

стройка частоты гетеродина (АПЧГ) и регулировка усиления (АРУ), подключение головных телефонов и магнитофона для записи звукового сопровождения, выключение громкоговорителей.

Благодаря применению импульсного источника питания, новых радиокomпонентов, схемных и конструктивных решений масса телевизора не превышает 25 кг, а потребляемая мощность более чем в 2 раза ниже, чем у телевизоров типа УСТ-61. Простота сборки и разборки телевизора, легкий доступ к элементам обеспечивают его высокую ремонтопригодность.

Структурная схема телевизора приведена на рис. 1. Он содержит блок приемника и разверток АЗ с селекторами каналов метровых СК-М-24-2 (АЗ.1) и дециметровых СК-Д-24 (АЗ.3) волн и submodule радиоканала СМРК-2-1 (АЗ.2); блок управления А2 с устройством сенсорного выбора программ СВП-4-10 (А1), импульсный источник питания А4 и кинескоп с закрепленными на нем отклоняющей системой А5 и платой панели кинескопа с выходным видеоусилителем АЗ.4.

Телевизионные радиосигналы усиливаются селекторами и преобразуются в сигналы ПЧ. Работой селекторов управляет устройство сенсорного выбора программ СВП-4-10 (см. статью Г. Мазуркевича и Л. Шепотковского «Горизонт Ц-257». Система управления в «Радио», 1984, № 12, с. 27—29). При нажатии любой из шести кнопок устройства на коммутирующие диоды селекторов каналов воздействует напряжение, включающее соответствующий поддиапазон, а на варикапы — предварительно установленное напряжение настройки, соответствующее выбранной программе.

Сигналы ПЧ изображения (38 МГц) и звука (31,5 МГц) с выхода селектора каналов диапазона МВ поступают в submodule радиоканала СМРК-2-1, где они усиливаются и детектируются. В состав submodule входит микросхема К174УР5, содержащая усилитель ПЧ изображения (УПЧИ), синхронный видеодетектор и устройства АПЧГ и АРУ. Для формирования амплитудно-частотной характеристики (АЧХ) тракта ПЧ изображения применен фильтр на ПАВах.

Устройство АПЧГ телевизора работает таким образом, что при отклонении ПЧ изображения от номинального значения на выходе появляется напряжение «ошибки» соответствующего знака, алгебраически складывающееся с напряжением настройки. Результирующее напряжение воздействует на варикапы селектора каналов и устраняет возникшую расстройку.

С выхода синхронного видеодетектора сигнал поступает на устройство АРУ, управляющее работой УПЧИ и селекторов каналов, и на усилитель ПЧ звука УПЧЗ-1м, выполненный на микросхеме К174УР4.

В УПЧЗ-1м выделяется сигнал разностной частоты — второй ПЧ звука (6,5 МГц), который затем усиливается, ограничивается и детектируется. Здесь же предварительно усиливаются и колебания ЗЧ, которые далее поступают на усилитель ЗЧ, собранный на микросхеме К174УН7.

Через эмиттерный повторитель submodule радиоканала видеосигнал проходит на микросхему К174ХА11, управляющую работой узлов строчной и кадровой разверток. Она обеспечивает выделение строчных и кадровых синхронимпульсов из полного телевизионного сигнала, автоматическую подстройку частоты и фазы строчной развертки, формирует стробирующие импульсы для привязки уровня черного и импульсы гашения. В ее состав входит также задающий генератор строчной развертки.

Импульсы строчной и кадровой частот поступают соответственно на каскады строчной и кадровой разверток, служащие для формирования пилообразного тока в катушках отклоняющей системы. Кроме задающего генератора, в узел строчной развертки входят предварительный и выходной каскады. Узел кадровой развертки состоит из задающего генератора и усилителя мощности на микросхеме К174УН7.

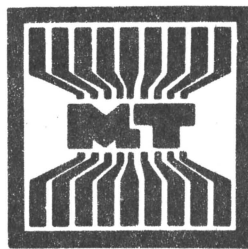
Высокое напряжение для питания анода кинескопа получается выпрямлением импульсов обратного хода в выходном строчном трансформаторе ТДКС-9.

Питание телевизора обеспечивается импульсным источником. В нем выпрямленное сетевое напряжение преобразуется в импульсное частотой около 25 кГц, которое затем трансформируется и выпрямляется. Блок питания включает в себя помехозащитный фильтр, выпрямитель сетевого напряжения, преобразователь, состоящий из блокинг-генератора и устройства стабилизации и защиты от перегрузок, а также выпрямители вторичных напряжений, необходимых для питания узлов телевизора.

(Продолжение следует)

Е. ГРИГОРЬЕВ,
В. ЛЕВИН,
Б. СТРЕЛЕЦ

г. Симферополь



Программирование на Бейсике

В предыдущей статье* мы познакомились с описанием интерпретатора языка Бейсик и разобрали ряд простейших программ. Опытный программист из простого перечисления набора операторов и функций языка может сделать вывод о его возможностях и представить себе круг задач, которые могут быть решены с его помощью. В этом ему помогает собственный опыт разработки программ на других языках и известные алгоритмы решения стандартных задач, неоднократно описанные в литературе по программированию.

Начинающему программисту для овладения техникой написания программ целесообразно разбирать и анализировать программы, написанные другими, подобно тому, как шахматисты набираются опыта, разбирая и анализируя партии, сыгранные мастерами. Аналогично поступим и мы: знакомство с приемами составления программ на Бейсике построим на разборе конкретных примеров. Этот путь хотя и не дает возможности овладеть программированием во всех тонкостях, но, как показала практика, позволяет достаточно быстро освоить ряд основных приемов. Рассматриваемые далее небольшие программы иллюстрируют некоторые особенности интерпретатора Бейсика для Микро-80.

* Бейсик для «Микро-80». — Радио, 1984, № 1—3.

ПРОГРАММЫ НА БЕЙСИКЕ

Одно из основных достоинств языка Бейсик — возможность разработки программ в диалоговом режиме. Вы, например, можете начать отладку программы, набрав всего одну строку. Убедившись в том, что эта строка «работает» так, как вы и предполагали, можно продолжить набор программы; если нет, то после анализа полученного результата можно быстро исправить обнаруженные ошибки. Кроме того, непосредственный режим работы интерпретатора предоставляет возможность проведения экспериментов по выявлению особенностей применения того или иного оператора или встроенной функции, что также способствует быстрому освоению языка.

В дальнейшем вы еще успеете оценить по достоинству преимущества, которые дает диалоговый режим при разработке программ, а пока попробуем сами составить программу, реализующую диалог с оператором. Начнем с постановки задачи. Допустим, необходимо разработать программу для проверки знаний арифметики у учащихся начальных классов. Сразу оговоримся, что описываемая ниже программа нужна нам прежде всего для ознакомления с возможностями языка Бейсик. Реальные программы проверки знаний требуют более тщательной проработки с привлечением специалистов по обучению и психологии.

Прежде всего разработаем «сценарий» работы программы. Мы умышленно пользуемся термином «сценарий», чтобы подчеркнуть диалоговый характер нашей будущей программы. Одним из возможных вариантов может быть такой. На экране появляется вопрос: КАК ТЕБЯ ЗОВУТ?

Ученик набирает на клавиатуре свое имя (например, АНТОН), и на экране тотчас возникает текст:

АНТОН, Я ХОЧУ ПРОВЕРИТЬ ТВОЕ ЗНАНИЕ АРИФМЕТИКИ. Я БУДУ ПРЕДЛАГАТЬ ПРИМЕРЫ, А ТЫ ПОСТАРАЙСЯ ИХ РЕШИТЬ.

ЕСЛИ ГОТОВ, НАЖМИ НА ЛЮБУЮ КЛАВИШУ.

После очистки экрана в его верхней части появляется следующая справочная информация:

ПОМНИ!
ЕСЛИ ПРИ ОТВЕТЕ ОШИБЕШЬСЯ И СЛУЧАЙНО НАЖМЕШЬ НЕ ТУ КЛАВИШУ, ТО НАЖМИ КЛАВИШУ «←», А ЗАТЕМ НУЖНУЮ. ПОСЛЕ НАБОРА ОТВЕТА НАЖМИ КЛАВИШУ «ВК».

Этот текст остается на экране до конца работы программы, а в центре экрана появляется вопрос:

АНТОН, СКОЛЬКО БУДЕТ 5 ПРИБАВИТЬ 2==?

При верном ответе на экран выводится:

МОЛОДЕЦ, АНТОН!
РЕШИ ЕЩЕ ПРИМЕР.
ЕСЛИ ГОТОВ, НАЖМИ ЛЮБУЮ КЛАВИШУ.

Неверный ответ приводит к появлению на экране сообщения:
НЕПРАВИЛЬНО!
ТЕБЕ, АНТОН, НАДО ПОДУМАТЬ.

Таблица 1

```

10 REM *****
20 REM * ПРОГРАММА ПРОВЕРКИ ЗНАНИЙ АРИФМЕТИКИ *
30 REM * У УЧЕНИКОВ МЛАДШИХ КЛАССОВ *
40 REM *****
50 CLS:PRINT:PRINT
60 INPUT "КАК ТЕБЯ ЗОВУТ ";IS:PRINT:PRINT
70 PRINT IS: ". Я ХОЧУ ПРОВЕРИТЬ ТВОЕ ЗНАНИЕ АРИФМЕТИКИ."
80 PRINT "Я БУДУ ПРЕДЛАГАТЬ ПРИМЕРЫ, А ТЫ ПОСТАРАЙСЯ ИХ РЕШИТЬ."
90 PRINT " ЕСЛИ ГОТОВ, НАЖМИ НА ЛЮБУЮ КЛАВИШУ."
100 A1=USR(-2045)
110 X=1:W=0
120 CLS:PRINT:PRINT
130 PRINT TAB(15):"ПОМНИ !"
140 PRINT "ЕСЛИ ПРИ ОТВЕТЕ ОШИБЕШЬСЯ И СЛУЧАЙНО НАЖМЕШЬ"
150 PRINT "НЕ ТУ КЛАВИШУ, ТО НАЖМИ КЛАВИШУ '←', А ЗАТЕМ НУЖНУЮ."
170 PRINT "ПОСЛЕ НАБОРА ОТВЕТА НАЖМИ КЛАВИШУ 'ВК':":PRINT
180 R=ABS(INT(RND(1)*20-10))
190 IF R=0 OR R=1 THEN 180
200 G=ABS(INT(RND(1)*20-10))
210 IF G=0 OR G=1 THEN 200
220 KL=INT(RND(1)*5)
230 IF KL=0 THEN 220
240 ON KL GOSUB 510,550,600,630
250 PRINT:PRINT IS;" СКОЛЬКО БУДЕТ ";P;U$;R;" = ";
260 INPUT D
270 IF D=Q THEN 370
280 PRINT:PRINT " НЕПРАВИЛЬНО !"
290 PRINT "ТЕБЕ, ";IS: ". НАДО ПОДУМАТЬ.":PRINT
300 PRINT IS;" СКОЛЬКО БУДЕТ ";P;U$;R;" = ";
310 INPUT D:PRINT
320 IF D=Q THEN 370
330 PRINT "ПЛОХО, ";IS: "! ТЫ НЕ СМОГ РЕШИТЬ ЭТОТ ПРИМЕР."
340 PRINT " ВОТ ПРАВИЛЬНОЕ РЕШЕНИЕ:"
350 PRINT TAB(8):P;U$;R;" = ";Q:PRINT
360 GOTO 390
370 W=W+1
380 PRINT "МОЛОДЕЦ ";IS: "! ПРАВИЛЬНО.":PRINT
390 IF X=10 THEN 450
400 X=X+1
410 PRINT "РЕШИ ЕЩЕ ПРИМЕР."
420 PRINT "ЕСЛИ ГОТОВ, ТО НАЖМИ ЛЮБУЮ КЛАВИШУ"
430 A1=USR(-2045)
440 GOTO 120
450 PRINT:PRINT
460 IF W=10 THEN PRINT"МОЛОДЕЦ, ";IS: ". ТЫ ЗНАЕШЬ АРИФМЕТИКУ НА 5!"
470 IF W=8 OR W=9 THEN PRINT"МОЛОДЕЦ, ";IS: ". ТЫ ЗНАЕШЬ АРИФМЕТИКУ НА 4!"
480 IF W=6 OR W=7 THEN PRINT IS: ". ТЫ ЗНАЕШЬ АРИФМЕТИКУ НА 3!"
490 IF W<6 THEN PRINT IS: ". ТЫ ПЛОХО ЗНАЕШЬ АРИФМЕТИКУ!"
500 STOP
510 U$=" ПРИБАВИТЬ "
520 G=G*3:R=R*3
530 P=G:Q=G+R
540 RETURN
550 U$=" ОТНЯТЬ "
560 IF G=R THEN G=G+3
570 G=G*3:R=R*3
580 IF G<R THEN Q=G:G=R:R=Q
590 P=G:Q=G-R:RETURN
600 U$=" УМНОЖИТЬ НА "
610 P=G:Q=G*R
620 RETURN
630 U$=" РАЗДЕЛИТЬ НА "
640 P=G*R:Q=G
650 RETURN

```

Если за две попытки учащийся так и не смог решить пример, то на экране появляется текст:

ПЛОХО, АНТОН! ТЫ НЕ СМОГ РЕШИТЬ ЭТОТ ПРИМЕР.
ВОТ ПРАВИЛЬНОЕ РЕШЕНИЕ:
5 ПРИБАВИТЬ 2=7
РЕШИ ЕЩЕ ПРИМЕР.
ЕСЛИ ГОТОВ, НАЖМИ ЛЮБУЮ КЛАВИШУ.

Ученик должен решить десять примеров. В зависимости от результатов в конце работы на экране высвечивается одно из следующих сообщений:

МОЛОДЕЦ, АНТОН, ТЫ ЗНАЕШЬ АРИФМЕТИКУ НА 5!
МОЛОДЕЦ, АНТОН, ТЫ ЗНАЕШЬ АРИФМЕТИКУ НА 4!
АНТОН, ТЫ ЗНАЕШЬ АРИФМЕТИКУ НА 3!
АНТОН, ТЫ ПЛОХО ЗНАЕШЬ АРИФМЕТИКУ!

Полный текст программы, выполняющей описанные действия, приведен в табл. 1. Рассмотрим ее особенности. Прежде всего обратим внимание на то, каким образом формируются текстовые сообщения, появляющиеся на экране дисплея. В соответствии с нашим сценарием возникает необходимость вывода на экран сообщений, в которых встречается имя отвечающего на вопросы (символьная переменная И_А)*. Для этой цели используется разделитель «;» (точка с запятой).

В строках 100 и 430 происходит вызов стандартной подпрограммы монитора «ввод кода символа с клавиатуры», адрес которой равен F803H в шестнадцатеричной системе счисления. Это вызвано необходимостью приостановки выполнения программы до тех пор, пока не будет нажата любая клавиша. Значение, возвращаемое подпрограммой, в данном случае не используется. В строках программы 180 и 200 формируются случайные числа, предназначенные для генерации операндов очередной задачи. Вид действия (сложение, вычитание, умножение или деление) определяется следующим образом. В строке 220 переменной К1 присваивается значение 1, 2, 3 или 4. В строке 240 в зависимости от значения этой переменной происходит вызов соответствующей подпрограммы. В этих подпрограммах формируются операнды предлагаемого примера. Символьной переменной У0 присваивается

определенное значение в зависимости от вида арифметической операции.

Переменная Х хранит значение, равное числу заданных примеров, а значение переменной W равно числу верных ответов.

При разработке программ и при работе с ЭВМ часто возникает необходимость перевода чисел из одной системы счисления в другую. Пользуясь известными правилами, это можно сделать и вручную. В табл. 2 приведен пример программы, позволяющей возложить эту трудоемкую работу на

65--70. В подпрограмме преобразования проводится проверка на корректность набранного числа (оно должно содержать только допустимые символы). При обнаружении ошибки ввода результат преобразования равен нулю.

В подпрограмме перевода из десятичной системы счисления в шестнадцатеричную вычисляется код очередной цифры и все число «накапливается» в символьной переменной А0. Здесь каких-либо специальных проверок не предусмотрено, так как сам интерпретатор осуществляет проверку корректности ввода числовых данных.

Таблица 2

```

10 REM *****
20 REM * ПРОГРАММА ПЕРЕВОДА ЧИСЕЛ ИЗ МЕСТ-
30 REM * НАДЦАТИРИЧНОЙ СИСТЕМЫ СЧИСЛЕНИЯ В *
40 REM * ДЕСЯТИЧНУЮ И НАОБОРОТ. *
50 REM *****
100 INPUT "ВВЕДИТЕ (Д)ЕС. ИЛИ (М)ЕСТ. ";S$
110 IF S$="M" OR S$="D" THEN 130
120 GOTO 100
130 IF S$="D" THEN 160
140 INPUT "ШЕСТНАДЦАТИРИЧНОЕ ЧИСЛО ";N$
150 N=0:GOSUB 1000:GOTO 100
160 INPUT "ДЕСЯТИЧНОЕ ЧИСЛО ";N
170 GOSUB 1100:GOTO 100
1000 REM ПЕРЕВОД МЕСТ. -> ДЕС.
1010 FOR I=1 TO LEN(N$)
1020 D=ASC(MID$(N$,I,1))-48
1030 IF D<10 THEN 1050
1040 D=D-7
1050 IF D<0 OR D>=16 THEN N=0:GOTO 1080
1060 N=N*16+D
1070 NEXT I
1080 PRINT "ДЕСЯТИЧНОЕ ЧИСЛО = ";N
1090 RETURN
1100 REM ПЕРЕВОД ДЕС. -> МЕСТ.
1110 A$=""
1120 L=INT(N/16)
1130 M=N-16*L
1140 IF M<10 THEN 1160
1150 M=M*7
1160 N=L:A$=CHR$(M+48)+A$
1170 IF N>=1 THEN 1120
1180 PRINT "ШЕСТНАДЦАТИРИЧНОЕ ЧИСЛО = ";A$
1190 RETURN


```

микро-ЭВМ. Программа переводит числа из десятичной системы счисления в шестнадцатеричную и наоборот, широко используя встроенные функции Бейсика для работы с символьными данными.

Начинается программа с запроса ввода с клавиатуры одной из букв «Д» или «Ш». При ошибочном вводе запрос повторяется. Далее, в зависимости от выбранного режима, оператор должен ввести десятичное или шестнадцатеричное целое число. При переводе чисел из шестнадцатеричной системы счисления в десятичную в программе используется то обстоятельство, что коды символов, отображающих шестнадцатеричные числа (0--9, A--F), равны соответственно 48--57,

В тексте описываемой программы процедуры преобразования оформлены в виде подпрограмм. Это позволяет использовать их в ваших программах на Бейсике, в которых требуется ввод и отображение данных в шестнадцатеричной системе счисления. К таким программам относятся МОНИТОР, АССЕМБЛЕР, ДИЗАССЕМБЛЕР и др. О назначении первых двух уже говорилось в предыдущих статьях. ДИЗАССЕМБЛЕР позволяет перевести программу, написанную в машинных кодах, в мнемонику ассемблера и может быть реализован на Бейсике.

Описанный в журнале интерпретатор имеет средства, позволяющие программам на Бейсике работать с подпрограммами, написанными на ассем-

* Далее по тексту вместо символа  использован символ 0, а в таблицах символ \$.

лере или в машинных кодах. Для вызова таких подпрограмм служит встроенная функция $USR(X)$. Рассмотрим пример, иллюстрирующий ее использование.

Алфавитно-цифровая клавиатура является в настоящее время самым распространенным устройством ввода данных в ЭВМ. Однако, чтобы быстро и безошибочно вводить данные с ее помощью, нужен немалый опыт. Кроме того, если использование такой клавиатуры для ввода символической информации вполне естественно, то ввод графических данных или указывание на какой-либо объект на экране с ее помощью весьма затруднены. В современных персональных микро-ЭВМ для выполнения этих действий используют ряд специальных устройств. К ним от-

носятся такие известные устройства, как «джойстик» и «мышь». Первое из них представляет собой электро-механическое устройство, в котором два или три переменных резистора управляются одной рукояткой, выполненной в виде рычажка. Иногда вместо переменных резисторов используют емкостные датчики. Недостатком этого устройства является довольно сложная конструкция.

Манипулятор «мышь» представляет собой небольшую коробочку с клавишами, соединяемую с ЭВМ кабелем-«хвостом». Определенное внешнее сходство и породило название манипулятора — «мышь». Перемещение манипулятора по поверхности стола приводит к взаимно однозначному перемещению курсора по экрану дисплея. Конечно, выдаваемую манипулятором информацию о положении программное

а в игровых программах управлять перемещением каких-либо объектов по экрану. Однако чаще всего манипулятор используют именно для указания.

Для Микро-80 разработано простое устройство ввода данных, функционально аналогичное описанному выше. Его принципиальная схема изображена на рис. 1, возможный вариант внешнего оформления — на рис. 2. Устройство состоит из двух одновибраторов ($DA1, DA2$), в частотозадающие цепи которых включены переменные резисторы $R4$ и $R5$. Длительность выходных импульсов пропорциональна углу поворота движка соответствующего резистора. При вводе данных резистором $R4$ (ось X) управляют большим пальцем правой руки, резистором $R5$ (ось Y) — средним, а кнопкой $SB1$ — указательным.

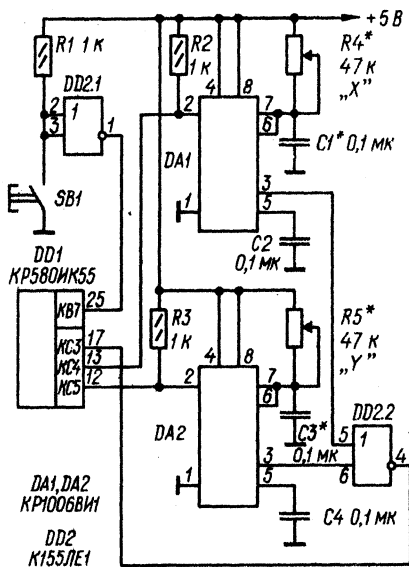


Рис. 1

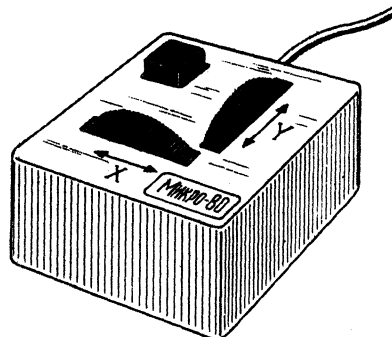


Рис. 2

Таблица 3

		:подпрограмма ROTX - ось X	
1960	1600	ROTX:	MVI D,0 ;счетчик длит. импульса
		:используем операцию устан./сброс битов канала C	
1962	3E08	MVI	A,8H ;формирование
1964	D304	OUT	4H ;импульса запуска
1966	3E09	MVI	A,9H ;на выводе KC4
1968	D304	OUT	4H ;порта клавиатуры
196A	C37719	JMP	ROTXU ;общие действия по X и Y
		:подпрограмма ROTY - ось Y	
196D	1800	ROTY:	MVI D,0
196F	3E0A	MVI	A,0AH ;формирование
1971	D304	OUT	4H ;импульса запуска
1973	3E0B	MVI	A,0BH ;на выводе KC5
1975	D304	OUT	4H ;порта клавиатуры
1977	DB05	ROTXU:	IN 5H
1979	14	INR	D ;счетчик + 1
197A	E608	ANI	8H ;выделяем 3 разряд
197C	CA7719	JZ	ROTXU ;импульс продолжается
197F	7A	MOV	A,D ;импульс закончился
1980	C9	RET	

Таблица 4

```

10 REM *****
20 REM * ПРОГРАММА РИСОВАНИЯ НА ЭКРАНЕ *
30 REM * С ПОМОЩЬЮ УСТРОЙСТВА УПРАВЛЕНИЯ *
40 REM * КУРСОРОМ. ИСПОЛЬЗУЕТ ПОДПРОГРАММЫ *
50 REM * В МАШИННЫХ КОДАХ ДЛЯ ОПРЕДЕЛЕНИЯ *
60 REM * КООРДИНАТ. *
70 REM *****
80 OUT4,131:REM НАСТРОЙКА ПОРТА
90 X=USR(6496):REM ВЫЗОВ ПОДПРОГРАММЫ ROTX
100 IF X>127 THEN X=127
110 Y=USR(6509):REM ВЫЗОВ ПОДПРОГРАММЫ ROTY
120 IF Y>63 THEN Y=63
130 REM ПРОВЕРКА СОСТОЯНИЯ КНОПКИ
140 IF INP(6) AND 128 THEN S=1:GOTO 150
150 S=0
160 GOTO 90

```

обеспечение может трактовать и иначе. Например, в графическом режиме с помощью манипулятора можно рисовать на экране различные фигуры,

Для подключения к микро-ЭВМ используют свободные порты ППА КР580ИК55, предназначенного для обслуживания клавиатуры. Можно под-

ключить устройство и иначе, но в этом случае необходимо соответственно изменить подпрограммы его обслуживания.

Запуск одновибраторов (по спаду импульса на входе 2) и определение длительности импульса производится подпрограммой, приведенной в табл. 3. После выдачи импульса запуска она считывает состояние порта ввода N5, к разряду KC3 которого через элемент DD2.2 подключены выходы одновибраторов. При выходе из подпрограммы содержимое аккумулятора соответствует углу поворота движка переменного резистора. Конденсаторы C1 и C3 подбирают таким образом, чтобы при полном повороте движков переменных резисторов R4 и R5 подпрограммы ROTX и ROTY возвращали числа 127 (размер по горизонтали) и 63 (размер по вертикали) соответственно.

Для хранения подпрограмм в машинных кодах, используемых совместно с интерпретатором, специально выделена небольшая область памяти с адресами с 1960H по 19FFH. Кроме приведенных подпрограмм, вы можете разместить здесь (если, конечно, хватит места) и свои. После этого на магнитофон целесообразно записать эту новую версию интерпретатора, что позволит в дальнейшем пользоваться библиотекой подпрограмм, расширяющих возможности интерпретатора.

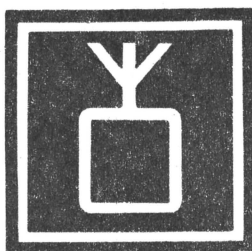
В табл. 4 приведен текст программы на Бейсике, которая предполагает использование описанного выше устройства управления курсором. Программа позволяет рисовать на экране дисплея разнообразные фигуры. В начале программы производится настройка порта (строка 80). Состояние кнопки, расположенной в устройстве управления курсором, определяется непосредственно в программе на Бейсике с помощью функции INP(X). В строках 90 и 110 происходит вызов подпрограмм в машинных кодах (функция USR(X)). Чтобы избежать в процессе работы программы появления ошибки 08, в ней предусмотрена коррекция значений, возвращаемых подпрограммами ROTX и ROTY.

Мы надеемся, что вы сумеете найти и более удачные варианты применения описанного устройства.

(Продолжение следует)

Г. ЗЕЛЕНКО,
В. ПАНОВ,
С. ПОПОВ

г. Москва



Стереodeкодер с кварцевым генератором

В современных условиях совершенствование техники приема программ стереофонического радиовещания в УКВ диапазоне идет по пути внедрения в стереофонические радиоприемные устройства элементов цифровой техники. Особенно отчетливо эта тенденция проявилась в конструировании стереофонических декодирующих устройств с временным переключением каналов и применением ФАПЧ. В публикуемой ниже статье вниманию читателей предлагается стереodeкодер, в системе ФАПЧ которого используется регулируемый генератор с кварцевым резонатором. Основные его достоинства — стабильное разделение стереоканалов, малые нелинейные искажения, хорошая фильтрация паразитных составляющих.

Наилучшее качество декодирования комплексного стереосигнала (КСС) обеспечивают, как известно, стереodeкодеры с временным переключением каналов [1]. Однако широкому их применению долгое время препятствовали трудности, связанные с формированием требуемых для таких устройств достаточно коротких импульсов, сопряженных с минимумами и максимумами сигнала поднесущей частоты (ПНЧ). Эти трудности удалось преодолеть, введя в формирователь коммутирующих импульсов фазовую автоподстройку частоты (ФАПЧ). Описания стереodeкодеров с ФАПЧ помещались на страницах журнала «Радио» [2, 3]. Анализ работы этих устройств выявил их существенный недостаток. Дело в том, что в системе ФАПЧ обоих стереodeкодеров использован обычный RC-генератор, не обладающий достаточной стабильностью частоты. По этой причине полосу захвата ФАПЧ пришлось расширить до единиц килогерц. Составляющие звуковых частот, попадающие в такую широкую полосу захвата, могут вызвать паразитную фазовую модуляцию сигнала генератора, что отрицательно сказывается на качестве демодуляции КСС. Чтобы этого не произошло, в стереodeкодеры дополнительно введен фильтр с добротностью порядка 100 и полосой пропускания 300 Гц. Однако и такой фильтр не может подавить все звуковые составляющие сигнала и не позволяет полностью избежать паразитной модуляции сигнала генератора, усугубляющейся при неточной настройке на частоту поднесущей.

Все это приводит к ухудшению разделения каналов в стереodeкодере.

Чтобы избавиться от этого недостатка, необходимо сузить полосу захвата ФАПЧ до 20...40 Гц, что требует применения высокостабильного управляемого генератора. В предлагаемом вниманию читателей стереodeкодере в качестве такого устройства применен генератор, стабилизированный кварцевым резонатором, с коэффициентом перестройки, достаточным для захвата и удержания частоты в возможном диапазоне нестабильности ПНЧ передатчика. Использование высокостабильного генератора позволило отказаться от достаточно сложных в изготовлении полосовых фильтров с высокой (более 100) добротностью и при этом не ухудшить, а улучшить помехозащищенность тракта ФАПЧ.

Еще одним отличием нового стереodeкодера от описанных в журнале является применение в синхронных детекторах дискретно-аналоговых линейно-интерполирующих фильтров вместо известных устройств с расширенным импульсом [2, 3].

Принцип работы этих устройств поясняется диаграммами напряжений, показанными на рис. 1. Ключевой детектор с расширением импульсов пропускает на конденсатор следующие с частотой 31, 25 кГц короткие импульсы, амплитуда которых соответствует мгновенным значениям КСС (рис. 1,а). Напряжение, соответствующее определенному импульсу, «запоминается» конденсатором и уровень его сохраняется не-