



## От редакции

Уважаемые читатели! Эти методические рекомендации открывают серию, адресованную пользователям "Электроники БК-0010" (0010.01). Надеемся, что содержание рекомендаций не обманет ваши ожидания, и вы ещё не раз воспользуетесь информационными услугами кооператива СБИС.

О появлении новых изданий в серии вы сможете узнать из выпусков информационного бюллетеня БИС. Заказать очередной и любой из предыдущих выпусков серии можно по адресу: 167024, Сыктывкар, а/я 430, СБИС, редакция БИС.

Желающих попробовать стать автором аналогичного пособия просим обращаться в редакцию. Заявка должна содержать план методических разработок, которые вы хотели бы и могли написать, и ваши точные координаты. С авторами будут заключены договоры на издание рекомендаций под редакцией СБИС.

Будем весьма благодарны всем, кто сообщит нам о замеченных ошибках или неточностях. Постараемся оперативно включить соответствующие поправки в текущие выпуски бюллетеня БИС.

(С) Научно-производственный кооператив СБИС, 1990.

(С) Радченко Л.И., 1990.

## ВВЕДЕНИЕ

Как бы ни увеличивался выпуск и ни расширялся ассортимент отечественных персональных ЭВМ, компьютер "Электроника БК-0010" наверняка останется в нашей памяти - это первая вычислительная машина, поступившая в магазины для широкой продажи. Говорят, что среди производителей, обсуждавших целесообразность серийного выпуска БК, высказывались сомнения: будут ли покупать этот компьютер "за наличные"? Стоит компьютер немало - 600 рублей, то есть во много раз дороже среднего микрокалькулятора. К тому же возможности БК довольно скромны в сравнении с тем, что предлагают пользователю хорошие зарубежные микроЭВМ.

В середине 1984 года первые БК-0010 появились на прилавках фирменных магазинов "Электроника" в Москве и Ленинграде, и все сомнения относительно спроса были сразу же развеяны - желающих иметь дома компьютер оказалось так много, что покупателю приходилось по нескольку месяцев ждать своей очереди. Эта невеселая ситуация сохраняется и в 1990 году: если БК-0010 не так уж сложно приобрести в Москве, то БК-0010.01 пока что в дефиците, хотя суммарный выпуск БК довольно высок.

Одним из устройств, с которым постоянно приходится сталкиваться пользователям микроЭВМ "Электроника БК-0010" - это магнитофон. Именно он призван играть роль накопителя информации, заменяя тем самым дисковод персонального компьютера. На обычной магнитной ленте приходится хранить данные, и, что гораздо чаще, программы. Наиболее сложными для начинающих, как правило, являются вопросы, связанные с копированием программ, использованием различных копировщиков и регенераторов. Не просто и обеспечить защиту программ от несанкционированного копирования и снятие защиты с интересных программ.

Пособие ответит на многие из этих вопросов. В частности, будут рассмотрены общие положения, касающиеся обмена информацией микроЭВМ с магнитофоном, а также формат записи программ на магнитофон. Не менее подробно описана работа драйвера магнитофона, заложенного в ПЗУ пускового монитора БК. Для желающих разработать копировщик, который отвечал бы индивидуальным требованиям программиста, приведены основы "экранного" копирования программ. В пособии обсуждаются также вопросы, связанные с защитой программ от несанкционированного копирования, и принципы регенерирования файлов программным способом.

Пособие адресовано всем владельцам "Электроники БК-0010", в том числе тем, кто не так давно начал работать с компьютером. Впрочем, немало интересного для себя найдут и опытные пользователи.

### Советы начинающим

Советы касаются того, как хранить программы. Прежде всего, кассет должно быть несколько, даже если у вас пока что всего около десятка программ. Несколько кассет отводят под запись программ в машинных кодах, другие - для программ на Бейсике- БК, Форт-программ или для программ на Т-языке. Одну из кассет выделяют под разработку новых программ. Можно завести специальную архивную кассету и записывать по одному дубли каждой программы, чтобы исключить возможность их утраты. Чтобы реже регулировать положение универсальной головки магнитофона, имеет смысл завести специальную обменную кассету для записи программ на других магнитофонах. Это позволит переписывать новые программы на свои кассеты в стандартном положении головки.

Каждую архивную программу желательно записывать, как минимум, дважды. Возможны два варианта, хотя оба они не лишены недостатков. Можно записывать дубли подряд, но тогда трудно будет находить нужную программу на ленте, если магнитофон не

имеет счётчика. Во втором случае записывают один дубль с одной стороны кассеты и, перевернув кассету, тотчас же, - второй дубль. В этом случае программы располагаются компактно, но механическое повреждение ленты почти наверняка повлечёт за собой повреждение обоих дублей. Можно предложить ещё немало приёмов: записывать после каждой программы серию тональных сигналов или, например, проговаривать в микрофон название программы.

## ОБМЕН С МАГНИТОФОНОМ

Пока у БК нет дисководов, единственный способ сохранить информацию (программу или данные) - это записать её на магнитофон. Магнитофон может быть любым: отечественным или зарубежным, новым или старым, кассетным или катушечным. Единственное условие: он должен надёжно воспроизводить последовательности импульсов. Надёжность зависит от многих причин, но есть несколько основных параметров, которые чаще всего определяют, насколько удобно работать с домашним вычислительным комплексом. Во-первых, это уровень сигнала - обычно в пределах от 0,25 до 0,8 В. Оптимум здесь располагается примерно посередине. Второе условие - верное положение головки магнитофона. Бывает так: программы, которые вы составляете самостоятельно, хорошо записываются и считываются с магнитофонной ленты, а кассету, полученную из другого города, прочесть не удаётся. Дело в том, что магнитные головки во всех магнитофонах должны находиться строго на одинаковой стандартной высоте по отношению к ленте. Только в этом случае запись, сделанная на одном магнитофоне, подойдёт к другому аппарату. К сожалению, так случается далеко не всегда, особенно когда речь идёт о портативных кассетных магнитофонах. И все же в большинстве случаев положение можно поправить.

Прежде всего разыщите кассету с эталонной записью (существуют специальные кассеты для измерений, однако для этой цели может быть использована любая кассета с записью, сделанной на заведомо отрегулированном, например, новом магнитофоне). Слегка поворачивая регулировочный винт, нужно выставить стандартную высоту головки. Ориентироваться при этом следует на качество воспроизведения высоких частот. Отметив эталонное положение винта краской, можно попытаться прочесть программы, записанные на "неправильных" магнитофонах, также ориентируясь на то положение головки, когда лучше всего воспроизводятся высокие частоты. Однако настроить магнитофон "на слух", довольно сложно. В этом деле вам поможет БК-0010, если воспользуетесь программой С. Гуторенко ("Информатика и образование" №5, 1988, с.90.).

Вот ещё несколько несложных приёмов, позволяющих выйти из многих затруднительных ситуаций с магнитофоном. Если магнитофон хорошо согласован с компьютером, то можно ускорить ввод программ в компьютер. Для этого программу записывают на кассетный магнитофон со скоростью 2,38 см/с, а воспроизводят со скоростью 4,76 см/с. Для катушечного магнитофона это 9,53 см/с и 19,05 см/с или 4,76 см/с и 9,53 см/с. В тех моделях, где переключателя нет, его можно сделать. Последовательно с подстроечным резистором R в блоке управления скоростью устанавливают цепочку из выключателя и резистора, включённых параллельно (см. Приложение, рис. 1).

Разомкнутое положение переключателя соответствует режиму "Запись с БК", замкнутое - "Загрузка в БК". Если считывание с магнитофона идёт плохо, попробуйте обратный приём: запись на большей скорости, а воспроизведение на меньшей.

Если слишком мала амплитуда сигналов на линейном выходе магнитофона, можно попытаться воспользоваться гнездом для подключения наушников. Амплитуду импульсов на выходе - 0,5 В устанавливают ручкой громкости (для этого лучше всего воспользоваться осциллографом). Манипулируя ручками тембра, нужно отыскать такое положение, когда

форма импульсов будет наиболее близка к прямоугольной.

Штатным магнитофоном БК-0010 является "Электроника 302". Условимся, что в нашем распоряжении именно эта модель. Прежде чем переходить к описанию тонкостей программного обращения к магнитофону, добавим ещё два совета, касающихся выбора уровня записи.

Для кассет МК-60 уровень записи выставляют не выше максимального рекомендованного уровня для записи речи. Для кассет типа МК-90 и импортных кассет уровень должен быть таким, чтобы стрелка индикатора находилась в положении, аналогичном тому, которое она занимает при индикации нормального напряжения питания.

Прежде чем рассматривать детали обмена информацией компьютера и магнитофона, вспомним структуру микроЭВМ "Электроника БК".

### Структура микроЭВМ "Электроника БК"

МикроЭВМ состоит из нескольких функциональных блоков. Перечислим основные: процессор К1801ВМ1, оперативное запоминающее устройство (ОЗУ), постоянное запоминающее устройство (ПЗУ), контроллеры внешних устройств (в частности, имеется контроллер магнитофона) и сами внешние устройства.

Программируя в машинных кодах или на ассемблере, можно рассматривать все внешние устройства, как множество системных регистров, расположенных по адресам 177600-177777 (см. Приложение, рис. 2).

Микропроцессор - это также довольно сложное устройство, состоящее из различных блоков. На рис. 2 показано, что процессор содержит:

- РА - регистр адреса памяти;
- РДП - регистр данных памяти;
- РИ - регистр инструкций;
- Р0, Р1-Р7 - восемь регистров общего назначения;
- PSW - слово состояния процессора;
- УУ - устройство управления;
- АЛУ - арифметико-логическое устройство.

Каждому регистру соответствует слово памяти. Слово обычно эквивалентно двум байтам, а каждый байт в свою очередь состоит из восьми бит.

В регистре PSW четыре младших бита отображают состояние АЛУ после выполнения операции. Они известны под названием "флаги состояния АЛУ":

бит 0	бит 1	бит 2	бит 3
С	V	Z	N
перенос	переполнение	ноль	отрицательный результат

В распоряжении программиста находятся восемь регистров общего назначения. Однако регистры R7 и R6 обычно выполняют роль соответственно программного счётчика - PC и указателя стека - SP.

Регистр адреса памяти (РАП) сохраняет адрес инструкции, находящейся в стадии выполнения. Функция регистра PC содержать адрес той инструкции, которая будет выполняться следующей.

Во многих случаях программисту желательно выделить в оперативной памяти специальную область, называемую стековой памятью или стеком, для временного хранения информации во время выполнения программы. В БК-0010 для этого обычно отводят область ОЗУ, начиная с адреса 400 и до адреса 1000. В этой области программист может использовать регистр SP в качестве адресного указателя.

Сказанное нуждается в пояснении. Предположим, программа расположена в ОЗУ пользователя, начиная с адреса 1000. Для запуска подобной программы достаточно отдать команду 1000 G.

Работа микроЭВМ начинается с того, что значение 1000 помещается в регистр PC. Далее это значение переносится в регистр RAП, а содержимое регистра PC одновременно увеличивается на 2 (байта). Содержимое ячейки памяти с адресом 1000 по системной магистрали переносится в регистр РДП, из которого направляется в РИ. Устройство управления интерпретирует содержимое регистра РИ и определяет, требуется ли дополнительная информация. Если информация требуется, то устройство управления помещает содержимое регистра PC в регистр RAП, увеличивает содержимое PC на 2 и извлекает содержимое ячейки памяти, на которую указывает регистр RAП.

Последовательность операций повторяется до тех пор, пока в процессоре не окажется вся информация, необходимая для одной инструкции. Затем процессор выполняет инструкцию и переходит к извлечению следующей инструкции, подлежащей исполнению.

Это продолжается до тех пор, пока процессор не получит инструкцию HALT. В этом случае он останавливается. Содержимое регистра PC при этом будет указывать на ячейку, где расположено слово, следующее за тем, в котором находится инструкция HALT.

Разумеется, разобраться с тонкостями работы микроЭВМ, программированием на языке ассемблера или в машинных кодах не так уж просто, тем более, это не входит в задачи этого методического пособия. Читателям, желающим усовершенствовать свои знания в указанных областях, рекомендуем обратиться к уже имеющейся литературе (Лин В. PDP-11 и VAX-11. Архитектура ЭВМ и программирование на языке ассемблера, М.: Радио и связь, 1989; Осетинский Л.Г., Осетинский М.Г., Писаревский А.Н. Фокал для микро- и мини-компьютеров. Л.: Машиностроение, 1988).

### Блок параметров для работы с магнитофоном

Работой компьютера с магнитофоном управляет специальная программа, записанная непосредственно в постоянную память (ПЗУ) компьютера - так называемый драйвер магнитофона. Эта программа обеспечивает запись информации на ленту, чтение с ленты, а также выдаёт команды, включающие и выключающие магнитофон.

Обмен информацией с магнитофоном осуществляется со скоростью  $1200_{10}$  бод (бит/сек), т.е. за одну секунду записывается или считывается 1200 единиц информации. При этом на одном миллиметре ленты умещается около 25 бит. Таким образом, объём информации, записанной на одной кассете типа МК-60, может достигать  $500_{10}$  Кбайт.

Обращение к драйверу магнитофона осуществляется с помощью команды EMT 36 (здесь и далее цифры без индекса приведены в восьмеричной системе). Перед обращением по этой команде необходимо задать блок параметров. Блок состоит из 52 байт, содержание которых поясняет табличка:

<i>номер байта</i>	<i>содержание байта</i>
0	команда
1	результат исполнения команды
2,3	адрес обрабатываемого массива

4,5	длина обрабатываемого массива
6-25	имя обрабатываемого массива
26,27	адрес найденного массива
30,31	длина найденного массива
32,51	имя найденного массива

Прежде чем раскрывать назначение тех или иных параметров, добавим, что блок параметров размещают в произвольной области ОЗУ, причём с чётного адреса. Система позволяет использовать для размещения блока параметров область ОЗУ с адресами 320-371, однако в этом случае надо быть уверенным, что во время работы драйвера магнитофона не произойдёт взаимное перекрытие блока параметров и системного стека (драйвер магнитофона может сохранять в стековой области памяти до семи значений, их общая длина не превышает 16 байт).

Первый параметр определяет, какая из пяти возможных команд будет исполнена. Код выбранной команды заносят в нулевой байт блока параметров:

<i>код</i>	<i>команда</i>
0	стоп
1	пуск двигателя
2	запись массива
3	чтение массива
4	фиктивное чтение массива

Байты со второго по двадцать пятый требуется заполнить информацией об адресе, длине и имени обрабатываемого массива, а адрес начала блока параметров помещают в регистр R1. Поясним сказанное на примере.

Предположим, в разрабатываемой программе требуется записать на магнитофон массив, начинающийся с адреса 1000 длиной 2000, присвоив ему имя "МАССИВ". Для эту понадобится короткая программка в машинных кодах, которую можно набрать, например, с адреса 3000:

АДРЕС	КОМАНДА	МНЕМОНИКА АССЕМБЛЕРА	ПОЯСНЕНИЯ
003000	012701	MOV #320, R1	Блок параметров будет располагаться с адреса 320. Запомним этот адрес в стеке. В блок параметров первой записывается команда 2 (запись). Адрес начала записываемого массива равен 1000. Длина данного массива равна 2000 байт.
003002	000320		
003004	010146	MOV R1, -(SP)	
003006	012721	MOV #000002, (R1)+	
003010	000002		
003012	012721	MOV #1000, (R1)+	
003014	001000		
003016	012721	MOV #2000, (R1)+	
003020	002000		
003022	012721	MOV #160755, (R1)+	
003024	160755		
003026	012721	MOV #171763, (R1)+	
003030	171763		
003032	012721	MOV #173751, (R1)+	
003034	173751		
003036	012702	MOV #5, R2	
003040	000005		
003042	012721	MOV #20040, (R1)+	
003044	020040		

003046	077203	SOB R2,003042	
003050	012601	MOV (SP)+,R1	Из стека извлекается адрес начала блока параметров (320) и записывается в R1.
003052	104036	EMT 36	Команда начать обмен с магнитофоном.
003054	000000	HALT	Останов после исполнения команды обмена с магнитофоном.

Если при чтении задан нулевой адрес (байты 2 и 3), то запись массива в ОЗУ производится, начиная с адреса, с которого программа была записана на ленту.

После выполнения операции первый байт блока параметров содержит информацию о результате. Возможны четыре варианта:

<i>код</i>	<i>ответ</i>
0	операция завершена без ошибок
1	имя текущего массива не совпадает с заданным в блоке параметров
2	ошибка контрольной суммы
4	останов по команде оператора

Выполняя команду EMT 36, драйвер магнитофона запускает двигатель магнитофона, производит операцию записи или считывания, а затем останавливает магнитофон.

Нажатие клавиши СТОП на клавиатуре БК прерывает работу драйвера магнитофона. При этом двигатель магнитофона останавливается, в байт ответа (байт 1) блока параметров помещается константа 4, и управление возвращается программе, вызвавшей драйвер магнитофона.

Если при чтении найден файл, имя которого отличается от заданного, в байт ответа помещается 1 и управление возвращается вызвавшей программе. При этом двигатель магнитофона остаётся включённым. В этом случае имя найденного массива целесообразно вывести на экран, а для поиска нужного массива вновь передать управление драйверу магнитофона, не изменяя блок параметров.

### Формат записи файла на магнитофон

Для записи информации на магнитную ленту БК формирует последовательность из логических нулей, логических единиц, синхроэлемента и маркеров. Каждый элемент при передаче на магнитофон преобразуется в два импульса одинаковой длины, но различной полярности. Используется, так называемая, широтно-импульсная модуляция: самыми короткими по длительности являются логический нуль и синхроэлемент, длительность логической единицы в два раза больше, ещё в два раза длиннее выходной маркер, а самым длинным является входной маркер, который в два раза шире выходного (см. Приложение, рис. 3).

При считывании информации компьютер измеряет длительность поступающих импульсов. Логические 0 и 1 различаются простым методом - измеренные длительности сравнивают с пороговым значением, в полтора раза большим длительности логического нуля. Если измеренная длительность меньше порога, то принимается решение, что получен ноль, в противном случае, компьютер считает, что принята единица.



Из перечисленных элементов формируется файл для записи на магнитофон. Файл состоит из следующих частей (см. Приложение, рис. 4):

- входной настроечной последовательности;
- настроечной последовательности заглавия;
- заглавия файла;
- настроечной последовательности массива;
- массива;
- контрольной суммы массива;
- выходной настроечной последовательности.

Каждая часть в свою очередь строится из определённой последовательности элементов (см. Приложение, рис. 5).

Настроечная последовательность состоит из входного маркера, некоторого числа синхроимпульсов, выходного маркера, логической единицы и синхроэлемента.

Во входной настроечной последовательности входной маркер в несколько раз длиннее выходного - 10000 синхроэлементов. Последовательность предназначена для определения начала файла, измерения порогового значения, синхронизации.

В настроечных последовательностях заглавия и массива длительность входного маркера равна длительности синхроэлемента. Эти последовательности предназначены для задержки по времени начала передачи этих частей и синхронизации. Задержка необходима для того, чтобы можно было принять решение, тот ли файл найден.

Выходная настроечная последовательность имеет 400 синхроэлементов, и её входной маркер более чем в два раза превышает длительность выходного маркера. Её назначение - завершить чтение файла, если по каким-либо причинам произошёл сбой синхронизации.

Заглавие файла, массив и контрольная сумма массива состоят из последовательности логических единиц и нулей, перемежаемых синхроэлементами.

В свою очередь заглавие состоит из слова, определяющего адрес загрузки файла, слова, определяющего длину файла, и 20 байт, определяющих имя файла.

Массив фактически определяет записываемый файл.

Контрольная сумма состоит из одного слова и служит для контроля правильности считывания.

При блочной записи файла, например, при записи программ на Бейсике, файл разбивается на блоки, причём каждый отдельный блок записывается в описанном формате.

Структура файла для записи на магнитофон формируется с помощью служебной программы - драйвера магнитофона, записанной в ПЗУ компьютера БК.

## **ДРАЙВЕР МАГНИТОФОНА**

Прежде чем говорить непосредственно о драйвере магнитофона, отметим, что для обмена информацией с магнитофоном в микроЭВМ БК-0010 имеется контроллер магнитофона, к которому можно обращаться через регистр управления системными внешними устройствами (см. Приложение, рис. 2).

Этот регистр используется для задания адреса начального пуска процессора, а также для управления внешними устройствами микроЭВМ. Адрес регистра - 177716. Поясним назначение отдельных разрядов регистра:

0-3: для задания режимов работы процессора (доступны только для чтения);

4-7: для управления внешними устройствами ЭВМ (имеют выход на четырёхразрядный выходной регистр системного порта и четырёхразрядный входной регистр системного порта. Выходной регистр в отличие от входного доступен только по записи).

#### Выходной регистр

1. Разряд 4 - передача информации на ТЛГ-линию (исходное состояние разряда "логическая 1");
2. Разряд 5 - передача информации на магнитофон, либо сигнала готовности на ТЛГ-линию (исходное состояние - "логический 0"), одновременный обмен информацией с магнитофоном и по последовательному каналу недопустимы;
3. Разряд 6 - передача информации на магнитофон и сигнала при нажатии клавиши микроЭВМ (исходное состояние - "логический 0");
4. Разряд 7 - для управления двигателем магнитофона, причём "логическая 1" соответствует команде "СТОП", "логический 0" - команде "ПУСК". Исходное состояние разряда - "логическая 1".

#### Входной регистр

1. Разряд 4 - для чтения информации с ТЛГ-линии;
2. Разряд 5 - для чтения информации с магнитофона;
3. Разряд 6 - индикатор нажатия клавиши ("логический 0" означает, что клавиша нажата, "логическая 1" - клавиша отжата) используется при реализации режима "Повтор";
4. Разряд 7 - для чтения сигнала готовности с ТЛГ-линии.
5. Разряды 8-15 для задания адреса пуска процессора при включении питания, при этом младший байт адреса принимается равным 0. В микроЭВМ БК-0010 адрес начального пуска процессора равен 100000. Разряды доступны только для чтения.

В процессе работы драйвер магнитофона использует некоторые ячейки системной области:

- 300 - индикатор полярности передачи импульсов;
- 301 - байт ответа выполнения операции обмена с магнитофоном;
- 302 - индикатор фиктивного чтения;
- 304 - шаг смещения байтов при чтении;
- 306 - адрес блока параметров;
- 310 - значение стека после входа в драйвер;
- 312 - значение контрольной суммы массива;
- 314 - пороговое значение, измеренное по настроечной последовательности.

Начинающим названия переменных могут казаться непонятными, однако их смысл должно прояснить дальнейшее изложение.

Рассмотрим подробнее драйвер магнитофона, записанный в ПЗУ пускового монитора. Подпрограммы, составляющие драйвер, будем записывать по уже знакомому нам шаблону: в первом столбце - реальные адреса команд, во втором столбце - содержимое соответствующих ячеек; далее будет приведена команда в мнемонике ассемблера, а в правом столбце даны пояснения, для чего служит та или иная команда.

### **Управляющая программа драйвера магнитофона**

Входные параметры: R1 - адрес начала блока параметров операции обмена с магнитофоном.

АДРЕС	КОМАНДА	МНЕМОНИКА	АССЕМБЛЕРА	ПОЯСНЕНИЯ
116076 116100	004467 172244	JSR R4,	110346	Начало драйвера магнитофона. Содержимое регистров R0-R4 запомнили в стеке.
116102 116104	010167 062200	MOV R1,	306	В 306 ячейку записали адрес блока параметров.
116106 116110	012703 177716	MOV #177716,	R3	В регистр R3 записали адрес регистра управления внешними устройствами.
116112 116114	005067 062162	CLR 300		Установили, что будем анализировать длительность положительных импульсов поступающих с магнитофона.
116116 116120	005067 062160	CLR 302		Установили, что речь идёт о чтении с магнитофона, а не о фиктивном чтении.
116122 116124	013746 000004	MOV @#4,-	(SP)	Запись в стек адреса обработки команды СТОП программы, из которой произошло обращение к драйверу магнитофона.
116126 116130 116132	012737 116214 000004	MOV #116214,@#4		Определили, что если работа драйвера будет прервана командой СТОП, то управление будет передано на адрес 116214.
116134 116136	010667 062150	MOV SP,	310	Значение указателя стека записывается в ячейку 310.
116140	111100	MOVB (R1),	R0	В регистр R0 из блока параметров считывается команда. Далее следует её анализ.
116142 116144 116146 116150	001003 012713 000220 000430	BNE 116152 MOV #220,	(R3)	Если команда - "0", то на магнитофон выдаётся команда СТОП и управление передаётся на выход драйвера (ячейка 116232).
116152 116154 116156 116160 116162	005300 001003 012713 000020 000423	DEC R0 BNE 116164 MOV #20,	(R3)	Если команда "1", то на магнитофон выдаётся команда "ПУСК" и управление передаётся на выход драйвера (ячейка 116232).
116164 116166 116170 116172 116174	005300 001003 004767 000062 000416	DEC R0 BNE 116176 JSR PC,	116256	Если команда "2", то файл записывается на магнитофон (подпрограмма 116256) и управление передаётся на выход драйвера (ячейка 116232).
116176 116200 116202 116204 116206 116210 116212	005300 001402 005267 062074 004767 000426 000407	DEC R0 BEQ 116206 INC 302 JSR PC,	116640	Если команда "3", то считывается файл с магнитофона (подпрограмма 116640), а если "4", то устанавливается в ячейке 302 флаг фиктивного чтения, производится фиктивное чтение (подпрограмма 116640), после чего управление переходит на выход драйвера.

## Обработка прерывания после нажатия клавиши "СТОП".

116214	016706	MOV 310, SP	Восстанавливаем значение указателя стека, бывшее при входе в драйвер.
116216	062070		
116220	112767	MOVB #4, 301	В байт 301 направляется сообщение "Останов по команде оператора".
116222	000004		
116224	062053		
116226	012713	MOV #220, (R3)	На магнитофон выдана команда СТОП.
116230	000220		

## Выход драйвера магнитофона

116232	016701	MOV 306, R1	Во второй байт блока параметров записан результат выполнения операции обмена информацией с магнитофоном.
116234	062050		
116236	116761	MOVB 301, 1 (R1)	
116240	062037		
116242	000001		
116244	012637	MOV (SP)+, @#4	Восстановлен адрес обработки прерывания после нажатия клавиши СТОП.
116246	000004		
116250	004467	JSR R4, 110362	Восстановлены значения регистров R0-R4, которые они имели до обращения к драйверу магнитофона.
116252	172106		
116254	000207	RTS PC	Вернулись в вызывавшую программу.

## Подпрограмма записи на магнитофон

**Входные параметры:** R1 - адрес блока параметров, R3 - адрес 177716.

116256	012713	MOV #20, (R3)	Отдана команда ПУСК на магнитофон. Магнитофон включён.
116260	000020		
116262	016105	MOV 2 (R1), R5	В регистре R5 установлен адрес начала записываемого массива.
116264	000002		
116266	016104	MOV 4 (R1), R4	В регистр R4 установлена длина записываемого массива. Если длина записываемого массива равна нулю, то осуществляется передача управления на выход подпрограммы.
116270	000004		
116272	001441	BEQ 116376	
116274	004767	JSR PC, 116622	Подсчитана контрольная сумма записываемого массива (подпрограмма 116622) и направлена в ячейку 312 на хранение.
116276	000322		
116300	010067	MOV R0, 312	
116302	062006		
116304	012700	MOV #10000, R0	Выдана входная настроечная последовательность.
116306	010000		
116310	004767	JSR PC, 116404	
116312	000070		
116314	012100	MOV (R1)+, R0	Записано заглавие файла вместе с настроечной последовательностью (при этом содержимое регистра R1 - это адрес ячейки в блоке параметров. в которой записан адрес загрузки
116316	012702	MOV #24, R2	
116320	000024		
116322	004767	JSR PC, 116474	
116324	000146		

			файла, а в R2 записана длина заглавия - 24 байта).
116326	016701	MOV 306,R1	Массив записан вместе с настроечной последовательностью массива. В регистре R1 - адрес начала массива, в регистр R2 - длина записываемого массива. (Адрес подпрограммы, записывающей массив с настроечной последовательностью, - 116474).
116330	061754		
116332	016102	MOV 4(R1),R2	
116334	000004		
116336	016101	MOV 2(R1),R1	
116340	000002		Записана контрольная сумма. В регистре R1 - адрес, равный 312, в регистре R2 - длина контрольной суммы, равная двум байтам. (Адрес подпрограммы записи массива без настроечной последовательности - 116504).
116342	004767	JSR PC,116474	
116344	000126		
116346	012701	MOV #312,R1	
116350	000312		
116352	012702	MOV #2,R2	Записана выходная настроечная последовательность. В ней 400 синхроэлементов (R0), длины положительного (R5) и отрицательного (R4) импульсов равны 400 тактам счета по команде SOB.
116354	000002		
116356	004767	JSR PC,116504	
116360	000122		
116362	012700	MOV #400,R0	
116364	000400		На магнитофон выдана команда СТОП.
116366	010004	MOV R0,R4	
116370	010005	MOV R0,R5	
116372	004767	JSR PC,116404	
116374	000006		
116376	012713	MOV #220,(R3)	Возврат в вызвавшую программу.
116400	000220		
116402	000207	RTS PC	

### Подпрограмма выдачи настроечной последовательности

**Входные параметры:** R5 - длительность положительного импульса входного маркера, R4 - длительность отрицательного импульса входного маркера, R0 - число синхроэлементов, R3 - адрес 177716.

116404	012713	MOV #160,(R3)	Выдача синхроэлемента. Первый синхроэлемент - это входной маркер. По адресу 116412 записана длительность положительного импульса синхроэлемента, а по адресу 116424 - длительность отрицательного импульса.
116406	000160		
116410	062705	ADD #27,R5	
116412	000027		
116414	077501	SOB R5,116414	
116416	012713	MOV #20,(R3)	
116420	000020		
116422	062704	ADD #26,R4	
116424	000026		
116426	077401	SOB R4,116426	
116430	077013	SOB R0,116404	Подсчёт числа синхроэлементов в последовательности.
116432	012713	MOV #120,(R3)	Запись выходного маркера. По адресу 116440 записана длительность положительного импульса выходного маркера. По адресу 116452 - длительность отрицательного импульса выходного маркера.
116434	000120		
116436	012705	MOV #151,R5	
116440	000151		
110442	077501	SOB R5,116442	
116444	012713	MOV #60,(R3)	
116446	000060		
116450	012705	MOV #144,R5	
116452	000144		
116454	077501	SOB R5,116454	
116456	005204	INC R4	Запись логической единицы

116460	010200	MOV R2,R0	вместе с синхроэлементом.
116462	010402	MOV R4,R2	[Адрес подпрограммы записи
116464	004767	JSR PC,116544	логической единицы - 116544, при
116466	000054		этом должен быть передан всего один
116470	010002	MOV R0,R2	байт (R4=1), а из этого байта
			должен быть записан всего
			один бит, R1=1].
116472	000207	RTS PC	Возврат в вызывавшую программу.

### Подпрограмма передачи массива на магнитофон

**Входные параметры:** R1 - адрес начала массива, R2 - его длина, R3 - адрес 177716.

116474	012700	MOV #10,R0	Передана настроечная
116476	000010		последовательность массива.
116500	004767	JSR PC,116404	
116502	177700		
116504	112100	MOVB (R1)+,R0	В R0 записано значение следующего
			байта передаваемого массива.
116506	012704	MOV #10,R4	Регистр R4 определяется, как счётчик
116510	000010		числа битов в байте.
116512	006200	ASR R0	Каково значение следующего бита?
116514	103413	BCS 116544	Если 1, то управление передаётся
			на блок выдачи логической 1
			(адрес 116544).
116516	012713	MOV #160,(R3)	В противном случае на магнитофон
116520	000160		выдаётся логический 0. По адресу
116522	012705	MOV #30,R5	116524 записана длительность
116524	000030		положительного импульса
116526	077501	SOB R5,116526	логического 0, по адресу 116536
116530	012713	MOV #20,(R3)	длительность отрицательного
116532	000020		импульса.
116534	012705	MOV #27,R5	
116536	000027		
116540	077501	SOB R5,116540	
116542	000412	BR 116570	Переход на выдачу синхроэлемента.
116544	012713	MOV #120,(R3)	Выдача логической 1.
116546	000120		По адресу 116552 записана
116550	012705	MOV #63,R5	длительность положительного
116552	000063		импульса логической 1,
116554	077501	SOB R5,116554	а по адресу 116564 -
116556	012713	MOV #60,(R3)	длительность отрицательного
116560	000060		импульса.
116562	012705	MOV #63,R5	
116564	000063		
116566	077501	SOB R5,116566	
116570	012713	MOV #160,(R3)	На магнитофон передан синхро-
116572	000160		элемент. По адресу 116576
116574	012705	MOV #30,R5	записана длительность
116576	000030		положительного импульса
116600	077501	SOB R5,116600	синхроэлемента, а по
116602	012713	MOV #20,(R3)	адресу 116610 - длительность
116604	000020		его отрицательного
116606	012705	MOV #26,R5	импульса.
116610	000026		
116612	077501	SOB R5,116612	

116614	077442	SOB R4,116512	Вычисляется число переданных битов.
116616	077246	SOB R2,116504	Вычисляется число переданных байт.
116620	000207	RTS PC	Возврат в вызвавшую программу.

### Подпрограмма подсчёта контрольной суммы

Входные параметры: R5 - адрес начала массива, R4 - его длина.

116622	005000	CLR R0	Очистка R0.
116624	005002	CLR R2	Очистка R2.
116626	152502	BISB (R5)+,R2	Текущий байт установлен в R2.
116630	060200	ADD R2,R0	Суммирование значений байтов. Если при суммировании получен перенос, то он прибавляется к сумме.
116632	005500	ADC R0	
116634	077405	SOB R4,116624	Подсчёт числа суммированных байт.
116636	000207	RTS PC	Возврат в вызывавшую программу.

### Подпрограмма чтения с магнитофона

Входные параметры: в R3 - адрес 177716.

116640	012705	MOV #40,R5	Запись в регистр R5 маски для чтения информации с магнитофона.
116642	000040		
116644	012713	MOV #20,(R3)	Выдача на магнитофон команды "ПУСК".
116646	000020		
116650	012767	MOV #1,304	Установка шага чтения равным 1 (значение запоминается в ячейке 304).
116652	000001		
116654	061426		
116656	005067	CLR 300	Длительность элементов будет измеряться по положительным импульсам, поступающим с магнитофона.
116660	061416		
116662	004767	JSR PC,116712	Поиск настроечной последовательности с помощью подпрограммы, начинающейся с адреса 116712.
116664	000024		
116666	004767	JSR PC,117070	Чтение заголовка найденного файла с помощью подпрограммы чтения заголовка программы, начинающейся с адреса 117070.
116670	000176		
116672	105767	TSTB 301	Если найден заданный файл, производится чтение массива, если имя найденного файла не совпадает с заданным, то следует переход на завершение подпрограммы.
116674	061403		
116676	001004	BNE 116710	
116700	004767	JSR PC,117154	
116702	000250		
116704	012713	MOV #220,(R3)	Выдача на магнитофон команды "СТОП".
116706	000220		
116710	000207	RTS PC	Возврат в вызвавшую программу.

## Подпрограмма поиска настроечной последовательности

Входные параметры: в R3 - адрес 177716, в R5 - маска 40.

116712	012702	MOV #4000,R2	Запись в R2 длины участка, по которому будет оцениваться, найдено ли начало файла.
116714	004000		
116716	005000	CLR R0	Обнуление счётчиков R0 и R4.
116720	005004	CLR R4	
116722	030513	BIT R5, (R3)	Пропуск отрицательного импульса.
116724	001776	BEQ 116722	
116726	005204	INC R4	Измерение длительности положительного импульса.
116730	030513	BIT R5, (R3)	
116732	001375	BNE 116726	
116734	160400	SUB R4,R0	Если разность измеренного ранее и вновь полученного значения превышает 2, вновь начинается поиск настроечной последовательности.
116736	100403	BMI 116746	
116740	020027	CMP R0,#2	
116742	000002		
116744	101362	BHI 116712	
116746	010400	MOV R4,R0	Запись измеренного значения.
116750	077215	SOB R2,116720	Счёт числа измеренных элементов.
116752	005000	CLR R0	Определение для данного файла порогового значения (по двумстам измерениям), необходимого для различения логических "0" и "Г", а также запись этого значения в ячейку 314.
116754	012702	MOV #200,R2	
116756	000200		
116760	004767	JSR PC,117376	
116762	000412		
116764	060400	ADD R4,R0	
116766	077204	SOB R2,116760	
116770	012702	MOV #7,R2	
116772	000007		
116774	006200	ASR R0	
116776	077202	SOB R2,116774	
117000	010004	MOV R0,R4	
117002	006204	ASR R4	
117004	060400	ADD R4,R0	
117006	010067	MOV R0,314	
117010	061302		
117012	005004	CLR R4	Поиск первого импульса выходного маркера. Если этот импульс оказывается положительным, то содержимое байта 300 остаётся неизменным. Если импульс оказывается отрицательным, то в байт 300 записывается 1. Если найденный импульс не является ни маркером и ни синхрозлементом, то восстанавливается значение указателя стека и осуществляется переход на новый поиск настроечной последовательности. После того, как маркер найден, производится фиктивный приём логической единицы и затем следует возврат в вызывавшую программу.
117014	005204	INC R4	
117016	030513	BIT R5, (R3)	
117020	001375	BNE 117014	
117022	020400	CMP R4,R0	
117024	101010	BHI 117046	
117026	005004	CLR R4	
117030	005204	INC R4	
117032	030513	BIT R5, (R3)	
117034	001775	BEQ 117030	
117036	020400	CMP R4,R0	
117040	103764	BLO 117012	
117042	105267	INCB 300	
117044	061232		
117046	006300	ASL R0	
117050	020400	CMP R4,R0	
117052	101004	BHI 117064	
117054	004767	JSR PC,117376	
117056	000316		



117060	000207	RTS PC
117062	005726	TST (SP)+
117064	005726	TST (SP)+
117066	000673	BR 116656

### Подпрограмма чтения заголовка файла

Входные параметры: R3 - адрес 177716, R5 - маска 40.

117070	016701	MOV 306,R1	Запись в регистр R1 адреса загрузки заголовка текущего файла. Адрес вычисляется, как сумма адреса блока параметров и константы 26. В регистр R2 записывается длина заголовка - 24 байта. После установки параметров следует переход на подпрограмму приёма массива с настроечной последовательностью.
117072	061212		
117074	062701	ADD #26,R1	
117076	000026		
117100	012702	MOV #24,R2	
117102	000024		
117104	004767	JSR PC,117260	
117106	000150		
117110	012702	MOV #10,R2	Имя найденного файла сравнивается с заданным на чтение.
117112	000010		Если найденное имя отлично от заданного, следует переход на адрес 117146.
117114	024161	CMP -(R1),-24(R1)	
117116	177754		
117120	001012	BNE 117146	
117122	077204	SOB R2,117114	
117124	014167	MOV -(R1),266	Если найден заданный на чтение файл, то в ячейку 266 записывается его длина, а в ячейку 264 - адрес, указывающий, куда будет загружен файл. Далее осуществляется передача управления на выход из подпрограммы.
117126	061136		
117130	016167	MOV -26(R1),264	
117132	177752		
117134	061126		
117136	001005	BNE 117152	
117140	014167	MOV -(R1),264	
117142	061120		
117144	000402	BR 117152	
117146	105267	INCB 301	Если не найден заданный файл, то запись в ячейку 301 ответа 1.
117150	061127		
117152	000207	RTS PC	Возврат в вызвавшую программу.

### Подпрограмма чтения массива файла

Входные параметры: R3 - адрес 177716, R5 - маска 40.

117154	016701	MOV 264,R1	В регистр R1 записывается адрес, по которому будет считываться массив.
117156	061104		
117160	016702	MOV 266,R2	В регистр R2 записывается длина считываемого массива.
117162	061102		
117164	166767	SUB 302,304	Проверка - было ли задано фиктивное чтение. Если "да", то фиктивное чтение будет производиться в ячейку 312.
117166	061112		
117170	061112		
117172	001002	BNE 117200	
117174	012701	MOV #312,R1	
117176	000312		
117200	004767	JSR PC,117260	Принят массив с настроечной последовательностью.
117202	000054		

117204	012701	MOV #312,R1	Контрольная сумма принята в ячейку 312. Подпрограмма приёма массива без настроечной последовательности начинается с адреса 117336).
117206	000312		
117210	012702	MOV #2,R2	
117212	000002		
117214	004767	JSR PC,117336	
117216	000116		
117220	005767	TST 302	Если задано обычное чтение файла с магнитофона, то в регистр R5 записывается адрес начала массива, в регистр R4 - длина массива. С помощью подпрограммы, начинающейся с адреса 116622, производится подсчёт контрольной суммы массива.
117222	061056		
117224	001014	BNE 117256	
117226	016705	MOV 264,R5	
117230	061032		
117232	016704	MOV 266,R4	
117234	061030		
117236	004767	JSR PC,116622	
117240	177360		
117242	020067	CMP R0,312	
117244	061044		
117246	001403	BEQ 117256	
117250	112767	MOVB #2,301	
117252	000002		
117254	061023		
117256	000207	RTS PC	Возврат в вызвавшую программу.

### Подпрограмма приёма массива с настроечной последовательностью

**Входные параметры:** R1 - адрес загрузки массива; R2 - длина загружаемого массива, R3 - адрес 177716, R5 - маска 40.

117260	016700	MOV 314,R0	Запись в R0 порогового значения.
117262	061030		
117264	105767	TSTB 300	Принятие решения - по какому импульсу (положительному или отрицательному) оценивать приём информации.
117266	061010		
117270	001007	BNE 117310	
117272	005004	CLR R4	Поиск положительного импульса маркера, при этом пропускаются синхрозлементы. Для этого ведётся поиск положительного импульса, длина которого больше порогового значения. Если такой импульс найден, то осуществляется переход на адрес 117324 для принятия решения о том, что найдено начало файла.
117274	005204	INC R4	
117276	030513	BIT R5,(R3)	
117300	001375	BNE 117274	
117302	020400	CMP R4,R0	
117304	103772	BLO 117272	
117306	000406	BR 117324	
117310	005004	CLR R4	Поиск выходного маркера по отрицательному импульсу аналогично тому, как это делалось для положительного импульса, то есть измеряется длительность импульса и сравнивается с пороговым значением. Синхроимпульсы при этом пропускаются.
117312	005204	INC R4	
117314	030513	BIT R5,(R3)	
117316	001775	BEQ 117312	
117320	020400	CMP R4,R0	
117322	103772	BLO 117310	
117324	006300	ASL R0	Длительность одного импульса маркера должна быть меньше удвоенного порогового значения. Если это не
117326	020400	CMP R4,R0	
117330	101254	BNI 117062	

117332	004767	JSR PC,117376	так, то управление вновь передается на поиск начала файла.
117334	000040		Если маркер найден, то осуществляется фиктивное считывание логической единицы настроечной последовательности и производится приём массива.

### Подпрограмма приёма массива без настроечной последовательности

**Входные параметры:** R1 - адрес загрузки, R2 - длина массива, R3 - адрес 177716, R5 маска 40.

117336	012700	MOV #10,R0	Запись длины байта в битах в регистр R0.
117340	000010		
117342	004767	JSR PC,117376	Измерение длительности принимаемого элемента (адрес подпрограммы 117376).
117344	000030		
117346	020467	CMP R4,314	Сравнение измеренных значений с пороговым.
117350	060742		
117352	101002	BNI 117360	Если измеренное значение меньше порогового или равно ему, то записывается логический ноль,
117354	000241	CLC	а если больше порогового,
117356	000401	BR 117362	то записывается логическая единица. Адрес байта, куда производится запись, указывает значение из регистра R1.
117360	000261	SEC	
117362	106011	RORB (R1)	
117364	077012	SOB R0,117342	Подсчёт числа битов в байте, в который производится запись.
117366	066701	ADD 304,R1	Как только все биты текущего байта установлены, то указатель записи (регистр R1) переводится на следующий байт.
117370	060712		
117372	077217	SOB R2,117336	Подсчёт числа принятых байт массива.
117374	000207	RTS PC	Возврат в вызвавшую программу.

### Подпрограмма измерения длительности принимаемого элемента

**Входные параметры:** R3 - адрес 177716, R5 - маска 40.

117376	005004	CLR R4	Очистка регистра R4.
117400	105767	TSTB 300	Определение - по какому импульсу положительному или отрицательному
117402	060674		необходимо измерять длительность элемента.
117404	001013	BNE 117434	Измерения по положительному импульсу.
117406	030513	BIT R5,(R3)	Пропускается синхроэлемент.
117410	001376	BNE 117406	
117412	030513	BIT R5,(R3)	
117414	001776	BEQ 117412	
117416	005204	INC R4	Измерение длительности положительного импульса. Суммируется полу-
117420	030513	BIT R5,(R3)	

117422	001375	BNE 117416	ценное значение и длительность
117424	005204	INC R4	отрицательного импульса. В реги-
117426	030513	BIT R5, (R3)	стре R4 подсчитывается длитель-
117430	001775	BEQ 117424	ность принимаемого элемента.
117432	000207	RTS PC	Возврат в вызвавшую программу.
117434	030513	BIT R5, (R3)	Измерения по отрицательному
117436	001776	BEQ 117434	импульсу. Суммируется полученное
117440	030513	BIT R5, (R3)	значение и длительность положитель-
117442	001376	BNE 117440	ного импульса. В регистре R4 под-
117444	005204	INC R4	считывается длительность принима-
117446	030513	BIT R5, (R3)	емого элемента.
117450	001775	BEQ 117444	
117452	005204	INC R4	
117454	030513	BIT R5, (R3)	
117456	001375	BNE 117452	
117460	000207	RTS PC	Возврат в вызвавшую программу.

Эта подпрограмма - последняя из тех, что в совокупности образуют драйвер магнитофона "Электроники БК-0010". Рассматривая копировщики, мы будем опираться на это описание. Отметим, что разработчики драйвера выбрали не самый удачный вид модуляции для передачи информации. Алгоритм обработки сигналов, поступающих с магнитофона, также не оптимален. Автору удалось доработать драйвер таким образом, что надёжность считывания файлов увеличилась в 16 раз.

## ОСНОВЫ "ЭКРАННОГО" КОПИРОВАНИЯ ФАЙЛОВ

Перезапись программ - эта операция, с которой пользователи микроЭВМ сталкиваются, пожалуй, чаще, чем с любой другой. В то же время именно эта операция вызывает наибольшие сложности у начинающих пользователей БК.

При копировании программ важно иметь возможность визуально контролировать процесс считывания программы с магнитной ленты в память машины. Эффективным приёмом является считывание программы в оперативную память экрана и последующая запись её на магнитофон также из экранной памяти.

Основная проблема при реализации этого способа связана с тем, что в заголовке файла указан вовсе не "экранный" (условимся в дальнейшем опускать кавычки) адрес загрузки, который необходимо сохранить, несмотря на то, что копирование осуществляется "через экран". Как разрешить эту проблему?

Очевидно, что для экранного копирования требуется специальный копировщик. Написать его не так сложно, как это обычно представляется начинающим программистам. Дело в том, что копировщик может использовать большинство подпрограмм драйвера магнитофона и лишь несколько подпрограмм придётся изменить.

Поясним сказанное на примере. Пусть вновь созданный копировщик будет записан в ОЗУ с адреса А. Необходимо прежде всего написать управляющую программу, аналогичную управляющей программе драйвера магнитофона. Остальные подпрограммы, входящие в копировщик, будут повторять соответствующие подпрограммы драйвера, за исключением значений нескольких ячеек. Чтобы не переписывать полностью подпрограммы драйвера, условимся, что адреса ячеек новых подпрограмм будем обозначать в виде А (адрес соответствующей ячейки драйвера ПЗУ). Например, начало управляющей программы вновь разрабатываемого копировщика, соответствующее началу управляющей программы драйвера, может находиться по любому адресу А. Однако в

тексте оно будет представлено в виде:

A (116076) 004467  
 A (116100) 172244  
 A (116102) 010167  
 A (116104) 062200

...

В то же время A (116076) может быть равно 1000.

Первая операция, которую должен выполнить копировщик, получив команду на считывание файла, это операция установки экрана. Для этого достаточно выполнить следующую последовательность команд, записанных, например, с адреса A:

A	012700	MOV #214, R0	Выдаётся команда перехода в
A+2	000214		режим расширенной памяти и, тем
A+4	104016	EMT 16	самым, установка отображения на
A+6	104016	EMT 16	экран с адреса 70000. Вторая
			команда EMT 16 обеспечивает об-
			ратный переход к обычному распре-
			делению памяти и установку отобра-
			жения на экран с адреса 40000.

В результате перехода в режим расширенной памяти и обратно экран очистится, и левый верхний угол экрана будет соответствовать адресу 40000. Именно с этого адреса будем записывать программу при чтении с магнитофона.

Ещё одна подпрограмма, которую требуется доработать - это подпрограмма чтения с магнитофона. Изменения незначительны - корректируется только адрес подпрограммы чтения массива файла. В копировщике по адресу A (116702) необходимо записать смещение адреса новой подпрограммы. Читатель, по-видимому, знаком с понятием "смещение адреса", однако, напомним, что в командах, использующих адресацию 67, следующее за командой слово содержит смещение адреса оператора, определяемое как разность между адресом операнда и адресом ячейки, следующей за ячейкой, содержащей смещение.

В копировщик должна входить и подпрограмма чтения массива файла. Исходная подпрограмма драйвера нуждается в небольшой корректировке. Прежде всего изменяют первую команду.

Вместо команды драйвера

117154	016701	MOV 264, R1
117156	061104	

вводят команду копировщика

A(117154)	012701	MOV #40000, R1
A(117156)	040000.	

Коррекция обеспечивает считывание файла в экранную память.

В копировщик должна входить также скорректированная подпрограмма записи массива файла, в которой необходимо заменить команду драйвера

116262	016105	MOV 4(R1), R5
116264	000004	

на команду

```
A(116262) 012705 MOV #40000, R1
A(116264) 040000,
```

а команду

```
116336 016101 MOV 2(R1), R1
116340 000002
```

на команду

```
A(116336) 012701 MOV #40000, R1
A(116340) 040000,
```

оставив все остальные команды без изменений.

После того, как файл считан в экранное ОЗУ, на экран требуется вывести параметры программы: имя, адрес загрузки, длину, адрес запуска, контрольную сумму, а также сообщения о правильности выполнения чтения. Для вывода этих параметров требуется выделить часть экранного ОЗУ.

Есть два способа решить эту задачу. Простейший предусматривает ограничение длины копируемых файлов - этим способом воспользовался автор нескольких известных программ В.А. Ярошенко, создавая свои программы COPY2, COPY4 и другие, где длина файлов ограничена 37000 байт. Второй способ основан на пересылке части копируемого файла в буфер, расположенный в пользовательской части ОЗУ. Этот способ записи положен в основу копировщиков, разработанных автором. Они позволяют копировать файлы длиной 40000 байт. При этом информация о программе отображается в 3-12 строках экрана. Возможно и дальнейшее развитие второго метода - считывая файл как в пользовательское ОЗУ, так и в ОЗУ экрана и пересылая в буфер часть информации после завершения чтения, можно копировать файлы длиной и более чем 40000 байт.

Пересылать в буфер лучше всего начало копируемого файла, так как в этом случае удаётся использовать служебную строку экрана для индикации режимов.

Объём требуемого буфера определяют исходя из следующих данных:

- отображение экрана установлено с адреса 40000;
- вывод информации в служебную строку осуществляется с адреса 40000 и занимает 2000 байт;
- под каждую информационную строку требуется 1200 байт.

Таким образом, рассчитывая выводить параметры в N строк, объём буфера V в пользовательском ОЗУ определяют по формуле:

$$V = 2000 + 1200 * N.$$

Например, для вывода  $10_{10}$  строк информации в КОПИРОВЩИКЕ в ОЗУ пользователя зарезервировано место под буфер объёмом 16400 байт.

Если выбран метод пересылки части файла в буфер, запись программ на магнитофон можно осуществлять двумя различными способами:

- перед записью на магнитофон часть файла из буфера пересылается на своё место в экранном ОЗУ и запись производится только из экранного ОЗУ;
- осуществляется запись части файла, находящейся в буфере, а затем без перерыва на магнитофон выводится часть файла, оставшаяся в экранном ОЗУ.

Хорошим примером может служить соответствующий фрагмент программы КОПИРОВЩИК, реализующей второй способ.

006500	012713	MOV #20, (R3)	Выдача на магнитофон команды ПУСК
006502	000020		
006504	012705	MOV #11400, R5	Запись в регистр R5 адреса начала буфера.
006506	011400		
006510	013704	MOV 324, R4	Запись длины файла в регистр R4.
006512	000324		
006514	020427	CMP R4, #16400	Проверка - умещается ли в буфер весь файл?
006516	016400		
006520	101413	BLOS 6550	
006522	012704	MOV #16400, R4	
006524	016400		Если файл не умещается в буфере, производится запись длины буфера в регистр R4,
006526	104114	EMT 114	Подсчёт контрольной суммы содержимого буфера (стандартная программа подсчёта контрольной суммы в КОПИРОВЩИКЕ выполняется по командному прерыванию EMT 114).
006530	012705	MOV #56400, R5	Подсчёт контрольной суммы части файла, находящейся в экранном ОЗУ и суммирование с контрольной суммой части файла, находящейся в буфере. Далее следует переход на запись файла.
006532	056400		
006534	013704	MOV 324, R4	
006536	000324		
006540	162704	SUB #16400, R4	
006542	016400		
006544	104153	EMT 153	
006546	000401	BR 6552	
006550	104114	EMT 114	Подсчёт контрольной суммы файла для случая, когда он весь помещается в буфере.
006552	010037	MOV R0, 312	Запись контрольной суммы в ячейку 312.
006554	000312		
006556	012700	MOV #10000, R0	Выдача на магнитофон настроечной последовательности. В КОПИРОВЩИКЕ по команде EMT 116 происходит обращение к подпрограмме драйвера, выдающей настроечную последовательность.
006560	010000		
006562	104116	EMT 116	
006564	012701	MOV #322, R1	Запись на магнитофон заголовка программы. Используется подпрограмма драйвера магнитофона, предназначенная для передачи на магнитофон массива.
006566	000322		
006570	012702	MOV #24, R2	
006572	000024		
006574	004737	JSR PC, 116474	
006576	116474		
006600	012701	MOV #11400, R1	
006602	011400		
006604	013702	MOV 324, R2	
006606	000324		
006610	020227	CMP R2, #16400	
006612	016400		
006614	101415	BLOS 6650	
006616	012702	MOV #16400, R2	
006620	016400		
006622	004767	JSR PC, 116474	
006624	116474		

006626	012701	MOV #56400,R1	Запись на магнитофон части файла из экранного ОЗУ. В регистр R1 записывается адрес начала сохраняемой в ОЗУ экрана части файла, в регистр R2 - длина этой части файла. Используется фрагмент подпрограммы записи на магнитофон массива.
006630	056400		
006632	013702	MOV 324,R2	
006634	000324		
006636	162702	SUB #16400,R2	
006640	016400		
006642	004737	JSR PC,116504	
006644	116504		
006646	000402	BR 6654	Переход на запись контрольной суммы.
006650	004737	JSR PC,116474	Если файл умещается в буфере, то он записывается только из буфера с помощью подпрограммы записи массива на магнитофон.
006652	116474		
006654	012701	MOV #312,R1	Запись на магнитофон контрольной суммы массива файла. Используется фрагмент подпрограммы записи на магнитофон массива.
006656	000312		
006660	012702	MOV #2,R2	
006662	000002		
006664	004737	JSR PC,116504	
006666	116504		
006670	012700	MOV #400,R0	Запись на магнитофон выходной настроечной последовательности. Используется подпрограмма драйвера магнитофона для выдачи настроечной последовательности.
006672	000400		
006674	010004	MOV R0,R4	
006676	010005	MOV R0,R5	
006700	004737	JSR PC,116404	
006702	116404		
006704	012713	MOV #220,(R3)	Выдача на магнитофон команды СТОП.
006706	000220		
006710	000207	RTS PC	Возврат в вызвавшую программу.

В приведённом фрагменте программы КОПИРОВЩИК используются командные прерывания ЕМТ с номерами, большими 100. Это связано с тем, что данная программа написана в системе КОФОКМ.

Система КОФОКМ имеет дополнительный набор системных командных прерываний с 100 до 200 номера, которые пользователь может изменять в соответствии со своими потребностями.

Переход от записи файла из буфера на запись файла из экранного ОЗУ происходит в тот момент, когда на магнитофон выдаётся синхрэлемент, поэтому никаких искажений в файл не вносится.

### Как выбрать скорость?

Вопрос о скорости, с которой можно записывать файлы, волнует многих. В самом деле, изменив только этот параметр, можно на одной и той же кассете записать гораздо больше программ, сэкономив одновременно время, требующееся на их считывание и запись.

Для того, чтобы изменить скорость, с которой файлы записываются на магнитную ленту, необходимо соответствующим образом скорректировать длительности всех элементов, из которых формируется файл.

В таблице представлены длительности импульсов, позволяющие записывать файлы



со скоростью 600, 1200 и 2400 бод (бит/с).

Элемент файла	Импульс	Адрес элемента	Длительность			
			для скорости	600	1200	2400
Синхроэлемент	положительный	116412		57	27	13
Синхроэлемент	отрицательный	116424		56	26	12
Маркер	положительный	116440		323	151	64
Маркер	отрицательный	116452		316	144	57
Логический 0	положительный	116524		60	30	14
Логический 0	отрицательный	116536		57	27	13
Логическая 1	положительный	116552		147	63	31
Логическая 1	отрицательный	116564		147	63	31
Синхроэлемент	положительный	116576		60	30	14
Синхроэлемент	отрицательный	116610		56	26	12

Платой за повышенную скорость является меньшая помехозащищённость элементов. В результате запись на скорости 2400 бод удаётся вести только на высококачественных магнитофонах. На бытовом магнитофоне "Электроника 302" и кассетах МК-60 это удаётся крайне редко.

И наоборот, запись со скоростями, меньшими стандартной, повышает помехозащищённость, поэтому такую запись целесообразно применять всякий раз, когда используются низкокачественные магнитофоны и кассеты.

Всякий раз после того, как файл записан на магнитную ленту, возникает вопрос - удалась ли запись? Поскольку качество записи зависит от качества кассеты и магнитофона, а также от скорости, с которой осуществляется запись, то надёжный ответ можно получить лишь в том случае, если считать только что записанную программу.

Обычно этот вопрос решается достаточно просто - сбрасывается файл, находящийся в копировщике, и считывается только что записанный.

Очевидно, что этот способ страдает значительным недостатком - всякий раз, когда копия файла, считанного с магнитной ленты, оказалась неудачной, придётся вновь загружать оригинальный файл. Более эффективны копировщики, позволяющие проверить качества записи копии, не сбрасывая оригинал.

Один из вариантов подобного теста реализован в программе КОПИРОВЩИК.

Приём заголовка тестируемого файла может осуществляться в любую свободную область памяти размером 24 байта, например, с адреса 346.

004620	012701	MOV #312,R1	При тестировании будет использоваться только один байт с адресом 312.
004622	000312		
004624	013702	MOV 350,R2	Запись длины тестируемого файла в регистр R2.
004626	000350		
004630	005000	CLR R0	Очистка регистра R0, который будет использоваться для подсчёта контрольной суммы.
004632	010046	MOV R0,-(SP)	Приём первого байта массива. При этом используется подпрограмма драйвера магнитофона, предназначенная для приёма массива с настроенной последовательностью.
004634	010246	MOV R2,-(SP)	
004636	012702	MOV #1,R2	
004640	000001		
004642	004737	JSR PC,117260	
004644	117260		
004646	012602	MOV (SP)+,R2	

004650	005302	DEC R2	
004652	012600	MOV (SP)+,R2	Определение контрольной суммы.
004654	154137	BISB -(R1),234	Байт 234 используется как промежуточная переменная.
004656	000234		
004660	063700	ADD 234,R0	
004662	000234		
004664	005500	ADC R0	
004666	010046	MOV R0,-(SP)	Приём следующего байта.
004670	012700	MOV #10,R0	Используется подпрограмма драйвера магнитофона, предназначенная для измерения длительности принимаемого элемента.
004672	000010		
004674	004737	JSR PC,117376	
004676	117376		
004700	020437	CMP R4,314	
004702	000314		
004704	101002	BHI 4712	
004706	000241	CLC	
004710	000401	BR 4714	
004712	000261	SEC	
004714	106011	RORB (R1)	
004716	077012	SOB R0,4674	
004720	012600	MOV (SP)+,R0	Замена предыдущего значения контрольной суммы (ячейка 234) на значение принятого байта.
004722	151137	BISB (R1),234	
004724	000234		
004726	063700	ADD 234,R0	
004730	000234		
004732	005500	ADC R0	
004734	077224	SOB R2,4666	Подсчёт числа принятых байтов.
004736	010037	MOV R0,312	Запись контрольной суммы в ячейку 312.
004740	000312		

Данный метод позволяет осуществлять тестирование файлов любой длины, используя практически всего один байт. Если после окончания тест-программы вновь вернуться к её началу, то с её помощью можно проверить все файлы, записанные на кассете подряд

### Как упростить загрузку и запись незнакомой программы?

В настоящее время для микроЭВМ БК-0010 (БК-0010.01) создано несколько сот интересных программ, написанных в кодах, на Фокале, Бейсике. Для того, чтобы использовать незнакомую программу, необходимо знать два адреса, поскольку для запуска программы необходимо дать команду на загрузку, где фигурирует адрес загрузки, а затем запустить считанный файл с адреса запуска.

Программы, написанные в кодах, нередко имеют различные адреса запуска, а программы на Фокале или Бейсике - свои строки запуска. В результате, в каталоге программ приходится хранить не только название программы, но и сведения, необходимые для её считывания и запуска.

В то же время загрузку и запуск программы можно упростить, например, ограничившись тем, что в пусковом мониторе достаточно будет только выдать команду М (чтение с магнитофона) и указать имя программы. Для этого программу дополняют участком автозапуска, а программу на языке высокого уровня, так называемой программой-инициализатором.

Автозапуск программы основан на том, что для запоминания адресов возврата из подпрограмм драйвера монитора при вызове его из пускового монитора используются ячейки с адреса 766 и меньше, а при вызове драйвера магнитофона из отладчика БК адреса

возврата запоминаются в ячейках с адресами меньше 732.

Достаточно считать программу в копировщик и дополнить её участком, начинающимся, например, с адреса 730. В ячейках, имеющих адрес с 730 по 766, записывается адрес запуска программы, тем самым, для неё определяется режим автозапуска. Теперь при считывании программы адрес возврата в пусковой монитор или отладчик, хранившийся до того в ячейке стека 730, будет заменён на адрес запуска считываемой программы. В результате, как только считывание будет завершено, вместо того, чтобы вернуть управление вызвавшей программе, процессор передаст его только что считанной программе, то есть произойдёт её автозапуск.

Для установки режима автозапуска программ на Фокале или на Бейсике, этими действиями не обойтись. Следует также осуществить инициализацию Фокала или Бейсика и дать соответствующую команду исполнения программы на этом языке (G или RUN). Для этого программу при копировании дополняют подпрограммой - инициализатором.

Рассмотрим в качестве примера инициализатор, который добавляет к программам на Фокале программа КОПИРОВЩИК.

### Подпрограмма установки инициализатора Фокал-программ

005660	012701	MOV #40000,R1	Перемещение Фокал-программы в экранном ОЗУ. Регистр R1 служит указателем местонахождения команд первоначального положения программы, регистр R2 - соответствующим указателем при новом положении программы, а регистр R3 - счётчиком числа пересланных байт. До перемещения в ячейке 40000 находилось значение байта программы, которое при исполнении программы должно находиться в ячейке с адресом 1752. После перемещения в ячейке 40000 будет находиться байт, который при считывании программы на исполнение будет загружаться в ячейку с адресом 760.
005662	040000		
005664	063701	ADD 324,R1	
005666	000324		
005670	010102	MOV R1,R2	
005672	062702	ADD #772,R2	
005674	000772		
005676	013703	MOV 324,R3	
005700	000324		
005702	114142	MOVB -(R1),-(R2)	
005704	077302	SOB R3,5702	
005706	104142	EMT 142	Пересылка в буфер части программы, вместо которой в экранном ОЗУ должна записываться служебная информация.
005710	012700	MOV #13304,R0	Запись в программу с её реального адреса 1664 параметров начальной установки Фокала, которые хранятся в ПЗУ Фокала начиная с адреса 120106.
005712	013304		
005714	012701	MOV #120106,R2	
005716	120106		
005720	012702	MOV #33,R2	
005722	000033		
005724	012120	MOV (R1)+,(R0)+	Байт программы, имеющий адрес 1664 при исполнении программы, будет находиться в буфере по адресу 13304.
005726	077202	SOB R2,5724	
005730	012700	MOV #11420,R0	Вставка инициализатора Фокал-программы в копируемую программу с реального адреса 1000.
005732	011420		
005734	012701	MOV #7434,R1	
005736	007434		
005740	012702	MOV #32,R2	
005742	000032		
005744	012120	MOV (R1)+,(R0)+	
005746	077202	SOB R2,5744	

005750	013737	MOV 322,13356	Уточнение параметров начальной установки Фокала с учётом длины данной программы.
005752	000322		
005754	013356		
005756	063737	ADD 324,13356	
005760	000324		
005762	013356		
005764	013737	MOV 13356,13360	
005766	013356		
005770	013360		
005772	062737	ADD #120,13360	Определение в блоке параметров адреса загрузки - 760. Длина программы увеличивается на 1000 за счёт участка автозапуска и вставки фрагмента, предназначенного для инициализации.
005774	000120		
005776	013360		
006000	012737	MOV #760,322	
006002	000760		
006004	000322		
006006	062737	ADD #1000,324	
006010	001000		
006012	000324		
006014	000207	RTS PC	Возврат в вызвавшую программу.

### Инициализатор программ на Фокале

007434	001000		Адреса запуска инициализатора, в программе записываемые по адресам 760-766.
007436	001000		
007440	001000		
007442	001000		
007444	000000		Служебные параметры в программе записываемые по адресам 770-776.
007446	000772		
007450	010000		
007432	000000		
007454	005000	CLR R0	В программе фрагмент, предназначенный для инициализации, начинается с адреса 1000. После автозапуска инициализатор записывает по адресам 0-36 необходимые для работы Фокала векторы прерывания.
007456	012701	MOV #120020,R2	
007460	120020		
007462	012702	MOV #14,R2	
007464	000014		
007466	012120	MOV (R1)+, (R0)+	
007470	077202	SOB R2,1012	
007472	012737	MOV #121136,34	
007474	121136		
007476	000034		
007500	012737	MOV #340,36	
007502	000340		
007504	000036		
007506	012737	MOV #-1,262	Определение выдачи кода 15 при нажатии на клавишу ВВОД.
007510	177777		
007512	000262		Выдача интерпретатору Фокала команды на запуск программы.
007514	104672	TRAP 272	

Программу на Фокале, снабжённую подобным инициализатором, можно загружать непосредственно из пускового монитора аналогично тому, как если бы она была написана в кодах.

Для программ на Бейсике можно применять аналогичный инициализатор, но он должен учитывать параметры конкретной версии Бейсика. Это сравнительно нетрудно, если говорить о Бейсике-85 (87), и весьма сложно для версии Бейсика-БК, вшитой в ПЗУ

БК-0010.01. Решение получается довольно громоздким, поскольку кроме инициализатора необходим и микродрайвер, позволяющий принимать файлы с блочной структурой. Ограниченный объём нашего методического пособия не позволяет нам останавливаться на рассмотрении этой задачи.

## ПРОГРАММЫ ЗАЩИЩАЮТСЯ

Разработка программы, будь то игровая, деловая или системная программа, - результат порой значительных интеллектуальных усилий программиста, требующих от него немалых затрат времени. Труд программиста должен быть оценён соответствующим образом.

Несмотря на эти, прямо скажем, очевидные истины, организации, копирующие программное обеспечение без какого бы то ни было разрешения автора, не торопятся поделиться с ним своими доходами. Это касается, как государственных, так и большинства кооперативных организаций и частных лиц. Несанкционированное тиражирование программ не вызывает восторга у авторов программ, поэтому они прилагают усилия по защите своих программ от копирования.

Знание некоторых способов защиты программ, а также приёмов, позволяющих преодолеть подобную защиту, поможет начинающим пользователям БК. В то же время мы считаем, что в основу защиты авторских прав программиста должно быть положено сознательное их уважение всеми пользователями БК и организациями, распространяющими программное обеспечение, а также Закон об охране прав на интеллектуальную собственность, необходимость принятия которого очевидна.

Можно считать программу защищённой, если её нельзя загрузить или запустить после копирования. Мы уже рассматривали формат записи на магнитофон, принятый в компьютере БК-0010. Этот формат используется и при копировании файлов. Читатель, по-видимому, догадывается, что для защиты программы от несанкционированного копирования, достаточно так изменить формат записи, чтобы при копировании "модифицированного" файла обычными копировщиками происходило искажение нестандартного формата или его параметров.

Простейший способ защиты - разбиение файла на две (или большее число) частей. Каждая из этих частей может иметь обычный формат. Первая часть файла является загрузчиком второй части. При этом адрес загрузки, длина и адрес запуска второй части хранятся в первой части. После считывания и запуска первая часть может отображать на экране заставку, сведения о программе и осуществлять загрузку второй части программы. Вторая часть файла должна иметь неочевидные параметры записи, например, трудно набираемое имя файла, нечётные адрес загрузки, длину файла, изменённые параметры настрочной последовательности.

Если, например, в имя файла между буквами вставить код 0, то его нельзя будет набрать с клавиатуры.

В результате подобный файл нельзя скопировать с помощью отладчика. Для этого необходим специальный копировщик, способных читать первый найденный файл без указания каких-либо дополнительных параметров.

Нечётный адрес загрузки затрудняет определение адреса, начиная с которого требуется загружать файл. Загрузка такого файла с чётного адреса полностью изменяет коды всех команд, а в результате программа становится невыполняемой.

В файлах адрес загрузки может быть 730 и меньше. В результате, если воспользоваться для копирования обычным копировщиком, загружающим файл с его

адреса загрузки, будет происходить автозапуск считанного файла. Для того, чтобы затруднить копирование, такие файлы снабжают ещё одной "деталью". Это может быть, например, условие, что при нажатии на клавишу СТОП, происходит повторный запуск программы (в этом случае по адресу 4 записан адрес запуска программы) или не происходит никаких действий (в ячейке 4 записан адрес команды RTI - возврата из прерывания, имеющей код 2) или обеспечивается стирание всей программы (для этого в ячейке 4 достаточно указать адрес 100 000).

## Как обойти защиту?

### Совет 1.

*Обойти защиту во всех перечисленных случаях удаётся с помощью экранных копировщиков разработанных, например, Радченко или Ярошенко.*

При загрузке файла параметры настроечных последовательностей не фиксируются в памяти машины, за исключением порогового значения, измеренного по входной настроечной последовательности. Впрочем, при записи файла не используется даже пороговое значение. Если изменить в файле параметры настроечных последовательностей и ввести в первую часть файла (загрузчик) анализатор настроечных последовательностей, то получим средство контроля за фактом копирования второй, основной части файла.

Изменять проще всего выходную настроечную последовательность.

### Совет 2.

*Такой файл можно копировать лишь с помощью специального копировщика, способного осуществлять запись с укороченной выгодной последовательностью или вообще без неё. Впрочем, возможен иной способ копирования, предусматривающий коррекцию первой части файла, поскольку она в большинстве случаев не снабжена защитой, и к тому же обладает малой длиной. В результате не так уж сложно отыскать участок, где осуществляется анализ настроечной последовательности, и отключить его, после чего копирование программы не вызовет трудностей. Впрочем, анализатор выходной настроечной последовательности может находиться и во второй, защищённой части файла.*

Значительно надёжнее выглядит способ, основанный на изменении формата записи отдельных частей файла. Рассмотрим один из вариантов.

Читатель помнит, что драйвер магнитофона, записав на ленту массив, выдаёт на магнитофон контрольную сумму. При считывании файла драйвер магнитофона направляет считанную контрольную сумму в ячейку 312. После того как считывание завершается, драйвер подсчитывает контрольную сумму массива и сравнивает её с той, что уже записана в ячейке 312. Результат сравнения записывается в байт ответа блока параметров операции обмена с магнитофоном.

Если файл оборудован автозапуском, то информация о результате сравнения не используется. Этим и можно воспользоваться, создавая эффективную защиту. Вместо контрольной суммы на магнитофон сначала записывают код защиты данной программы, а лишь затем - контрольную сумму. Программу при этом дополняют блоком команд, который периодически сравнивает содержимое ячейки 312 со значением кода защиты.

При считывании подобной защищённой программы любой копировщик будет выдавать сообщение типа "Ошибка контрольной суммы", поскольку код защиты не равен контрольной сумме файла. При записи этого файла на магнитофон копировщик запишет

после массива файла контрольную сумму, а не код защиты. В итоге содержимое ячейки 312 будет являться своего рода индикатором - пытались ли копировать файл. После запуска скопированной программы блок защиты, обнаружив в ячейке 312 значение, отличное от кода защиты, должен стереть всю программу.

Блок защиты встраивают в программу таким образом, чтобы его было трудно обнаружить с помощью отладчика или дизассемблера.

### Совет 3.

*Перечислим несколько, несложных признаков, позволяющих обнаружить, оборудована ли программа блоком защиты подобного типа:*

- программа автоматически запускается после считывания;
- при считывании программы в отладчик или копировщик следует сообщение "Ошибка контрольной суммы";
- контрольная сумма файла, считанного в отладчик, не совпадает с значением, записанным в ячейку 312 при чтении.

*Для снятия блока защиты приходится дизассемблировать всю программу. Чтобы найти фрагмент программы, занимающийся защитой, иной раз требуется разобраться в принципе действия едва ли не всех блоков, из которых состоит программа.*

*В то же время есть более эффективный способ обойти защиту.*

*Достаточно отыскать свободную область в ОЗУ (можно даже в ОЗУ экрана), и дополнить программу небольшим фрагментом, который записывал бы код защиты в ячейку 312.*

*Внимательный читатель возразит, что для этого необходимо знать код защиты. Узнать его несложно, считав программу в отладчик БК с адреса 1000. В ячейке 312 при этом окажется искомое значение. После того как программа будет дополнена блоком, снимающим защиту, файл вновь записывают на ленту, а уж затем перезаписывают с помощью копировщика, предварительно изменив в заголовке файла адрес загрузки с 1000 на первоначальный. Полученный в итоге файл можно копировать без проблем.*

Разумеется, можно предложить ещё немало разнообразных методов защиты. Если говорить о каких-то универсальных способах, позволяющих справиться с многими способами защиты, то есть способ, который можно считать скорее аппаратным, нежели программным.

### Совет 4.

*Защищённую программу загружают обычным способом и запускают (или она запускается автоматически). Далее с помощью переключателя, установленного в отсеке пользователя, выключают процессор, а микросхему ПЗУ с Фокалом вынимают из её гнезда. Процессор включают вновь, и микроЭВМ автоматически выходит в пусковой монитор. Из пускового монитора переходят в отладчик БК (нажать клавишу T, затем ВВОД; после появления на экране символа "+" набирается ТС (буквы кириллицы, а не латиницы)) и записывают программу на магнитофон из отладчика обычным образом. Во многих случаях этот приём даёт желаемый результат.*

Формат файла также можно изменять различными способами, однако в большинстве случаев это влечёт за собой увеличение длины программы, а в итоге снижается помехозащищённость элементов файла и снижаются потребительские качества такой

программы.

Кроме того, опытный программист, хорошо знающий микроЭВМ и программирование в кодах, всегда способен снять защиту или обойти её. Впрочем, вряд ли это указывает на то, что защита нецелесообразна. Опытный программист, зная цену своего и чужого труда, вряд ли воспользуется своими знаниями для вульгарной спекуляции чужими разработками.

Уровень защиты можно считать достаточным, если опытному программисту приходится составлять специальную программу, причём достаточно сложную, чтобы обойти защиту. В этом случае программа считается надёжно защищённой, и она вряд ли будет распространяться без санкции автора.

Из всего спектра методов, позволяющих защищать программы, предпочтение следует отдавать тем, которые не увеличивают число операций, необходимых для запуска программы. Желательно также, чтобы при этом не возрастало время загрузки программы и не снижалась помехоустойчивость, а также не вносились ошибки в исполнение программы. Важно также, чтобы защиту нельзя было снять выключением процессора, а блоки защиты нельзя было прочесть прямым дизассемблированием.

Кроме того, разрабатывая программу, следует заранее предусматривать установку защиты, например, обеспечить, чтобы при нажатии на клавишу СТОП не происходил выход в отладчик или пусковой монитор, а при остановке процессора с помощью тумблера в отсеке пользователя часть программы терялась.

### Совет 5.

*Если все способы снять защиту с программу перепробованы, а результат не достигнут, то программу можно попытаться регенерировать.*

## РЕГЕНЕРИРОВАНИЕ ПРОГРАММ

Переписать файл с одной магнитной ленты на другую можно двумя способами: перезаписывая программу с ленты на ленту без специальной обработки сигналов или подвергая их обработке с помощью регенератора.

Каждый из этих способов обладает своими достоинствами и недостатками.

Для перезаписи с магнитофона на магнитофон необходим только магнитофонный шнур, входящий в комплект магнитофона. Данный способ не изменяет основных соотношений, характеризующих широтно-импульсную модуляцию, но в этом случае форма импульсов неминуемо будет искажаться, шумы станут накапливаться, возникнут колебания амплитуд отдельных импульсов и уровня сигнала в целом.

Для регенерирования файлов необходимо дополнительное оборудование. Речь идёт об устройстве, которое восстанавливало бы форму и другие параметры сигнала. Подобные устройства называют регенераторами. Можно предложить немало регенераторов для сигнала, используемого в БК для записи программ, различающихся сложностью. Лучшее качество обеспечивают регенераторы, позволяющие восстанавливать не только прямоугольную форму импульсов, но и их длительность, а также синхронизировать их по фазе и обеспечивать стабильность амплитуды. Однако такие генераторы будут довольно сложными, а значит и дорогими.

Значительно проще регенераторы, восстанавливающие только прямоугольную форму импульсов. Такие регенераторы можно реализовать не только аппаратно, но и программно-аппаратно. В качестве примера рассмотрим регенератор программного типа, который обеспечивает считывание информации с одного магнитофона через регистр



управления внешними устройствами и записывает эту информацию на другой магнитофон через порт ввода-вывода.

### Программа - регенератор

001000	012702	MOV #256,R2	Регистр R2 - указатель рекомендуемой в инструкции по эксплуатации микроЭВМ копии порта ввода-вывода.
001002	000256		
001004	012703	MOV #177716,R3	Регистр R3 - указатель регистра управления системными внешними устройствами (см. Приложение, рис.2).
001006	177716		
001010	012704	MOV #177714,R4	Регистр R4 - указатель порта ввода-вывода.
001012	177714		
001014	012705	MOV #40,R5	Установка маски в регистре R5.
001016	000040		
001020	005012	CLR (R2)	Очистка регистров R2-R4 и выдача на магнитофон команды ПУСК. (Известно, что команда ПУСК выдаётся записью логического "0" в разряд 7 регистра управления системными внешними устройствами, а здесь обнулены все разряды, в том числе и седьмой).
001022	005013	CLR (R3)	
001024	005014	CLR (R4)	
001026	030513	BIT R5, (R3)	Проверка - какой импульс поступил с магнитофона?
001030	001005	BNE 1044	
001032	030512	BIT R5, (R2)	Проверка - не ноль ли поступил с магнитофона перед этим опросом?
001034	001774	BEQ 1026	
001036	040512	BIC R5, (R2)	Если ноль не поступал, выдаётся ноль в порт ввода-вывода и организуется задержка.
001040	011214	MOV (R2), (R4)	
001042	000404	BR 1054	
001044	030512	BIT R5, (R2)	Проверка - не единица ли поступила с магнитофона перед этим опросом.
001046	001367	BNE 1026	
001050	050512	BIS R5, (R2)	Если единица не поступала, то на порт ввода-вывода выдаётся единица и организуется задержка
001052	011214	MOV (R2), (R4)	
001054	012701	MOV #10,R1	Организуется задержка в опросе.
001056	000010		
001060	077101	SOB R1,1060	
001062	000461	BR 1026	Вновь начинаем опрос.

Оптимальная длительность задержки составляет 10-14 единиц счёта по команде SOB (записывается по адресу 1056).

Сигнал на запись передаётся через порт ввода-вывода. Для этого сигнал на магнитофон подаются через делитель, состоящий из двух резисторов R1 и R2. Резистор R1 (сопротивлением 9 кОм) подключают к контакту В7, а резистор R2 (сопротивлением 620 Ом) - к контактам А11 или В11, то есть к общему выводу порта ввода-вывода (см. Приложение, рис. 6).

Подобный регенератор не улучшает соотношения между амплитудами и длительностями синхроимпульсов, логических нулей и единиц, а восстанавливает лишь прямоугольную форму импульсов. Более того, по сравнению с перезаписью "магнитофон-магнитофон" данный регенератор вносит дополнительную широтную модуляцию. Применять его стоит лишь в тех случаях, когда имеются колебания амплитуды импульсов, например, запись производилась на магнитофоне с автоматическим регулятором уровня записи, вносящим паразитную амплитудную модуляцию.

В любом случае, для качественного копирования рекомендуется вести перезапись через микроЭВМ.

## **ЗАКЛЮЧЕНИЕ**

Мы рассмотрели практически все аспекты, связанные с записью программ на магнитную ленту. Что касается защиты программ, то, разумеется, в небольшом по объёму пособии невозможно было бы описать все возможные способы защиты программ от несанкционированного доступа и копирования. Средства защиты постоянно совершенствуются, как совершенствуются и становятся все более изощрёнными методы копирования. Очевидно, что защищать авторское право разработчиков программ должны сознательность пользователей и Закон, а технические и программные способы защиты должны лишь помогать такой защите.

Автор надеется, что подробное описание драйвера магнитофона и принципов "экранного" копирования файлов помогут желающим создать более совершенные, в том числе и полностью автоматические копировщики файлов и кассет.

## **РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА ДЛЯ ВЛАДЕЛЬЦЕВ БК**

- БИС. Информационный бюллетень для владельцев БК. Сыктывкар, СБИС.
- "Информатика и образование" (раздел "Клуб БК").
- Лин В. PDP-11 и VAX-11. Архитектура ЭВМ и программирование на языке ассемблера, М. Радио и связь, 1989.
- Математическое обеспечение микроЭВМ "Электроника БК-0010". Руководство системного программиста, т.1, кн.2, 1985.
- Математическое обеспечение микроЭВМ "Электроника БК-0010". Язык Фокал. руководство пользователя, т.1, кн.1, 1985.
- "Наука и жизнь" (раздел "Человек и компьютер").
- "Новое в жизни, науке, технике". Серия "Вычислительная техника и её применение" (раздел "БК-за рога").
- Осетинский Л.Г., Осетинский М.Г., Писаревский А.Н. Фокал для микро- и мини компьютеров. Л.: Машиностроение, 1988.
- Осетинский Л.Г., Осетинский М.Г., Писаревский А.Н. Фокал - диалоговый язык для мини-ЭВМ. Л.: Машиностроение, 1985 - 195 с, ил.
- Текст программы КОПИРОВЩИК, 1989.

## Оглавление

От редакции.....	2
ВВЕДЕНИЕ .....	3
Советы начинающим .....	3
ОБМЕН С МАГНИТОФОНОМ .....	4
Структура микроЭВМ "Электроника БК" .....	5
Блок параметров для работы с магнитофоном .....	6
Формат записи файла на магнитофон.....	8
ДРАЙВЕР МАГНИТОФОНА.....	9
Управляющая программа драйвера магнитофона .....	10
Обработка прерывания после нажатия клавиши "СТОП".....	12
Выход драйвера магнитофона .....	12
Подпрограмма записи на магнитофон .....	12
Подпрограмма выдачи настроечной последовательности.....	13
Подпрограмма передачи массива на магнитофон .....	14
Подпрограмма подсчёта контрольной суммы .....	15
Подпрограмма чтения с магнитофона .....	15
Подпрограмма поиска настроечной последовательности .....	16
Подпрограмма чтения заглавия файла.....	17
Подпрограмма чтения массива файла.....	17
Подпрограмма приёма массива с настроечной последовательностью.....	18
Подпрограмма приёма массива без настроечной последовательности.....	19
Подпрограмма измерения длительности принимаемого элемента.....	19
ОСНОВЫ "ЭКРАННОГО" КОПИРОВАНИЯ ФАЙЛОВ.....	20
Как выбрать скорость? .....	24
Как упростить загрузку и запись незнакомой программы?.....	26
Подпрограмма установки инициализатора Фокал-программ .....	27
Инициализатор программ на Фокале.....	28
ПРОГРАММЫ ЗАЩИЩАЮТСЯ.....	29
Как обойти защиту? .....	30
Совет 1. ....	30
Совет 2. ....	30
Совет 3. ....	31
Совет 4. ....	31
Совет 5. ....	32
РЕГЕНЕРИРОВАНИЕ ПРОГРАММ .....	32

Программа – регенератор.....	33
ЗАКЛЮЧЕНИЕ.....	34
РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА ДЛЯ ВЛАДЕЛЬЦЕВ БК.....	34
Приложение.....	37

## Приложение

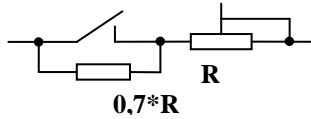


Рис. 1 Схема доработки магнитофона.

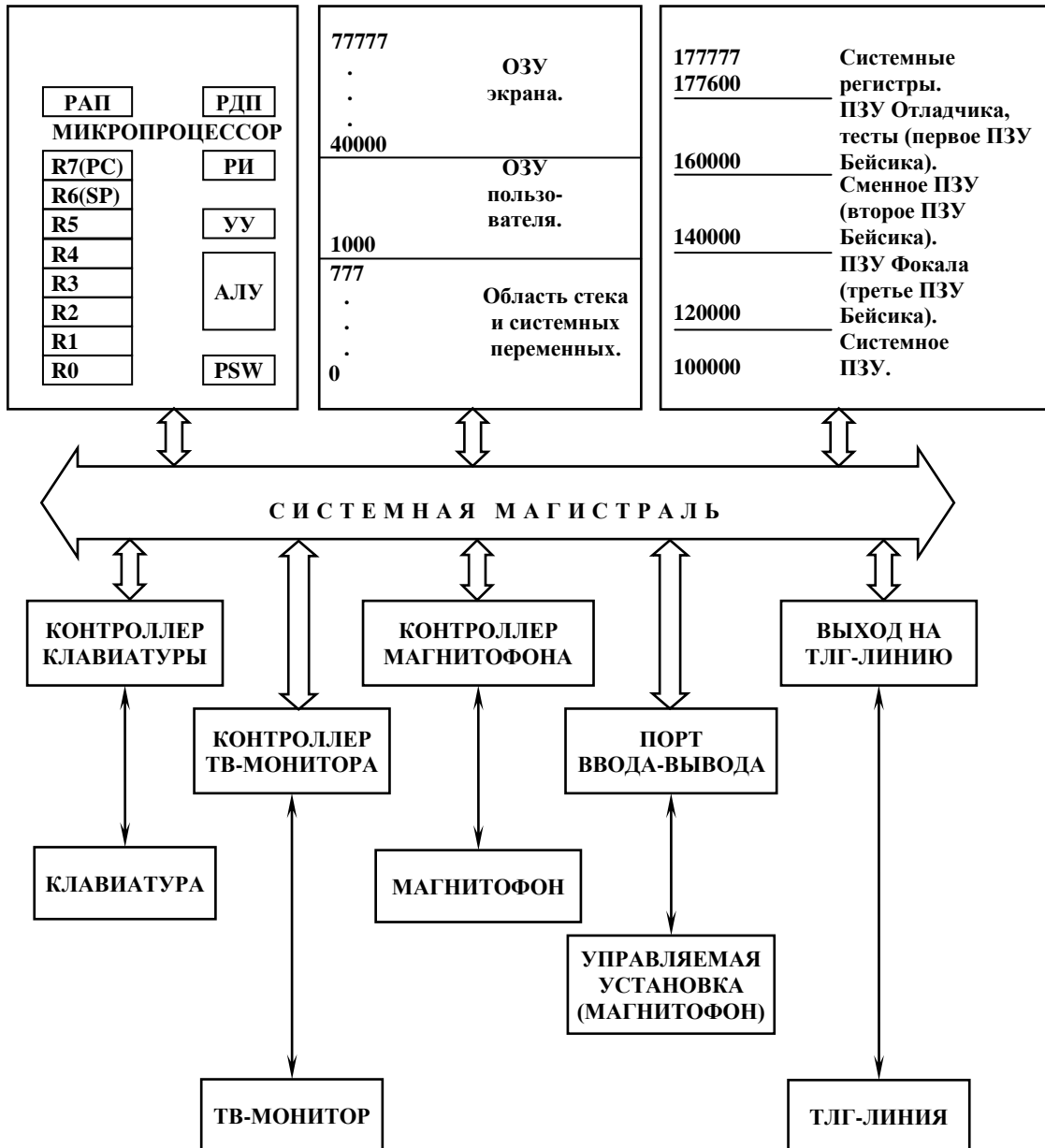


Рис. 2 Структура микроЭВМ БК-0010 (БК-0010.01).

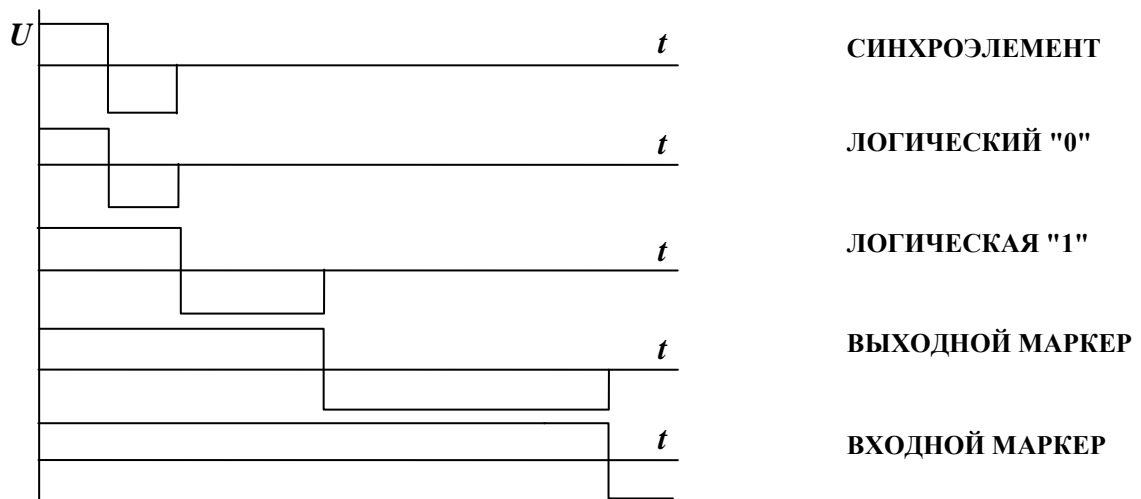
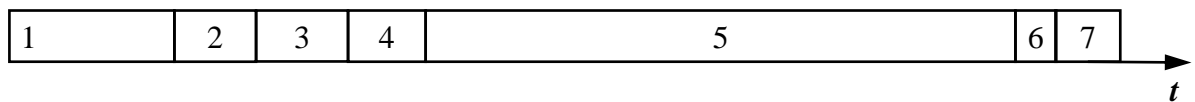


Рис. 3 Элементы записи информации на магнитофон.



- 1 – ВХОДНАЯ НАСТРОЕЧНАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ
- 2 – НАСТРОЕЧНАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ ЗАГЛАВИЯ
- 3 – ЗАГЛАВИЕ ФАЙЛА
- 4 – НАСТРОЕЧНАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ МАССИВА
- 5 – МАССИВ
- 6 – КОНТРОЛЬНАЯ СУММА МАССИВА
- 7 – ВЫХОДНАЯ НАСТРОЕЧНАЯ ПОСЛЕДОВАТЕЛЬНОСТЬ

Рис. 4 Формат записи файла на магнитофон.

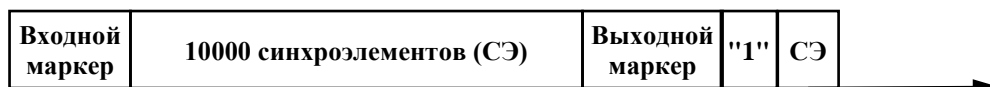
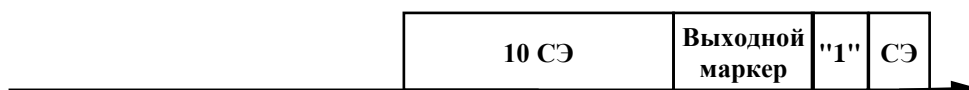
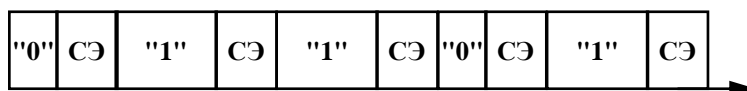
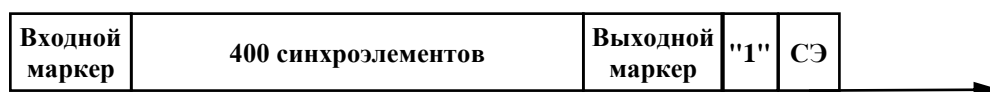
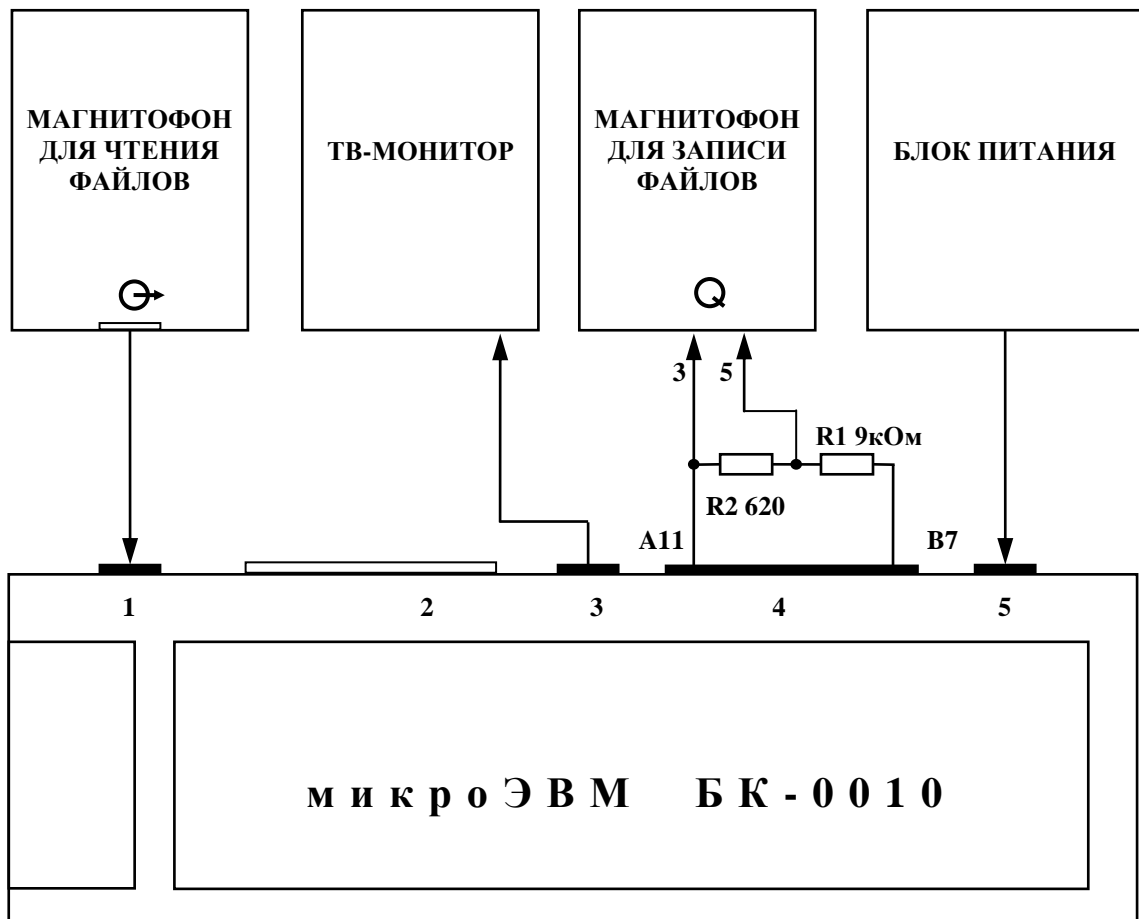
**Входная настроечная последовательность****Настроечная последовательность заглавия****Заглавие, массив, контрольная сумма****Выходная настроечная последовательность**

Рис. 5 Форматы последовательностей при записи на магнитофон



- 1 - разъём для подключения магнитофона, ТЛГ-линии
- 2 - разъём общей шины микроЭВМ
- 3 - разъём для подключения ТВ-монитора (видеовыход)
- 4 - порт ввода-вывода
- 5 - разъём для подключения блока питания

Рис. 6. Схема подключения магнитофонов при регенерации файлов

