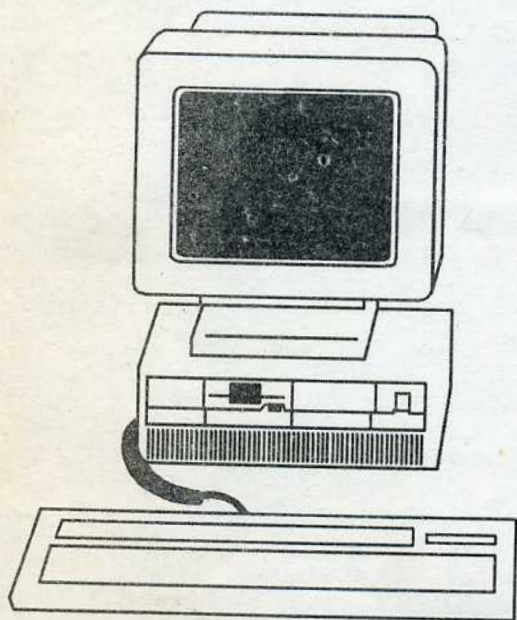


БОЛЬШИЕ
ВОЗМОЖНОСТИ
ВАШЕГО
«СПЕКТРУМА»



ZX

Spectrum

1. Введение

Мы начнем с того, что посвятим несколько слов истории слов истории создания "Спектрума". Для тех наших читателей, кто недавно начал работу с этой замечательной машиной, будет, по-видимому, небезынтересно узнать о взлетах и падениях, пережитых ее создателем сэром Клайвом Синклером и, уж во всяком случае, здесь Вы найдете ответ на вопрос, почему ZX-Spectrum стал самым популярным домашним компьютером и не сдает завоеванных позиций, хотя столько фирм выпускают более мощные машины.

Клайв Марлз Синклер родился 30 июня 1940 года. Еще школьником он начал печатать неплохие статьи в журнале "Практическое радио". Окончив школу, он не стал поступать в университет, а был принят в этот журнал в качестве помощника заместителя редактора, затем работал в издательстве, а в 1961 году зарегистрировал свою первую компанию "Синклер Радионикс". Первое изделие - микроусилитель, - он выпустил в 1963 году и во всех дальнейших разработках ставил перед собой две сверхзадачи - минимальные габариты и минимальная цена. Успех Синклера всегда основывался на том, что он со своим товаром был первым, причем часто ориентировался на рынок, который еще не существовал.

В 1979 году фирма "Коммодор" выпустила свой первый бытовой компьютер "PET" ценой 700 фунтов. Газета "Файнэншл Таймс" тогда предсказывала, что цены на персональные компьютеры опустятся ниже 100 ф.ст., не ранее, чем через 5 лет, а Синклер уже через полгода выпустил ZX-80 ценой 79 фунтов.

Резкому снижению цены содействовала идея использования телевизора в качестве дисплея, а бытового магнитофона в качестве внешней памяти.

ZX-80 сломил расхожее мнение об ЭВМ, как о чем-то доступном лишь для избранных. В первые 8 месяцев было продано 20 тыс. компьютеров, и в марте 1981 года была выпущена новая модель ZX-81 ценой 69 фунтов, а еще через несколько месяцев и принтер к нему. В эти дни американская фирма "Таймекс" купила право на распространение ZX-81 в Японии. Решительным рывком вперед стал договор с британской книготорговой сетью о реализации компьютеров по их торговым каналам. За один год товарооборот фирмы вырос с 4,6 млн.ф.ст. до 30 млн.ф.ст., а Синклер уже готовил новую модель "Спектрум" (март 1982г.). Были разработаны две версии - 16К и 48К. Эта машина сильно отличалась от своих предшественников, и ее популярность превзошла все ожидания. "Спектрумы" продавались по 15 тыс. штук в неделю.

Задумывался этот компьютер как учебный для изучения программирования, но фирмы, выпускающие программное обеспечение, быстро поняли, что программирование на уровне команд процессора позволяет получить неплохую динамичную графику, и для этого компьютера стали выпускаться увлекательные видеоигры. Получилась своего рода положительная обратная связь. Чем больше "Спектрумов" покупалось населением, тем активнее выпускались для него программы, а чем больше на рынке высококачественных программ для компьютера, тем активнее он покупается. Такой же процесс охватил и фирмы "третьего рынка", выпускающие периферийные устройства и аксессуары для компьютеров. К 1984 году, когда фирмы "Атари", "Коммодор" и "Амстрад" выпустили компьютеры, превосходящие "Спектрум 48", рынок уже был смещен в пользу Синклера, что продолжает чувствоваться и по сей день, а сам "Спектрум" уже выпускался более чем в 30 странах мира.

В 1984 году Синклер выпустил модель "Спектрум+", отличающуюся усовершенствованной клавиатурой, а к концу 1985 года "Спектрум+128" ("Дерби"), имеющий 128К оперативной памяти и 32К ПЗУ. Кроме этого новая модель имела звуковой процессор.

В 1986 году компания SINGLAIR RESEARCH LIMITED была вынуждена под давлением финансовых и организационных трудностей продать все права на производство "Спектрум" - совместимых моделей французской фирме "Амстрад". Проблемы были связаны с неоправдавшей себя 32-разрядной моделью "SINGLAIR-QL". Она задумывалась как дешевая альтернатива американским IBM PC, но в ее концепцию был заложен ряд просчетов (например использование в качестве внешней памяти микродрайвов вместо дисководов). Получилось так, что бытовым компьютером эта машина не стала по цене, а профессиональным - по своему аппаратному обеспечению. К тому же фирмы, выпускающие программы, не поддерживали эту в общем-то замечательную модель, опередившую идеи "АМИГИ-500" и "АТАРИ-520 СТ" как минимум на три года.

Положение компании усугублялось и недостаточной практической хваткой К.Синклера. Замечательный инженер, он так и не сумел стать бизнесменом. В самые напряженные дни весь штат его фирмы не превышал 12 человек (в том числе и представительство в Бостоне, США), а доход от продажи одного компьютера не превышал 1 ф.ст.

Продав все права на производство и реализацию своих изделий, Синклер оставил за собой исследовательскую лабораторию в Кембридже.

Последующие модели "ZX-SPECTRUM+2"(1986) - со встроенным магнитофоном и "ZX-SPECTRUM+3"(1987) - со встроенным

дисководом, выпускались уже фирмой "Амстрад". Поэтому они так похожи по внешнему виду на компьютер "Амстрад-6128". Основным их преимуществом является полноценная клавиатура, в то время как встроенные магнитофон и дисковод воспринимаются скорее как "нагрузка", непропорционально увеличивающие цену. Особенно, если принять во внимание нестандартный диаметр дискет 3,0 дюйма, малую их емкость (180К) и практическую сложность переноса имеющихся кассетных версий программ на диск, граничащую с нецелесообразностью.

С 1986 года фирма "Таймекс" на своих заводах в Португалии начала выпускать для Европы компьютер "ТАЙМЕКС-2048", практически полностью совместимый со "Спектрумом", но имеющий ряд преимуществ: улучшенную клавиатуру, встроенный порт манипулятора "джойстик", световиндикатор и выключатель питания, две экранные области памяти, режим расширенной цветной графики. Для продажи в США эта фирма выпустила компьютер "ТАЙМЕКС-2068", имеющий сменные картриджи. При подключении одного из них компьютер становится Спектрум-совместимым, а при подключении другого - CP/M-совместимым. (Американцы так же не представляют 8-разрядный компьютер без системы CP/M, как и 16-разрядный без MS DOS).

В заключении этого вступления упомянем только о перспективной разработанной модели, планировавшийся к выпуску в 1987 году. В основу "Суперспектрума" ("ЛОККИ") был положен процессор Z-80H, который может работать с частотой 7 МГц. При такой скорости удастся организовать и обслужить два банка памяти по 64К и экран емкостью более 51К. Он имел разрешающую способность 192*256 с возможностью одновременного воспроизведения 64-х цветов (для каждой точки). Эта машина была программно совместима со "Спектрумом", стоила менее 200 фунтов и была бы серьезным конкурентом для "АМИГИ". Но фирма "Амстрад", пользуясь своими правами, опасаясь конкуренции для своих машин, отказала в разрешении на его производство.

2. Подключение компьютера

Те, кто работает с фирменным компьютером ZX- SPECTRUM 48 или ZX- SPECTRUM+, могут увидеть на задней панели следующие разъемы:

EAR - для подключения магнитофона при загрузке

MIC - для подключения магнитофона при выгрузке программ

TV - для подключения телевизора через антенный вход

9VDC - для подключения внешнего блока питания

Кроме того, там же расположен 54-контактный краевой разъем который служит для подключения периферийных устройств. Если у Вас нет никакой периферии, лучше сразу заклеить этот разъем какой-либо липкой лентой, так как случайное попадание металлического предмета на контактные дорожки во время работы компьютера почти наверняка приведет к выходу его из строя. (Это может быть металлический колпачок от авторучки, магнитофонный штекер и т.п. К сожалению, мы знаем множество таких случаев, особенно когда с компьютером работают дети.)

Помните, что НИКАКУЮ периферию, в том числе и джойстик, НИКОГДА нельзя подключать к работающему компьютеру. Все шины небуферированы. Об этом должны помнить те, кто работает со "Спектрумом".

В компьютерах "ТАИМЕКС-2048", кроме этих разъемов, имеется также гнездо "VIDEO" для подключения к видеомонитору или к телевизору, имеющему низкочастотный вход "VIDEO". Впрочем, организовать такое подключение несложно и для обычного "Спектрума" (см. ниже).

В компьютерах "ZX-SPECTRUM+128" и "ZX-SPECTRUM+2" кроме антенного выхода имеется также и выход RGB (Red - красный, Green - зеленый, Blue - синий), который служит для подключения к RGB-монитору или на выход цветowych усилителей телевизора. Такой же способ подключения применяется в абсолютном большинстве самодельных компьютеров. Для подключения же фирменных "Спектрума" и "Спектрума+" по RGB-входу нужен специальный интерфейс (довольно сложное дорогое устройство). Для интересующихся ИНФОРКОМ выпускает обзор фирменных периферийных устройств (около 200 наименований с указанием назначения, фирмы-изготовителя, цены, технических деталей).

Еще одной особенностью компьютера "Спектрума+128" является то, что он передает звуковое сопровождение по антенному тракту на телевизор, и потому не имеет своего динамика. Правда звуковой сигнал модулирован в европейском стандарте и на отечественном телевизоре без переделки не воспроизводится.

2.1 Подключение к магнитофону

Соедините гнездо EAR компьютера с выходом магнитофона, предназначенным для подключения наушников. Если у Вас нет нужного выхода, то можете, в крайнем случае, вскрыть магнитофон и снять сигнал непосредственно с динамика.

Гнездо MIC компьютера соединяется с микрофонным входом магнитофона. По этой линии программы записываются на ленту. очень важно при работе с фирменным компьютером помнить, что при выгрузке программы на ленту загрузочный шнур EAR

должен быть обязательно отключен, в противном случае произойдет закольцовывание сигнала и запись будет испорчена. На компьютере "ТАЙМЕКС" и на большинстве самодельных это ограничение не распространяется. При загрузке не могут быть включены оба шнура.

2.2 Подключение к телевизору

Фирменные компьютеры работают в дециметровом диапазоне (36-ой канал), поэтому подключать их нужно ко входу ДМВ телевизора. При этом надо помнить, что цветные сигналы идут в системе PAL, поэтому на отечественном телевизоре, работающем в системе СЕКАМ, Вы цвета не увидите, а вместо этого будут воспроизводиться различные оттенки серого цвета. Одним из вариантов является встраивание в телевизор декодера ПАЛ/СЕКАМ, что пригодится Вам и при эксплуатации видеомагнитофона.

Если Вы сами не обладаете необходимыми знаниями и опытом по электронике, то встраивание декодера или подключение по RGB необходимо поручить специалисту.

Если в Вашем телевизоре нет блока ДМВ, то компьютер можно легко переделать на работу в метровом диапазоне (качество изображения при этом, как правило, повышается).

Вскройте компьютер. Предельно аккуратно выньте пластиковые шлейфы клавиатуры из разъемов и отложите клавиатуру в сторону. В левом дальнем углу платы Вы увидите металлический кожух размером несколько меньше спичечного коробка. Вскройте его. Внутри находится катушка индуктивности. Аккуратно выпаяйте ее. На ней намотан один неполный виток. Если намотать примерно 5 витков тем же проводом, то компьютер будет работать в метровом диапазоне (где-то между 5-ым и 8-ым каналом). Отрегулируйте катушку так, чтобы не было накладок на канал, по которому идут телевизионные передачи.

Если Ваш телевизор имеет вход "VIDEO", то качество изображения можно еще улучшить, если сделать видеовыход из компьютера.

Из модулятора выходят два провода. На одном из них +58, а на другом - низкочастотный сигнал, который можно непосредственно подать на вход "VIDEO" телевизора через коаксиальный кабель. Экран кабеля при этом припаивается к "земле" компьютера. Регулировать этот сигнал можно, если произвести такое подключение через транзистор (любой доступный типа p-p-n). Тогда низкочастотный сигнал подается на базу транзистора, сигнал "+58" - на коллектор, а эмиттер через резистор 150 Ом подключается "на землю". Сигнал для видеовыхода снимается с эмиттера через резистор 27 Ом. Резисторы можно подбирать для получения наиболее

приемлемого изображения. Помните, что от качества изображения зависит состояние Ваших глаз и глаз Ваших детей.

3. Клавиатура. Режимы работы. Курсоры.

Клавиатура Спектрума имеет всего 40 клавиш. В принципе это очень мало, и здесь наглядно проявился результат неуклонной борьбы К.Синклера за миниатюризацию и дешевизну своих изделий. Тем не менее, те, кто привык относиться к ней с уважением даже с любовью.

Каждая клавиша может выполнить по 6-7 различных действий в зависимости от того, в сочетании с какими клавишами она нажимается. Рассмотрим некоторые характерные клавиши.

ENTER - нажатие этой клавиши обычно завершает ввод в компьютерах чего-либо (команды, данных, программной строки). На компьютерах других систем эта клавиша может иметь другое наименование **RETURN**, **ВК** (возврат каретки) и др.

CFPS SHIFT - переключение регистра на печать прописными буквами (аналогично пишущей машинке).

SYMBOL SHIFT - переключение регистра для печати символов (+, -, /, и т.д.), а также некоторых ключевых слов (**THEN**, **TO**, **OR**, **AND**, **STEP** и др.)

SPACE - пробел. Эта же клавиша выполняет **BREAK** (прерывание исполнения программы), если ее нажать вместе с **CAPS SHIFT**.

Остальные клавиши имеют буквенное или цифровое обозначение.

Если Вы уже подключили компьютер, то на экране должно появиться исходное сообщение 1982 Sinclair Research Ltd. Это означает, что первичные проверки в компьютере прошли нормально и, скорее всего, он исправен и готов к работе.

Нажмите клавишу **ENTER**. На экране появится черный квадрат с буквенным обозначением на нем. Это курсор. Во-первых, он показывает то место на экране, в котором будет помещаться очередной набранный Вами символ, а, во-вторых, указывает в каком режиме находится клавиатура. Этих режимов - четыре; и им соответствуют пять разных курсоров.

3.1 Командный режим.

Курсор - "К". Он означает, что сейчас при нажатии клавиши будет введена команда, которая закреплена за этой клавишей, или цифра. Компьютер сам понимает, что строка может начинаться только либо с номера строки, либо с команды, поэтому дает курсор "К". После того, как команда набрана, и дальше должны пойти ее

параметры, режим автоматически переключается на литерный (курсор "L"). Если Вы введете ":" (двоеточие), то курсор опять переключится на "K", т.к. двоеточие является разделителем между несколькими командами, если они записываются в одной строке.

3.2 Литерный режим.

Курсоры - L, C. Курсор L соответствует печати строчными буквами, а курсор C - прописными. Переключиться на курсор "C" можно командой CAPS LOCK. Это выполняется одновременным нажатием CAPS SHIFT и цифры "2". В этом же режиме набираются символы и служебные слова, связанные с клавишей "SYMBOL SHIFT".

3.3 Расширенный командный режим.

Курсор "E". В этом режиме набираются команды, записанные над клавишами или под клавишами. Переход в режим "E" выполняется одновременным нажатием "CAPS SHIFT" и "SIMBOL SHIFT". Он действует только на одно нажатие. Если в режиме "E" нажать какую-либо клавишу, то появится слово, записанное над клавишей, а если нажать эту клавишу совместно с CAPS SHIFT, то слово, записанное под клавишей. Здесь имеется исключение для цифровых клавиш (верхний ряд). Чтобы набрать слово, записанное под клавишей, надо в режиме "E" нажать не CAPS SHIFT, а SIMBOL SHIFT совместно с клавишей. Такая уникальная способность "Спектрума" вводить операторы и функции не по буквам, а одним нажатием клавиши, называется токенизированной формой записи ключевых слов. Сначала это выглядит несколько сложновато, но опыт приходит быстро, а с ним удобство работы. По крайней мере Вы не сделаете ошибку в правописании английских слов, если этот язык Вам знаком плохо.

3.4. Графический режим.

Курсор "B". В этом режиме набираются символы блочной графики, расположенной на цифровых клавишах, а также символы графики пользователя (UDG - user defined graphics), но о них речь пойдет позже.

Переход в графический режим выполняется одновременным нажатием клавиш CAPS SHIFT и цифры "9". Если Вы в этом режиме нажмете, например, клавишу "5", то на экране появится квадрат, левая половина которого черная, а правая - белая. Нажатие клавиши "5" совместно с CAPS SHIFT изобразит тот же квадрат в инвертированном виде, т.е. левая половина будет белая, а правая черная.

Выход из графического режима выполняется так же, как и вход.

Пример.

Если Вы уже набрали что-то произвольное на экране, то скорее всего это неправильная строка и при нажатии ENTER на экране появится знак "?". Чтобы начать работу, надо эту строку стереть. Это выполняется командой DELETE (одновременное нажатие CAPS SHIFT и цифры 0). Когда все символы будут уничтожены, курсор встанет на исходную позицию и примет вид "K", то есть компьютер ждет от Вас команду.

Нажмите клавишу "R". На экране появится команда RUN. Курсор изменится на "L". Еще раз нажмите ту же клавишу - появится буква г. Если клавишу задержать в нажатом положении, то буква будет повторена многократно. Эта удобная функция называется "автоповтор" (кстати он действует и при стирании).

Нажатие этой клавиши совместно с CAPS SHIFT даст прописную букву R, а совместно с SYMBOL SHIFT - знак "<".

Перейдите в расширенный командный режим - CAPS SHIFT + SYMBOL SHIFT одновременно. Появится курсор "E". Теперь нажатие той же клавиши даст оператор INI, снова войдите в режим "E" и нажмите клавишу совместно с CAPS SHIFT. Получите команду VERIFY.

Те, кто работает с компьютерами "Спектрум+", "+128", "+2" и "+3", могут упростить свою работу благодаря наличию дополнительных клавиш, которые выполняют ряд функций одним нажатием.

DELETE - стирание символа.

GRAPH - переход в графический режим.

EXTENDED MODE - переход в расширенный командный режим.

EDIT - выполнение редактирования строки.

CAPS LOCK - переключение регистра на печать прописными буквами.

INV. VIDEO - включение инверсного режима (печать белых по черному).

TRUE VIDEO - возвращение из инверсного режима.

Кроме того, там имеется дополнительные клавиши для набора знаков препинания: точка, запятая, точка с запятой, кавычки, а также четыре клавиши для управления перемещением курсора при редактировании (см. ниже).

Для тех, кто работает с самодельными компьютерами и лишен возможности иметь замечательную инструкцию, прилагающуюся к "Спектрум'48" (к сожалению, качество инструкции к "+" и к "+128" не выдерживает никакой критики), мы приводим расписание кла-

виатуры "Спектрума", а также список ключевых слов компьютера с указанием назначения и примеров применения.

Таблица 1.

Расписание клавиатуры компьютера

Клавиша		Курсор				
		K	L		E	
			CAPS SHIFT	SYMBOL SHIFT		CAPS SHIFT
A	NEW	a	A	STOP	READ	~
B	BORDER	b	B		BIN	BRIGHT
C	CONTINUE	c	C	?	LPRINT	PAPER
D	DIM	d	D	STEP	DATA	\
E	REM	e	E	>=	TAN	ATN
F	FOR	f	F	TO	SGN	{
G	GO TO	g	G	THEN	ABS	}
H	GO SUB	h	H	^	SQR	CIRCLE
I	INPUT	i	I	AT	CODE	IN
J	LOAD	j	J	-	VAL	VAL\$
K	LIST	k	K	+	LEN	SCREEN\$
L	LET	l	L	=	URS	ATTR
M	PAUSE	m	M	.	PI	INVERSE
N	NEXT	n	N	,	INKEV\$	OVER
O	POKE	o	O	;	PEEK	OUT
P	PRINT	p	P	"	TAB	копирайт
Q	PLOT	q	Q	<=	SIN	RSN
R	RUN	r	R	<	INT	VERIFY
S	SAVE	s	S	NOT	RESTORE	
T	RANDOMIZE	t	T	>	RND	MERGE
U	IF	u	U	OR	CHR\$	
V	CLS	v	V	/	LLIST	FLASH
W	DRAW	w	W	<>	COS	ACS
X	CLEAR	x	X	!	EXP	INK
Y	RETURN	y	Y	AND	STR\$	
Z	COPY	z	Z	:	LN	BEEP
0	0	0	DELETE	_		FORMAT

Клавиша		Курсор				
		K	L			E
			CAPS SHIFT	SYMBOL SHIFT		CAPS SHIFT
1	1	1	EDIT	!	■	DEF FN*
2	2	2	CAPS LOCK	@	■	FN*
3	3	3	TRUE VIDEO	#	■	LINE*
4	4	4	INV. VIDEO	\$	■	OPEN#*
5	5	5	курсор влево	%	■	CLOSE#*
6	6	6	курсор вниз	&	■	MOVE*
7	7	7	курсор вверх	'	■	ERASE*
8	8	8	курсор вправо	(	■	POINT*
9	9	9	GRA- PHICS	)	■	CAT*

*Примечание: Символы верхнего ряда в расширенном командном режиме нажимаются не с CAPS SHIFT, а с SYMBOL SHIFT.

Таблица 2.

Ключевые слова Спектрума

Условные обозначения: адрр - адрес от 0 до 65535;

A - действительное число; N, M, X, Y, R - целое число.

Ключевое слово	Пример применения	Назначение
Функции		
ABS	LET A = ABS -3.5	Дает абсолютную величину числа.
ACS	LET A = ACS 0.5	Аркосинус в радианах.
AND		Логическое "И".
ASN	LET A = ASN 0.5	Арсинус в радианах.
ATN	LET A = ATN 1.5	Арктангенс в радианах.

Ключевое слово	Пример применения	Назначение
ATTR	PRINT ATTR (10,5)	Дает число,показывающее состояние цветовых атрибутов в заданном знакоместе экрана. Если это число разложить на биты,то бит 7 (старший) указывает на признак мигания. 1 - есть; 0 - нет. Бит 6 - признак яркости. биты 3...5 - цвет фона (от 0 до 7); биты 0...2 - цвет символа.
BIN	PRINT BIN 10011001	Перевод числа из двоичной формы в десятичную.
CHRS	PRINT CHRS N	Дает символ,код которого равен N.
CODE	PRINT CODE ABCD\$	Дает код первого символа строки (стринга).
COS	LET A = COS 0.5	Косинус числа.
EXP	LET A = EXP 0.5	Экспонента числа.
FN	PRINT FN W(X,Y...)	Вызов функции пользователя W,заданной DEF FN W. Аргумент должен быть в скобках. Если аргумента нет,скобки все равно должны быть.
IN	PRINT IN X	Ввод данных с внешнего порта номер X.
INKEY\$	LET A\$ = INKEY\$	Читает клавиатуру. Выдает символ,равный тому,клавиша которого была нажата; если ничто не нажато,то выдает пустой стринг. INKEY\$ не ждет нажатия. Это обходят например так: 10 LET A\$ = INKEY\$ 20 IF A\$ = "" THEN GO TO 10 30 ... продолжение работы после нажатия.

Ключевое слово	Пример применения	Назначение
INT	LET N = INT A	Выделение целой части числа (округление вниз).
LEN	LET N = LEN ABCD	Дает длину строки в символах.
LN	PRINT LN 2.345	Натуральный логарифм.
NOT		Логическое отрицание "НЕ".
OR		Логическое "ИЛИ".
PEEK	LET N = PEEK адр	Дает содержимое ячейки памяти по ее адресу.
PI		3.14159265
POINT	PRINT POINT (X,Y)	Проверяет включена или выключена точка экрана с координатами X,Y. Если включена - 1, если нет - 0. X = 0...255; Y = 0...175.
RND	LET N = INT (5*RND)	Дает псевдослучайное число между 0 и 1.
SCREEN\$	PRINT SCREEN\$(X,Y)	Дает символ, равный тому, который находится на экране в знакоместе с координатами X,Y. Если символ не распознаем, то выдается пустой строка. X = 0...23; Y = 0...31.
SGN	PRINT SGN A	Знак числа. 1 - положительное, -1 - отрицательное, 0 - если число равно нулю.
SIN	LET A = SIN 2.345	Синус угла, заданного в радианах.
SQR	PRINT SQR 25	Квадратный корень.
STR\$	PRINT STR\$ 100	Перевод числа в строковую переменную (строка), изображающую то же число. STR\$ 100 = "100".
TAN	LET A = TAN 2.345	Тангенс угла, заданного в радианах.

Ключевое слово	Пример применения	Назначение
USR addr.	RANDOMIZE USR 30000	Вызывает подпрограмму в машинном коде и стартует ее с указанного адреса.
USR A\$	LET addr. = USR "A"	Дает адрес в области графики пользователя UDS, в котором расположено изображение символа графики пользователя, соответствующего "A".
VAL	LET A = VAL "100"	Функция, обратная STR\$. Переводит стринг в соответствующее число.
VAL\$	PRINT VAL\$ ""100""	Рассматривает свой стринговый параметр без ограничительных кавычек как новый стринг.
Операторы		
BEEP	BEEP 20.0, 0.5	Воспроизводит в динамике компьютера звуковой сигнал продолжительностью 20 секунд и с высотой на 0.5 полутона выше, чем "ДО" первой октавы.
BORDER	BORDER N	Установка цвета бордюра. N = 0...7
BRIGHT	BRIGHT i	Установка яркости. 0 - Нормальная, 1 - повышенная.
CAT		Распечатать каталог картриджа микродр-йва.
CIRCLE	CIRCLE X,Y,R ..	Вычерчивание окружности с центром X,Y и с радиусом R.
CLEAR	CLEAR ADDR	Вычищает все переменные, выполняет RESTORE и CLS. Устанавливает новую верхнюю границу памяти по адресу ADDR.

Ключевое слово	Пример применения	Назначение
GLOSE #		Команда ИНТЕРФЕЙСА-1 (закрывать поток).
CSL		Очистка экрана.
CONTINUE		Продолжить исполнение программы после прерывания.
COPY		Команда ZX-принтера. Выполняет копирование экрана на бумагу.
DATA		Данные для оператора READ.
DEF FN	DEF FN W(X,Y,...)	Задаёт функцию пользователя. Имя функции W должно быть однобуквенным, а параметры - стоять в скобках. Если параметров нет - скобки все равно должны быть.
DIM	DIM A(N1,N2,...NK)	Задаёт k-мерный массив A. Устанавливает исходные значения элементов массива в нуль. Уничтожает все массивы с тем же именем.
	DIM A\$(N1,N2,...NK)	Задаёт символьный k-мерный массив. Исходное значение элементов массива устанавливается в " ". Его можно рассматривать как k-1-мерный массив стрингов фиксированной длины, равной NK.
DRAW	DRAW X,Y	Выполняется вычерчивание прямой линии от точки, в которой производилась печать в последний раз до точки, отстоящей от нее по горизонтали на X и по вертикали на Y.

Ключевое слово	Пример применения	Назначение
	DRAW X,Y,Z	То же самое, но между этими двумя точками вычерчивается не прямая, а дуга окружности, охватывающая угол Z, где угол измеряется в радианах.
ERASE		Команда микродрайва.
FLASH	FLASH I	Признак мигания. I - включено; 0 - выключено.
FOR	FOR I = x TO	Составная часть оператора FOR..TO..NEXT.
	FOR I = x TO y STEP z	Организует цикл с начальным значением параметра цикла x, конечным значением y и шагом z. Шаг может быть и отрицательным. В этом случае начальное значение больше, чем конечное.
FORMAT		Команда микродрайва.
GO SUB	GO SUB N	Переход к исполнению подпрограммы, расположенной начиная со строки N. Подпрограмма должна завершаться командой RETURN.
GO TO	GO TO N	Переход к исполнению строки N.
IF X THEN Y	IF A > 5 THEN GO TO N	Условный оператор. Если выражение X справедливо, то выполняется выражение после THEN. В противном случае выполняется переход к следующей строке. (Не к следующему оператору в этой строке, а именно к следующей строке! Типичная ошибка!)
INK	INK N	Устанавливает цвет печати N=0...7.

Ключевое слово	Пример применения	Назначение
INPUT	INPUT A	Служит для ввода с клавиатуры числа во время исполнения программы. Может вводиться и символьный стринг. В этом случае оператор должен иметь форму INPUT LINE.
INVERSE	INVERSE 0	Признак инверсии. 1 - включено (печатать белым по черному). 0 - нормальная печать.
LET	LET A = X	Оператор присваивания. Переменной A присваивается значение X.
LIST	LIST	Выполняется печать листинга (текста) программы на экране. В форме LIST N листинг выдается, начиная со строки N. Если текст занимает более чем один экран, то после заполнения экрана выдается запрос SCROLL? Если нажать клавиши BREAK или N
LLIST	LLIST N	Команда ZX-принтера. То же, что и LIST, но распечатка выполняется на ZX-принтере.
LOAD	LOAD "имя"	Загрузка БЕЙСИК-программы с магнитофона. Если имя не указано (LOAD ""), то загрузится первая встреченная программа. Обратите внимание: между кавычками не должно быть пробела.

Ключевое слово	Пример применения	Назначение
	LOAD "имя" CODE M,N	Команда на загрузку блока в машинных кодах. Загружает N байтов, начиная с адреса M. Если эти параметры не указаны, то блок будет загружаться в те адреса, из которых он был когда-то выгружен.
	LOAD "имя" DATA ()	Загрузка блока данных.
	LOAD "имя" DATA\$ ()	Загрузка символьного массива.
	LOAD "имя" SCREEN\$	Загрузка экрана. То же самое, что и LOAD "имя" CODE 16384,6912. Экранная область памяти "СПЕКТРУМА" всегда начинается с 16384 и имеет 6912 байтов (6144 - графика; 768 - атрибуты).
LPRINT		Команда ZX-принтера. СМ. PRINT.
MERGE	MERGE "имя"	Аналогична LOAD, но при загрузке новой программы не уничтожает старую, а производит их слияние. При этом остается старая программа с добавлением таких строк из новой программы, каких в старой программе не было. Для блоков, записанных в машинных кодах, этот оператор неприменим.
MOVE		Команда микродрайва.
NEW		Уничтожает БЕЙСИК-программу, устанавливает в исходное положение системные переменные, очищает память до границы RAMTOP, установленной CLEAR.

Ключевое слово	Пример применения	Назначение
NEXT	см. FOR	Часть оператора FOR..NEXT. Заканчивает цикл.
OPEN#		Команда интерфейса-1 и микродрайва.
OUT	OUT M,N	Выдает байт N на внешний порт M.
OVER	OVER I	Управляет условием наложения символов друг на друга при печати. 0 - новый символ полностью закрывает старый. 1 - изображение получается смещением символов.
PAPER	PAPER N (N=0...)	То же, что INK, но задает цвет фона.
PAUSE	PAUSE N	Пауза в работе программы на время N/50 секунд или до нажатия какой-либо клавиши.
	PAUSE 0	Пауза до нажатия клавиши.
PLOT	PLOT C;M,N	Выполняет печать точки на экране с цветом C в координатах M,N. Если параметр C не задан, то устанавливается текущий цвет. M = 0...255. N = 0...175. Обычно эта команда предшествует DRAW.
POKE	POKE адр,N	Засылает байт по указанному адресу.
PRINT	PRINT N	Печать числа.
	PRINT X	Печать результата расчета выражения X.
	PRINT "ABCD..."	Печать текста, стоящего в кавычках.
	PRINT A\$	Печать строки (строковой переменной).

Ключевое слово	Пример применения	Назначение
		Одной командой PRINT может выполняться печать сразу нескольких переменных или чисел. При этом между ними должен стоять разделитель. Таких разделителей три. Это запятая (,), точка с запятой (;) или апостроф('). Если стоит запятая, то печать выполняется на той же строке, но начиная с 16-ой колонки, т.е. в правой половине экрана. Точка с запятой означает печать в той же строке без зазора. Апостроф дает переход к новой строке печати. Пример PRINT A,B,C;D;E\$'F\$'N. Просто PRINT дает пустую строку. Так, PRINT: PRINT: PRINT эквивалентно PRINT ""
	PRINT AT M,N	Печать в заданной позиции (строка,столбец).
	PRINT TAB N	Печать с табуляцией. N показывает величину табуляции. Оператор PRINT может сопровождаться операторами задания цветовых атрибутов INK,PAPER,FLASH,BRIGHT, INVERSE,OVER. В этом случае они вводятся через точку с запятой. Пример: PRINT INK 6; PAPER 1; "VERY WELL", FLASH 1;" "

Ключевое слово	Пример применения	Назначение
RANDOMIZE N		Устанавливает системную переменную SEED (см. ниже). Эта системная переменная используется для работы генератора псевдослучайных чисел (см. RND)
RANDOMIZE		То же, что RANDOMIZE 0.
RANDOMIZE USR	RANDOMIZE USR адрр.	Вызывает и стартует процедуру, записанную в машинных кодах, находящуюся по указанному адресу. RANDOMIZE USR 0 инициализирует компьютер, т.е. выполняет эффективный сброс, эквивалентный выключению питания. Сравни с NEW.
READ	READ V1,V2,V3...VN	Присваивает переменным значения, находящиеся в соответствующем операторе DATA.
REM		Пустой оператор. Открывает строку комментариев. Все, что записано за REM (в пределах строки) не исполняется, а только распечатывается по LIST. Это помогает впоследствии разобраться с программой.
RESTORE N		Устанавливает указатель DATA, привязывающий DATA к соответствующему READ, так, чтобы он указывал на DATA, расположенной в строке N или, если такой нет, то к ближайшей старшей.

Ключевое слово	Пример применения	Назначение
RESTORE		То же, что и RESTORE 0.
RETURN		Пустой оператор. Завершает подпрограмму, к которой обращались по GO SUB. Передает управление к строке, следующей за вызывающей.
RUN	RUN N	Команда на исполнение программы. Все переменные, существовавшие до нее, уничтожаются и начинается исполнение со строки N. Эквивалентна CLEAR: GO TO N.
	RUN	То же, что и RUN 0.
SAVE	SAVE "имя"	Выполняет выгрузку БЕЙСИК-программы на магнитную ленту. Имя давать обязательно. Требуется нажатия ENTER. После ENTER появляется указание "START TAPE, THEN PRESS ANY KEY" (Запустите ленту и нажмите любую клавишу).
	SAVE "имя" LINE N	Записывает программу на ленту так, что она после загрузки будет автостартовать со строки N.
	SAVE "имя" CODE M,N	Запись на ленту блока машинных кодов, начиная с адреса M, в количестве N байтов.
	SAVE "имя" DATA ()	Запись числового массива.
	SAVE "имя" DATA\$ ()	Запись на ленту символьного массива.

Ключевое слово	Пример применения	Назначение
	SAVE "имя" SCREENS	Запись экрана. То же самое
STOP		Останавливает исполнение программы. Продолжить можно оператором CONTINUE.
VERIFY	VERIFY " "	Аналогично LOAD, но в отличие от нее не производит загрузку в ОЗУ, а только сверяет то, что записано на ленте с тем, что находится в памяти компьютера и в случае несоответствия выдает сообщение об ошибке. Применяется для проверки качества записи. Имеет все те же формы записи, что и оператор LOAD.

Примечание. Выше мы не рассматривали ряд операторов, указав, что они имеют отношение к Интерфейсу-1 и к микродрайву. Интерфейс-1 - это устройство, выполняющее одновременно три функции - контроллера локальной сети, порта RS232 и контроллера микродрайва. Выпускается той же фирмой SINCLAIR RESEARCH LTD. Имеет собственное ПЗУ и служит для подключения компьютера к другим компьютерам (практически любым), к глобальной сети (через модем) и для объединения класса "СПЕКТРУМОВ" в локальную сеть (до 64-х шт.) Команды Интерфейса-1, строго говоря, уже не являются операторами БЕЙСИКа как языка программирования, а командами операционной системы.

ZX-микродрайв - устройство, работающее с высокоскоростной закольцованной магнитной лентой, находящейся на сменных картриджах. Оставаясь ленточным носителем, он создает эффект работы с диском. Емкость картриджа порядка 100К. Одновременно можно подключить до восьми микродрайвов.

4. Загрузка программ с магнитофона

Если Вы недавно приобрели компьютер, то вполне понятно желание начать работу с готовыми фирменными программами, а самостоятельное программирование отложить на несколько более поздний срок.

Загрузка и запуск фирменных программ выполняются весьма просто:

LOAD "" <ENTER>

Помните, что между кавычками не должно быть пробела.

Основную проблему на этом этапе представляет плохая загрузаемость программ, о чем свидетельствует сообщение об ошибке TAPE LOADING ERROR, зависание программы после загрузки или самопроизвольный сброс. Причина не обязательно связана с плохой загрузкой, но если лента получена Вами из стороннего источника, то это скорее всего так. Здесь мы приведем некоторые принципы, которыми надо руководствоваться, чтобы улучшить загрузаемость программ.

1. По-настоящему надежную загрузку может обеспечить только высококачественная лента. Если она записана на том же магнитофоне, на котором и воспроизводится. Поскольку стоимость программ обычно значительно превышает стоимость ленты, то не надо на ней экономить, а понравившиеся Вам программы надо откопировать на своем магнитофоне (см. ниже).

2. Для работы с компьютером годится любой бытовой кассетный магнитофон, причем чем он проще, тем легче добиться надежной работы. В то же время желательно, чтобы он имел регуляторы тембров (по-возможности отдельные). Очень хорошо, если есть регулировка скорости движения ленты. Компьютер весьма чувствителен к отклонениям скорости. Монофонические магнитофоны во всех случаях предпочтительнее, чем стерео. Обязательным элементом является счетчик ленты, позволяющий быстро находить нужную программу.

3. Приступая к загрузке программы, установите уровень громкости на магнитофоне примерно на 70%-80% от максимального и, если загрузка не пошла, поэкспериментируйте с ним. Помните, что программы чаще страдают от избыточного уровня, чем от недостаточного. Поиск ведите в сторону уменьшения уровня. Иногда на фирменных компьютерах помогает следующий прием. Загрузочный шнур вставляют не в гнездо EAR компьютера, а в гнездо MIC. На самодельных аппаратах этот метод не сработает, т.к. там эти цепи электрически разорваны.

4. Установка регуляторов тембра также имеет большое значение. Нормально "высокие" надо ставить на 80% от максимума, а "низкие" примерно на 20%. Если программа решительно не загружается, а регулировка громкости ничего не дает, то попробуйте установку 50%-50% и 20%-80%.

5. Загрузка программы сопровождается перемещением цветных полос по бордюроному полю телевизора. По нему можно также многое сказать о настройке магнитофона. Сначала идут в течение 3-х - 5-ти секунд широкие красные и голубые полосы. Это так называемый пилоттон. Он нужен для того, чтобы процессор подготовился к загрузке. Ширина красных и голубых полос должна быть одинаковой. Если голубые полосы шире, то значит уровень слишком большой. Если шире красные полосы - слишком низкий. Эти полосы должны медленно перемещаться по экрану. Слишком быстрое их перемещение говорит о несоответствии скоростей Вашего магнитофона и того, на котором была сделана запись. Неравномерное перемещение (с ускорением и рывками) свидетельствует о некачественности лентопротяжной системы магнитофона либо Вашего, либо того, на котором делалась запись.

Бывают случаи, когда широкие красные и голубые полосы "превращаются" в широкие желтые и синие. Это может указывать на то, что следует поменять местами установку регуляторов тембра.

6. Если у Вас стереомагнитофон и программа не загружается, попробуйте загружать ее отдельно по левому и правому каналам.

7. Если все пути регулировки исчерпаны, но программа не загружается и при этом точно известно, что копия работоспособна, по всей видимости причина в настройке головок у писавшего и воспроизводящего магнитофона. Как правило, недорогие монофонические магнитофоны имеют в крышке отверстие для узкой крестовой отвертки, чтобы можно было при воспроизведении записи на слух поворотом подружженного винта регулировать азимутальное положение головки. При этом следует добиваться наиболее четкого и резкого звучания сигналов. Обычно бывает достаточно поворота в ту или иную сторону на угол до 120 градусов. Если у Вас только один магнитофон, и Вы используете его также и для записи программ, то регулировать головку нежелательно, либо придется периодически сталкиваться с необходимостью юстировки. Самый лучший вариант - использовать для воспроизведения программ, полученных со стороны, отдельный магнитофон, выбрав его из самых простых и дешевых, а для записи программ - наиболее качественный, по возможности специализированный аппарат.

8. Обычно программы состоят из нескольких блоков, которые загружаются последовательно один за другим. При плохой загрузке

бывает очень трудно загрузить программу, разные блоки которой требуют разной установки органов управления магнитофона. Для того, чтобы обойти эту проблему, надо проводить загрузку в копирующую программу. Если, предположим, при загрузке второго блока произойдет сбой, то первый блок уже загружен и остается так, что его повторять не надо. К тому же, по окончании длительной трудоемкой загрузки, можно сразу же выполнять качественную копию и больше не иметь с этой программой проблем.

9. Сделать процесс настройки магнитофона наглядным и значительно более простым можно с помощью специальных программ, например, с помощью программы TAPER.

Программа TAPER

Программа предназначена для настройки кассетного магнитофона. Она анализирует (раскладывает по частотам сигнал, проходящий на вход компьютера и изображает его на экране в виде спектра. Вертикальная шкала имеет три метки. Против этих меток должны находиться пики сигналов во время загрузки. Самая верхняя метка указывает на пик во время считывания "единиц". При прослушивании цифровой записи через программу TAPER пики в участках, отмеченных метками, должны быть ярко выражены и никаких других побочных всплесков не должно быть.

5. Как выполнить запасную копию программы

Если Вы сами написали какую-либо программу на БЕЙСИКе, то выполнить ее копию несложно. Для этого достаточно дать команду `SAVE "имя" LINE N`. Если в программе есть блоки, записанные в машинном коде, то дается команда `SAVE "имя" CODE M,N`, где `M` - адрес, с которого начинается блок кодов, а `N` - длина этого блока в байтах. При загрузке программ блоки, машинных кодов как `BYTES`... Если Вы загружаете свою разработку, то Вы, конечно, знаете те адреса, в которых располагаются Ваши блоки. Однако, если Вы имеете дело с чужой (фирменной) программой, то загрузив ее, Вам не так-то просто ее выгрузить. Во-первых, она автостартует и остановить ее сложно. Во-вторых, Вам неизвестны адреса блоков, из которых она состоит.

Упростить процесс копирования и автоматизировать его можно с помощью специальных копирующих программ. Эти программы имеют свою нестандартную загрузочную систему. Таких программ очень много. Мы здесь рассмотрим несколько наиболее широко распространенных.

COPY 86/M

Это, по-видимому, наиболее удобный и наглядный копировщик общего применения. В исходном состоянии он имеет объем свободного пространства 45000 байтов, и способен компрессировать данные при загрузке. Компрессирование состоит в том, что, например, вместо последовательности из N нулевых байтов записывается один "нуль", а следующий байт указывает сколько их (N). Это позволяет загружать в копировщик значительно более 45К. Особенно много места экономится при загрузке экранов, т.к. в графических изображениях часто встречаются длинные последовательности нулей.

Результат компрессии данных отправляется на хранение в область экрана, что изображается появлением на экране полос и точек, напоминающих телеграфный код.

Сводка команд программы COPY 86/M.

BREAK - переход в исходное положение.

L - загрузить файлы.

C - выгрузить (скопировать файлы). Выгрузка производится с паузой 1.5 секунд между записями.

M - то же, но с паузой 3 секунды.

V - VERIFY - проверка выгруженных записей.

D - DELETE - удаление ненужных записей.

H - перевод в шестнадцатичную систему.

X - просмотр записей с отбоем ранее поданных команд.

S - то же, но без отбоя команд.

B - просмотр БЕЙСИК-программы.

A - ALL - после этой команды очередная команда будет одновременно распространяться на все записи.

Все команды подтверждаются ENTER.

На экране файлы записываются в "окне", имеющем 4 строки. Если записей более чем 4, то одновременно их вывести на экран нельзя, поэтому и необходимы клавиши S и X для "протягивания" записей через "окно".

При использовании программы следует помнить, что она может не работать с некоторой периферией. Так, она работать не будет, если к компьютеру подключен кемпстон-джойстик.

К сожалению, эта программа не работает и со многими самодельными компьютерами. В таком случае как правило пользуются другой компрессирующей программой TF COPY.

TF COPY

После загрузки программы на экране появляется основное меню программы, которое выглядит так:

LOAD SAVE DELETE VERIFY MODE.

Нажав первую букву нужной команды, Вы входите в нужный Вам режим.

LOAD - загрузка файлов.

SAVE - выгрузка файлов.

DELETE - удаление файлов.

VERIFY - проверка выгруженный файлов.

MODE - переключение режима. В программе имеются 3 режима работы.

Режим 1 - 41984 байта свободной памяти.

Режим 2 - 44032 байта свободной памяти.

Режим 3 - 44288 байтов свободной памяти.

Все команды должны завершаться нажатием ENTER.

Обратите внимание, что при переходе на другой режим происходит очистка памяти, т. е. загруженная информация будет уничтожена.

После выбора одного из вышеуказанных режимов программа выдает запрос. Например, нажатие "D" дает режим DELETE и появляется запрос:

DELETE FROM ...TO Удалить запись с ... по

В ответ надо дать номера тех записей, которые Вам нужны. Например, если Вы хотите уничтожить записи с третьей по шестую, то соответственно надо ввести 3 и 6.

COPY-COPY (PIRATE 02)

Эта программа находится в эксплуатации довольно давно и представляет классический образец копировщика. Она не выполняет компрессирования и имеет довольно ограниченный объем свободной памяти (42К), но в ней есть несколько оригинальных вспомогательных режимов, которые в ряде случаев делают ее незаменимой.

Сводка команд

Все команды являются ключевыми словами "СПЕКТРУМА" и потому не набираются по буквам. Они требуют завершения нажатием клавиши ENTER.

LOAD - выполнить загрузку очередной записи (очередного блока).

LOAD N TO M - выполнить загрузку записей, начиная с номера N до номера M. При этом ранее существовавшие записи с номером N и выше будут уничтожены, т.е.

LOAD 1 уничтожит все ранее существовавшие записи и подготавливает компьютер к загрузке новых.

Возможные вариации этой команды:

LOAD N TO

LOAD TO M

SAVE - выполнение выгрузки записи (записей) на ленту. Может иметь форму:

SAVE N TO M
SAVE TO M
SAVE N TO
SAVE N TO M STEP K

Параметр K после оператора **STEP** показывает величину паузы в секундах, которую компьютер выдержит между отдельными блоками.

VERIFY - проверка выгруженных записей. Имеет все те же формы, что и команда **LOAD**.

LOAD AT ADDR - загрузка блока, начиная с **ADDR**.

Например

LOAD AT 16384 - загрузка экрана.

LIST ADDR - выдача на экран содержимого памяти, начиная с указанного адреса. **LIST = LIST 0**.

POKE ADDR,N - поместить по указанному адресу байт N.

Этой командой часто пользуются совместно с командой **LIST**. Они позволяют вносить изменения в машинный код программы. В принципе с подобными задачами лучше справляются специализированные программы для отладки машинного кода (их часто называют **МОНИТОРЫ**), но наличие такой функции в копировщике делает его весьма удобным.

COPY 16384 - команда, которая позволяет копировать блоки длиной 49K. Этот режим делает данную программу незаменимой во многих случаях. Дело в том, что если программа имеет один значительный блок длиной порядка 49K и не поддается компрессии, то откопировать его копировщиками типа **COPY 86/M** и **TF COPY** не удастся, т.к. он не помещается в память. (Например, программы **FIRELORD** и **URIDIUM** фирмы **HUSON CONSULTANTS** и другие.) В режиме **COPY 16384** программа **COPY-COPY** удаляет саму себя из памяти и оставляет только небольшой блок длиной несколько байтов, предназначенный для выгрузки загруженного блока. Выгрузка выполняется нажатием клавиши **CAPS SHIFT**. Возможна только однократная выгрузка, после этого копировщик необходимо снова загрузить. Разумеется магнитофон должен быть включен на запись до нажатия **CAPS SHIFT**.

Дополнительные сведения.

Мы рассмотрели только три наиболее распространенные копирующие программы. Н самом деле их очень много, но принципы их работы отличаются не очень значительно. В то же время, надо помнить, что с помощью копировщиков можно только копировать программы, в которых не принято специальных мер для защиты от

копирования. Многие фирмы такие меры применяют. Это, например "спидлок" (ускоренный загрузчик), джеркитон (пилоттон, сопровождающийся рывками), блоки избыточной длины (более 50К), фальшхедеры (блоки длиной 17 байтов, которые воспринимаются копировщиком как заголовок очередного блока, в то время, как они таковым не являются), ускоренный пилоттон, удлиненный пилоттон, слишком широкий или слишком узкий пилоттон, блоки с замеряемой паузой между ними и многие другие методы. Принцип их действия состоит в том, что первый блок программы является БЕЙСИК-загрузчиком, который подготавливает загрузку второго блока. Второй блок в машинных кодах подготавливает нестандартную загрузку прочих блоков, которые уже не могут быть загружены никуда, где предварительно не отработал второй блок, в том числе и в копировщик. Копирование таких программ довольно трудоемкий процесс, но со многими защитами справляется, например, специальная программа LERM-7 (ТС-7).

Другой подход к таким программам состоит в том, что используются специальные периферийные устройства. Их назначение - остановить ("заморозить") программу в каком-либо месте, а затем сделать полный дамп памяти (полную выгрузку всей памяти). Широко известны устройства типа MULTIFACE, позволяющие кроме этого производить выгрузку экрана в любом месте программы, вносить в программу изменения и продолжать исполнение программы с места прерывания. Аналогичными устройствами снабжаются иногда и дисковые системы, например, BETA ДИСК ИНТЕРФЕЙС (ТР ДОС) фирмы TECHNOLOGI RESEARCH. Он имеет специальную кнопку, называемую MAGIC BUTTON (волшебная кнопка) для прерывания работы программы и сброса ее на диск. Правда запустить откопированную таким образом программу через LOAD не удастся. Для этого интерфейс реализует нестандартный подход и выполняет это по команде GO TO "имя".

Можно предложить и такой способ копирования с помощью двух магнитофонов. Один магнитофон включается на вход компьютера, а другой - на выход. Компьютер считывает информацию бит за битом и тут же транспортирует ее на выход, при этом он выполняет роль компаратора и выравнивает фронты сигналов. Несложная программа в машинных кодах будет предложена Вам в следующей главе.

Правда, этот метод не очень надежно работает на фирменных "СПЕКТРУМАХ", т.к. у них вход и выход имеют электрическую связь, но на компьютерах "ТАЙМЕКС" и на большинстве самодельных дает отличный результат.

6. Составление и редактирование программ

Компьютер "СПЕКТРУМ" может работать в двух режимах. Это режим непосредственного исполнения команд и программный режим.

В режиме непосредственного исполнения команды выполняются после нажатия клавиши ENTER и после исполнения в памяти не хранятся. Если Вам надо повторить команду, ее придется снова набрать с клавиатуры.

Например:

```
PRINT 2*2
```

4 О.К. - лишний раз убедитесь, что дважды два - четыре. Сообщение О.К.(о'кей) говорит о том, что команда выполнена и все в порядке. Команды, подаваемые в режиме непосредственного исполнения, называют прямыми командами. Прямая команда может состоять и из нескольких команд. В этом случае они должны разделяться двоеточием. Например, для создания таблицы квадратов целых чисел от 1 до 20 дайте прямую команду:

```
FOR i=1 TO 20 : PRINT i, i*i: NEXT i
```

В программном режиме команды записываются в виде строк программы. Каждая строка начинается с номера. Строки в программе размещаются в порядке возрастания номеров, поэтому если Вы в программе после строки номер 10 наберете строку номер 5, то она встанет в программе до строки номер 10. Ввод строки в программу завершается нажатием ENTER, а начать работу программы можно прямой командой RUN или RUN N, где N - номер строки, с которой Вы хотите, чтобы программа начала работу. Можно запустить программу с нужной строки и прямой командой GO TO N.

Обычно строки программы нумеруют через десяток - 10, 20, 30, ... Это делают для того, чтобы в случае необходимости можно было бы легко вставить пропущенную строку. Так, если Вам понадобилось бы вставить строку между строкой 1 и строкой 2, Вам пришлось бы их переделать, а чтобы вставить строку между строкой 10 и строкой 20, Вам достаточно присвоить ей номер от 11 до 19, например 15.

В программе для "СПЕКТРУМА" Вы можете использовать любые номера строк от 1 до 9999. В одной строке могут быть записаны несколько команд или операторов, но они должны разделяться двоеточием.

Если Вы наберете строку с некоторым номером N, а строка с таким номером у Вас уже есть, то после нажатия ENTER новая строка встанет на свое место в программе, а старая будет уничтожена полностью. Это дает эффективный метод стирания ненужных строк.

Для этого достаточно набрать номер строки, подлежащей уничтожению, и нажать ENTER.

Программа для печати квадратов целых чисел от 1 до 20 будет выглядеть так:

```
10 FOR i=1 TO 20
```

```
20 PRINT i, i*i
```

```
30 NEXT i
```

Запускается программа командой RUN. После того как программа отработает, она, в отличие от прямой команды, остается в памяти и может быть повторена сколько угодно раз. Прервать исполнение работающей программы можно командой BREAK. Если Вам надо внести изменение в программу, то надо ее остановить, затем командой LIST или LIST M вывести на экран текст программы, а затем выполнить редактирование.

СПЕКТРУМ имеет строчный программный редактор. Это означает, что редактировать можно каждую строку отдельно. Для того, чтобы редактировать (переделывать) строку, надо сначала вызвать ее в позицию редактирования, которая расположена в нижней части экрана. Это делается так. Предположим, что вам надо устранить ошибку в работающей программе. Прервите ее нажатием BREAK. Дайте команду LIST для печати текста на экране. Если программа достаточно длинная и на экране целиком не помещается, то после того, как экран будет заполнен, Распечатка прекратится и появится запрос SCROLL? Если строка, которую Вы хотите переделывать, есть на экране, нажмите в ответ на запрос N (нет) или BREAK, а если ее нет, то любую другую клавишу и распечатка текста программы на экране будет продолжена до получения очередного запроса SCROLL? .

Когда нужная строка найдена и Вы увидели ее номер, Вы должны подвести к ней курсор. Этот курсор имеет вид стрелки "" и перемещается курсорными клавишами. CAPS SHIFT 6 вниз и - CAPS SHIFT 7 - вверх. Если программа длинная, то перегонять курсор из одного места в другое довольно утомительно. В этом случае лучше сразу обратиться по номеру строки N командой LIST N, а в ответ на запрос SCROLL? нажать BREAK. После этого курсор "" сразу будет установлен против нужной строки.

Теперь надо эту строку вызвать в позицию редактирования. Это делается ключевым словом EDIT (CAPS SHIFT 1). Строка будет повторена в нижней части экрана. Здесь установите курсор в строке с помощью курсорных клавиш справа от символа, подлежащего удалению и нажмите DELETE (CAPS SHIFT 0). Курсорные клавиши в данном случае - CAPS SHIFT 5 - влево; CAPS SHIFT 8 - курсор вправо. Ввод новых символов ключевых слов производится по месту

установки курсора. После того как редактирование строки закончено, нажмите ENTER и если отредактированная строка не содержит синтаксических ошибок, она займет свое место в программе.

Процесс редактирования упрощается несколько за счет того, что если после остановки программы нажать не LIST, а сразу ENTER, то на экране появляется текст программы, причем именно в том фрагменте, который последним редактировался или вводился. Это сокращает время на поиск нужной строки.

Конечно, стандартный редактор СПЕКТРУМа очень удобным не назовешь, но к нему можно быстро привыкнуть. Кроме того, существуют другие версии БЕЙСИКа, например, БЕТА БЕЙСИК (версия 1.0; 1.8; 3.0). Эта программа загружается с кассеты. Она предоставляет неплохие возможности для эффективного редактирования программ. Например, она имеет автоумерацию строк, возможность перенумерации строк, команду EDIT N для немедленного редактирования строки N, возможность быстрой переброски курсора, отладочные операторы, дополнительные операторы организации цикла (как в ПАСКАЛе), возможность работы с "окнами", развитые средства структурного программирования (процедуры) и еще несколько десятков удобных функций и команд. Опыт показывает, что при программировании на БЕЙСИКе эта программа сокращает время примерно в два раза на малых программах (за счет средств редактирования) и примерно в три-четыре раза на больших (более 200 строк) программах за счет средств структурного программирования и отладочных операторов.

Программа BETA BASIC занимает определенное место в оперативной памяти, но во-первых это критично только для очень больших программ, а во-вторых именно для них эта программа и предоставляет наибольшие возможности по экономии памяти за счет структурного программирования. Поэтому занятая память на больших программах с лихвой окупается, а на малых не имеет значения.

В заключение этой главы мы предлагаем Вам набрать небольшую программу, предназначенную для копирования программ с одного магнитофона на другой через компьютер.

```
10 DIM a(12)
20 FOR i = 1 TO 12
30 READ a(i)
40 POKE (29999+i), a(i)
50 NEXT i
60 DATA 243, 175, 219, 254, 31, 31, 31, 211, 254, 195, 50, 117
70 RANDOMIZE USR 30000
```

В принципе эта программа написана в машинном коде. БЕЙ-СИК служит только для того, чтобы организовать этот код по адресу 30000 и стартовать его.

7. Системные переменные.

Компьютер выделяет место в оперативной памяти от адреса 23552 по адрес 23733 для размещения в нем некоторых переменных, которые нужны для нормального функционирования компьютера. Тем не менее, пользователь может использовать информацию, содержащуюся в них для повышения эффективности программирования. Первоначальные значения системных переменных устанавливаются после включения компьютера. Их устанавливают те процедуры, которые содержатся в ПЗУ и стартуют после включения. Вы можете в любое время посмотреть содержание однобайтной системной переменной с помощью команды PRINT PEEK ADDR, где ADDR - это ее адрес, а для двухбайтной - PRINT PEEK ADDR + 256*PEEK (ADDR+1).

Вы можете и сами заслать какие-либо значения (от 0 до 255) в адрес системной переменной оператором POKE, но в некоторых случаях такое изменение будет иметь лишь кратковременный характер и предыдущее значение будет само восстановлено (такие системные переменные помечены буквой N), а в некоторых случаях изменение приведет к сбросу компьютера (такие переменные помечены буквой X).

Таблица системных переменных

Длина	Адрес	Имя	Содержание
N8	23552	KSTATE	Используется при сканировании клавиатуры.
N1	23560	LAST K	Запоминает последнюю нажатую клавишу.
1	23561	REPDEL	Время, в течении которого надо задерживать нажатую клавишу, чтобы произошел автоповтор. Измеряется в 50-х долях секунды. Исходная установка - 35, но может заслать сюда свое значение.
1	23562	REPPER	Интервал между повторениями нажатой и задержанной клавиши. Измеряется в 50-х долях секунды. Исходное значение - 5.

Дли- на	Адрес	Имя	Содержание
N2	23563	DEFADD	При расчете функции, заданной пользователем, здесь хранится адрес ее аргументов. В противном случае 0.
N1	23565	K DATA	Второй байт кода управления цветом при вводе с клавиатуры.
N2	23566	TV DATA	Здесь хранится информация о цвете и о параметрах AT и TAB для телевизора.
X38	23568	STRMS	Адрес каналов, подключенных к потокам.
2	23606	CHARS	Эта системная переменная указывает на адрес, который лежит на 256 байтов ниже, чем набор символов (знаков), начинающийся с символа "пробел" и кончающийся символом "копирайт". Обычно этот набор находится в ПЗУ, но Вы можете организовать свой набор в ОЗУ и изменить CHARS, чтобы она на него указывала.
1	23608	RASP	Продолжительность предупредительного звукового сигнала.
1	23609	PIP	Продолжительность звукового щелчка при нажатии клавиши.
1	23610	ERR NR	На единицу меньше, чем код сообщения об ошибке. В исходном состоянии - 255.
X1	23611	FLAGS	Различные флаги, необходимые для работы операционной системы.
X1	23612	TV FLAG	Флаги, связанные с работой телевизора.
X2	23613	ERR SP	Адрес параметра на машинном стеке, который используется как адрес возврата после ошибки.
N2	23615	LIST SP	Адрес возврата после автоматического листинга.
N1	23617	MODE	Указывает на тип курсора K,L,C или B.
2	23618	NEWPPC	Номер строки, к которой следует переход.

Дли- на	Адрес	Имя	Содержание
1	23620	NSPPC	Номер оператора в строке, к которому следует переход. Заслав нужное значение в NEWPPC, а затем в NSPPC, Вы можете вызвать переход к нужному оператору в нужной строке.
2	23621	PPC	Номер строки, в который расположен исполняемый в данный момент оператор.
1	23623	SUBPPC	Номер оператора в строке, который в данный момент исполнится 1 23624 BORDCR Цвет бордюра, умноженный на 8. Здесь же содержатся атрибуты нижней части экрана.
2	23625	E PPC	номер текущей строки (в котором находится курсор).
X2	23627	VAR\$	Адрес программных переменных.
N2	23629	DEST	Адрес переменной при исполнении оператора присвоения.
X2	23631	CHANS	Адрес информации о каналах.
X2	23633	CURCHL	Адрес информации, которая в настоящий момент вводится или выводится.
X2	23635	PROS	Адрес, с которого начинается программа, написанная на БЕЙСИKe
X2	23637	NXTLIN	Адрес следующей строки программы.
X2	23639	DATADD	Адрес разделителя после последнего введенного параметра из блока DATA.
X2	23641	E LINE	Адрес вводимой команды.
2	23643	K CUR	Адрес курсора.
X2	23645	CH ADD	Адрес следующего интерпретируемого символа.
2	23647	X PTR	Адрес символа после маркера "?".
X2	23649	WORKSP	Адрес рабочего пространства.
X2	23651	STKBOT	Адрес основания стека калькулятора.
X2	23653	STKEND	Адрес вершины стека калькулятора.
N1	23655	BRES	Содержимое регистра "B" калькулятора.

Дли- на	Адрес	Имя	Содержание
N2	23656	MEM	Адрес области, используемой в качестве области памяти калькулятора. Обычно совпадает с MEMBOT, но это не всегда.
1	23658	FLAGS2	Флаги.
X1	23659	DF SZ	Количество строк в нижней части экрана (в системном окне). 2 23660 S TOP Номер верхней программной строки при автоматическом листинге. 2 23662 OLDPPC Номер строки, к которой выполняется переход по CONTINUE. 1 23644 OSPPC Номер оператора в строке, к которому выполняется переход по CONTINUE.
N1	23665	FLAGH	Вспомогательные флаги.
N2	23666	STRLEN	Длина строковой переменной (строинга) в операторе присвоения.
N2	23668	T ADDR	Адрес следующего параметра в синтаксической таблице.
2	23670	SEED	Здесь хранится величина для функции RND. Исходное значение можно установить оператором RANDOMIZE N.
3	23672	FRAMES	Трехбайтная переменная. (Старший байт - первый). Получает приращение на единицу каждую пятидесятую долю секунды. Может использоваться при организации контроля времени. Так, команда PRINT (256*256*PEEK23672+256*PEEK23673+PEEK23674)/50 покажет сколько времени (в секундах) прошло с того момента, как Вы включили компьютер.
2	23675	UDG	Адрес 1-го символа графики пользователя. Вы можете его изменить.
1	23677	CODRDS	Координата X последней помещенной на экран точки.
1	23678		Координата Y последней помещенной на экран точки.
1	23679	P POZN	Номер колонки позиции, в которой находится принтер.

Дли- на	Адрес	Имя	Содержание
1	23680	PR CC	Младший байт адреса следующей позиции печати по LPRINT (в буфере принтера).
1	681		Не используется.
2	23682	ECHO E	Номер колонки и номер строки конца буфера клавиатуры.
2	23684	DF CC	Адрес в экранной области памяти позиции печати.
2	23686	DFCC	То же, но для нижней части экрана.
X1	23688	S POSN	Номер колонки позиции печати.
X1	23689		Номер строки позиции печати.
X1	23690	SPOSNL	То же, что и S POSN, но для нижней части экрана.
1	23692	SCR CT	На единицу больше, чем необходимое количество передвижек строк на экране до остановки с запросом SCROLL? Если Вы организуете постоянную подачу после PRINT какого-либо числа больше, чем 1 в эту ячейку, то печать будет идти непрерывно, без запроса SCROLL?
1	23693	ATTR	Текущие цветовые атрибуты (для всего экрана).
1	23694	MASK P	Маска локальных атрибутов. Каждый бит этой переменной, если он равен единице, показывает, что соответствующий цветовой атрибут надо брать не из ATTR P, а из того, что уже есть на экране в позиции, для которой выполняется печать.
N1	23695	ATTR T	Временные атрибуты. Их установка справедлива только для выполнения текущего оператора PRINT, INPUT.
N1	23696	MASK T	То же, что и MASK P, но для временных атрибутов.
1	23697	PFLAG	Флаги атрибутов.

Дли- на	Адрес	Имя	Содержание
N30	23698	MEMBOT	Область памяти калькулятора. Используется для хранения чисел, которые неудобно помещать в стек калькулятора.
2	23728		Не используется.
2	23730	RAMTOP	Адрес вершины области, доступной для размещения программы на БЕЙСИКе.
2	23732	P-RAMT	Адрес физической вершины памяти компьютера.

8. Организация памяти компьютера

Для возможности работы операционной системы компьютера в его памяти выделено несколько областей. Эти области отделены друг от друга некоторыми границами, которые могут иметь постоянный (фиксированный) характер, а могут изменяться в зависимости от конкретных требований.

Карта памяти компьютера.

Фиксированные границы обозначены числом, а переменные - именем системной переменной, в которой они хранятся.

0...16383 - ПЗУ.

16384...22527 - экранная область. Изображение хранится здесь в черно-белом виде.

22528...23295 - область атрибутов. Цвета, признаки яркости и мигание.

23296...23551 - область буфера ZX-принтера. При отсутствии такого принтера часто используются для хранения небольших процедур в машинных кодах. Здесь они не повреждаются БЕЙСИКом, а, кроме того, размещение их в нижней половине памяти компьютера повышает их быстродействие.

В компьютерах 128К в этой области хранятся дополнительные системные переменные для поддержания повышенных возможностей этих машин. Там категорически нельзя ничего засылать в эту область памяти.

23552...23733 - область системных переменных.

23734...CHANS-1 - карты памяти микродрайва. При работе с микродрайвом здесь хранится информация, например, об испорченных секторах на ленте и т.п. Если микродрайва нет, то эта область не организуется.

CHAS...PROG-2 - здесь размещается информация о каналах.

PROG...VARS-1 - текст программы, записанной на БЕЙСИКЕе. Отметим, что он хранится несколько в иной форме, чем то, что Вы видите на экране.

VARS...ELINE-2 - область хранения переменных, с которыми оперирует БЕЙСИК-программа.

ELINE...WORKSP-1 - буфер редактора. Сюда помещается строка программы, когда Вы выполняете ее ввод или редактирование.

WORKSP...STKBOT-1 - буфер инструкции INPUT.

STKBOT...STKEND-1 - стек калькулятора. К нему обращается интерпретатор БЕЙСИКа. Калькулятор служит для выполнения арифметических операций калькулятор может вызываться из машинного кода командой ассемблера RST 40. По окончании операции на вершину стека помещается ее результат. **STKEND...** sp - свободное пространство. Сюда развиваются навстречу друг другу стек калькулятора и машинный стек. sp - это не системная переменная. Это указатель машинного стека. Его можно вызвать только из машинного кода. Из БЕЙСИКа он недоступен. **sp...ERR SP** - машинный стек. Нужен процессору Z 80 для выполнения обработки данных, адресов и пр.

ERR SP+1...RAMTOP - стек возвратов. (GO SUB-стек). Здесь хранится информация о номере строки, к которой надо вернуться после выполнения подпрограммы по RETURN.

RAMTOP...UDG - свободная область. Может быть использована для хранения процедур в машинных кодах.

URG...PRAMT - область графики пользователя.

PRAMT - физическая вершина памяти компьютера. Для СПЕКТРУМА 48 - **PRAMT** = 65535, а для СПЕКТРУМА 16 - это 32767.

9. Методы введения русского шрифта

Обеспечение возможности работы компьютера с символами русского алфавита является, наверное, одной из первейших задач, которые возникают перед тем, кто хочет сам писать программы на СПЕКТРУМе.

Существуют два принципиально различных подхода для решения этой задачи. Первый подход состоит в использовании символов графики пользователя, а второй - в создании нового набора символов, размещение его в оперативной памяти и назначении его в качестве действующего с помощью системной переменной **CHARS**.

Использование графики пользователя.

В графическом режиме Вы можете, во-первых, набирать графические символы блочной графики, расположенные на цифровых

клавишах, а также можете воспользоваться символами графики пользователя. Их может быть до 21; они берутся нажатием на клавиши от "А" до "U" в графическом режиме (курсор "G"). Изображение символов графики пользователя должны быть сформированы в специальной области памяти, на которую указывает системная переменная UDG. Адрес, в котором расположено изображение скажем буквы Я, мы будем иметь возможностью нажатием клавиши "А" в графическом режиме печатать на экране букву "Я".

Учитывая, что многие буквы русского и латинского алфавита имеют сходное начертание (например О, А, Х, Н, Р и др.), можно задать прописные буквы, не имеющие аналогов и при этом уложиться в разрешенный 21 символ.

Изображение буквы "Я" формируется из точек экрана (пикселей), например, так:

```
00000000 0
00xxxxx0 0*128+0*64+1*32+1*16+1*8+1*4+1*2+0*1=62
0x0000x0 0*128+1*64+0*32+0*16+0*8+0*4+1*2+0*1=66
0x0000x0 0*128+1*64+0*32+0*16+0*8+0*4+1*2+0*1=66
00xxxxx0 0*128+0*64+1*32+1*16+1*8+1*4+1*2+0*1=62
00x000x0 0*128+0*64+1*32+0*16+0*8+0*4+1*2+0*1=34
0x0000x0 0*128+1*64+0*32+0*16+0*8+0*4+1*2+0*1=66
```

00000000 0 Таким образом, в результате такого синтеза пикселей на буквы "Я", мы можем представить ее последовательностью из 8 байтов, разместив их в соответствующем месте - 0, 62, 66, 66, 62, 34, 66, 0.

Точно так же, можно изобразить и остальные желаемые символы. Для тех, кто не хочет глубоко вдаваться в суть процесса, мы предлагаем небольшую готовую программу, которую Вы можете набрать и выгрузить на ленту. Всякий раз, когда вы хотите составить какую-либо программу, в которой возможна необходимость в русских буквах, загрузите эту программу и стартуйте ее (RUN). После останова программы дайте LIST и продолжайте со строки 100 набирать свою программу.

```
10 FOR Z=1 TO 20
20 READ A$
30 FOR X=0 TO 7
40 READ Y
50 POKE URS A$ + X,Y
60 NEXT X
70 NEXT Z
80 DATA "B",0,126,64,124,66,66,124,0
81 DATA "G",0,126,64,64,64,64,0
82 DATA "D",0,28,36,36,36,36,126,66
```

83 DATA "J", 0,65,73,62,73,73,65,0
 84 DATA "I", 0,66,70,74,82,98,66,0
 85 DATA "L", 0,30,34,34,34,98,0
 86 DATA "P", 0,126,66,66,66,66,66,0
 87 DATA "O", 0,66,66,36,24,16,96,0
 88 DATA "C", 0,68,68,68,68,68,126,2
 89 DATA "H", 0,66,66,66,126,2,2,0
 90 DATA "N", 0,65,73,73,73,73,127,0
 91 DATA "M", 0,65,73,73,73,73,127,1
 92 DATA "E", 0,60,66,30,2,66,60,0
 93 DATA "U", 0,76,82,114,82,82,76,0
 94 DATA "A", 0,62,66,66,62,34,66,0
 95 DATA "S", 24,66,70,74,82,98,66,0
 96 DATA "R", 0,66,64,124,66,66,124,0
 97 DATA "T", 0,192,64,124,66,66,124,0
 98 DATA "F", 0,127,73,73,73,127,8,0
 99 DATA "Q", 0,66,66,114,74,74,114,0

В результате работы этой программы устанавливается следующее соответствие между клавишами в графическом режиме и символами русского алфавита:

"Q" = "Ы" "R" = "Ь" "U" = "Ю" "O" = "У" "A" = "Я"
 "D" = "Д" "G" = "Г" "J" = "Ж" "C" = "Ц" "N" = "Ш"
 "E" = "Э" "T" = "Ъ" "I" = "И" "P" = "П" "S" = "М"
 "F" = "Ф" "H" = "Ч" "L" = "Л" "B" = "Б" "M" = "Щ"

Изменения набора знаков

Набор знаков компьютеров "СПЕКТРУМ" состоит из 96 символов и занимает $96 \times 8 = 768$ байтов. Стандартный набор находится в ПЗУ по адресу 15616. Этот адрес можно определить с помощью системной переменной CHARS.

PRINT PEEK 23606+256*PEEK 23607+256<ENTER>
 15616 О.К.

При этом по адресу 23606 находится 0, а по адресу 23607 - 60. Изменение набора знаков состоит в том, чтобы сформировать с помощью какой-либо специализированной программы или вручную желаемый набор знаков, поместить его в любое свободное место оперативной памяти и изменить содержимое ячеек 23606 и 23607, чтобы оно указывало на адрес места расположения нового набора. Рассмотрим эти процедуры на конкретном примере.

1. Для формирования набора знаков очень удобно использовать графический редактор "ARISTUDIO". Эта программа не только является идеальным графическим редактором, но и позволяет конструировать шрифты. Выгрузим сформированный шрифт на ленту.

2. Допустим, мы хотим разместить новый шрифт, начиная с адреса 30000. Загрузим его с ленты LOAD ""CODE 30000,768.

3. Системная переменная CHARS должна указывать на адрес, находящийся на 256 байтов ниже, т.е. на 30000-256=29744.

4. Определим старший байт двухбайтной переменной CHARS.
 $02 = \text{INT}(29744/256) = 116$.

5. Определим младший байт: $81 = 29744 - 116 * 256 = 48$.

6. Переключение на печать сформированным Вами шрифтом выполняется командой

POKE 23606,48: POKE 23607,116.

Обратное переключение на латинский шрифт - командой
POKE 23606,0: POKE 23607,60.

10. Защита программ

Защита программ состоит в исключении возможности просмотра текста программы и внесения в него изменений лицом, не уполномоченным на это автором программы.

Методов защиты программ существует очень много, и здесь мы рассмотрим только основные принципы и обозначим подходы к этой проблеме.

Можно выделить три основных направления:

1. Исключить возможность остановки (прерывания работы программы).

2. Исключить возможность подачи команды LIST для распечатывания текста программы.

3. Сделать листинг программы нечитаемым.

Исключение возможности остановки программы

Самое первое, что Вы должны сделать - это выполнить программу-автостартующей со строки номер N. Это выполняется при выгрузке программы командой SAVE "имя" LINE N. Теперь программа после загрузки будет сама стартовать с указанной строки. Тем не менее, она может быть остановлена командой BREAK или по сообщению по ошибке.

Отключить клавишу BREAK можно, если в той строке, с которой происходит автостарт, поместить команду
POKE 23613, PEEK 23670 - 5.

Если программа Вами хорошо отлажена, то остановку по ошибке можно исключить. Надо помнить, что чаще всего при взломе программы сознательно вносят ошибку во время исполнения оператора INPUT. Так, если программа просит от пользователя ввести какое-либо число, он сознательно нажимает на клавишу с буквой. Программа останавливается с сообщением VARIABLE NOT

FOUND. Рекомендуем не использовать оператор INPUT, а организовывать опрос клавиатуры на основе функции INKEYS.

Если в строке автостарта поместить нижеприведенные команды, то при попытке сделать BREAK или при появлении ошибки, программа будет сбрасываться:

```
LET ERR = 256*PEEK 23614 + PEEK 23613: POKE ERR,0: POKE  
ERR+1,0
```

Примерно тот же эффект дает помещение в стартовой строке команды POKE 23659,0. Одним из способов остановки автостартующей программы является использование команды MERGE "" вместо LOAD "" при загрузке.

Если Ваша программа имеет размер больше 7К, то защитить ее от этого приема можно, если перед выгрузкой подать следующие прямые команды:

```
LET X = 256*PEEK 23636 + PEEK 23635: POKE X,60: POKE X+1,0.
```

Другой способ защиты от MERGE состоит в том, чтобы перед выгрузкой Вашей программы на ленту, дать ложную информацию о том, что якобы первая строка Вашей программы имеет чрезмерно большую длину. Это делается прямой командой:

```
POKE (PEEK 23635 + 256*PEEK 23636 + 3),255.
```

Отключение команды LIST

Введите первой строкой программы 1 REM. Перед выгрузкой программы на ленту дайте прямую команду:

```
POKE (PEEK 23635 + 256*PEEK 23636),100
```

Теперь программу можно запускать (RUN), но читать нельзя.

Как сделать текст программы нечитаемым

Простейший прием, применяемый во многих программах состоит в том, что устанавливают одинаковым цвета символов (INK) и фона (PAPER).

Например:

```
10 INK 7: PAPER 7
```

В тех же местах, где программа должна сделать вывод на экран в операторе PRINT вставляют в качестве временных правильные цвета:

```
20 PRINT INK 0: "ZX Spektrum"
```

Другой способ состоит в искажении набора символов путем задания "фальшивого" значения системной переменной CHARS. Попробуйте, например:

```
POKE 23606,8: PRINT "ZX Spektrum": POKE 23606,0
```

Если адресовать системную переменную CHARS в те области памяти, где вообще ничего нет, например POKE 23607,200, то все символы будут выглядеть как пробелы и на экране вообще ничего

не будет. Вам же нужно перед всяким оператором PRINT включать истинное значение CHARS и выключать его после PRINT.

Еще один регулярно используемый прием состоит в организации "нулевой" строки, которая реально исполняется программой, но не может быть удалена при редактировании. Чтобы первая строка стала нулевой, дайте команду
POKE 23755,0: POKE 23756,0.

11. Операции с экраном

Типичной задачей для тех, кто начинает создавать собственные программы на "СПЕКТРУМе" является создание красочных изображений в графике высокого разрешения, хранение их в памяти компьютера и вывод на экран.

Мы не будем рассматривать графические операции с операторами DRAW, CIRCLE, PLOT, а также печать символов графики пользователя, а рассмотрим операции с экраном, созданным в графическом редакторе. Наиболее удобным в работе является графический редактор ARTSTUDIO, но можно использовать и какой-либо другой доступный.

Созданное в редакторе изображение Вы можете выгрузить на ленту в виде блока кодов, длина которого 6912 байтов. Если теперь загрузить его командой

```
LOAD "" CODE 16384, 6912
```

то изображение сразу будет загружено на экран, но хранить его там, естественно, нельзя. Можно загрузить изображение в произвольное место и вызывать его оттуда на экран по мере необходимости. Причем адрес для хранения картинки 30000.

```
LOAD "" CODE 30000, 6912
```

Переброску на экран можно организовать в БЕЙСИKe.

```
500 FOR I = 1 TO 6912
```

```
510 POKE 16383+I, PEEK 29999+I
```

```
520 NEXT I
```

```
530 RETURN
```

Всякий раз, когда Вам надо вызвать изображение на экран, Вы можете делать это командой GO SUB 500.

К сожалению, этот метод очень медленно работает, что связано с ограниченными возможностями БЕЙСИКа. Построение экрана происходит более 20 секунд.

Возможность быстрой переброски экрана (менее 1 сек.) представляет программирование в машинном коде. Процессор Z-80 для этой цели имеет мощную команду перемещения блоков LDIR (ее код 237,176). Для ее работы необходимо, чтобы в регистре HL процессора находился адрес начала блока, подлежащего переброске (в нашем

случае - 30000), в регистре BC - длина блока (6912), а в регистре DE - адрес места назначения (16304). Тогда программа в машинном коде будет выглядеть так:

Машинный код	Ассемблер	Комментарий
33 48 117	LD HL, 48 117	Загрузить в регистр HL адрес 30000. Обратить внимание на то, что $48 + 117 * 256 = 30000$
1 0 27	LD BC, 0 27	Загрузить в регистр BC 6912.
17 0 64	LD DE, 0 64	Загрузить в регистр DE 16384.
237 176	LDIR	Выполнить переброску блока в новое место.
201	RET	Возврат в вызывающую программу.

Таким образом, программа в машинном коде выглядит так:
33, 48, 117, 1, 0, 27, 17, 0, 64, 237, 176, 201.

Поместим ее в произвольное место оперативной памяти, например, начиная с адреса 63000 следующей программой, написанной на БЕЙСИКЕ.

```
10 FOR I=1 TO 12
20 READ A
30 POKE 59999+I,A
40 NEXT I
```

```
50 DATA 33,48,117,1,0,27,17,0,64,237,176,201
```

Теперь всякий раз, когда Вам нужно поместить картинку на экран, Вам надо стартовать эту процедуру, что выполняется командой RANDOMIZE USR 65000.

Если же у Вас в памяти компьютера хранится не одно, а несколько графических изображений, то их можно также перебрасывать на экран этой процедурой, надо только изменить адрес, поступивший в регистр процессора HL, что можно сделать командой POKE по адресу 65001 и по адресу 65002.

Предположим, что у Вас есть вторая картинка, помещенная начиная с адреса 37000. Тогда $INT(37000/256) = 144$, а $37000 - 256 * 144 = 136$. Выполним POKE 65001,136 и POKE 65002,144. Команда RANDOMIZE USR 65000 выполнит переброску второй картинки и т.д.

Очевидно, что поскольку каждый экран занимает 6912 байтов, то одновременно в памяти нельзя хранить более 6 изображений, что во многих случаях бывает недостаточно. Если учесть, что каждое изображение имеет длительные последовательности нулей, то име-

ется возможность компрессировать каждое изображение. Это позволяет во многих случаях вдвое сократить объем памяти, занимаемой графикой, а иногда даже больше.

Мы предлагаем Вашему вниманию две процедуры в машинном коде. Первая процедура называется FN s и служит для компрессирования изображения с экрана и размещения его в заданном пользователем адресе. Она имеет два параметра h и l, которые задают адрес для хранения экрана. h - старший байт адреса, а l - младший байт.

Сама процедура FNs расположена начиная с адреса 56600. Длина - 60 байтов. Вторая процедура FNt служит для декомпрессирования изображения и переброске его на экран. Она имеет два тех же параметра h,l. Адрес размещения этой процедуры - 56500. Длина - 40 байтов.

Чтобы создать процедуру FNs, Вам надо набрать и запустить следующую БЕЙСИК-программу. После того, как она отработает, выгрузите созданную процедуру на ленту командой SAVE"FNs" CODE 56600,60 и в дальнейшем можете использовать в своих программах. Точно так же исполняется копия процедуры FNt: SAVE "FNt" CODE 56500,40.

Процедура FNs

```
8300 LET b=56600: LET l=60: LET z=0: RESTORE 8310
8301 FOR i=0 TO l-l: READ a
8302 POKE (b+i), a: LET z=z+a
8303 NEXT i
8304 LET x=INT(((z/l)-INT(z/l))*l)
8305 READ a: IF a<>z THEN PRINT "???": STOP
8310 DATA 42, 11, 92, 1, 4
8311 DATA 0, 9, 86, 14, 8
8312 DATA 9, 94, 237, 83, 82
8313 DATA 221, 33, 0, 64, 6
8314 DATA 1, 126, 44, 32, 8
8315 DATA 36, 245, 124, 254, 91
8316 DATA 40, 16, 241, 78, 185
8317 DATA 32, 4, 4, 32, 238
8318 DATA 5, 18, 19, 120, 18
8319 DATA 19, 24, 227, 241, 18
8320 DATA 19, 120, 18, 237, 83
8321 DATA 22, 221, 201, 0,0
8322 DATA 56, 0, 0, 0, 0
```

После того, как процедура выполнит компрессирование экрана и перебросит его в место, отведенное для длительного хранения, Вы можете узнать адрес, в который можно помещать следующий экран.

PRINT PEEK 23296 + 256* PEEK 23297. Но помните, что делать это можно только немедленно после окончания компрессирования и переброски блока.

Процедура Fnt

```
8350 LET b=56500: LET l=35: LET z=0: RESTORE 8360
8351 FOR i=0 TO l-1 : READ a
8352 POKE (b+i),a: LET z=z+a
8353 NEXT i
8454 LET z=INT(((z/l)-INT(z/l))*l)
8355 READ a: IF a<>z THEN PRINT "???": STOP
8360 DATA 42, 11, 92, 1, 4
8361 DATA 0, 9, 86, 14, 8
8362 DATA 9, 94, 33, 0, 64
8363 DATA 26, 245, 19, 26, 19
8364 DATA 71, 241, 119, 35, 16
8365 DATA 252, 124, 254, 91, 32
8366 DATA 240, 201, 0, 0, 0
8367 DATA 28, 0, 0, 0, 0
```

Пример применения процедур компрессии и декомпрессии

Предположим, что Вам надо компрессировать и отправить на хранение три экрана scr1, scr2, scr3, причем первый из них идет в адрес 30000 (48, 117).

Тогда основная программа может выглядеть так:

```
10 DEF FN s(h,l) = USR 56600
20 DEF FN t(h,l) = USR 56500
100 LET a=23296: LET b=23297
110 LET h1=117: LET l1=48
120 LET ad=l1+(h1*256)
130 PRINT ad: LOAD "scr"CODE 16384,6912
140 RANDOMIZE FNs(h1,l1)
150 LET l2=PEEK a: LET h2=PEEK b
155 LET ad=12+h2*256
160 PRINT ad: LOAD "scr2"CODE 16384,6912
170 RANDOMIZE FNs(h2,l2)
180 LET l3=PEEK a: LET h3=PEEK b
195 LET ad=13+256*h3
190 PRINT ad: LOAD "scr3"CODE 16384,6912
200 RANDOMIZE FNs(h3,l3)
210 PAUSE 50
300 RANDOMIZE FNs(h1,l1)
310 PAUSE 50
```

320 RANDOMIZE FNs(h2,l2)
330 PAUSE 50
340 RANDOMIZE FNs(h3,l3)
350 PAUSE 50: GO TO 300

12. Что нужно знать для подключения принтера

По-видимому, принтер - это наиболее желаемое периферийное устройство, перспективу возможности приобретения и подключения которого к СПЕКТРУМу имеет в виду большинство пользователей. Сразу заметим, что фактически к этому компьютеру можно подключить практически любой принтер, но при этом могут возникать те или иные трудности.

Прежде всего надо определиться для чего он Вам нужен. Если основное направление Вашей работы - программирование, то Вам может подойти небольшой узкопечатный ZX-принтер, который подключается к краевому разьему компьютера без каких-либо интерфейсов и не требует никакого программного обеспечения для работы (все уже содержится в ПЗУ компьютера). Такой принтер имеет ширину бумажной ленты 10....13 см и выводит 32 символа в строке. Он также может выполнить графическую копию экрана. Качество печати далеко не идеальное, но такие устройства сравнительно дешевы, просты и удобны в работе и, если их применять в основном для распечатки текстов программ или результатов расчетов, то вполне соответствуют поставленным задачам.

Наиболее распространены 4 модели таких принтеров. Это ZX printer, ALPHACOM 32, TIMEX-2040, SEIKOSHA GP 50S.

ZX printer - стандартное устройство. Печать выполняется на алюминизированной рулонной бумаге. Качество печати наилучшее из этих 4-х моделей.

ALPHACOM 32 и TIMEX 2040 - практически идентичны. Они работают со специальной термочувствительной бумагой и очень широко распространены.

SEIKOSHA GP 50S - выпускался в Японии. Он имеет наиболее эстетичный дизайн и наилучшее качество печати. Бумага - обычная рулонная шириной 13 см. Применяется узкая красящая лента.

Если по роду Вашей деятельности необходимо получение качественных документов на листах стандартного формата, то такой принтер Вам не подойдет. Полноразмерные принтеры выпускаются двух типов - это точно-матричные принтеры и принтеры типа "ромашка". Разница в конструкции печатающего элемента. В первом случае это набор иглол, оставляющий через красящую ленту мозаичный отпечаток на бумаге. Из точек складывается изображе-

ние символа. В принтерах второго типа печатающие литеры закреплены на вращающемся прорезном диске ("ромашка"). Этот диск может быть сменным для замены набора литер.

Разница в конструкции вызывает и различия в эксплуатации. Матричные принтеры значительно быстрее работают, могут гибко (программно) перестраиваться с одного шрифта на другой, но по качеству печати только самые лучшие экземпляры могут приближаться к принтерам типа "ромашка".

В основном принтеры "ромашка" применяются в тех случаях, когда имеется необходимость в большом количестве деловой переписки, и качество исполненного документа свидетельствует об уровне представительства. Надо также отметить, что выдача графических изображений на принтерах этого типа исключена.

Наиболее гибким и многофункциональным устройством является точечно-матричный принтер. Далее мы остановимся на этих устройствах и отметим какие вопросы Вы должны для себя решить, прежде чем приобрести и подключать такой принтер.

Тип используемого интерфейса.

Этот пункт является весьма важным, т.к. от него зависит способ подключения к компьютеру. Существуют два основных метода обмена информацией между компьютерами и периферийными устройствами - параллельный и последовательный. При параллельном подключении все биты, составляющие байт посылки, передаются одновременно по параллельным шинам данных. При последовательном байт передается бит за битом по очереди. Исторически за интерфейсами, реализующими параллельный метод передачи закрепились названия "Центроникс", а за последовательными - RS232.

Большинство современных компьютеров уже имеют встроенный интерфейсный порт того или иного типа, но Спектрум-48 его не имеет. Поэтому, приобретая принтер, надо подумать и об интерфейсе. Наибольшее распространение получили следующие модели интерфейсов: 1. ИНТЕРФЕЙС-1. Мы о нем уже упоминали. Он имеет последовательный порт RS232, к которому может подключаться принтер, имеющий такой же интерфейс. Никакой программной поддержки этот интерфейс не требует, она находится в его ПЗУ.

ZX-LPRINT III. Этот интерфейс может настраиваться и поддерживать аппаратуру как с интерфейсом "Центроникс", так и с RS232.

TASPRINT - интерфейс типа "Центроникс". Требуется специальной программы (драйвера) для своей работы.

KEMPSTON-E - интерфейс типа "Центроникс", также требует драйвера.

Компьютеры "Спектр+128" и "Спектр+2" имеют уже встроенный порт RS232. Зато они не могут работать с ZX-принтерами, т.к. область буфера принтера там занята дополнительными системными переменными, хотя существуют программные пути устранения этого недостатка.

Обратим Ваше внимание, однако на то, что использовать этот порт непросто. Дело в том, что он поддерживается только в режиме 128К, а наибольшая часть прикладных программ (текстовые редакторы, графические редакторы и пр.), с которыми хотелось бы использовать принтер, работают в режиме 48К.

Наличие дополнительного набора знаков.

Далеко не все импортные принтеры имеют встроенный русский набор знаков, что необходимо учитывать. Наилучшие модели имеют возможность загрузки шрифта от компьютера.

Применяемая бумага.

Некоторые принтеры используют термобумагу, что создает весьма ощутимые неудобства в связи с отсутствием ее в широкой продаже. Кроме того, качество исполнения документа в этом случае ниже.

Красящая лента.

Многие принтеры работают с лентой, находящейся в специальных неразборных кассетах. Это дает удобство при смене ленты, но может поставить Вас в тупик, если доступа к таким кассетам Вы не имеете и применение обычной ленты на катушках конструктивно невозможно.

Скорость печати.

Если Вы предполагаете исполнение многочисленных копий документов, этот параметр может быть важным. Обычно точечно-матричный принтер имеет скорость печати 80...100 знаков в секунду. Даже если этот параметр Вам безразличен, имейте в виду, что чем он выше, тем совершеннее принтер. Наилучшие модели имеют скорость выше 180 знаков в секунду.

Количество игло́к в печатающей головке.

Как правило, принтер среднего качества имеет 9 игло́к в печатающей головке. Если у него только 7 игло́к, то некоторые буквы, например, у, j, p могут лишиться своих нижних элементов. Наиболее совершенные модели имеют до 24 игло́к.

Наличие режима NLQ.

Режим NLQ - NORMAL LETTER QUALITY обеспечивают печать символами, приближающимися по качеству к стандартной

пишущей машинке. Такой режим имеется далеко не у всех принтеров, и печать в этом режиме выполняется значительно медленнее, зато он позволяет получить представительные документы.

Коды управления принтером.

Принтер может не только распечатывать текст, который поступает в него из компьютера, но выполняет дополнительно и целый ряд специфических действий (подчеркивать текст, переключаться с одного набора знаков на другой, изменять размер шрифта, выполнять печать двойным ударом и еще многие другие действия). Эти переключения делаются с помощью управляющих кодов, которые в нужных местах вставлены в текст. Как правило, управляющие коды состоят из последовательности кодов, начинающихся с символа 27 (ESCAPE), поэтому часто коды управления принтером называют ESCAPE-кодами. В зависимости от того, какому коду какое действие принтера соответствует, различают несколько стандартов этих кодов. Наиболее широко распространена система кодов фирмы EPSON, которая применяется на принтерах этой фирмы. В зависимости от того, поддерживает ли принтер этот стандарт кодов, он может быть или не быть EPSON-совместимым. Желательно, чтобы такая совместимость была, тогда Вы сможете распечатывать подготовленные документы не только на своем принтере, а также упростите задачу настройки фирменных программ на работу с принтером, т.к. они, как правило, готовятся в расчете на EPSON-совместимые принтеры.

DIP-переключатели.

Эти переключатели служат для задания определенных режимов работы принтера. Многие их функции дублируются и могут быть изменены кодами управления принтером, но те условия, которые заданы этими переключателями, являются исходными в момент включения принтера. Чем их больше, тем больше возможностей имеет принтер. Нормально их должно быть порядка 24 шт.

Буфер принтера.

Принтер должен иметь определенный объем свободной оперативной памяти, где он размещает текст, поступающий от компьютера и выполняет с ним преобразования перед выводом на печать. Именно наличие буфера позволяет принтеру, например, печатать при обратном проходе головки (справа налево), что вдвое повышает его быстродействие. При наличии достаточно емкого буфера компьютер очень быстро может перебросить в него выводимый текст, и Вы сможете, например, готовить к печати очередной блок, пока принтер ведет печать. Если же емкость буфера мала, компьютеру приходится все время подзагружать буфер, и Вы не можете исполь-

зовать его для других целей. В принципе, для большинства практических задач емкость порядка 0,5К достаточна, но лучше, если она больше 2К, а совершенные модели имеют буфер больше 4К.

Выбор текстового редактора.

Текстовые редакторы предоставляют наиболее удобные условия для работы с принтером, поэтому, предлагая подключение принтера, надо иметь в виду и необходимость наличия редактора. Наиболее широко распространены различные версии текстового редактора TASWORD. Эта программа отличается простотой в работе, предоставляет возможность изображения на экране как 32 символов в строке, так и 64-х. Имеет несколько кодов управления принтером, которые вставляются в текст в виде символов блочной графики. Достаточно просто переделывается на работу с русским или другим национальным шрифтом.

Наиболее совершенным и гибким в работе является редактор THE LAST WORD 2. Среди программ для бытовых компьютеров любых моделей трудно найти сравнимый по возможностям. Он может воспроизводить текст на экране в режимах 40, 48, 60, и 80 знаков в строке. Имеет очень удобное меню настройки условий печати, обширный набор операторов управления принтером, который к тому же может гибко перестраиваться без выхода из программы. Имеет встроенный таймер и калькулятор, средства поиска информации и мощные средства для оформления текста и составления комплексных документов.

Строго говоря, это не просто программа, а пакет программ, т.к. к нему прилагаются дополнительные файлы, позволяющие перенастроить редактор на работу в дисковых системах "TR DOS", "OPUS", "GORDON" и др., а также дополнительные программные средства для обеспечения возможности работы с текстами, выполненными в других редакторах и базах данных.

Правда, надо отметить, что программа значительно сложнее в освоении по сравнению с редактором "TASWORD" в силу своих гораздо более глубоких возможностей и имеет весьма объемную инструкцию по работе. Кроме того, переделка шрифта на национальный представляет серьезную трудность из-за наличия нестандартных генераторов символов.