (один или два, а в некоторых случаях до четырех);
- контроллер дисковода (его официальное наименование: КНГМД — контроллер накопителя на гибких магнитных дисках), для БК-0010(.01) желательнее, чтобы в КНГМД было установлено дополнительное ОЗУ емкостью не менее 16 кб;
- блок питания дисковода (питание контроллера осуществляется за счет компьютера);
- кабель питания, соединяющий дискковод с его блоком питания (нужен специальный разъем для подключения этого кабеля к

дисководу, обычно он имеется в комплекте дисковода);

- многожильный плоский кабель («шлейф») для передачи информации и разъемы для его подключения к КНГМД и дисководу (покупаются отдельно);
- дискровая операционная система — специальная программа, обеспечивающая работу с дисководом, и ряд дополнительных служебных программ — утилит;
- дискеты.

А теперь обсудим каждую из перечисленных комплектующих более подробно и дадим общие рекомендации по их выбору.

## Дискководы

В настоящее время широко распространены дискководы двух типоразмеров: 5.25- и 3.5-дюймовые. Последние, хотя и привлекают своей малогабаритностью и надежностью дискет, защищенных от механических воздействий жестким пластмассовым корпусом, на БК используются мало: в нашей стране «трехдюймовки» почти не выпускаются, а импортные не всем доступны по цене. «Пятидюймовые» дискководы, напротив, в последнее время гораздо чаще можно видеть на БК и других отечественных малых компьютерах, чем на IBM-совместимых ПЭВМ. Выпускаемые в России модели при не очень высокой цене достаточно надежны и не предъявляют слишком больших требований к условиям эксплуатации.

Дискководы 5.25" бывают одно- и двухсторонние (в последнее время односторонние модели практически не используются), а также 40- и 80-дорожечные. Большинство из них можно без особых проблем подключить к БК, но наиболее удобно использовать 80-дорожечные двухсторонние дискководы, обеспечивающие информационную емкость дискет 800 кб. Возможность же подключения к БК мегабайтных дискководов (1.2 Мб для 5.25-дюймовых и 1.4 Мб для 3.5-дюймовых) связана с определенными трудностями, так как стандартные контроллеры дисководов для БК такую емкость не поддерживают.

Сегодня в продаже можно встретить следующие модели «пятидюймовых» дискководов:

- **MC5301:** двухсторонний 40-дорожечный дискковод с «широкой» передней панелью (размер 150×85 мм; большинство других моделей имеют «узкую» переднюю панель 150×42 мм), скорее являющийся «анти-

кварной редкостью», нежели действительно популярный среди БКманов;

- **MC5305 и MC5311:** «узкие» 80-дорожечные двухсторонние дискководы, различающиеся только заводами-изготовителями и практически одинаковые по принципиальной схеме и рабочим характеристикам. По общепринятому мнению, MC5311 более надежен, чем MC5305\*;
- **MC5313:** также двухсторонний 80-дорожечный, но, в отличие от MC5305 и MC5311, не имеет электромагнита поднятия-опускания магнитных головок (это отличие будет подробно рассмотрено ниже);
- **MC5323 и MC5321:** болгарские дискководы, ранее устанавливавшиеся почти на всех выпускавшихся в СССР и странах соцлагеря IBM-подобных компьютерах (ЕС1840, ЕС1841, «Нейрон», «Искра» и т. п.), однако крайне ненадежные.

Следует особо отметить, что большинство из названных моделей в настоящее время сняты с производства (или выпускаются в ограниченных количествах), поэтому практически все имеющиеся в продаже дискководы — б/у. Тем не менее не стоит особо смущаться этим фактом: изделия (по крайней мере имеющиеся в продаже при редакции нашего журнала) тщательно проверяются, а при выходе из строя в течение гарантийного срока — заменяются на новые.

Из всех перечисленных «пятидюймовых» дискководов для подключения к БК с наибольшей уверенностью можно рекомендовать MC5313. Он максимально надежен, кроме того, отсутствие электромагнита поднятия/опуска-

\* Впрочем, это относится лишь к модели MC5305 производства Армении. Российские дискководы с тем же названием показали себя достаточно надежными при сравнительно низкой стоимости.

ния головок создает более щадящие условия эксплуатации дискет, чем на МС5305 и МС5311.

Поясним это важное отличие более подробно. В дисководах пятой и одиннадцатой моделей магнитные головки даже при установленной дискете и закрытом «флажке» (рукоятке, запирающей щель дисковода после установки диска) отжаты от поверхности дискеты упругой металлической пластиной (пружиной поднятия). Когда же нужно произвести чтение информации с диска или запись на нее, специальный электромагнит прижимает головки к поверхности дискеты, для чего одновременно с сигналами чтения или записи на него подается управляющий импульс. Предполагалось, что такое решение повысит долговечность дискеты (головки не «трутся» о магнитный слой входостую). На практике же при каждом срабатывании электромагнита головки ударяют по дискете с такой силой, что иной раз выбивают из поверхности частицы магнитного слоя. А на модели МС5313 магнитные головки опускаются на поверхность дискеты одновременно с закрытием «флажка» и до его открывания постоянно скользят по поверхности диска (соответственно, форма самих головок более обтекаемая и гладкая, чем на МС5305 и МС5311), что позволяет избежать их биения по магнитному слою.

**(Примечание.** «Народные умельцы» часто дорабатывают дисководы МС5305 и МС5311, удаляя поднимающую головки пружину и та-

ким образом ликвидируя их биения о поверхность диска. Но делать это **НЕ РЕКОМЕНДУЕТСЯ**, так как нескругленные края установленных на этих моделях магнитных головок царапают поверхность дискет, постепенно выводя их из строя.)

В приобретаемом дисковом месте вместо дискеты может быть установлен транспортировочный картонный вкладыш, предохраняющий магнитные головки от вибрации. При работе этот вкладыш вынимается, если же требуется перевести дисковод в другое место, вкладыш нужно вставить снова и закрыть рукоятку замка.

Напоследок отметим еще одну особенность. Как правило, дисководы поступают в продажу без наружного корпуса, его нужно приобрести отдельно. (Дисковод крепится в корпусе на четырех винтах, ввинчиваемых в резьбовые отверстия металлической рамы.) Если вы хотите подключить два дисковода, можно купить общий корпус под них. Другой вариант — пара дисководов МС5305, смонтированных в одном корпусе вместе с блоком питания и имеющих торговое наименование МС5310.

**(Примечание.** Некоторые владельцы БК используют дисковод без корпуса, заменяя его упаковочным картонным футляром. Это в общем-то допустимо (хотя и не слишком удобно), нужно лишь оберегать находящиеся на печатной плате элементы от механических воздействий и следить, чтобы расположенный снизу вращающийся маховик не касался поверхности стола.)

## Контроллер дисковода (КНГМД)

Для подключения дисковода к БК необходим специальный дополнительный блок — контроллер, на который возложены функции управления обменом информацией с дисковыми накопителями. Для БК-0011 и БК-0011М имеются стандартные контроллеры, внешне похожие на блок МСТА и подключаемые к разъему системной магистрали БК (МПИ). Во многих случаях они входят в комплект поставки компьютера, так что никаких проблем с подключением не возникает. Поэтому дальнейший разговор будет касаться владельцев БК-0010 и БК-0010.01.

Удобнее всего использовать для подключения к десятой модели доработанные стандартные контроллеры от БК-0011 или БК-0011М. Подойдет и контроллер КМД-УК от УКНЦ, но из-за необходимости его значительной доработки (перепайки разъема сопряжения с компьютером, подключения внешнего тактового генератора на 4 МГц и установки ПЗУ со специальной управляющей программой —

драйвером БКшного типа) этот вариант вряд ли доступен для начинающих и здесь не обсуждается (см. №1 за 1993 г., с. 118).

Контроллеры для одиннадцатой модели БК, построенные на основе БИС 1801ВП1-128, выпускаются в двух вариантах: блок КНГМД 3.059.001 для старой модификации БК-0011 и блок КНГМД 3.059.014 для БК-0011М. Различаются они только прошитым в ПЗУ драйвером дисковода, (микросхемы ПЗУ 1801РЕ2-253 и 1801РЕ2-326 соответственно). При подключении к БК-0010(.01) предпочтительнее использовать контроллер от БК-0011М (с 326-й прошивкой), так как его драйвер позволяет загружать операционную систему автоматически. (Поскольку на БК-0011 система ЕМТ-функций монитора несовместима с применяемой на БК-0010(.01), при работе с КНГМД от БК-0011 для загрузки операционной системы на десятой модели БК требуется каждый раз вводить с клавиатуры или подгружать с магнитофона специальную «пусковую» программу.)

Для подключения контроллера от БК-0011М к БК-0010.01 требуется внести в его схему небольшие изменения: установить перемычки между контактами А19 и А2 разъема подключения КНГМД к порту МПИ БК и подпаять резистор сопротивлением 300 Ом между контактами А29 и А4 того же разъема. Резистор при подключении блока КНГМД к БК блокирует работу внутреннего ПЗУ компьютера по адресу 160000 (именно здесь располагается ПЗУ драйвера дисковода, установленное в контроллере). Аналогичная доработка контроллера нужна и при подключении к старой модели БК-0010, но в ней не предусмотрено подключение к системной магистрали каких-либо блоков (сигнал отключения внутренних ПЗУ на разъем МПИ не выведен), и поэтому необходимо внести соответствующие изменения в схему компьютера. То же нужно учитывать и при подключении обсуждаемого ниже дополнительного ОЗУ.

С доработанным указанным способом контроллером уже можно работать на БК-0010(.01) с дисководом. Но при этом возникает одно неудобство: дисковая операционная система (ОС) располагается в основном ОЗУ компьютера и любая запущенная из нее пользовательская программа неизбежно затирает собой систему. В результате всякий раз приходится снова загружать ДОС с содержащей ее специальной системной дискеты. Для выхода из создавшегося положения внутрь контроллера в первые годы «дисководного бума на БК» устанавливались еще одно ПЗУ (по адресу 120000) и дополнительные перемычки, отключающие внутреннюю микросхему памяти по тому же адресу. В результате сразу после включения питания БК при подключенном контроллере дисковая операционная система начинает работу (как ФОКАЛ или БЕЙСИК при «магнитофонной» конфигурации), не требуя установки системной дискеты. Аналогично управление сразу же передается операционной системе после выхода из пользовательской программы или перезапуска БК по кнопке RESET (либо переключателем «СТОП-ПУСК»).

Однако использование ПЗУ для хранения операционной системы имеет свои недостатки, из которых наиболее заметный — невозможность перехода на другую ОС без замены микросхемы ПЗУ. Поэтому в настоящее время внутрь контроллера принято устанавливать дополнительное энергонезависимое ОЗУ по адресам 120000—157777, «вытесняющее» из этой области адресов внутренние ПЗУ БК. Термин «энергонезависимое» означает, что при подключении батарейки на 1.5—3 В к выведенным из корпуса КНГМД проводам содержащаяся в ОЗУ информация сохраняется и после выключения питания БК. В

результате операционная система, как и в случае с ПЗУ, постоянно готова к работе, но в любой момент можно загрузить с системной дискеты другую ОС, после чего уже она будет постоянно храниться в дополнительной памяти. Большинство последних версий операционных систем занимает все дополнительное ОЗУ, при работе же со старыми версиями меньшего объема в оставшуюся свободную область памяти можно загружать для постоянной работы и другие программы (отладчик, драйвер копирования экрана в файл и т. д.).

Говоря о дополнительном ОЗУ, мы отметили, что оно при подключении контроллера к БК блокирует установленные внутри компьютера ПЗУ с ФОКАЛОМ и БЕЙСИКОМ. Так что же дисковод лишает пользователя возможности работы с этими языками? Конечно же нет! С ФОКАЛОМ дело обстоит наиболее просто: в дополнительное ОЗУ с адреса 120000 вместо ДОС временно загружается немного доработанная копия транслятора ФОКАЛА БК-0010 из ПЗУ, в которой все операторы магнитофонного обмена переадресованы на работу с дисководом. Файл «дискового ФОКАЛА» входит в комплект поставки практически любой современной операционной системы. Что же касается БЕЙСИКА, здесь необходима аппаратная доработка контроллера, сводящаяся к установке взамен блокирующей ПЗУ БК-0010.01 перемычек логической микросхемы, которая подключает на адреса 120000—177777 ПЗУ БЕЙСИКА, а для чтения-записи файлов — ПЗУ контроллера. (Эта доработка, именуемая «дисковым БЕЙСИКОМ», подробно описана в №2 за 1995 г., с. 72.) Специальная программа, управляющая этими переключениями, как и «дисковый ФОКАЛ», входит в комплект операционных систем и при работе загружается в дополнительное ОЗУ.

Все перечисленные доработки хотя и не очень сложны, но для начинающего пользователя, которому редко приходится держать в руках паяльник, могут представлять определенные трудности. Проще и удобнее приобрести готовый дисковод, в котором уже установлено дополнительное ОЗУ 16 кб (или больше, но с ростом объема ДОЗУ существенно возрастает и стоимость блока) и микросхема «дискового БЕЙСИКА». Подобную модель КНГМД вы можете купить в открытом совместными силами редакции и фирмы «АльтПро» Центре БК. (Кроме всего прочего, на контроллерах «АльтПро» установлена и кнопка RESET. Тем не менее удобнее монтировать такую кнопку не на КНГМД, а непосредственно в корпус БК, как рассказано в №1 за 1994 г.; в применении редактора на с. 71.)

## Блок питания

Для обеспечения работы двигателей и электронной схемы дисководов необходим стабилизированный блок питания, имеющий на выходе напряжения +5 и +12 В. Поскольку сбои питания почти всегда приводят к ошибкам чтения-записи, самостоятельное изготовление блока питания дисководов можно рекомендовать лишь хорошо подготовленным радиолюбителям. Лучше всего купить готовый блок, тем более что цены на них невелики по сравнению со стоимостью дисководов и контроллера. Как правило, имеющиеся в продаже дисководные блоки питания представляют собой

немного доработанные «штатные» выпрямители из комплекта БК, достаточно надежные и хорошо защищенные от помех. Существуют также «сдвоенные» блоки питания, к которым одновременно подключается БК и один дисковод, что дает возможность разместить весь полученный вычислительный комплекс более компактно.

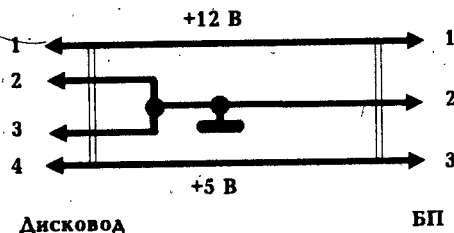
Что же касается контроллера дисководов, напряжение питания для него снимается непосредственно с контактов порта системной магистрали (МПИ), так что отдельный блок питания для КНГМД не требуется.

## Кабель питания дисководов

Подключение дисководов к блоку питания осуществляется трехпроводным кабелем (его длина не должна быть слишком велика, но и чрезмерное натяжение также нежелательно). К этому кабелю подпаиваются два разъема: для подсоединения к блоку питания (как правило, пятиштырьковая вила типа ОНЦ-ВГ-4-5/16-В, как на прочих соединительных кабелях БК) и к дисководу (специальная разборная

вила плоской формы: провода кабеля расплавляются на отдельные контактные штырьки, которые затем вставляются в корпус вилок и фиксируются в нем упругими выступами). Распайка разъема питания дисководов приведена на рисунке.

Гнездо для подключения кабеля питания к дисководу закреплено справа на заднем торце его печатной платы. (Разумеется, нужно снять заднюю крышку корпуса.) Скосы в углах вилок обеспечивают правильность ее установки, следует лишь учесть, что никаких фиксаторов вилок питания в гнезде дисководов не предусмотрено, поэтому кабель должен иметь некоторую глубину (во избежание выдергивания вилок при неумелом движении). Если же дисковод отказывается работать (при обращении к нему не включается двигатель и не загорается индикатор), нужно проверить, не выпала ли из гнезда вила питания.



## Интерфейсный (сигнальный) кабель

Передача управляющих сигналов и данных для чтения-записи от контроллера к дисководу и обратно осуществляется через 40-жильный плоский кабель (шлейф). Один его конец закрепляется в 40-контактном разъеме ОНП-КГ-56-40-Р без использования паяльника: имеющиеся на тыльной стороне разъема специальные иглы при нажатии на прижимную планку и ее фиксации прокалывают изоляцию жил кабеля и обеспечивают электрический контакт. На другой стороне шлейфа аналогичным способом крепится разъем ОНП-НГ-79-34-МЭК-О или аналогичный импортный. Он имеет лишь 34 контакта и устанавливается на шлейф так, чтобы первый контакт разъема контроллера был соединен с первым контак-

том разъема дисководов. Оставшиеся лишними шесть жил шлейфа не используются, их нужно отделить, осторожно надрезав конец шлейфа вдоль. Установку прижимной планки на разъеме (особенно зарубежного производства) нужно производить крайне осторожно: при малейшем превышении допустимого усилия хрупкие боковые щечки планки, фиксирующие ее на корпусе разъема, могут отломиться. Если же это все-таки произошло, можно попытаться закрепить планку, наложив поверх нее П-образную металлическую пластинку или кусок жесткого монтажного провода и плотно обмотав разъем с боков изоляционной лентой.

Установка разъемов не сложна даже для начинающего пользователя, но при желании

вы можете купить и готовый интерфейсный кабель. Подключить же его к контроллеру и дисководу еще проще. Разъем контроллера подсоединяется к гнезду на заднем торце КИГМД, при этом боковые фиксаторы должны защелкнуться на выступах вилки. Второй разъем надевается прямо на фигурный выступ печатной платы дисковода (роль контактов выполняют печатные дорожки). Если вы использовали разъем отечественного производства, имеющаяся в нем перегородка и прорезь печатной платы дисковода обеспечивают

стоцентную правильность его подключения. Импортный аналог, как правило, не имеет перегородки, так что при подключении нужно быть внимательным: первый контакт разъема должен располагаться со стороны прорези. Если же обмен информацией с дисководом не производится (например, сразу после подключения разъема не удается загрузить с заведомо хорошей дискеты операционную систему), проверьте правильность подключения разъема интерфейсного кабеля.

### Подключение к БК двух дисководов

Для подключения второго дисковода на тот же интерфейсный шлейф нужно установить второй дисководный разъем ОНП-НГ-79-34-МЭК-О, отступя от первого на небольшое расстояние (15—20 см). Второй разъем устанавливается аналогично первому (пусть вас не смущает то, что одни и те же провода шлейфа окажутся подключены одновременно к обоим дисководам). На самих дисководах нужно установить (в соответствии с документацией)

перемычки номера привода: одному дисководу присваивается номер 0 (в большинстве ОС он обозначается как «А:» и именно с него по умолчанию загружается операционная система), другому — 1 («В:») и т. д. Блок и кабель питания должны быть индивидуальны для каждого дисковода (можно купить и более мощный блок питания, рассчитанный на два дисковода сразу\*).

### Дискеты

Для работы на БК с перечисленными в начале статьи моделями дисководов используются любые пятидюймовые двухсторонние дискеты двойной плотности (обозначение DS/DD, 2S/DD или 2S/2D). На них, как правило, указывается емкость 360 кб, стандартная для IBM-совместимых ПЭВМ, но большинство имеющихся в продаже марок дискет (ГМД-130, MaxiData, MiniDisk, Basf, 3M и другие) допускает хранение на них до 800 кб информации и пригодно для работы на БК.

Учитывая, что большинство пользователей, впервые подключивших дисковод и ранее работавших только с магнитофоном, незнакомы с принципами хранения информации на дискете, поговорим об этом более подробно. При чтении или записи данных дискета вращается с большой скоростью вокруг оси, перпендикулярной ее плоскости, а магнитные головки, прижатые к поверхности дискеты с обеих сторон, перемещаются по шагам вдоль радиуса от периферии к центру. В результате магнитная запись располагается на диске в виде концентрических окружностей — дорожек. Крайняя дорожка обозначается номером 0, а ближай-

шая к центру — 80 (или 40, если у вас 40-дорожечный дисковод).

Новую чистую дискету нужно прежде всего отформатировать, для чего служит специальная программа из комплекта поставки операционной системы. (Иногда функцию форматирования дискет встроена в файловую оболочку и вызывается из ее меню, подробнее об этом мы поговорим чуть позже.) При этом на диске формируются дорожки, каждая из которых разбивается на 10 секторов служебными магнитными метками. Секторы нумеруются начиная с 0 (первый сектор на верхней стороне нулевой дорожки):

Дорожка	Сторона	Секторы
0	верхняя	0—9
	нижняя	10—19
1	верхняя	20—29
	нижняя	30—39
.....		
79	верхняя	1580—1589
	нижняя	1590—1599

\* Доработанный типовый блок питания от БК («кирпич»), рассчитанный на подключение одного дисковода, несложно самостоятельно модифицировать для двух дисководов, установив радиатор охлаждения большей площади на «словом» транзисторе КРЕН8Б и подключив второй кабель с разъемом питания дисковода параллельно первому.

Первые секторы каждой дорожки автоматически распознаются дисководом по прохождении мимо установленного на его плате фотоэлемента индексного отверстия дискеты при ее вращении (когда оно совмещается с маленьким круглым отверстием конверта справа).

После форматирования на дискету можно записывать, а затем читать с нее какую-либо информацию. При записи контроллер разбивает массив байтов на блоки по 0.5 кб (длиной 1000b) и заносит каждый блок в отдельный сектор взамен его прежнего содержимого. (Во время форматирования для контроля качества дискеты в каждый сектор заносится повторенный нужное количество раз какой-нибудь байт, обычно нулевой.) Если длина записываемого файла не кратна 1000b, последний сектор будет неполным, при чтении же файла контроллер автоматически объединяет содержимое нужных секторов в единое целое, а лишний «хвост» в конце последнего сектора отбрасывается. Таким образом получаем, что на отформатированную дискету можно записать до 800 кб информации при 80-дорожечном дисководе или до 400 кб при 40-дорожечном.

При записи на дискету нескольких файлов они располагаются в идущих подряд секторах друг за другом, например:

- файл 1 — секторы 0—15;
- файл 2 — секторы 16—18;
- файл 3 — секторы 19—45 и т. д.

Если удалить какой-либо файл (кроме последнего по счету), на его место можно записать другой, такой же или меньшей длины, либо между файлами остается незанятое пространство, которое можно убрать («уплотнив» расположение файлов) с помощью специальной программы — сквизера.

А как определить, с какого места дискеты начинается тот или иной файл? Разумеется, вам не нужно каждый раз запоминать номера начального и конечного секторов — компьютер все делает сам. На нулевой дорожке дискеты выделяется служебная область, именуемая каталогом. В его первом секторе (с номером 0) размещается загрузчик, обеспечивающий чтение операционной системы, а далее следуют записи о хранящихся на дискете файлах: имя файла (с расширением), начальный адрес и длина, номер начального сектора и количество занимаемых секторов. Когда нужно загрузить файл, компьютер прежде всего считывает каталог и определяет, содержимое каких секторов нужно загружать в ОЗУ. Аналогично при записи компьютер определяет по каталогу номер первого свободного сектора, а после выполнения операции в каталог заносятся все сведения о нем. (Из-за наличия каталога фай-

лы записываются начиная не с сектора 0, как было показано в примере выше, а с другого, например с 20. Конкретный номер зависит от используемой операционной системы, а дорожки дискеты для записи собственно файлов, в отличие от каталога, называются ее областью данных.)

Такой «поточный» способ записи характерен для «исконно БКшного» (и используемого на большинстве подобных ей машин — ДВК, УКНЦ и пр.) формата. На IBM же используется другой способ, позволяющий обойтись без сквизирования и произвольно изменять длину файлов (например при редактировании текста можно под тем же именем записать новую версию большей длины взамен старой). Здесь каждый файл может быть произвольно «разрублен» на фрагменты разной длины, размещаемые компьютером в различных местах дискеты наиболее оптимальным образом (впрочем, могут встречаться и «неразрезанные» файлы), а в начале дискеты кроме каталога размещается «карта», где отмечены секторы каждого файла. То же делается на БК в операционной системе ANDOS, что позволяет вести обмен файлами (текстами и графикой) с IBM. Подобный формат записи информации на диск называется кластеризованным. (Конечно, эта картина несколько упрощена по сравнению с реальной. За подробностями следует обратиться к книгам по IBM, например: *Нортон П.* Программно-аппаратная организация IBM PC. М.: Радио и связь, 1991.)

Напоследок перечислим основные правила работы с дискетами, которые должен помнить и неукоснительно соблюдать каждый пользователь, имеющий дело с дисководом (в виде пиктограмм и кратких пояснений эти правила даются на оборотной стороне бумажного пакета, в который вкладывается дискета).

1. Дискета боится магнитных полей. Ни в коем случае нельзя подносить близко к коробке с дискетами магниты или намагниченные металлические предметы либо класть дискеты рядом с излучающими электромагнитное поле бытовыми приборами (повредить записи может даже блок питания магнитофона).

2. Дискеты нельзя оставлять на холоде и под прямыми солнечными лучами, допускать попадание воды на нее. Зимой на улице коробку с дискетами лучше спрятать за пазуху, а войдя в помещение, не начинать с ними работать сразу, а немного подождать, пока дискеты не отогреются до комнатной температуры.

3. Еще один враг дискет — механические воздействия. Дискеты нельзя перегибать, не рекомендуется плотно набивать в коробку большее количество дискет, чем то, на которое она рассчитана. Нельзя также писать на наклейке

дискеты жестким карандашом или шариковой ручкой — только фломастером. А еще лучше сначала сделать надпись на новой наклейке, а затем наклеить ее на дискету.

4. *Будьте аккуратны.* Нельзя касаться руками магнитного слоя, видимого в прорези запаивно-

го конверта дискеты (для перемещения магнитных головок). Вообще, лучше всего брать дискету за края конверта, осторожно вставлять в щель дисковода (не допуская перекосов), а вынув диск из дисковода, сразу же вкладывать его обратно в защитный бумажный пакетик.

## Операционная система

После того как весь аппаратный комплекс (БК, дисковод, контроллер, блоки питания и соединительные кабели) собран, можно начинать работу с диском. Включаем питание... «Но постойте! — скажет читатель. — Ведь БКшка «привыкла» работать только с магнитофоном! Как же «научить» ее осуществлять информационный обмен с дисководом?»

Для этого нам понадобится специальная программа — дисковая операционная система (ОС), записанная на отдельной дискете (которую называют системной или загрузочной). Включив питание дисковода, монитора и БК,ждемся появления на экране приглашения монитора «?». (Если вопросительный знак мигает, несколько раз нажмите клавишу «СТОП», пока мигание не прекратится. Заметим также, что это описание требуемых для запуска БК в дисковом варианте действий относится только к «пустому» контроллеру дисковода или оснащеному дополнительным ОЗУ. В случае наличия в КНГМД встроенного ПЗУ с операционной системой она готова к работе сразу же после включения питания.)

В ответ на приглашение «?» набираем команду: S160000. При этом запускается прошитая в основном ПЗУ КНГМД программа загрузки системы. (Это ПЗУ, размещаемое по адресам 160000—177777, содержит так называемый драйвер дисковода. Не путайте его с устанавливаемым в меньших адресах памяти в некоторых моделях контроллеров ПЗУ с операционной системой.) Компьютер вначале считывает в ОЗУ с адреса 1000 содержимое нулевого сектора дискеты (системная дискета должна быть уже вставлена в дисковод перед набором команды S160000) и запускает его на исполнение. Эта программа называется загрузчиком и производит считывание в ОЗУ и запуск содержащегося на дискете основного файла (собственно ОС). Если же вы по ошибке вставили несистемную дискету, ничего страшного не произойдет: по отсутствию загрузчика в нулевом секторе БК сразу определит ошибку и вновь выдаст приглашение «?». (Внимание! Никаких текстовых сообщений здесь не предусмотрено. Поэтому, если БК после команды

S160000 не загружает ОС, проверьте, системная ли дискета установлена в дисковод.)

После успешной загрузки ОС «берет управление на себя» и все дальнейшие действия — загрузка файловых оболочек, драйверов и т. д. зависят от конкретной версии.

(Примечание. В контроллерах дисковода, изготавливаемых фирмой «АльтПро», предусмотрен дополнительный сервис: после включения питания встроенная в КНГМД программа выполняет тестирование контроллера, а после вывода текстовой подсказки и ожидания нажатия пользователем любой клавиши приступает к загрузке ОС с вставленной системной дискеты без ввода команды S160000. Впрочем, эта команда также сохранена для обеспечения совместимости со стандартными КНГМД от БК-0011М.)

Наверное, у кого-то из читателей возник вопрос: зачем такие сложности? Программа из ПЗУ КНГМД считывает загрузчик, он читает ОС, та в свою очередь драйверы... Не проще ли было бы сразу поручить все действия содержимому ПЗУ?

Отнюдь не проще, и вот почему. Микросхема ПЗУ устанавливается в контроллер на заводе и, как правило, впоследствии не заменяется. А различных операционных систем для БК существует не меньше десятка и продолжают разрабатываться все новые. Процесс же загрузки, как мы уже отметили, для каждой версии ОС свой. Поэтому-то и выбран такой на первый взгляд неоправданно сложный метод. Чтение и запуск загрузчика из нулевого сектора всегда выполняются одинаково, поэтому программу для этой операции можно разместить в ПЗУ. Загрузчик же создается конкретно для данной версии ОС, так что на одной и той же БК с одним и тем же контроллером можно загружать разные операционные системы.

Теперь поговорим о самих ОС: какие они бывают, чем различаются, какую лучше всего выбрать? (Ниже приведены только краткие сведения и рекомендации, более подробное описание имеется в статье Д. Ю. Усенкова в №1 за 1993 г. и в обзоре В. П. Юрова в №1 за 1993 г. и в №3 за 1994 г.)



Прежде всего надо учесть возможности имеющегося у вас КНГМД. Если это «пустой» контроллер (в котором нет ни дополнительного ОЗУ, ни ПЗУ), нужно искать простейшую ОС типа «Нортон», загружаемую в основное ОЗУ. («Пустые» КНГМД сегодня почти не используются, поэтому легче заказать установку в контроллер дополнительного ОЗУ, чем найти требуемую версию ОС.)

Если в контроллере имеется ПЗУ с прошитой в нем операционной системой, все хлопоты с загрузкой ОС с системной дискеты вам ни к чему — достаточно включить питание. Но тогда вы не сможете установить взамен имеющейся в ПЗУ другую версию ОС. Выход из этой ситуации тот же, что и при «пустом» контроллере: установка дополнительного ОЗУ вместо микросхемы ПЗУ.

И наконец, при имеющемся дополнительном ОЗУ вы можете запускать практически любую из существующих ОС из числа способных работать на БК-0010(.01). (Владельцам одиннадцатой модели гораздо проще: им годится стандартный «пустой» контроллер, ибо свободным ОЗУ БК-0011(М) располагает в избытке. Надо лишь купить ОС, предназначенную для БК-0011(М) или универсального применения.) Конечно же, нужно учитывать и объем имеющегося дополнительного ОЗУ — для более старых версий ОС достаточно 8 кб, новейшие же работоспособны только при 16 кб. А если загруженная ОС вам не понравилась (или она не подходит для конкретной задачи), тем же путем можно загрузить иную версию с другой системной дискеты (из текущей ОС нужно выйти в монитор и в ответ на приглашение «?» снова набрать S160000).

Предположим наилучший вариант: у вас имеется КНГМД с дополнительным ОЗУ не менее 16 кб. Теперь можно выбрать любую из перечисленных ОС (в скобках указаны номера новейших версий): **NORD** (3.2), **МК-DOS** (3.15), **ANDOS** (3.1). (Есть и другие, но они либо устарели, либо слишком мало распространены и притом несовместимы с другими.) Обсудим характеристики каждой из названных систем.

**NORD** — довольно удобная система, оснащенная встроенной файловой оболочкой наподобие Norton Commander для IBM и позволяющая работать с подкаталогами. Недостатками ее являются нестандартные ключи (что приводит к путанице при переходе с какой-то другой ОС на **NORD** и обратно) и особенности самого формата хранения информации (невозможность записи нового варианта файла поверх старого, из-за чего дискета быстро забивается одноименными копиями, и необхо-

димость сквизирования дискеты). Очевидный вывод: **NORD** наиболее пригоден для хранения различных игр и готовых прикладных программ, в том числе включающих одноименные файлы. Комплекты файлов для каждой программы раз и навсегда размещаются в поддиректориях, а затем только запускаются в работу без каких-либо изменений. Тем же, кто собирается использовать БК в качестве «рабочей лошади» (редактировать тексты, активно заниматься программированием и т. д.), операционная система **NORD** вряд ли подходит.

(В качестве небольшого отступления поясним, что такое подкаталог (поддиректория). Представьте себе обширную картотеку, например предметный каталог в большой библиотеке. Множество карточек разложено по алфавиту (пусть для простоты они лежат в одном длинном ящике). Найти требуемое название несложно, но еще проще и быстрее это сделать, если карточки в ящике разложены по секциям в зависимости от тематики. Теперь, если вам нужны сведения о синхрофазотроне, не потребуются заглядывать в секции по темам «сельское хозяйство» или «бытовые приборы». Можно пойти еще дальше: как следует продумав систематику каталога, разложить карточки «по нескольку уровней» — пакетики по конкретным темам вложить в коробочки, объединив их в рамках более общих вопросов, коробочки — в секции, секции — в ящички, ящички — в шкафы... И теперь поиск нужной книги будет совсем простым и быстрым, например, «прикладные науки» — «физика» — «ядерная физика» — «инструменты исследований» — «синхрофазотрон».)

Аналогичные трудности могут встретиться и при поиске нужного файла на дискете. Если все они находятся в основном списке («корневом каталоге»), приходится просматривать его весь. Еще сложнее отыскать все файлы, входящие в отдельный комплект (например, чтобы скопировать его на другой диск). Поэтому и были придуманы поддиректории. В корневом каталоге системной дискеты мы видим имена обычных файлов (которые можно загрузить и попытаться запустить, наведя на имя файла инверсную подсветку — курсор с помощью клавиш со стрелками и нажав «ВВОД»), а также имена поддиректорий, отличающиеся от файлов заглавными буквами или особым символьным обозначением. Если навести курсор на имя поддиректории и нажать «ВВОД», мы как бы откроем тематическую коробочку картотеки и увидим список файлов, хранящихся в данной поддиректории. Среди имен этого списка также могут быть вложенные поддиректории (более низкого уровня), их тоже можно

раскрыть и просмотреть содержащиеся файлы. А можно вернуться «на один уровень вверх», выйдя из поддиректории и снова увидев только ее имя. Таким образом можно распределить все записанные на дискете файлы по вложенным друг в друга поддиректориям, как мы делали это с карточками книг. Например, комплект файлов графического редактора BKPAINT будет находиться в отдельной поддиректории, она вместе с GRAF и другими графическими редакторами (а также, может быть, файлы документации к ним) будут находиться внутри поддиректории «Графические системы». Последняя, в свою очередь, вместе с рисунками войдет в состав поддиректории «Графика», а она вместе с подкаталогами «Тексты», «Копировщики» и т. д. — в поддиректорию «Прикладные программы», которая будет находиться в корневом каталоге вместе с поддиректорией «Игры». И теперь, чтобы запустить BKPAINT, вы легко пройдете все это «дерево поддиректорий»: «Прикладные» — «Графика» — «Графические системы» — «BKPAINT» — нужный запускаемый файл.)

Продолжим обсуждение операционных систем. Следующая ОС MK-DOS отличается от NORDa более высоким качеством оформления интерфейса с пользователем и набором функциональных возможностей, а также отдельной (а не встроенной в саму ОС) файловой оболочкой. Формат хранения файлов на диске в MK-DOS тот же что в NORDe, поэтому и здесь не обойтись без сквизирования. Рекомендации к применению MK-DOS в общем-то те же, что и для NORD: хранение и запуск неизменяемых программ. Так что на выбор между MK-DOS и NORD скорее будет влиять стоимость дискеты с ОС и личные пристрастия.

И наконец, ANDOS принципиально отличается от двух предыдущих систем. Формат хранения файлов на диске в нем тот же, что и в MS-DOS на IBM-совместимых ПЭВМ. В результате, с одной стороны, на БК появляется возможность многократно переписывать новые варианты текстов, листингов программ и т. д. поверх старых (причем никакого сквизирования не требуется), а с

другой — свободно обмениваться информацией с IBM (читать и записывать файлы). Конечно, программы для БК на IBM не запустятся (как и IBMовские — на БК), но графику, а тем более текст можно переносить с одного компьютера на другой без проблем. Довершают список преимуществ ANDOSa принципиальная возможность работы с файлами любой длины, наличие структуры подкаталогов, совместимость с другими ОС (по крайней мере возможность чтения и запуска файлов с дискет в формате NORD и MK-DOS) и удобная файловая оболочка. Из недостатков же можно назвать разве только невозможность хранения на одной и той же дискете (даже в разных подкаталогах!) нескольких файлов с одним и тем же именем. Вывод: ANDOS наиболее подходит для серьезной работы с БК. Вообще же ОС ANDOS можно считать наиболее приемлемой в любом случае, в том числе и для первоначального освоения БК с дисководом.

Завершая обсуждение операционных систем, заметим и такое, часто смущающее желающих заменить магнитофон на дисковод пользователей обстоятельство. Многие из них опасаются, что старые программы, рассчитанные на магнитофон, не будут функционировать с диском без доработки. На самом деле никаких проблем такого рода чаще всего не возникает. Современные дисковые операционные системы оснащены так называемым «EMT-перехватом», т. е. автоматически отслеживают операции стандартного обращения любой программы к магнитофону (функция EMT 36) и перенаправляют поток информации на дисковод. Таким образом, любая программа, рассчитанная на магнитофон (текстовые и графические редакторы, многофайловые игры и т. п.), будет без изменений работать и с дисководом. Исключение составляют лишь немногие копировщики и игры, использующие нестандартное обращение к магнитофону «в обход» EMT 36 (непосредственно через прошитые в ПЗУ монитора БК подпрограммы магнитофонного драйвера). Но таких исключений очень немного, и, кроме того, в большинстве случаев они и не нужны для работы с диском.

## Дисковые утилиты

Кроме операционной системы для работы с дисководом вам потребуется еще ряд служебных программ, реализующих сравнительно редко используемые и потому не включенные в состав самой ОС функции. Как правило, большинство таких вспомогательных программ, называемых утилитами, ориентировано на работу с конкретной опе-

рационной системой и поставляется вместе с ней на одном и том же системном диске. Но есть и универсальные программы, работающие в любой ОС и приобретаемые отдельно. Поскольку описать все обилие существующих сегодня разработок в одной статье невозможно, ограничимся обсуждением только основных классов утилит.

**Форматировщики.** О том, что такое форматирование, мы уже говорили, когда обсуждали типы используемых на БК дискет. Там же было сказано, что операция форматирования иногда включается в состав файловой оболочки. Но чаще всего используются отдельные форматировщики (пример — ANFORMAT из комплекта ANDOS), либо эта функция включается в состав дисковых сервисных программ более общего назначения (например, DiskTools — набор функций для тестирования дискеты или DiskDoctor — программа для восстановления плохо читаемых файлов). По-видимому, обсуждать этот класс утилит подробнее не имеет смысла, так как все сведения о прилагаемом к той или иной ОС форматировщике всегда указываются в записанной на системной дискете документации.

**«Диск-докторы» и утилиты для сохранения нулевой дорожки.** Иногда какой-либо файл на дискете перестает читаться. Это может произойти по самым разным причинам: царапина на магнитном слое из-за попавшей песчинки, жирный след от пальцев, сбой при записи программы, из-за чего стирается поставленная при форматировании магнитная метка или контрольная сумма записанной в секторе информации не соответствует реальной... (Еще одна часто встречающаяся причина порчи информации на дискетах заключается в недоработке конструкции самого дисководов. При выключении питания БК раньше, чем дисковод, магнитные головки «клатают» по дискете и затирают попавшийся «под горячую руку» сектор. Поэтому нужно взять за правило всегда вынимать дискету до выключения питания и вставлять ее после его включения.) Разумеется, если вы строго выполняете правила работы с дискетами, вероятность порчи информации очень мала, но стопроцентной гарантии не может дать никто (известно ведь, что «и на старуху бывает проруха»).

Конечно, при царапине или другом физическом повреждении магнитного слоя содержимое данного сектора, скорее всего, восстановить не удастся. Но при ошибке «на программном уровне» «диск-доктор» иногда может после некоторых ухищрений корректно считать поврежденный фрагмент файла. Если это удастся (или если пользователю надоест многократный вывод сообщений о невозможности прочесть сектор и он нажатием клавиши откажется от попытки восстановления), «диск-доктор» форматирует поврежденную дорожку заново и записывает на нее ту часть содержимого, которую удалось считать.

Если окажется нечитаемым сектор, содержащий фрагмент одного какого-нибудь фай-

ла, вы можете потерять только этот файл. Но если будет повреждена нулевая дорожка, где, как мы помним, находится каталог содержимого дискеты, вы рискуете потерять все файлы, ибо найти и «вытянуть» их на другую дискету вручную не просто даже для опытного пользователя. (Особенно это касается дисков в формате операционной системы ANDOS, где файлы могут быть фрагментированы и их приходится «собирать по кусочкам». Поэтому многие БКманы предпочитают ANDOSy NORD или MK-DOS, где файлы записываются на диск целиком подряд друг за другом.)

Чтобы уберечься от такой напасти, проще всего сохранить копию нулевой дорожки в другом месте дискеты. Чаще всего эта копия размещается на обычно неиспользуемой, но доступной на большинстве дисководов 80-й или 81-й дорожке. При повреждении основного каталога можно восстановить его по ранее сделанной копии. Но следует помнить, что копию нулевой дорожки нужно сохранять каждый раз заново после любого изменения файловой структуры на дискете (скажем, после записи новых файлов). Иначе после восстановления каталога дискета примет тот же вид, как перед сохранением последней копии и уже не будет совпадать с истинным содержимым дискеты.

Для сохранения копии нулевой дорожки разработаны специальные программы (например, DigKeeper). Эта же функция входит в состав большинства «диск-докторов», а также имеется в файловой оболочке DiskMaster для ANDOS версии 3.1.

**Копировщики.** До того как подключить дисковод, вы наверняка накопили множество программ на магнитной ленте и хотели бы работать с ними и в дисковом варианте. Ранее уже говорилось, что большинство старых «магнитофонных» программ без изменений могут работать и с дисководом, нужно лишь переписать их на дискету. Для этого необходим копировщик, адаптированный для обмена информацией с диском.

Как правило, в комплекте большинства ОС имеются простейшие копировщики для переписывания программ с кассеты на диск, а иногда и обратно (в некоторых системах такая функция встроена в файловую оболочку). Но лучше всего эта проблема решена в ANDOSe: специальная программа HELP-DISK (к сожалению, она не может работать в новой версии ANDOS 3.1) позволяет свободно копировать любые файлы с кассеты на диск и обратно, в том числе в «архивном» формате HELP7.

Другая полезная разработка (особенно для пользователей, чей контроллер дисководов не оснащен «дисковым БЕЙСИКом») — SCREW

BACKUP SYSTEM. Она позволяет переписывать с кассеты на диск листинги программ на вильнюсском БЕЙСИКЕ, «шивая» ASC-блоки в единый файл, и, наоборот, «разрезать» на блоки файл на диске и записывать его на кассету в ASC-формате.

И наконец, вам понадобятся «чисто дисковые» копировщики. Функция переписывания файлов с одной дискеты на другую в пределах формата одной и той же ОС имеется в составе большинства файловых оболочек. Если же нужно копировать файлы из одного формата в другой, чаще всего требуется отдельный копировщик.

*Каталогизаторы.* Если у вас всего одна или две дискеты, местонахождение каждого файла на них нетрудно запомнить. Но если их накопилось несколько коробок, даже несмотря на быстроту доступа к каталогам дискет, поиск требуемого файла — работа достаточно долгая и нудная. Значительно облегчить жизнь могут специальные программы — каталогизаторы. Хорошим примером является созданный минскими программистами «ANDOS INFORMER». Он позволяет автоматически сохранять на специально выделенной дискете нулевую дорожку любого диска в формате ANDOS (достаточно вставить его в дисковод и нажать клавишу), восстанавливать поврежденный каталог (как и DirKeeper, используя ранее сохраненную копию нулевой дорожки), а также искать в полученной базе данных всего по нескольким буквам имени файла, на каких дискетах он записан, и просматривать содержимое любой дискеты.

## Дисковод уже есть. А что дальше?

Итак, дисковод стал неотъемлемой частью вашей БКшки. К хорошему привыкают быстро, и вот вы уже начинаете подумывать о дальнейшем расширении возможностей компьютера. Что же еще можно подключить к БК?

На IBM помимо дисководов имеется такая полезная, с точки зрения работающего с файлами пользователя, вещь, как винчестер (жесткий диск). Всего год назад о возможности подключения винчестера к БК можно было только мечтать. Но теперь за сравнительно небольшую плату (около 100 долларов) к БК-0011(M) и даже к БК-0010(.01) можно подключить жесткий диск емкостью 20, 40, 80 и более мегабайт. В результате вам уже не потребуется постоянно переставлять дискеты в ди-

*Конвертор текста.* Как отмечено выше, ANDOS позволяет свободно переносить файлы с IBM на БК и обратно. Но кодировка русских букв на двух этих компьютерах различна, поэтому, прежде чем читать на IBM текст с БК, его нужно перекодировать. Аналогично, перенеся текст с IBM на БК, нужно преобразовать его в БКшный формат и только потом загружать в текстовый редактор. Для конвертирования текстов из формата БК в IBM и наоборот в комплект ANDOS введена утилита CONTXT. (При переносе текстов с IBM нужно также заранее разбить их на фрагменты приемлемой для БК длины. То же можно сделать и уже на БК с помощью утилиты TOPOR.) Впрочем, новейшие разработки еще более сближают БК с IBM-совместимыми ПЭВМ. Встроенная в файловую оболочку DiskMaster для ANDOS версии 3.1 функция просмотра текстовых файлов и недавно появившийся текстовый редактор Vortex! 4.0 могут работать с файлами любой длины в «альтернативной» кодировке IBM (с помощью собственного перекодировщика).

Конечно же, спектр видов существующих программных средств для работы с диском не ограничивается перечисленными выше. Прочитать о них вы можете в разных выпусках журнала «Персональный компьютер БК-0010 — БК-0011М». В частности, о преимуществах и недостатках того или иного программного дискового обеспечения; а также об обслуживании дисководов рассказано в статье «БК-0010: опыт работы с диском» в №1 за 1993 г.

сководе: все имеющиеся программы будут доступны сразу (достаточно указать нужную литеру — обозначение «логического раздела» жесткого диска). Кроме того, можно разместить на винчестере несколько операционных систем одновременно и быстро переходить от одной из них к другой. Дискеты же останутся только средством переноса файлов с вашего БК на другие и наоборот. Подробнее о винчестере на БК рассказано в №2 за 1995 г.

\* \* \*

*По вопросам приобретения большинства из описанного в этой статье программного и аппаратного обеспечения, а также для получения консультаций обращайтесь в редакцию по телефону: (095) 151-19-40.*

**Редакция благодарит за помощь в подготовке данной статьи  
В. Г. Александрова, А. С. Анфалова, А. М. Надежина,  
С. М. Неробеева и А. В. Сорокина.**

## Поговорим о зависаниях

Зависания пользуются у компьютерщиков дурной славой. Каждому, кто работал на ЭВМ, наверняка приходилось сталкиваться с леденящей душу ситуацией, когда компьютер вдруг перестает реагировать на какие бы то ни было команды клавиатуры, издавать привычные звуки, менять картинку на экране, — в общем, зависает. В памяти в этот момент могут находиться ценнейшие данные, плод долгих трудов, и нет никакой возможности их спасти...

В борьбе с этим опасным явлением пользователи изобрели разные приемы и хитрости, позволяющие выйти из зависания и вернуть данные из «небытия». Можно, например, обрывать компьютер специальной кнопкой, при нажатии на которую происходит перезапуск процессора (кнопка Reset). При этом на БК-0010 надо еще часто-часто нажимать клавишу «СТОП», чтобы обеспечить выход в монитор и спасти память от обнуления ФОКА-Лом (а на БК-0010.01 без блока МСТА — БЕЙСИКом). К сожалению, даже такие «крутые» меры помогают не всегда: данные могли быть уничтожены и самой причиной, вызвавшей зависание. Поэтому вместо борьбы с собственно зависаниями лучше устранять вызывающие их причины. Как гласит восточная мудрость, предотвращенная схватка — выигранная схватка.

Причины, вызывающие зависания, могут иметь «искусственное» и «естественное» происхождение. Некоторые программисты, чтобы защитить свою разработку от постороннего любопытства, устраивают на выходе из программы зависание. В этом есть что-то от дурного тона: злостные нарушители все равно найдут способ снять защиту, а простых пользователей такое свойство программы только раздражает. Другим примером «искусственной» причины зависаний являются проделки вирусов, созданных компьютерными хулиганами. Предотвратить подобные причины невозможно, остается только призвать программистов к благодарности. (Конечно, на каждый вирус в конце концов найдется свой антивирус, но и это не абсолютная панацея.)

«Естественной» причиной зависаний могут быть ошибки и неточности. От этого не застрахована ни одна программа, в том числе системная. Иногда ошибка долгое время остается незамеченной, дожидаясь, подобно вирусу, «своего часа», и, когда создадутся подходящие условия, приводит к фатальным последствиям, в том числе к зависанию. Ошибку, обнаруженную в программе пользователя, можно исправить, заменив соответствующие коды в оперативной памяти. К сожалению, коварные ошибки встречаются и в системных программах, «защитных» в постоянной памяти и потому не подлежащих исправлению. Про такие ошибки необходимо помнить и при работе «обходя их стороной».

Многие пользователи БК-0010 сталкивались с зависаниями при работе в графическом режиме. Сбой здесь может возникнуть при попытке вывести графический курсор за верхнюю или нижнюю границу экрана. Особенно эффектно это смотрится в цветном режиме отладочного монитора ПЗУ тестов (на БК-0010.01 нужно подключить блок МСТА). Находясь в нем (приглашение «И»), установите графический режим («СУ»+«AP2»+«U»), нормализуйте экран (два раза «AP2»+«СБР»), а затем наберите 13 и нажмите клавишу «стрелка вверх». Экран при этом мгновенно покроется цветными полосами, а компьютер безнадежно зависнет.

Причиной этого зависания является неточность в программе монитор-драйверной системы. При отработке вертикальных перемещений графического курсора монитор проверяет адреса его отображаемых курсора на выход за границы экрана. Допустим, что в регистре R3 — адрес верхней части изображения курсора, в ячейке 202 — адрес начала экранной области, например 40000 (здесь и далее числа восьмеричные), в ячейке 206 — размер экранной области (40000). При отображении графического курсора монитор производит следующую проверку:

```

106640: CMP R3, 202 ;Сравнить адрес верхней части
;курсора с адресом начала экрана
106644: BNI 106652 ;Если выше, то все в порядке,
106646: ADD 206, R3 ;иначе прибавить размер экрана
106652: MOV #7, R2 ;Подготовить размер курсора
106656: XOR R4, (R3) ;Начать отображение курсора
...

```

Неточность здесь заключается в команде по адресу 106644 ВНИ — «переход, если ВЫШЕ». Вместо нее следовало поставить команду ВНИС — «переход, если ВЫШЕ ИЛИ РАВНО». В самом деле, иногда адрес верхней части графического курсора может оказаться в точности равным адресу начала экрана, который к тому же равен 40000. Перехода по условию ВЫШЕ не произойдет, адрес курсора будет увеличен на 40000 (размер экрана) и составит 100000.

Дальнейшее воспринимается как детективный сюжет из мира программ. По адресу 100000 в БК-0010(.01) начинается, как известно, системное ПЗУ, изменять информацию в котором невозможно. Тем не менее программа вывода изображения графического курсора вынуждена обратиться по адресу ПЗУ (100000) по записи. Попытка выполнения по адресу 106656 операции «исключающее ИЛИ» над словом из ПЗУ вызывает прерывание по вектору 4 — обращение по записи к постоянной памяти. Все бы ничего, но мы все еще находимся в графическом режиме, и программа обработки вектора 4 опять сталкивается с необходимостью отобразить (переместить) графический курсор при тех же обстоятельствах, что опять вызывает прерывание по неверному обращению к ПЗУ.

Процесс заикливается, при каждом прерывании в стек заносятся новые данные, стек переполняется, его указатель «спускается» все ниже, приближаясь к началу памяти, вся информация системной области портится. Наконец стек, опустившись ниже адреса 30, портит вектор ЕМТ, заменяя его случайным числом. Программа обработки прерывания по вектору 4, используя команду ЕМТ, передает управление по случайному адресу...

Разобраться в дальнейшей цепочке переходов можно, только пользуясь дедуктивным методом Шерлока Холмса. В нашем примере компьютер в конце концов выходит из подпрограммы закраски экранной строки (из системного монитора). В регистре, который является ее аргументом и в котором должно находиться число закрашиваемых байтов, как назло в данный момент записан ноль. В результате программа закраски 65536 раз выполняет команду SOB, заполняя всю оперативную память произвольным числом. Компьютер окончательно и безнадежно зависает. Такому итогу мог бы «позавидовать» любой компьютерный вирус! Пользователям же можно посоветовать проявлять большую осторожность и, находясь в графическом режиме, избегать пересечения курсором границ экрана.

Другим примером опасной неточности в программе системного ПЗУ является функция PAINT БИЙСИКа БК-0010.01. Как известно, PAINT производит закраску замкнутой области и требует для своей работы значительных размеров стека. Многими это воспринимается просто как свойство функции PAINT с кото-

рым ничего не поделаешь. Однако это именно программная ошибка, способная привести к зависанию по причине переполнения стека. Дело в том, что подпрограмма PAINT использует стек очень неэффективно, помещая туда много лишней информации. И несмотря на то что в БИЙСИКе указатель стека настроен на адрес 2000 (вместо обычного для монитора 1000), при выполнении PAINT стек зачастую бывает переполнен, портится системная область и происходит зависание.

Какую же информацию и когда подпрограмма PAINT помещает в стек? Алгоритм закраски требует сохранения данных о текущей строке в случае изменения направления линии контура, для того чтобы, закончив закраску одной части, можно было вернуться к другой и завершить процесс. Программа чисто формально помещает данные в стек всякий раз, когда левая или правая граница следующей строки (точка закрашиваемого контура) оказывается более чем на пиксел дальше, чем текущей. Подразумевается, что при этом вполне возможно изменение направления контура. В этом и состоит ее ошибка. Интервал, данные о котором сохраняются в стеке, может целиком состоять из пикселей цвета границы, и в его сохранении нет никакой необходимости. Тем не менее программа откладывает проверку этого факта «на потом» и «занимает» в стек все новую и новую нижнюю информацию. Не удивительно, что стек в конце концов может переполниться.

Особенно часто сохранение ненужной информации происходит, если границами контура являются прямые с угловым коэффициентом, меньшим или равным 1/3. Попробуйте запустить на исполнение следующую простую программу на БИЙСИКе:

```
10 CLS
20 LINE (0,0)-(255,85),1
30 LINE -(0,170)
40 PAINT (0,1),1
```

Программа начертит клин из двух прямых и попытается его закрасить. Инструкция предупреждает о возможности сбоя в случае сложного контура, но в данном случае произойдет зависание, хотя контур совсем не сложен! Зато он имеет 170Δ (буква Δ означает десятичное число) ступенек, на каждой из которых в стек «занимается» 4 числа, и он увеличивается на 8 байт. Пока закраска доберется до нижней границы контура, стек успеет разрастись до 170·8=1360 байт, что с лихвой превышает отведенные для него и для системной области 1024Δ байт. Зависание не заставит себя ждать.

К счастью, имеется возможность исправить этот программный дефект. В статье Д. Ю. Усенкова «Библиотека графических функций для ассемблера БК-0010(.01)» в журнале «Персональный компьютер БК-0010 — БК-0011М», № 1 за 1994 г.,

приводится ассемблерный листинг варианта подпрограммы PAINT, размещаемой в ОЗУ. Нетрудно внести в него необходимые дополнения, обеспечивающие более правильную работу функции закраски. Для этого нужно вставить в текст специальную подпрограмму, проверяющую сохраняемый интервал и освобож-

дающую стек, если интервал состоит только из точек цвета границы. Команды вызова этой подпрограммы необходимо разместить по меткам 26050 и 26214, а текст, ранее там находившийся, сдвинуть вниз. Имя добавочной подпрограммы %TEST, остальные обозначения — как в статье Д. Ю. Усенкова:

26050: JSR R7, %TEST

26214: JSR R7, %TEST

```

; Подпрограмма проверки интервала
%TEST: MOV R1, -(R6)
        MOV R2, -(R6)
        MOV 14(R6), R1 ;Координата X левой границы
        MOV 12(R6), R2 ;сохраняемого интервала
        ADD 18(R6), R2 ;Координата Y
2:      INC R1 ;Продвинуть точку интервала
        CMP 16(R6), R1 ;Интервал весь просмотрен?
        BLE 1 ;Да - на выход с освобождением стека
        JSR R7, %A52 ;Точка принадлежит контуру?
        BCS 2 ;Да - продолжать просмотр
        MOV (R6)+, R2 ;Нет - на выход
        MOV (R6)+, R1
        RTS R7
1:      MOV (R6)+, R2 ;Освобождение стека
        MOV (R6)+, R1 ;от ненужной
        MOV (R6)+, 6(R6) ;информации
        MOV (R6)+, 6(R6)
        CMP (R6)+, (R6)+
        RTS R7 ;и выход

```

## Дополнение к опубликованной статье

Хотелось бы сделать небольшое дополнение к статье Д. Ю. Усенкова «ПЗУ для БК» (№2 за 1994 г., с. 38). Старшие разряды адреса ПЗУ, названные в статье «чип-селектором», для всех типов указанных там микросхем программируются пережиганием расплавляемых перемычек. Поэтому исходным состоянием ячеек «чип-селектора» является логическая единица, а при программировании значения отдельных (выбранных пользователем) ячеек становятся нулевыми. Вернуть же ошибочно запрограммированные на логический нуль ячейки «чип-селектора» в исходное единичное состояние уже невозможно. (Напомню для тех, кто программирует ПЗУ вручную или самостоятельно разрабатывает и изготавливает программаторы, что в БК логическому нулю соответствует напряжение питания (+5 В), а логической единице — «общий» провод («заземление»).

Р. Аскеров, г. Брянск



## СОДЕРЖАНИЕ

	3	От редакции
<i>Б. Н. Сергеев</i>	8	Практикум на ассемблере
	23	Руководство системного программиста БК-0011(М). Базовая операционная система

### HARD & SOFT

<i>А. И. Кузнецов</i>	31	Доработка принтеров серии МС-6312
<i>В. В. Зыков</i>	33	Математический сопроцессор для БК
<i>С. М. Неробеев, А. В. Сорокин</i>	43	Трехканальный музыкальный синтезатор для БК-0010(.01)
<i>А. В. Кузнецов</i>	48	Дополнительный блок ДОЗУ и ПЗУ для БК-0010(.01)
<i>П. П. Животовский</i>	51	Базы данных для БК: сравнительный обзор
<i>М. И. Кондратович</i>	54	Суперкаталогизатор дискет для ANDOS

### НАМ ПИШУТ...

<i>А. В. Бриндеев</i>	57	Доработка программ для дискового БЕЙСИКА
<i>Р. Аскеров</i>	58	Условные переходы в ассемблере
<i>Ю. А. Зальцман</i>	59	Виновник найден!
<i>В. Г. Шаклеин</i>	59	Доработка канала звука в БК-0011
<i>Д. Р. Насыров</i>	60	«Секретные» пароли в играх для БК

### ГЛЮКАДЕМИЯ

	61	Доработка драйвера принтера в БК-PAINT
	62	Новые фактуры закраски для БК-PAINT

### НАЧИНАЮЩИМ ПОЛЬЗОВАТЕЛЯМ

	63	Подключение дисководов к БК
<i>С. К. Румянцев</i>	76	Поговорим о зависаниях





**ПЕРСОНАЛЬНЫЙ  
КОМПЬЮТЕР  
БК-0010 —  
БК-0011М**

**Главный редактор**  
Васильев Б. М.

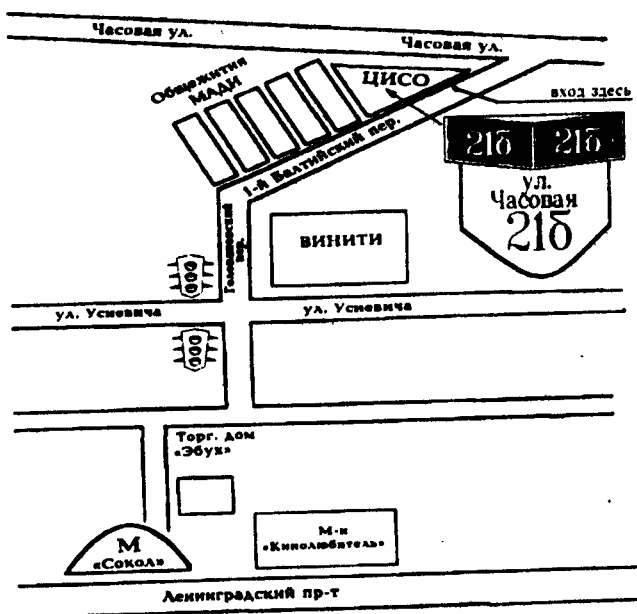
**Редактор**  
Усенков Д. Ю.

**Корректор**  
Антонова В. С.

**Компьютерная верстка**  
Усенков Д. Ю.

Наш адрес: Москва, ул. Часовая, 21Б, помещение Центра  
Интерактивных Средств Обучения (ЦИСО), комн. 36  
Телефон: (095) 151-19-40

Как к нам добраться:



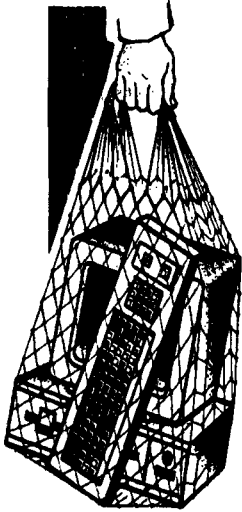
Адрес для переписки: 125315, Москва, а/я 17.

**ПЕРСОНАЛЬНЫЙ КОМПЬЮТЕР  
БК-0010 - БК-0011М**

Подписано в печать с оригинал-макета издательства  
«Компьютика» .95. Тираж 1500 экз. Формат 70×100  
1/16. Бумага офсетная. Усл.печ.л. 6,5. Заказ №  
. Цена 5000 руб. (по подписке). В розничной продаже  
цена договорная.

# УВАЖАЕМЫЕ ЧИТАТЕЛИ!

Приглашаем вас в Центр БК, открытый совместными усилиями редакции журнала и фирмы «АльтПро». Вашему вниманию предлагается новейшее аппаратное и программное обеспечение для БК-0010(.01) и БК-0011(М):



- мониторы, принтеры, блоки питания;
  - дисководы и контроллеры НГМД с ДОЗУ различного объема;
  - винчестерские диски емкостью от 20 до 540 Мб с контроллерами для БК-0010(.01) и БК-0011(М);
  - кабели, разъемы и прочие комплектующие
- а также
- новые версии операционных систем ANDOS, NORD и MK-DOS;
  - база данных BASIS 2 и BASIS 3;
  - графический редактор TECHNO ART 3;
  - текстовый процессор Vortex 4.0 (тексты любой глины);
  - трассировщик двухсторонних печатных плат VCAD;
  - программы высококачественной печати COMFORT и WRITER;
  - разнообразные драйверы и утилиты, системные и прикладные программы;
  - богатый выбор игр и многое другое.

Специалисты Центра БК предоставят вам консультации по подключению периферийных устройств и использованию программного обеспечения, окажут помощь в ремонте БК и периферии.

Центр БК расположен в помещении редакции журнала по адресу: Москва, ул. Часовая, д. 21б, комн. 36. Для заказа каталогов имеющегося программного и аппаратного обеспечения по почте достаточно прислать в редакцию письмо с заявкой и вложенным пустым конвертом с надписанным обратным адресом и наклеенными почтовыми марками

Здесь же вы можете приобрести отдельные выпуски журналов «Персональный компьютер БК-0010 — БК-0011М» и «Персональный компьютер УКНЦ», а также оформить полугодовую подписку на эти издания с получением экземпляров в редакции или по почте.

Заключаем с авторами договоры на рекламу и коммерческое распространение программных и аппаратных разработок. Приглашаем к сотрудничеству книготорговые организации, фирмы и заинтересованных лиц для реализации нашей печатной продукции.

Адрес для переписки: 125315, Москва, а/я 17

Телефон: (095) 151-19-40

E-Mail: mail@infoobr.msk.su