

SOUNDSYNTHESIZER

Use the Hi-Res graphics screen to create sound effects or simple melodies with this utility program. A combination of Applesoft and machine language lets you create, edit, and test sounds before you save them to disk.

by James R. Geschwender
3855 Orchard St.
Lincoln, NB 68503

Most people know that the Apple is capable of producing a wide variety of sounds and sound effects. However, if you program in Applesoft, chances are that you are not able to produce much more than simple beeps and buzzes. If you have ever thought that your programs could be spiced up with the addition of music or sound effects, wait no longer. SOUND SYNTHESIZER is designed to allow you to easily create and save sounds ranging from a complete musical score to explosions, machine guns, and even bird calls. In other words, it is an open-ended sound creation system, and any sound you create can be re-played from an Applesoft program with a single command.

Using the Program

SOUND SYNTHESIZER produces a high resolution representation of the sound as you create it. In this display, the top of the screen represents higher pitches, and the bottom represents lower pitches. When played, the sound is read from left to right. The program has two modes of sound creation. The first is for making all varieties of sound effects, and the second is for synthesizing music. I will begin by describing mode one.

To use the program, simply run SYNTHE-SIZER. From the introductory menu, select Instructions first, as these include demonstrations of some of the simpler effects that are possible. When the introductory menu is displayed again, select Begin New Sound Table, and when prompted, enter any legal file name for your Sound Table. The Hi-Res screen will clear, and after a short pause, the cursor will appear on the left-hand side of the screen. You may now begin to create a sound.

You will find that you can move the cursor up one dot with the 'I' key, five dots with the 'U' key, and twenty-five dots with the 'Y' key. Similarly, you can move the cursor down one, five, and twenty-five dots with the 'M', 'N', and 'B' keys respectively. The 'S' key sets the pitch of a single tone and advances the cursor

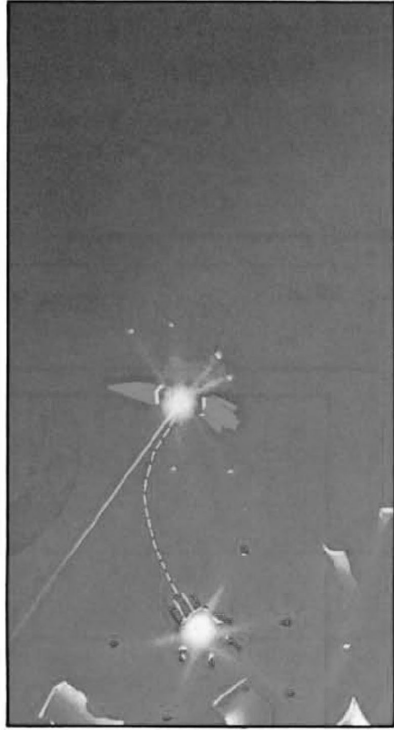
one dot to the right. Optionally, the vertical position of the cursor can be controlled with a game paddle or joystick, with the paddle button replacing the function of the 'S' key. This change is accomplished from the menu.

Pressing the space bar advances the cursor and leaves no dot, resulting in a pause when the sound is played. The right and left arrow keys can be used to move the cursor left or right, without changing existing tone dots. The cursor may also be moved with a single press of any of the number keys. Each represents a different horizontal position on the screen, proportional to their relative positions on the keyboard.

The other options in sound creation and editing are SHIFT and MENU. SHIFT will be explained later. Pressing <RETURN> brings up the main menu.

The Menu Options

The main menu has seven options as shown in Figure 1. The first of these, TEST SOUND, allows you to hear the sound you have designed. Within this option you choose the overall pitch, the number of repetitions, and the length of each tone. By adjusting these parameters, one sound pattern can be played hundreds of different ways. If you are satisfied



Music Synthesis

In the music synthesis mode, you will notice certain changes in the vertical movement of the cursor. In this mode, instead of having the full range of pitches, you are limited to the fourteen specific musical notes shown in Figure 3.

A musical staff is provided on the Hi-Res screen in order to distinguish the notes. To use this mode most effectively, you should have a simple piece of sheet music at hand. Notice that there are notes between the staff lines, and on the lines. You will find that the cursor will have a vertical position that corresponds to each of these notes. Sharps and flats are not included in this mode for the sake of clarity and ease of use. By placing tone dots in the same positions as the melody notes on the sheet music, you will be able to duplicate the music. You must keep in mind, however, that because of the way the Sound Reader operates, high notes play faster than low notes.

In practice, the highest C plays about three times faster than the lowest D. In order to compensate, you must make the note lines for high notes longer than those for low notes. For notes of average length, Figure 3 suggests the number of tone dots to use per note (i.e., nine tone dots for a high C). Remember that in music not all notes are meant to have the same duration, so you will not want to use these values for all notes. With this in mind, you may copy the music directly into the Sound Table. For a long piece, you may wish to continue on another screen. All options can be used the same way in the music synthesis mode as they were in the sound synthesis mode.

The Shift Option

There is one additional editing feature that is particularly useful in music synthesis, SHIFT. For instance, what if you had entered an entire musical piece, and on testing, found you had left out a note near the beginning? The SHIFT option can remedy this kind of problem. This is not a menu option. It can be accessed during sound creation and editing. Using the number and arrow keys, move the cursor to the horizontal position of the missing note, and press <ESC>. SHIFT then allows you to move all the dots that are to the right of the cursor, further right, so that you may insert the missing note. If you had inadvertently added an extra note, you could use SHIFT to move the same dots to the left instead, over the incorrect note. SHIFT can also be used if you want to shorten or lengthen a note.

After making any corrections, go back to the menu and test the sound again. Note that the cursor is regarded by the SOUND .READER as the end of sound marker. So if you leave the cursor in the middle of the sound when you return to the menu, you will only hear half of the sound when you test it. This is easily fixed by choosing option 3 (ADD TO SOUND) and moving the cursor to the end of the sound.



Saving the sound allows you to go on and work on the next sound for the table. One table can contain a maximum of 46 sounds, each of which can be played hundreds of different ways.

Error Handling

The program is designed to be user friendly. All nonreversible menu options require verification before being carried out. Entering a <SHIFT> 0 allows you to escape from the SHIFT option. Also, illegal entries of all kinds are generally ignored.

Using the Sounds You Create

After having created and saved a Sound Table, you may want to use these sounds in one of your Applesoft programs. In order to do this, you need only include the following lines in your program:

```
BLOAD SOUND.READER
```

and,

```
BLOAD soundfile
```

You can then execute any of the sounds in your sound file with the CALL command that you recorded during sound testing (CALL 768,1,3,5,10 for example).

The general form of the command is:

```
CALL 768,n,p,r,l
```

where n is the number of the sound in the table and must be between 1 and 46; p is the overall pitch and must be between 1 and 9; r is the number of repetitions and l is the length of each tone, both of which must be between 1 and 255. Figures 4-7 show just some of the possible sounds (and their parameters) that you can create. (These sounds, along with ten others, are included on disk. See the coupon on the last page of this issue for ordering details.)

Entering the Program

The program consists of two parts, the machine language sound reader, and the main Applesoft program. First enter the machine language listing, SOUND.READER (shown in Listing 1.) For help in entering machine language listings, see "A Welcome to New Nibble Readers" in the beginning of this issue.

When you have finished typing in the listing, save the program to disk with the command:

```
BSAVE SOUND.READER,AS300,LS70
```

with the sound, write down the CALL statement that is displayed at this point (see Figure 2); this is the statement you will use to invoke the sound from within an Applesoft program. Then return to the menu and use the SAVE SOUND option, which will save the entire Sound Table as it now stands, under the file name you specified earlier.

If you are not satisfied, you may DELETE SOUND, which clears the screen to allow you to start over. This does not, however, delete the disk file if you have saved it. You may also use the ADD TO SOUND option, which returns you to sound creation and editing.

The other possibilities from the menu are DISPLAY PREVIOUSLY CREATED SOUND, QUIT, and CHANGE MODES. The DISPLAY option allows you to call up and view any other sound in your Sound Table. If, for instance, you have nine sounds in your table, this allows you to add to, change, test, or delete any of them, and resave the table if you want to make any of the changes permanent.

QUIT allows you to leave the program. After having done so, you may wish to run the program again in order to work on, or begin a different Sound Table.

CHANGE MODES presents you with two options. One allows you to choose between keyboard entry and paddle entry to set the pitch. The other lets you choose between sound synthesis and music synthesis.

Next, enter the Applesoft program shown in Listing 2. You must save the program before attempting to run it, because of the way that the program relocates itself above the Hi-Res graphics screen. It does this (in program lines 90-110) because the program is too large to fit entirely below the page 1 Hi-Res screen. When you have entered the program, save it under the name SYNTHESIZER. With this done, you will be ready to use the program.

The Program

The layout of the program is as follows:

80-320 Initialization

Sound Creation and Editing

330-430 Clear for next sound
 440-600 Main sound creation loop
 610-690 Process button press
 700-1040 Process key press

Main Menu and Menu Routines

1050-1130 Main menu
 1140-1210 Test sound
 1220-1280 Save sound
 1290-1310 Add to sound
 1320-1430 Display previously created sound
 1440-1460 Delete sound
 1470-1560 Change modes
 1570-1590 Quit

Other Routines

1600-1800 Shift sound
 1810-1910 Load saved Sound Table
 1920-2630 Instructions
 2640-2760 Error Trap

The program's major working section is the sound creation and editing loop. It is located in lines 330-430. This loop monitors whether a key has been pressed. In paddle entry mode, the loop maintains the vertical position of the cursor, as well as reading the status of the paddle button. If a key is pressed, the program jumps to the routine in lines 700-1040, to determine which key has been pressed, and what to do about it. If the paddle button is pressed in paddle entry mode, the program jumps instead to the routine in lines 610-690. In order to understand these two routines and all their PEEKs and POKEs, let me introduce some variables.

CX, CY — the X- and Y-coordinates of the cursor.

FIGURE 1: MAIN MENU

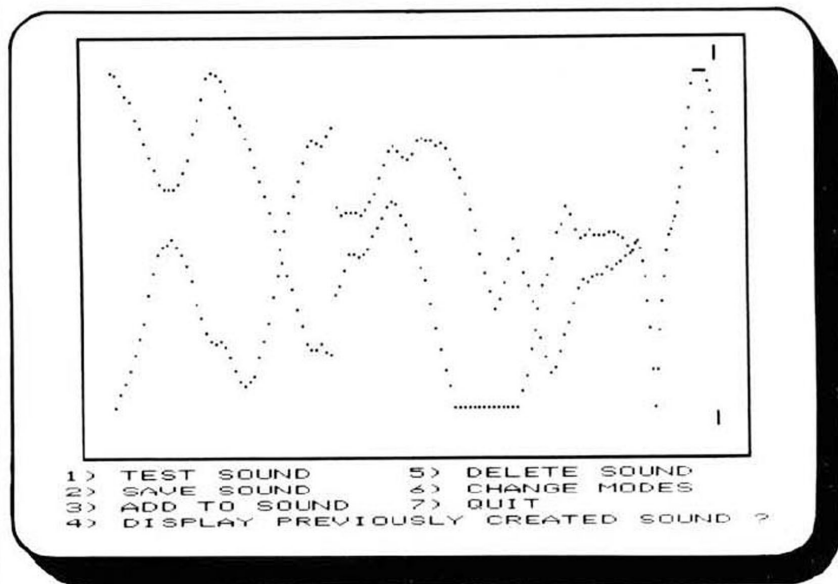
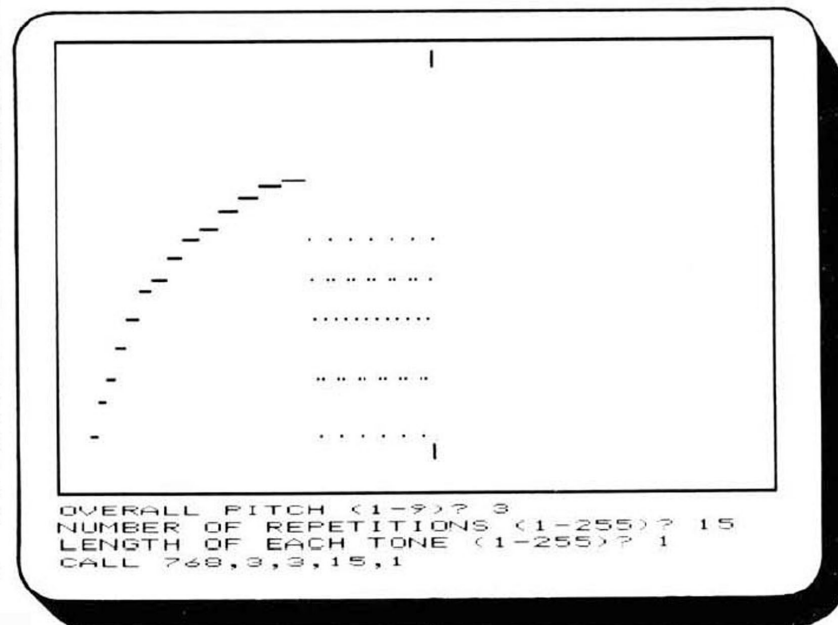


FIGURE 2: CALL PARAMETERS



PITCH — The pitch number; 1 represents the highest pitch, 255 the lowest, and 0 represents a pause.

TNUM — The number of tone dots to the left of the cursor (pauses included).

SNUM — The sound number.

STADD — The starting address of the Sound Table.

The Sound Table itself is simply a list of the pitch number of each tone dot in hexadecimal. Each sound in the table is allotted \$100 (or 256) bytes. The first (or \$00) byte of each sound gives the number of the allotted bytes actually used for that sound; so it is a number between \$01 and \$FF. Each of the following bytes, up to and including the \$FF byte, contains the pitch number of one of the tone dots, or a zero if there is no tone dot in that position. The next sound begins with the next \$100 byte interval, so a Sound Table with four sounds will be \$400 bytes long. **Listing 4** shows an example of a Sound Table.

You can see that a large Sound Table takes up quite a bit of memory. Luckily, it is easy to put the Sound Table in a section of memory that wouldn't otherwise be used. **SOUND SYNTHESIZER** builds the Sound Table beginning at memory address \$6400, which is a little above high resolution graphics screen 2. This location should be fine under most circumstances. If you have a long program though, and you want to place the program above the graphics screens, you may wish to locate the Sound Table at \$800. In order to do this you simply use the following lines in your program:

```
BLOAD SOUND.READER
BLOAD soundfile, A$800
POKE 789, 8
```

The **POKE** changes the byte of the Sound Reader program that identifies the location of the Sound Table. If you want the Sound Table somewhere else in memory, remember:

- The low byte of the hexadecimal address must be \$00 (i.e., \$800, \$4000, \$6100).
- The second parameter of the **POKE** must be the decimal equivalent of the high byte of the hex address.

FIGURE 3: MUSICAL STAFF

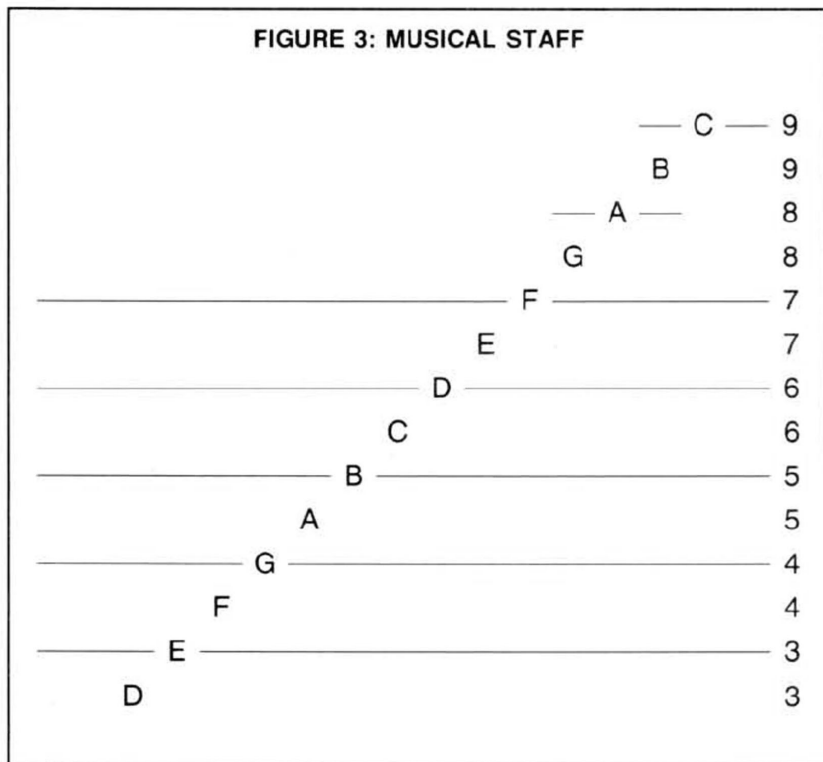


FIGURE 4: LASER BATTERY

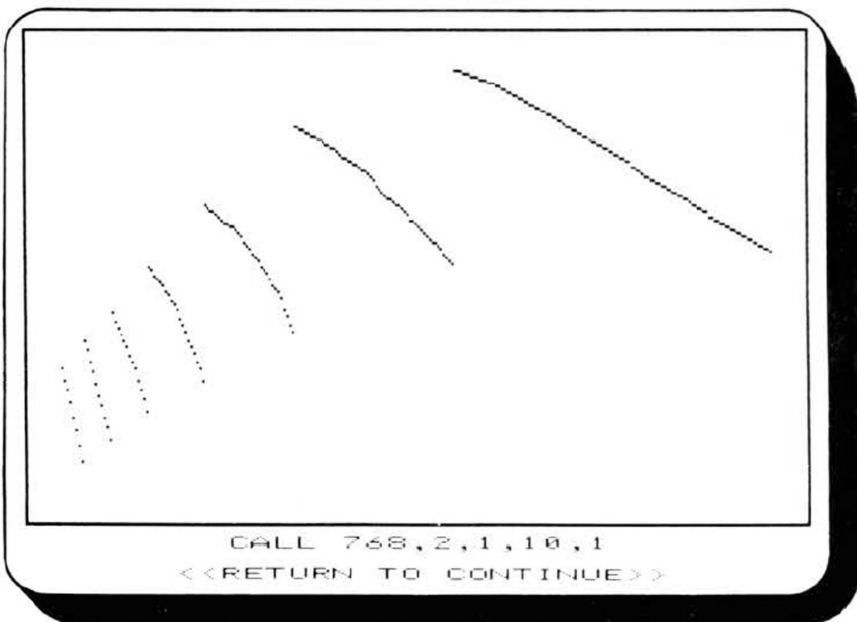


FIGURE 5: BIRD CALL

The Sound Reader

The basis of all sound effects produced by the Apple is memory location -16336, or \$C030 in hexadecimal. Accessing this location will produce a single click of the Apple's speaker. From machine language these clicks can be performed rapidly enough to generate a tone, with the pitch dependent on the length of the pause between clicks. The SOUND .READER program uses the data stored in a sound file created by SYNTHESIZER, to determine the length of these pauses. The specifics of how this is done can be better understood by examining the assembly listing of the program (Listing 1).

Lines 20-31 store the parameters n, p, r, and l from the CALL command in SNUM, OPITCH, REP, and TLEN respectively.

Lines 32-42 calculate the high byte of the address of the beginning of the sound, based on the sound number, and store it at memory locations \$324 and \$33F within the program. This corrects STSOUND, which is originally set at \$FF00.

Lines 43-45 store the length of the sound in SLEN.

Lines 46-55 compute the correct value for BRANCH, based on the overall pitch p. Once corrected, line 85 will produce a conditional branch to one of the lines 75-83. A branch to line 75 results in the longest pause, and occurs with an overall pitch of 9.

Lines 56-96 are the sound producing loop. Within this loop, a pitch number is loaded into the X-Register from the Sound Table. If that number is zero, line 72, which clicks the speaker, is skipped. Lines 73-83 execute the pause between clicks, which determines the pitch. Following the pause are the conditional branches for the three nested loops that determine whether to repeat the same pitch number, go on to the next pitch number, repeat the sound or end.

The diversity of the sounds generated by this routine, comes from the variables OPITCH, REP, and TLEN, and from the nature of the Sound Tables it reads. The only question that remains is how to make a Sound Table that will produce the sounds you desire.

The Sound Demo

The SYNTHESIZER and SOUND .READER programs are complete in themselves; and SYNTHESIZER includes instructions on using the system. But the varie-

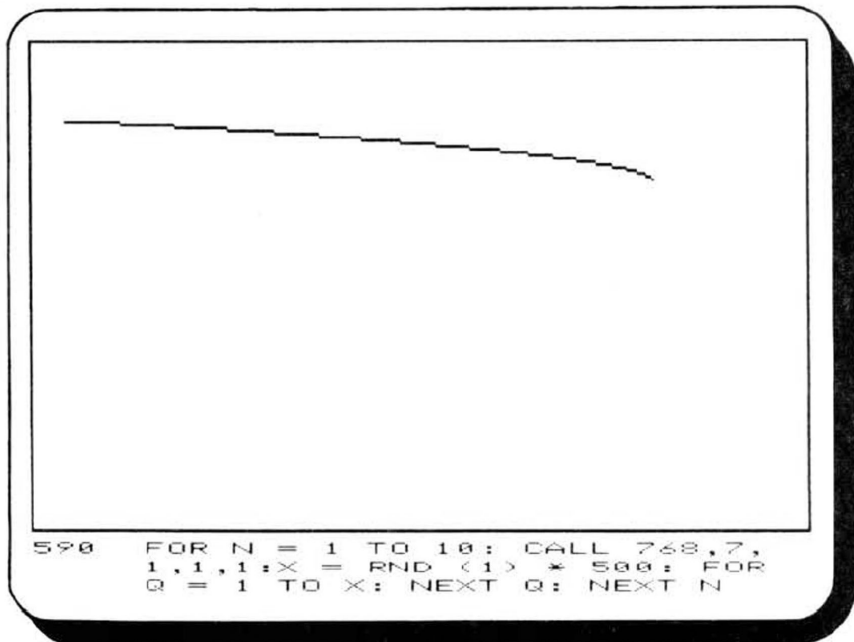
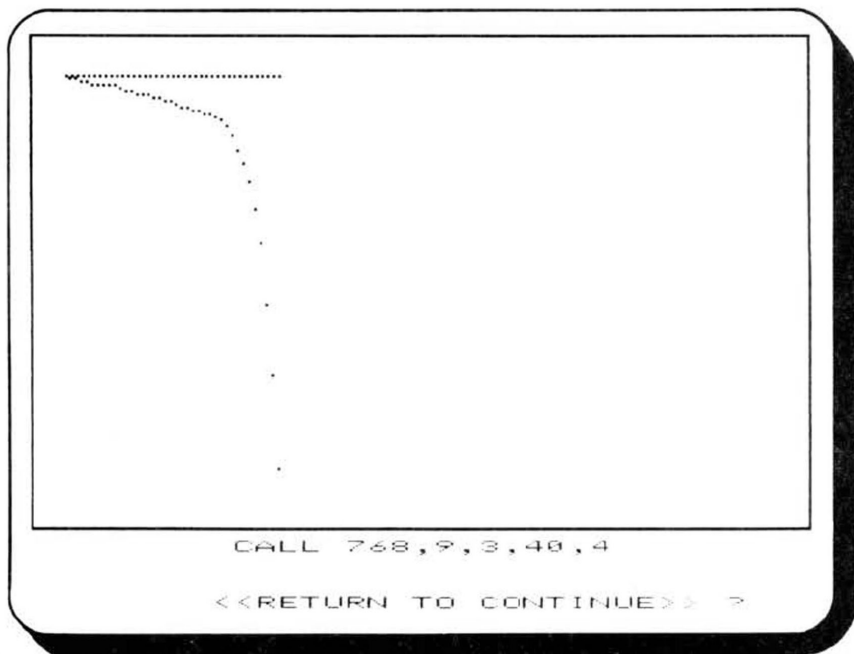


FIGURE 6: MACHINE GUN



LISTING 1: SOUND.READER

ty of techniques possible is great enough that a good set of examples will help get you on the right track. To this end, the Applesoft program SOUND.DEMO and the binary file SOUNDS are also presented here as Listings 3 and 4. SOUNDS must be entered from the Monitor, and when complete can be saved to disk with the command:

BSAVE SOUNDS, A\$6400, L\$E00

SOUNDS is an actual sound file created with SYNTHESIZER and includes such sounds as a laser battery, bird calls, an explosion, machine guns, the Marine Hymn and more. (See Figures 3-7.) SOUND.DEMO displays each of these, as well as describing the methods used in creating them. It also invokes each sound and shows the command or commands used to do so. In all, it is intended to enhance your understanding of sound creation so that you can more easily design custom sounds for your own programs.

User Modifications

Once you fully understand SOUND SYNTHESIZER, you may decide to modify the program. It is, therefore, important to note that the end of the program comes quite close to the beginning of the sound file it creates. This means that if your modifications significantly increase the length of the program, you will probably want to change the following first.

- In line 210 of SYNTHESIZER, change the value of STADD to 28672
- In the SOUND.READER listing, change address \$315 to \$70

These changes will give you more room to make modifications, but they will also reduce to 34, the number of sounds you may have in one Sound Table. The Sound Table will now be built at \$7000 instead of \$6400.

One modification you may wish to try is to allow the entry of sharp and flat notes for music synthesis. Some method of distinguishing them from natural notes would be necessary in order to prevent confusion.

Conclusion

When you begin experimenting with creating sounds, use your imagination. You will be surprised at the wide variety of effects you can achieve, if you try different techniques. Short, simple patterns can often be the most effective, but artistry is the key. If you are like me, you will have the most fun just making up sounds to see what the Apple can do.

```

SOURCE FILE: SOUND.READER.SOURCE
0000: 1 *
0000: 2 * SOUND.READER *
0000: 3 * BY JAMES R. GESCHWENDER *
0000: 4 * COPYRIGHT (C) 1984 *
0000: 5 * BY MICROSPARC, INC. *
0000: 6 * ASSEMBLER: DOS TOOL KIT *
0000: 7 *
-----
NEXT OBJECT FILE NAME IS SOUND.READER
0300: 8 ORG $0300
0006: 9 SNUM EQU $06
0007: 10 OPITCH EQU $07
0008: 11 REP EQU $08
0009: 12 TLEN EQU $09
001C: 13 TNUM EQU $1C
001B: 14 TNUM2 EQU $1B
001A: 15 TLEN2 EQU $1A
FF00: 16 STSOUND EQU $FF00
E74C: 17 COMBYTE EQU $E74C
C030: 18 SPKR EQU $C030
0300: 19 *
0300: 20 * STORE SOUND NUMBER
0300:20 4C E7 21 JSR COMBYTE
0303:86 06 22 STX SNUM
0305: 23 * STORE OVERALL PITCH
0305:20 4C E7 24 JSR COMBYTE
0308:86 07 25 STX OPITCH
030A: 26 * STORE NUMBER OF REPETITIONS
030A:20 4C E7 27 JSR COMBYTE
030D:86 08 28 STX REP
030F: 29 * STORE LENGTH OF EACH TONE
030F:20 4C E7 30 JSR COMBYTE
0312:86 09 31 STX TLEN
0314: 32 * CALCULATE HIGH BYTE OF THE
0314: 33 * ADDRESS OF THE BEGINNING
0314: 34 * OF THE SOUND
0314:A0 64 35 LDY #$64
0316:88 36 DEY
0317:C8 37 INCADD INY
0318:C6 06 38 DEC SNUM
031A:D0 FB 39 BNE INCADD
031C: 40 * CORRECT STSOUND
031C:8C 24 03 41 STY $0324
031F:8C 3F 03 42 STY $033F
0322: 43 * GET SLEN FROM SOUNDFILE
0322:AE 00 FF 44 LDX STSOUND
0325:86 1C 45 STX TNUM
0327: 46 * CALCULATE AND CORRECT 'BRANCH'
0327: 47 * BASED ON THE VALUE OF THE
0327: 48 * OVERALL PITCH
0327:A0 00 49 LDY #$00
0329:88 50 CHBR DEY
032A:88 51 DEY
032B:88 52 DEY
032C:C6 07 53 DEC OPITCH
032E:D0 F9 54 BNE CHBR
0330:8C 61 03 55 STY $0361
0333: 56 *
0333: 57 * SOUND PRODUCING LOOP
0333: 58 *

```

```

0333:A0 01 59 REPEAT LDY #501
0335:A6 1C 60 LDX TNUM
0337:86 1B 61 STX TNUM2
0339:A5 09 62 NEXTPN LDA TLEN
033B:85 1A 63 STA TLEN2
033D: 64 * LOAD X WITH A PITCH NUMBER
033D: 65 * FROM SOUNDFILE
033D:BE 00 FF 66 SAMEPN LDX STSOUND,Y
0340:E8 67 INX
0341:CA 68 DEX
0342: 69 * IF PITCH NUMBER=0 SKIP CLICK
0342:F0 1B 70 BEQ DECPIT
0344: 71 * CLICK SPEAKER
0344:AD 30 C0 72 LDA SPKR
0347: 73 * PAUSE BETWEEN CLICKS
0347: 74 * (LONG PAUSE = LOW PITCH)
0347:4C 4A 03 75 JMP J1
034A:4C 4D 03 76 J1 JMP J2
034D:4C 50 03 77 J2 JMP J3
0350:4C 53 03 78 J3 JMP J4
0353:4C 56 03 79 J4 JMP J5
0356:4C 59 03 80 J5 JMP J6
0359:4C 5C 03 81 J6 JMP J7
035C:4C 5F 03 82 J7 JMP DECPIT
035F:CA 83 DECPIT DEX
0360: 84 * IF X<>0, CONTINUE PAUSE
0360:D0 FE 85 BRANCH BNE BRANCH
0362:C6 1A 86 DEC TLEN2
0364: 87 * IF TLEN2<>0, SAME PITCH NUMBER
0364:D0 D7 88 BNE SAMEPN
0366:C8 89 INY
0367:C6 1B 90 DEC TNUM2
0369: 91 * IF TNUM2<>0, NEXT PITCH NUMBER
0369:D0 CE 92 BNE NEXTPN
036B:C6 08 93 DEC REP
036D: 94 * IF REP<>0, REPEAT SOUND
036D:D0 C4 95 BNE REPEAT
036F:60 96 RTS

```

KEY PERFECT 4.0
RUN ON
SOUND READER

CHECK CODE 3.0

```

=====
CODE      ADDR# - ADDR#
-----
27B4      0300 - 034F
0F07      0350 - 036F
PROGRAM CHECK IS : 70

```

ON: SOUND READER
TYPE: B
LENGTH: 0070
CHECKSUM: 87

LISTING 2: SYNTHESIZER

```

10 REM *****
20 REM * SYNTHESIZER *
30 REM * BY JAMES R. GESCHWENDER *
40 REM * COPYRIGHT (C) 1984 *
50 REM * BY MICROSPARC, INC. *
60 REM * LINCOLN, MA. 01773 *
70 REM *****
80 REM INITIALIZATION
90 IF PEEK (104) = 64 THEN 120
100 POKE 103,1: POKE 104,64: POKE 16384,0
110 PRINT CHR$(4);"RUN SYNTHESIZER"
120 LOMEM: 37376
130 DIM Q(14),Y(6),M$(2),I$(2):D$ = CHR$(4)
140 PRINT D$;"BLOOD SOUND READER"
150 REM DATA FOR MUSICAL NOTES
160 FOR N = 0 TO 14: READ Q(N): NEXT : DATA
73,77,86,96,108,116,128,144,152,171,192,
213,230,254,0
170 REM DATA FOR SCALE LINES
180 FOR N = 0 TO 6: READ Y(N): NEXT : DATA 4
8,54,65,77,89,107,128

```

```

190 REM DATA FOR SHAPE TABLE
200 FOR N = 912 TO 924: READ A: POKE N,A: NEXT
: POKE 232,144: POKE 233,3: DATA 2,0,6,0
,8,0,4,0,172,30,7,32,0
210 Z = 12.5:M = 1:I = 1:PITCH = 125:SNUM = 1
:NUM = 1:STADD = 25600:TEXT : HOME : SCALE=
1: ROT= 0
220 M$(1) = "SOUND SYNTHESIS":M$(2) = "MUSIC
SYNTHESIS"
230 I$(1) = "KEYBOARD ENTRY":I$(2) = "PADDLE
ENTRY"
240 HTAB 11: PRINT "SOUND SYNTHESIZER": VTAB
22: PRINT "** COPYRIGHT 1984 BY MICROSPA
RC, INC. **"
250 VTAB 10: HTAB 7: PRINT "1) LOAD EXISTING
SOUND TABLE"
260 HTAB 7: PRINT "2) BEGIN NEW SOUND TABLE"
270 HTAB 7: PRINT "3) INSTRUCTIONS":
280 GET A$:X = VAL (A$): IF X < 1 OR X > 3 THEN
250
290 HOME : ON X GOTO 1820,300,1930
300 PRINT "ENTER NAME FOR NEW SOUND TABLE.":
INPUT NAMES: ONERR GOTO 2650
310 IF VAL (NAMES) > 0 OR LEN (NAMES) > 15
THEN PRINT "ILLEGAL FILE NAME. TRY AGA
IN": GOTO 300
320 PRINT D$"BSAVE";NAMES$;"A";STADD;"L1": POKE
216,0
330 REM CLEAR FOR NEXT SOUND
340 HGR : HCOLOR= 3: HPLLOT 0,0 TO 279,0 TO 2
79,159 TO 0,159 TO 0,0
350 GOSUB 1560:TNUM = 1
360 FOR N = 0 TO 255: POKE STADD + SNUM * 25
6 - 256 + N,0: NEXT
370 HOME : VTAB 21: PRINT "RETURN FOR MENU":
380 HTAB 25: INVERSE : PRINT "SOUND NUMBER"
:SNUM: NORMAL
390 IF I = 1 THEN PRINT "KEYS YUI FOR UP, B
NM FOR DOWN, S FOR DOT": GOTO 410
400 PRINT "PADDLE AND BUTTON TO SET PITCH"
410 PRINT "SPACE BAR FOR PAUSE":
420 HTAB 26: PRINT "ESC FOR SHIFT"
430 PRINT "KEYS 0-9 AND ARROW KEYS TO PLACE
CURSOR":
440 REM MAIN SOUND MAKING LOOP
450 IF I = 1 THEN 580
460 REM PADDLE ENTRY
470 IF M = 1 THEN 510
480 N = INT (( PDL (1) + 21.23) / 21.24):PIT
CH = Q(N)
490 HCOLOR= (N < 1): HPLLOT 10,Y(0) TO 269,Y(
0)
500 HCOLOR= (N < 3): HPLLOT 10,Y(1) TO 269,Y(
1): GOTO 520
510 PITCH = PDL (1) + 1: IF PITCH > 255 THEN
PITCH = 255
520 CX = Z + TNUM:CY = Z + PITCH / 2
530 XDRAW 2 AT CX,CY: HCOLOR= 0
540 X = PEEK (- 16384): IF X > 127 THEN POKE
- 16368,0: GOTO 890
550 X = PEEK (- 16286): IF X > 127 THEN 620
560 XDRAW 2 AT CX,CY: GOTO 470
570 REM KEYBOARD ENTRY
580 CX = Z + TNUM:CY = Z + PITCH / 2: XDRAW 2
AT CX,CY
590 X = PEEK (- 16384): IF X > 127 THEN POKE
- 16368,0: GOTO 720
600 GOTO 590
610 REM PROCESS BUTTON PRESS
620 XDRAW 2 AT CX,CY
630 A = STADD + SNUM * 256 + TNUM - 256
640 HPLLOT CX, PEEK (A) / 2 + Z
650 POKE A,PITCH: POKE A - TNUM,TNUM
660 TNUM = TNUM + 1 - (TNUM > 254)
670 XDRAW 1 AT CX,CY
680 IF PEEK (- 16286) < 128 OR I = 1 THEN
450
690 GOTO 680
700 REM PROCESS KEY PRESS
710 REM KEYBOARD ENTRY
720 IF X < 193 THEN 890
730 IF X = 211 THEN 620

```

```

740 XDRAW 2 AT CX,CY
750 X = - 50 * (X = 217) - 10 * (X = 213) -
      2 * (X = 201) + 50 * (X = 194) + 10 * (X
      = 206) + 2 * (X = 205)
760 IF M = 2 THEN 800
770 PITCH = PITCH + X: IF PITCH < 1 THEN PITC
      H = 1
780 IF PITCH > 255 THEN PITCH = 255
790 GOTO 450
800 FOR N = 0 TO 13: IF PITCH > Q(N) THEN NEXT

810 X = SGN (X) * INT ( ABS (X) ^ .4)
820 N = N + X: IF N < 0 THEN N = 0
830 IF N > 13 THEN N = 13
840 PITCH = Q(N)
850 HCOLOR= (N < 1): HPLOT 10,Y(0) TO 269,Y(
      0)
860 HCOLOR= (N < 3): HPLOT 10,Y(1) TO 269,Y(
      1)
870 HCOLOR= 0: GOTO 450
880 REM PADDLE AND KEYBOARD ENTRY
890 XDRAW 2 AT CX,CY
900 X = X - 128: IF X = 21 THEN 990
910 IF X = 32 THEN 970
920 IF X > 47 AND X < 58 THEN 1020
930 IF X = 8 THEN 1030
940 IF X = 27 THEN GOSUB 1610: GOSUB 1300
950 IF X = 13 THEN 1060
960 GOTO 450
970 A = STADD + SNUM * 256 - 256 + TNUM
980 HPLLOT CX, PEEK (A) / 2 + Z: POKE A,0
990 POKE STADD + SNUM * 256 - 256,TNUM
1000 TNUM = TNUM + 1 - (TNUM > 254)
1010 GOTO 450
1020 X = X - 49 + 10 * (X = 48):TNUM = INT (
      X * 28.3): GOTO 990
1030 IF TNUM < 2 THEN 450
1040 TNUM = TNUM - 2: GOTO 990
1050 REM MAIN MENU
1060 HCOLOR= 3: HPLLOT CX,4 TO CX,9: HPLLOT CX
      ,142 TO CX,147
1070 HOME : VTAB 21: PRINT " 1) TEST SOUND"
      ;; HTAB 21: PRINT "5) DELETE SOUND"
1080 PRINT " 2) SAVE SOUND":; HTAB 21: PRINT
      "6) CHANGE MODES"
1090 PRINT " 3) ADD TO SOUND":; HTAB 21: PRINT
      "7) QUIT"
1100 PRINT " 4) DISPLAY PREVIOUSLY CREATED
      SOUND ":; GET AS$
1110 X = VAL (AS$): IF X < 1 OR X > 7 THEN 10
      60
1120 ON X GOSUB 1150,1230,1300,1330,1450,148
      0,1580
1130 GOTO 1060
1140 REM SOUND TEST
1150 HOME : VTAB 21: HTAB 7: PRINT "**TEST S
      OUND**"
1160 INPUT "OVERALL PITCH (1-9)? ";P: IF P <
      1 OR P > 9 THEN 1160
1170 INPUT "NUMBER OF REPETITIONS (1-255)? "
      ;R: IF R < 1 OR R > 255 THEN 1170
1180 INPUT "LENGTH OF EACH TONE (1-255)? ";L
      : IF L < 1 OR L > 255 THEN 1180
1190 PRINT "CALL 768,"SNUM","P","R","L: CALL
      768,SNUM,P,R,L
1200 PRINT "<<PRESS RETURN FOR MENU>> ":;
      GET AS$: IF ASC (AS$) = 13 THEN RETURN

1210 GOTO 1150
1220 REM SAVE SOUND
1230 HOME : VTAB 21: PRINT "VERIFY SAVE (Y/N
      )? ":; GET AS$: IF AS$ < > "Y" THEN RETURN

1240 HOME : VTAB 21: PRINT " **SAVING SOUN
      D TABLE**": PRINT "TABLE NAME - ":NAMES$
1250 A = ( PEEK (STADD + NUM * 256 - 256) = 0
      )
1260 X = 256 * (NUM - A):NUM = NUM + 1 - A
1270 PRINT DS$;"BSAVE ";NAMES$;"A";STADD;"L"
      ;X
1280 POP : GOTO 1870
1290 REM ADD TO SOUND

1300 HCOLOR= 0: HPLLOT CX,4 TO CX,9: HPLLOT CX
      ,142 TO CX,147
1310 POP : GOTO 370
1320 REM DISPLAY PREVIOUS SOUND
1330 IF NUM < 2 THEN RETURN
1340 HOME : VTAB 21: PRINT "DISPLAY WHICH SO
      UND NUMBER (1-"NUM":; INPUT "? ";AS$
1350 X = VAL (AS$): IF X < 1 OR X > NUM THEN
      1340
1360 HPLLOT 13,11 TO 267,11: HCOLOR= 0: HPLLOT
      Z,1 TO Z,158
1370 SNUM = X:TNUM = PEEK (STADD + SNUM * 25
      6 - 256)
1380 FOR N = 1 TO 255
1390 CX = Z + N:Q = PEEK (STADD + SNUM * 256
      - 256 + N):CY = Z + Q / 2
1400 HPLLOT CX,1 TO CX,158: IF Q > 0 THEN XDRAW
      1 AT CX,CY
1410 NEXT
1420 IF M = 2 THEN GOSUB 1560
1430 CX = Z + TNUM: HCOLOR= 3: HPLLOT CX,4 TO
      CX,9: HPLLOT CX,142 TO CX,147: RETURN
1440 REM DELETE SOUND
1450 HOME : VTAB 21: PRINT "VERIFY DELETE (Y
      /N)? ":; GET AS$: IF AS$ < > "Y" THEN RETURN

1460 POP : GOTO 340
1470 REM CHANGE MODES
1480 HOME : VTAB 21: INVERSE : PRINT " CURRE
      NT MODE ": PRINT MS(M): PRINT IS(I): NORMAL

1490 VTAB 22: HTAB 18: PRINT "1) CHANGE SOUN
      D MODE"
1500 HTAB 18: PRINT "2) CHANGE ENTRY MODE"
1510 HTAB 12: PRINT "<<RETURN FOR MENU>> ":;
      GET AS$
1520 X = ASC (AS$): IF X = 13 THEN RETURN
1530 IF X = 49 THEN M = 2 - (M = 2): GOSUB 1
      560: GOTO 1480
1540 IF X = 50 THEN I = 2 - (I = 2): GOTO 14
      80
1550 GOTO 1480
1560 HCOLOR= M - 1: FOR N = 2 * M - 2 TO 6: HPLLOT
      10,Y(N) TO 269,Y(N): NEXT : RETURN
1570 REM QUIT
1580 HOME : VTAB 21: PRINT "VERIFY QUIT (Y/N
      )? ":; GET AS$: IF AS$ < > "Y" THEN RETURN

1590 TEXT : HOME : END
1600 REM SHIFT DISPLAY
1610 HCOLOR= 3: HPLLOT CX,4 TO CX,9: HPLLOT CX
      ,142 TO CX,147
1620 HOME : VTAB 21: PRINT "WITH THIS FEATUR
      E YOU MAY SHIFT ALL OF"
1630 PRINT "THE DOTS THAT ARE TO THE RIGHT O
      F THE": PRINT "CURSOR,"
1640 PRINT "SHIFT LEFT OR RIGHT (L OR R)? ":
      ; GET AS: PRINT AS$
1650 IF AS$ = "R" THEN AS$ = "RIGHT":X = 1: GOTO
      1680
1660 IF AS$ < > "L" THEN 1610
1670 AS$ = "LEFT":X = - 1
1680 PRINT "SHIFT HOW FAR "AS$" (0-9)? ":; GET
      AS$: PRINT AS$:
1690 PRINT : PRINT : INVERSE : HTAB 16: PRINT
      "SHIFTING":; NORMAL
1700 X = X + VAL (AS$): ON SGN (X) + 2 GOTO
      1710,1800,1730
1710 IF - X > = TNUM THEN X = 1 - TNUM
1720 FOR N = TNUM TO 255: GOTO 1750
1730 IF X + TNUM > 255 THEN RETURN
1740 FOR N = 255 - X TO TNUM STEP - 1
1750 Q = STADD + 256 * SNUM - 256 + N: POKE Q
      + X, PEEK (Q): NEXT
1760 IF X > 0 THEN FOR N = TNUM TO TNUM + X
      : POKE STADD + SNUM * 256 - 256 + N,0: NEXT
      : GOTO 1780
1770 FOR N = 255 + X TO 255: POKE STADD + SN
      UM + 256 - 256 + N,0: NEXT
1780 POKE STADD + SNUM * 256 - 256,TNUM + X:
      X = SNUM
1790 GOSUB 1360

```

```

1800 RETURN
1810 REM LOAD SAVED SOUND TABLE
1820 PRINT "ENTER NAME OF EXISTING SOUND TABLE."
1830 PRINT "TYPE 'CAT' FOR A CATALOG": ONERR
GOTO 2660
1840 INPUT NAMES$: IF NAMES$ = "CAT" THEN 1900
1850 PRINT DS;"BLOAD";NAMES$;"A";STADD: POKE
216,0
1860 NUM = PEEK (43617) + 1
1870 HOME : VTAB 21: PRINT "FILE "NAMES$ " CURRENTLY"
1880 PRINT "CONTAINS "NUM - 1" SOUNDS."
1890 SNUM = NUM: GOTO 340
1900 PRINT DS;"CATALOG"
1910 GOTO 1820
1920 REM INSTRUCTIONS
1930 HOME : HTAB 14: PRINT "INSTRUCTIONS"
1940 VTAB 7: PRINT "THIS PROGRAM HAS TWO SOUND MAKING MODES."
1950 PRINT "THE FIRST IS FOR MAKING SOUNDS EFFECTS OF";
1960 PRINT "ALL VARIETIES, AND THE SECOND IS FOR";
1970 PRINT "SYNTHESIZING MUSIC."
1980 PRINT : PRINT "TO USE MODE ONE YOU SIMPLY LAY OUT THE"
1990 PRINT "DESIRED SOUND ON THE HI-RES DISPLAY"
2000 PRINT "USING A PADDLE CONTROLLER OR JOY STICK"
2010 PRINT "AND THE KEYBOARD, OR THE KEYBOARD ONLY."
2020 PRINT "THE TOP OF THE SCREEN REPRESENTS HIGH"
2030 PRINT "PITCH, AND THE BOTTOM REPRESENTS LOW"
2040 PRINT "PITCH."
2050 FOR N = 1 TO 255
2060 Q = (N < 127) * (125 - 60 * SIN (N * 0.5))
2070 POKE STADD + N,Q: NEXT
2080 GOSUB 2610: PRINT "AN EXAMPLE MIGHT LOOK LIKE THIS."
2090 HGR : HCOLOR= 3: HPLLOT 0,0 TO 279,0 TO 279,159 TO 0,159 TO 0,0
2100 X = 1:M = 1: POKE STADD,126: GOSUB 1360
2110 VTAB 21: PRINT "WHEN YOU WISH TO HEAR THE SOUND, GO TO"
2120 PRINT "THE MENU, AND USE THE 'TEST' OPTION.": GOSUB 2610
2130 PRINT "YOU CONTROL THE OVERALL PITCH."
2140 PRINT "THE NUMBER OF REPETITIONS."
2150 PRINT "AND THE LENGTH OF EACH NOTE.": GOSUB 2610
2160 PRINT "SO, WITH THE SOUND PATTERN ABOVE, THIS.": CALL 768,1,1,1,30
2170 PRINT "AND THIS.": CALL 768,1,2,10,1
2180 PRINT "AND THIS.": CALL 768,1,9,3,3
2190 PRINT "ARE POSSIBLE.": GOSUB 2610
2200 PRINT "IF YOU ARE NOT SATISFIED WITH THE SOUND.":
2210 PRINT "USE THE 'ADD TO SOUND' OPTION.": GOSUB 2610
2220 PRINT "YOU MAY MOVE THE CURSOR HORIZONTALLY"
2230 PRINT "LEFT OR RIGHT WITH THE ARROW KEYS, AS"
2240 PRINT "WELL AS WITH THE NUMBER KEYS (0-9).": GOSUB 2610
2250 PRINT "YOU MAY WRITE OVER AN INCORRECT SECTION"
2260 PRINT "WITH DIFFERENT TONE DOTS, OR ERASE A"
2270 PRINT "SECTION BY SPACING OVER IT.": GOSUB 2610
2280 TNUM = 30: CX = Z + 126: CY = 60
2290 PRINT "YOU MAY ALSO SHIFT THE RIGHT HAND"
2300 PRINT "PORTION OF THE SCREEN LEFT OR RIGHT."

```

```

2310 PRINT "PLACE THE CURSOR HERE FOR INSTANCE, AND"
2320 PRINT "PRESS ESC TO TRY IT. <<PRESS ESC>> "
2330 HCOLOR= 0: HPLLOT CX,4 TO CX,9: HPLLOT CX,142 TO CX,147: CX = Z + TNUM: XDRAW 2 AT CX,CY
2340 GET AS: IF ASC (AS) < > 27 THEN 2340
2350 XDRAW 2 AT CX,CY: GOSUB 1610
2360 HOME : VTAB 21: IF X = 0 THEN PRINT "PLEASE TRY AGAIN WITH A NON-ZERO SHIFT.": GOSUB 2610: GOTO 2320
2370 PRINT "NOTE THAT THE SOUND READER READS ONLY UP";
2380 PRINT "TO THE 'END OF SOUND' SLASHES.": CALL 768,1,2,3,70
2390 GOSUB 2610: PRINT "SO PLACE THE CURSOR AT THE END OF THE"
2400 PRINT "SOUND BEFORE TESTING.": GOSUB 2610
2410 PRINT "MODE TWO SETS UP A MUSICAL SCALE WHICH"
2420 PRINT "ALLOWS YOU TO TRANSCRIBE DIRECTLY FROM"
2430 PRINT "SHEET MUSIC, TO A FORMAT LIKE THIS."
2440 Q = 1: FOR N = 0 TO 8
2450 READ X,A: DATA 7,9,5,11,3,14,14,1,5,9,6,8,5,9,7,7,14,60
2460 FOR TNUM = Q TO Q + A: POKE STADD + TNUM,Q(X)
2470 NEXT TNUM: Q = TNUM
2480 NEXT N
2490 POKE STADD,79: X = 1: M = 2: GOSUB 1360
2500 GOSUB 2610: PRINT "NOTE THAT HIGH NOTES PLAY FASTER SO THAT";
2510 PRINT "THE NOTE LINE MUST BE LONGER FOR EQUAL"
2520 PRINT "DURATION.": CALL 768,1,4,1,12
2530 GOSUB 2610: CALL 768,1,1,1,25: TEXT : VTAB 7
2540 PRINT "WHEN YOU ARE SATISFIED WITH THE SOUND."
2550 PRINT "RECORD THE CALL COMMAND DISPLAYED, AND"
2560 PRINT "SAVE THE SOUND. TO USE THE SOUND IN AN"
2570 PRINT "APPLESOFT PROGRAM, BLOAD THE SOUND FILE"
2580 PRINT "AND THE SOUND READER PROGRAM, AND USE"
2590 PRINT "THE CALL COMMAND TO EXECUTE THE SOUND."
2600 GOSUB 2610: RESTORE : GOTO 160
2610 VTAB 24: HTAB 9: PRINT "<<RETURN TO CONTINUE>> "
2620 GET AS: IF ASC (AS) < > 13 THEN 2610
2630 HOME : VTAB 21: RETURN
2640 REM DISK ERROR TRAP
2650 EL = 1: GOTO 2670: REM ENTRY POINT FOR DISK WRITE
2660 EL = 2: REM ENTRY POINT FOR DISK READ
2670 ER = PEEK (222)
2680 IF ER = 4 THEN PRINT "DISK WRITE PROTECTED": GOTO 2760
2690 IF ER = 6 THEN PRINT "FILE NOT FOUND": GOTO 2760
2700 IF ER = 8 THEN PRINT "DISK I/O ERROR": GOTO 2760
2710 IF ER = 9 THEN PRINT "DISK FULL": GOTO 2760
2720 IF ER = 10 THEN PRINT "FILE LOCKED": GOTO 2760
2730 IF ER = 11 THEN PRINT "ILLEGAL FILE NAME": GOTO 2760
2740 IF ER = 13 THEN PRINT "FILE TYPE MISMATCH": GOTO 2760
2750 PRINT "ERROR #";ER"IN LINE "; PEEK (218) + PEEK (219) + 256
2760 ON EL GOTO 300,1820

```


KEY PERFECT 4.0
RUN ON
SYNTHESIZER

CODE	LINE# - LINE#
6CFB	10 - 100
A5C1	110 - 200
C29E	210 - 300
A7F3	310 - 400
70F0	410 - 500
733D	510 - 600
545B	610 - 700
6FD5	710 - 800
58A2	810 - 900
51A9	910 - 1000
9069	1010 - 1100
A0DB	1110 - 1200
8067	1210 - 1300
7C6D	1310 - 1400
73CA	1410 - 1500
72BC	1510 - 1600
8DBD	1610 - 1700
7760	1710 - 1800
834B	1810 - 1900
907E	1910 - 2000
8009	2010 - 2100
ADB5	2110 - 2200
A63D	2210 - 2300
AC5D	2310 - 2400
8DCF	2410 - 2500
9A93	2510 - 2600
7BA2	2610 - 2700
5712	2710 - 2760

PROGRAM CHECK IS : 1F94

CHECK CODE 3.0

ON: SYNTHESIZER
TYPE: A

LENGTH: 1C4D
CHECKSUM: B9

LISTING 3: SOUND.DEMO

```

10 REM .....
20 REM * SOUND DEMO *
30 REM * BY JAMES R. GESCHWENDER *
40 REM * COPYRIGHT (C) 1984 *
50 REM * BY MICROSPARC, INC. *
60 REM * LINCOLN, MA. 01773 *
70 REM .....
80 DIM Y(4):DS = CHR$(4):STADD = 25600:Z =
  12.5
90 PRINT DS;"BLOAD SOUND READER"
100 FOR N = 0 TO 4: READ Y(N): NEXT : DATA 6
  5,77,89,107,128
110 TEXT : HOME : VTAB 10: HTAB 8: PRINT "**
  SOUND DEMONSTRATION**": VTAB 22: PRINT "
  ** COPYRIGHT 1984 BY MICROSPARC, INC. **
  "
120 PRINT DS;"BLOAD SOUNDS"
130 FOR N = 0 TO 3000: NEXT : CALL 768,6,1,1
  ,30
140 CALL 768,13,5,2,13: CALL 768,14,5,1,13: FOR
  N = 1 TO 450: NEXT : CALL 768,13,5,1,13
150 CALL 768,9,3,40,4: GOSUB 210
160 CALL 768,9,3,15,4: GOSUB 210: GOSUB 210
170 FOR N = 1 TO 1500: NEXT
180 CALL 768,11,2,1,30: FOR N = 1 TO 500: NEXT
190 FOR N = 1 TO 10: CALL 768,7,1,1,1: X = RND
  (1) * 440: FOR Q = 1 TO X: NEXT Q: NEXT
  N
200 GOTO 220
210 CALL 768,8,2,1,8: FOR N = 1 TO 9: CALL 7
  68,10,N,1,1: POKE - 16299,0: CALL 768,1
  0,N,1,1: POKE - 16300,0: CALL 768,10,N,
  1,1: NEXT : RETURN
220 HOME : VTAB 5: PRINT "THE PURPOSE OF THI
  S PROGRAM IS TO SHOW"
230 PRINT "SOME OF THE CAPABILITIES OF THE S
  OUND"
240 PRINT "SYNTHESIZER. ALL THE SOUNDS YOU
  HAVE"

```

```

250 PRINT "JUST HEARD WERE CREATED WITH THE
  SOUND"
260 PRINT "SYNTHESIZER, AND ARE SAVED ON THI
  S DISK"
270 PRINT "UNDER THE FILE NAME 'SOUNDS'. YO
  U WILL"
280 PRINT "BE SHOWN SOME USEFUL TECHNIQUES T
  O AID"
290 PRINT "YOU IN DESIGNING YOUR OWN SOUNDS.
  FOR"
300 PRINT "EACH SOUND PLAYED, YOU WILL BE SH
  OWN"
310 PRINT "WHAT THE SOUND LOOKS LIKE, AND TH
  E"
320 PRINT "COMMAND OR COMMANDS USED TO INVOK
  E IT"
330 PRINT "FROM AN APPLESOFT PROGRAM.": GOSUB
  1280
340 X = 1: GOSUB 1300
350 PRINT "EVEN THE SIMPLEST PATTERNS CAN GI
  VE"
360 PRINT "INTERESTING EFFECTS WHEN REPEATED
  "
370 PRINT "RAPIDLY.": P = 1: R = 10: L = 7: GOSUB
  1240
380 P = 4: R = 10: L = 4: GOSUB 1250
390 P = 9: R = 10: L = 2: GOSUB 1250
400 PRINT "OF COURSE GREATER COMPLEXITY GIVE
  S MORE"
410 PRINT "POSSIBILITIES.": X = 2: GOSUB 1300
420 P = 1: R = 10: L = 1: GOSUB 1240
430 P = 9: R = 7: L = 1: GOSUB 1250
440 X = 3: GOSUB 1300
450 P = 2: R = 15: L = 1: GOSUB 1240
460 P = 1: R = 20: L = 1: GOSUB 1250
470 X = 4: GOSUB 1300: P = 4: R = 8: L = 20: GOSUB
  1240
480 X = 5: GOSUB 1300: P = 1: R = 1: L = 50: GOSUB
  1240
490 X = 6: GOSUB 1300: P = 1: R = 1: L = 30: GOSUB
  1240
500 PRINT "A SIMPLE PATTERN CAN ALSO SOUND L
  IKE A"
510 PRINT "BIRD.": X = 7: GOSUB 1300: P = 1: R =
  10: L = 1: GOSUB 1240
520 PRINT "IF YOU WANT IT TO SOUND MORE NATU
  RAL."
530 PRINT "INTRODUCE RANDOM PAUSES LIKE THIS
  ": GOSUB 1280
540 PRINT "590 FOR N = 1 TO 10: CALL 768,7,
  "
550 HTAB 6: PRINT "1,1,1: X = RND (1) * 500:
  FOR"
560 HTAB 6: PRINT "Q = 1 TO X: NEXT Q: NEXT
  N"
570 FOR N = 1 TO 10: CALL 768,7,1,1,1: X = RND
  (1) * 440: FOR Q = 1 TO X: NEXT Q: NEXT
  N
580 GOSUB 1280: PRINT "SOMETIMES MOVING THE
  SOUND UP OR DOWN ON"
590 PRINT "THE SCREEN CAN MAKE A SMALL BUT"
600 PRINT "IMPORTANT DIFFERENCE IN THE SOUND
  ": GOSUB 1280
610 PRINT "SO, FOR THE INCOMING SHELL YOU HE
  ARD"
620 PRINT "EARLIER, I USED THIS.": X = 8: GOSUB
  1300
630 P = 2: R = 1: L = 8: GOSUB 1240
640 PRINT "FOR SOME SOUNDS, IT IS LESS OBVIO
  US HOW"
650 PRINT "TO LAY THEM OUT. TRIAL AND ERROR
  TAKES"
660 PRINT "ON A GREAT DEAL OF IMPORTANCE.": GOSUB
  1280
670 PRINT "FOR INSTANCE, WOULD YOU HAVE GUES
  SED"
680 PRINT "THE PATTERN FOR A MACHINE GUN COU
  LD LOOK"
690 PRINT "LIKE THIS?": X = 9: GOSUB 1300
700 P = 3: R = 40: L = 4: GOSUB 1240
710 PRINT "OTHER SOUNDS CAN BE BASICALLY SIM
  PLE."
720 PRINT "BUT REQUIRE INOVATION IN THE PLAY
  ING IN"
730 PRINT "ORDER TO MAKE THEM SOUND RIGHT.":
  GOSUB 1280
740 X = 10: GOSUB 1300: PRINT "TAKE THIS FOR
  INSTANCE."
750 PRINT "ORDINARY TECHNIQUES DON'T YIELD A
  NYTHING"
760 PRINT "PARTICULARLY INTERESTING.": P = 5:
  R = 3: L = 20: GOSUB 1240
770 P = 9: R = 30: L = 1: GOSUB 1250
780 P = 2: R = 20: L = 3: GOSUB 1250
790 PRINT "BUT WHAT IF YOU MADE A ROUTINE IN
  YOUR"

```

```

800 PRINT "APPLESOFT PROGRAM THAT PLAYED IT
AT EACH";
810 PRINT "OF THE PITCHES, 1 TO 9, CONSECUTI
VELY?": GOSUB 1280
820 PRINT "A LINE LIKE THIS WOULD ACCOMPLISH
IT."
830 PRINT "890 FOR N = 3 TO 27:X = INT (N
/"
840 HTAB 6: PRINT "3): CALL 768,10,X,1,1: NE
XT"
850 GOSUB 1280
860 PRINT "AND WHEN EXECUTED WOULD SOUND LIK
E THIS;": GOSUB 1280
870 FOR N = 3 TO 27:X = INT (N / 3): CALL 7
68,10,X,1,1: NEXT
880 GOSUB 1280: PRINT "OR ADD ANOTHER SOUND
AND SOME FLASHES."
890 PRINT "AND YOU GET THIS.": GOSUB 1280: GOSUB
210: GOSUB 1280
900 PRINT "SIMPLE MUSIC CAN BE REPRODUCED BY
"
910 PRINT "ENTERING IT NOTE BY NOTE FROM SHE
ET"
920 PRINT "MUSIC. HERE IS AN EXAMPLE.": X =
11: GOSUB 1300: GOSUB 1230
930 P = 2: R = 1: L = 30: GOSUB 1240
940 PRINT "HERE IS A TUNE THAT IS HIGHER ON
THE"
950 PRINT "SCALE.": X = 12: GOSUB 1300: GOSUB
1230
960 P = 1: R = 1: L = 85: GOSUB 1240
970 PRINT "NOTE THAT FOR THE SAKE OF CLARITY
"
980 PRINT "SHARPS AND FLATS ARE NOT INCLUDED
IN"
990 PRINT "MODE TWO.": GOSUB 1280
1000 PRINT "THIS IS BECAUSE THESE WOULD NOT
BE"
1010 PRINT "EASILY DISTINGUISHABLE ON THE MU
SICAL"
1020 PRINT "STAFF PROVIDED.": GOSUB 1280
1030 PRINT "HOWEVER, YOU MAY WISH TO ATTEMPT
TO ADD"
1040 PRINT "SHARPS OR FLATS IN MODE ONE, BUT
YOU"
1050 PRINT "MUST DO IT BY EAR.": GOSUB 1280
1060 PRINT "IF YOU HAVE A LONG OR REPETITIOU
S TUNE."
1070 PRINT "YOU MAY WANT TO PUT IT ON TWO OR
MORE"
1080 PRINT "SCREENS. LIKE THIS:": X = 13: GOSUB
1300: GOSUB 1230
1090 P = 5: R = 1: L = 13: GOSUB 1240
1100 PRINT "NOTE THAT THE AVERAGE LENGTH OF
A NOTE"
1110 PRINT "LINE IS A MATTER OF CHOICE, HERE
, LONGER";
1120 PRINT "NOTE LINES ARE USED.": GOSUB 128
0
1130 PRINT "HERE IS THE REST OF THE 'MARINE
HYMN'"
1140 PRINT "AND WHAT IT SOUNDS LIKE PLAYED A
LONE.": X = 14: GOSUB 1300: GOSUB 1230
1150 P = 5: R = 1: L = 13: GOSUB 1240: PRINT "N
OTE THAT LEDGER LINES ARE PROVIDED WHEN"
;
1160 PRINT "YOU ENTER NOTES HIGH ON THE SCAL
E."
1170 HCOLOR= 1: HPLLOT 10,48 TO 269,48: HPLLOT
10,54 TO 269,54: GOSUB 1280
1180 PRINT "USING THESE TECHNIQUES, AND ANY
OF YOUR"
1190 PRINT "OWN, A GREAT VARIETY OF SOUNDS C
AN BE"
1200 PRINT "CREATED."
1210 CALL 768,13,5,2,13: CALL 768,14,5,1,13:
FOR N = 1 TO 450: NEXT : CALL 768,13,5,
1,13
1220 : GOSUB 1280: TEXT : HOME : END
1230 HCOLOR= 1: FOR N = 0 TO 4: HPLLOT 10,Y(N
) TO 269,Y(N): NEXT : HCOLOR= 3: RETURN
1240 GOSUB 1280: GOTO 1260
1250 HOME : VTAB 21
1260 HTAB 10: PRINT "CALL 768,"X","P","R","L
": CALL 768,X,P,R,L: GOSUB 1280: RETURN
1270 END
1280 VTAB 24: HTAB 9: PRINT "<<RETURN TO CON
TINUE>> ": GET AS$: IF ASC (AS$) < > 13
THEN 1280
1290 HOME : VTAB 21: RETURN
1300 HGR : HCOLOR= 3: HPLLOT 0,0 TO 279,0 TO
279,159 TO 0,159 TO 0,0
1310 SNUM = X: TNUM = PEEK (STADD + SNUM * 25
6 - 256)
1320 FOR N = 1 TO TNUM: CX = Z + N: Q = PEEK
(STADD + SNUM * 256 - 256 + N): CY = Z +
Q / 2: IF Q > 0 THEN HPLLOT CX,CY
1330 NEXT : RETURN

```

KEY PERFECT 4.0

RUN ON
SOUND DEMO

```

-----
CODE          LINE# - LINE#
-----
83EB          10   -   100
9E7A          110  -   200
CD12          210  -   300
999D          310  -   400
7A6C          410  -   500
B661          510  -   600
A31A          610  -   700
AAEF          710  -   800
AEE7          810  -   900
91EA          910  -  1000
A21E         1010 -  1100
B4C9         1110 -  1200
8A9B         1210 -  1300
36C7         1310 -  1330
PROGRAM CHECK IS : 1106
    
```

CHECK CODE 3.0

ON: SOUND DEMO
TYPE: A

LENGTH: 103A
CHECKSUM: C1

LISTING 4: SOUNDS

6400-	3E	01	01	01	03	03	03	05	6650-	49	49	49	49	49	00	74	90	6768-	00	00	00	00	00	00	00	00
6408-	05	07	07	09	09	0B	0D	0F	6658-	AB	D5	FE	D5	AB	90	74	90	6770-	00	00	00	00	00	00	00	00
6410-	11	15	17	19	1B	1F	21	25	6660-	AB	D5	FE	D5	AB	90	74	90	6778-	00	00	00	00	00	00	00	00
6418-	29	2B	2F	33	37	39	3D	41	6668-	AB	D5	FE	D5	AB	90	74	90	6780-	00	00	00	00	00	00	00	00
6420-	45	49	4B	4F	51	55	59	5B	6670-	AB	D5	FE	D5	AB	90	74	90	6788-	00	00	00	00	00	00	00	00
6428-	5F	63	67	6B	6F	71	75	79	6678-	AB	D5	FE	D5	AB	90	74	90	6790-	00	00	00	00	00	00	00	00
6430-	7F	85	89	8D	91	95	9B	A1	6680-	AB	D5	FE	D5	AB	90	74	90	6798-	00	00	00	00	00	00	00	00
6438-	A7	AD	B9	C5	D5	E9	FF	00	6688-	00	00	00	00	00	00	00	00	67A0-	00	00	00	00	00	00	00	00
6440-	00	00	00	00	00	00	00	00	6690-	00	00	00	00	00	00	00	00	67A8-	00	00	00	00	00	00	00	00
6448-	00	00	00	00	00	00	00	00	6698-	00	00	00	00	00	00	00	00	67B0-	00	00	00	00	00	00	00	00
6450-	00	00	00	00	00	00	00	00	66A0-	00	00	00	00	00	00	00	00	67B8-	00	00	00	00	00	00	00	00
6458-	00	00	00	00	00	00	00	00	66A8-	00	00	00	00	00	00	00	00	67C0-	00	00	00	00	00	00	00	00
6460-	00	00	00	00	00	00	00	00	66B0-	00	00	00	00	00	00	00	00	67C8-	00	00	00	00	00	00	00	00
6468-	00	00	00	00	00	00	00	00	66B8-	00	00	00	00	00	00	00	00	67D0-	00	00	00	00	00	00	00	00
6470-	00	00	00	00	00	00	00	00	66C0-	00	00	00	00	00	00	00	00	67D8-	00	00	00	00	00	00	00	00
6478-	00	00	00	00	00	00	00	00	66C8-	00	00	00	00	00	00	00	00	67E0-	00	00	00	00	00	00	00	00
6480-	00	00	00	00	00	00	00	00	66D0-	00	00	00	00	00	00	00	00	67E8-	00	00	00	00	00	00	00	00
6488-	00	00	00	00	00	00	00	00	66D8-	00	00	00	00	00	00	00	00	67F0-	00	00	00	00	00	00	00	00
6490-	00	00	00	00	00	00	00	00	66E0-	00	00	00	00	00	00	00	00	67F8-	00	00	00	00	00	00	00	00
6498-	00	00	00	00	00	00	00	00	66E8-	00	00	00	00	00	00	00	00	6800-	FE	01	FF	03	F6	0C	ED	13
64A0-	00	00	00	00	00	00	00	00	66F0-	00	00	00	00	00	00	00	00	6808-	E3	18	D8	26	CC	2C	CO	38
64A8-	00	00	00	00	00	00	00	00	66F8-	00	00	00	00	00	00	00	00	6810-	A9	41	99	48	8B	55	86	59
64B0-	00	00	00	00	00	00	00	00	6700-	21	2D	FF	2D	F7	2D	EE	2D	6818-	83	5A	80	5A	85	55	8C	4E
64B8-	00	00	00	00	00	00	00	00	6708-	E5	2D	DD	2C	D3	2D	CB	2C	6820-	94	43	9F	30	B1	25	BB	15
64C0-	00	00	00	00	00	00	00	00	6710-	C4	2D	C1	19	C3	19	C8	18	6828-	C8	06	CE	02	CF	04	CE	08
64C8-	00	00	00	00	00	00	00	00	6718-	D1	19	DB	18	E4	19	EE	18	6830-	D1	0F	D9	1C	E4	24	EC	2A
64D0-	00	00	00	00	00	00	00	00	6720-	FB	18	00	00	00	00	00	00	6838-	EF	33	EC	3C	E7	49	DA	53
64D8-	00	00	00	00	00	00	00	00	6728-	00	00	00	00	00	00	00	00	6840-	CE	60	BF	6C	AC	79	9C	88
64E0-	00	00	00	00	00	00	00	00	6730-	00	00	00	00	00	00	00	00	6848-	91	7A	9B	6B	A5	60	B6	54
64E8-	00	00	00	00	00	00	00	00	6738-	00	00	00	00	00	00	00	00	6850-	C1	46	CF	3A	D3	36	D3	38
64F0-	00	00	00	00	00	00	00	00	6740-	00	00	00	00	00	00	00	00	6858-	D0	39	D5	32	D8	2B	AB	64
64F8-	00	00	00	00	00	00	00	00	6748-	00	00	00	00	00	00	00	00	6860-	A4	6D	98	6C	8C	6A	8C	6C
6500-	FE	C2	CA	D0	D8	E1	EA	F4	6750-	00	00	00	00	00	00	00	00	6868-	8D	6D	8B	6A	87	64	7E	5D
6508-	FD	AF	B6	BE	C3	CB	D3	D9	6758-	00	00	00	00	00	00	00	00	6870-	78	52	70	47	68	3D	64	39
6510-	E1	E8	EF	9E	A4	A7	AC	B0	6760-	00	00	00	00	00	00	00	00	6878-	66	3E	6E	42	76	43	80	3F
6518-	B3	B9	BD	C1	CA	D0	D5	DE										6880-	89	35	95	34	A0	36	B2	35

69A8- 00 00 01 2E 1A 1C 1F 1E
69B0- 23 23 25 28 20 1E 22 1D
69B8- 1C 1F 1B 00 3B 3A 3B 3E
69C0- 3B 37 36 32 32 2E 2C 27
69C8- 22 28 2E 22 1B 1B 1B 25
69D0- 2E 35 38 38 00 00 00 00
69D8- 00 00 00 00 00 00 00 00
69E0- 00 00 00 00 00 00 00 00
69E8- 00 00 00 00 00 00 00 00
69F0- 00 00 00 00 00 00 00 00
69F8- 00 00 00 00 00 00 00 00
6A00- D3 1C 1C 1C 1C 1C 1C 1C
6A08- 1C 1B 1B 1B 1B 1B 1B 1B
6A10- 1C 1C 1C 1C 1C 1D 1D 1D
6A18- 1D 1D 1D 1D 1D 1D 1D 1D
6A20- 1D 1D 1D 1D 1D 1D 1D 1D
6A28- 1F 1F 1F 1F 1F 1F 1F 1F
6A30- 1F 1F 1F 1F 1F 1F 1F 1F
6A38- 1F 1F 1F 21 21 21 21 21
6A40- 21 21 21 21 21 21 21 21
6A48- 21 21 21 21 24 24 24 24
6A50- 24 24 24 24 24 24 24 24
6A58- 24 24 24 24 26 26 25 25
6A60- 25 25 25 25 25 25 25 25
6A68- 25 25 26 28 28 28 28 28
6A70- 28 28 28 28 28 28 28 28
6A78- 28 29 29 29 29 29 29 29
6A80- 29 29 29 29 29 29 2B 2B
6A88- 2C 2B 2B 2B 2C 2C 2B 2C
6A90- 2B 2B 2E 2E 2E 2E 2E 2E
6A98- 2D 2D 2D 2D 2E 30 30 2F
6AA0- 2F 2F 2F 30 2F 30 30 32
6AA8- 32 32 32 31 32 31 31 31
6AB0- 33 33 33 33 33 33 33 34
6AB8- 35 35 35 35 35 35 35 37
6AC0- 38 37 38 37 37 3A 39 39
6AC8- 39 3A 3C 3B 3C 3C 3E 3D
6AD0- 3E 40 3F 42 00 00 00 00
6AD8- 00 00 00 00 00 00 00 00
6AE0- 00 00 00 00 00 00 00 00
6AE8- 00 00 00 00 00 00 00 00
6AF0- 00 00 00 00 00 00 00 00
6AF8- 00 00 00 00 00 00 00 00
6B00- 97 5E 5E 5E 5E 5E 5E 5E
6B08- 5E 5E 5D 5D 5D 5F 5F 5F
6B10- 5F 5F 5F 5F 5F 5F 5F 62
6B18- 61 61 61 61 61 61 61 61
6B20- 64 64 63 63 63 63 63 63
6B28- 64 66 66 66 66 65 66 65
6B30- 65 68 68 67 68 68 67 68
6B38- 67 6A 6A 69 69 6A 6A 6A
6B40- 6B 6B 6B 6B 6C 6B 6B 6D
6B48- 6D 6D 6E 6D 6D 6D 70 6F
6B50- 70 6F 6F 6F 71 72 72 72
6B58- 72 72 73 74 73 74 74 74
6B60- 75 75 75 76 76 75 77 77
6B68- 77 77 77 77 79 79 79 79
6B70- 79 7C 7B 7B 7B 7B 7D 7D
6B78- 7E 7E 7E 80 7F 7F 7F 80
6B80- 82 81 82 81 84 83 83 83
6B88- 86 86 86 86 87 88 87 87
6B90- 8A 89 89 89 8B 8B 8C 8B
6B98- 00 00 00 00 00 00 00 00
6BA0- 00 00 00 00 00 00 00 00
6BA8- 00 00 00 00 00 00 00 00
6BB0- 00 00 00 00 00 00 00 00
6BB8- 00 00 00 00 00 00 00 00
6BC0- 00 00 00 00 00 00 00 00
6BC8- 00 00 00 00 00 00 00 00
6BD0- 00 00 00 00 00 00 00 00
6BD8- 00 00 00 00 00 00 00 00
6BE0- 00 00 00 00 00 00 00 00
6BE8- 00 00 00 00 00 00 00 00

6BF0- 00 00 00 00 00 00 00 00
6BF8- 00 00 00 00 00 00 00 00
6C00- 4D 01 04 01 03 02 06 02
6C08- 06 01 07 01 08 01 08 01
6C10- 08 01 07 01 09 01 0B 01
6C18- 0B 01 0E 01 0E 01 0D 01
6C20- 0F 01 10 01 12 01 11 01
6C28- 13 01 16 01 15 01 18 01
6C30- 17 01 19 01 19 01 1B 01
6C38- 1D 01 22 01 27 01 31 01
6C40- 39 01 46 01 57 01 6E 01
6C48- 95 01 C3 01 FF 01 00 00
6C50- 00 00 00 00 00 00 00 00
6C58- 00 00 00 00 00 00 00 00
6C60- 00 00 00 00 00 00 00 00
6C68- 00 00 00 00 00 00 00 00
6C70- 00 00 00 00 00 00 00 00
6C78- 00 00 00 00 00 00 00 00
6C80- 00 00 00 00 00 00 00 00
6C88- 00 00 00 00 00 00 00 00
6C90- 00 00 00 00 00 00 00 00
6C98- 00 00 00 00 00 00 00 00
6CA0- 00 00 00 00 00 00 00 00
6CA8- 00 00 00 00 00 00 00 00
6CB0- 00 00 00 00 00 00 00 00
6CB8- 00 00 00 00 00 00 00 00
6CC0- 00 00 00 00 00 00 00 00
6CC8- 00 00 00 00 00 00 00 00
6CD0- 00 00 00 00 00 00 00 00
6CD8- 00 00 00 00 00 00 00 00
6CE0- 00 00 00 00 00 00 00 00
6CE8- 00 00 00 00 00 00 00 00
6CF0- 00 00 00 00 00 00 00 00
6CF8- 00 00 00 00 00 00 00 00
6D00- 0D FF F2 E7 D5 C2 AB 95
6D08- 7B 5F 3A 19 01 00 00 00
6D10- 00 00 00 00 00 00 00 00
6D18- 00 00 00 00 00 00 00 00
6D20- 00 00 00 00 00 00 00 00
6D28- 00 00 00 00 00 00 00 00
6D30- 00 00 00 00 00 00 00 00
6D38- 00 00 00 00 00 00 00 00
6D40- 00 00 00 00 00 00 00 00
6D48- 00 00 00 00 00 00 00 00
6D50- 00 00 00 00 00 00 00 00
6D58- 00 00 00 00 00 00 00 00
6D60- 00 00 00 00 00 00 00 00
6D68- 00 00 00 00 00 00 00 00
6D70- 00 00 00 00 00 00 00 00
6D78- 00 00 00 00 00 00 00 00
6D80- 00 00 00 00 00 00 00 00
6D88- 00 00 00 00 00 00 00 00
6D90- 00 00 00 00 00 00 00 00
6D98- 00 00 00 00 00 00 00 00
6DA0- 00 00 00 00 00 00 00 00
6DA8- 00 00 00 00 00 00 00 00
6DB0- 00 00 00 00 00 00 00 00
6DB8- 00 00 00 00 00 00 00 00
6DC0- 00 00 00 00 00 00 00 00
6DC8- 00 00 00 00 00 00 00 00
6DD0- 00 00 00 00 00 00 00 00
6DD8- 00 00 00 00 00 00 00 00
6DE0- 00 00 00 00 00 00 00 00
6DE8- 00 00 00 00 00 00 00 00
6DF0- 00 00 00 00 00 00 00 00
6DF8- 00 00 00 00 00 00 00 00
6E00- FE FE FE FE FE FE FE FE
6E08- 00 C0 C0 C0 C0 C0 C0 C0
6E10- C0 00 00 00 00 00 00 00
6E18- 00 00 00 00 00 FE FE FE
6E20- 00 C0 C0 C0 C0 98 98
6E28- 98 98 98 98 98 98 98 98
6E30- 00 00 00 00 00 00 00 00

6E38- 00 00 00 00 FE FE FE 00
6E40- 00 C0 C0 C0 C0 98 98 98
6E48- 98 98 98 98 98 98 98 00
6E50- FE FE FE 00 C0 C0 C0 C0
6E58- 00 98 98 98 98 98 98 98
6E60- 98 98 98 00 FE FE FE 00
6E68- C0 C0 C0 C0 00 98 98 98
6E70- 98 98 98 98 98 98 98 00
6E78- 00 00 00 00 00 00 00 00
6E80- 00 00 00 00 C0 C0 C0 C0
6E88- 00 98 98 98 98 98 00 80
6E90- 80 80 80 80 80 80 80 80
6E98- 80 80 80 80 80 80 80 80
6EA0- 80 00 00 98 98 98 98 98
6EA8- 98 98 98 98 98 00 C0 C0
6EB0- C0 C0 C0 C0 C0 C0 00 FE
6EB8- FE FE FE FE FE 00 00 00
6EC0- 00 00 00 00 00 00 00 00
6EC8- 00 FE FE FE 00 FE FE FE
6ED0- 00 C0 C0 C0 C0 C0 C0 C0
6ED8- C0 00 00 C0 C0 C0 C0 00
6EE0- C0 C0 C0 C0 00 C0 C0 C0
6EE8- C0 C0 C0 C0 C0 00 00 C0
6EF0- C0 C0 C0 C0 C0 C0 C0 00
6EF8- 00 C0 C0 C0 C0 C0 C0 00
6F00- FE 90 90 90 90 74 74 74
6F08- 74 60 60 60 60 60 60 60
6F10- 60 60 60 56 56 56 56 56
6F18- 56 60 60 60 60 60 74 74
6F20- 74 74 90 90 90 90 80 80
6F28- 80 80 74 74 74 74 00 74
6F30- 74 74 74 80 80 80 80 90
6F38- 90 90 90 80 80 80 80 80
6F40- 80 00 00 90 90 90 90 74
6F48- 74 74 74 60 60 60 60 00
6F50- 60 60 60 60 60 56 56 56
6F58- 56 56 56 60 60 60 60 60
6F60- 74 74 74 74 90 90 90 90
6F68- 80 80 80 80 74 74 74 74
6F70- 00 74 74 74 74 80 80 80
6F78- 80 00 80 80 80 80 90 90
6F80- 90 90 90 90 90 6C 6C
6F88- 6C 6C 6C 6C 6C 6C 6C
6F90- 6C 6C 6C 6C 6C 56 56 56
6F98- 56 00 56 56 56 56 56 56
6FA0- 56 56 60 60 60 60 00 60
6FA8- 60 60 60 74 74 74 90 90
6FB0- 90 90 90 80 80 80 80 80
6FB8- 80 80 00 90 90 90 90 90
6FC0- 74 74 74 74 60 60 60 60
6FC8- 00 60 60 60 60 60 56 56
6FD0- 56 56 56 56 60 60 60 60
6FD8- 60 74 74 74 74 90 90 90
6FE0- 90 80 80 80 80 74 74 74
6FE8- 74 00 74 74 74 74 80 80
6FF0- 80 80 00 80 80 80 80 90
6FF8- 90 90 90 90 90 90 00 00
7000- F3 90 90 90 90 90 90 00
7008- 74 74 74 74 74 74 00 00
7010- 60 60 60 60 60 60 60 60
7018- 60 60 60 60 60 60 60 60
7020- 00 60 60 60 60 60 60 60
7028- 60 60 60 60 60 60 60 60
7030- 60 60 60 60 60 60 60 60
7038- 60 60 60 60 60 60 60 60
7040- 60 60 00 60 60 60 60 60
7048- 60 60 60 60 60 60 60 60
7050- 60 60 60 00 60 60 60 60
7058- 60 60 60 60 60 60 60 60
7060- 60 60 60 60 60 60 60 60
7068- 60 60 60 60 00 49 49 49
7070- 49 49 49 49 49 49 00 60
7078- 60 60 60 60 60 60 60 60

7080- 60 60 60 60 60 60 60 00
 7088- 74 74 74 74 74 74 74 00
 7090- 6C 6C 6C 6C 6C 6C 6C 00
 7098- 60 60 60 60 60 60 60 60
 70A0- 60 60 60 60 60 60 60 60
 70A8- 00 60 60 60 60 60 60 60
 70B0- 60 60 60 60 60 60 60 60
 70B8- 60 00 6C 6C 6C 6C 6C 6C
 70C0- 6C 6C 6C 6C 6C 6C 6C 6C
 70C8- 00 80 80 80 80 80 80 80
 70D0- 80 80 80 80 80 80 00 90
 70D8- 90 90 90 90 90 90 90 90
 70E0- 90 90 90 90 90 90 90 90
 70E8- 90 90 90 90 90 90 00 00
 70F0- 00 00 00 00 00 00 00 00
 70F8- 00 00 00 00 00 00 00 00
 7100- FF 49 49 49 49 49 49 49
 7108- 49 49 00 4D 4D 4D 4D 4D
 7110- 4D 4D 4D 4D 00 56 56 56
 7118- 56 56 56 56 56 56 56 56
 7120- 56 56 56 56 56 00 6C 6C
 7128- 6C 6C 6C 6C 6C 6C 6C 6C
 7130- 6C 6C 6C 6C 00 56 56 56
 7138- 56 56 56 56 56 56 56 56
 7140- 56 56 56 56 56 00 49 49
 7148- 49 49 49 49 49 49 49 49
 7150- 49 49 49 49 49 49 49 49
 7158- 00 60 60 60 60 60 60 60
 7160- 60 60 60 60 60 60 60 60
 7168- 60 60 60 60 60 60 60 60
 7170- 60 00 74 74 74 74 74 74
 7178- 74 00 60 60 60 60 60 60
 7180- 60 60 60 60 60 60 60 60
 7188- 60 60 00 49 49 49 49 49
 7190- 49 49 49 49 00 4D 4D 4D
 7198- 4D 4D 4D 4D 4D 00 56

71A0- 56 56 56 56 56 56 56 56
 71A8- 56 56 56 56 56 56 00
 71B0- 6C 6C 6C 6C 6C 6C 6C
 71B8- 6C 6C 6C 6C 6C 00 56
 71C0- 56 56 56 56 56 56 56
 71C8- 56 56 56 56 56 56 00
 71D0- 49 49 49 49 49 49 49
 71D8- 49 49 49 49 49 49 49
 71E0- 49 49 00 60 60 60 60 60
 71E8- 60 60 60 60 60 60 60
 71F0- 60 60 60 60 60 60 60
 71F8- 60 60 60 60 60 60 00
 7200- C6

KEY PERFECT 4.0
RUN ON
SOUNDS

CODE	ADDR#	-	ADDR#
1FDE	6400	-	644F
00	6450	-	649F
00	64A0	-	64EF
1D7C	64F0	-	653F
2957	6540	-	658F
23EE	6590	-	65DF
25AC	65E0	-	662F
28F9	6630	-	667F
2A5C	6680	-	66CF
0EDB	66D0	-	671F
29B4	6720	-	676F
00	6770	-	67BF
08B1	67C0	-	680F
28F5	6810	-	685F
2347	6860	-	68AF

2518 68B0 - 68FF
 2B6B 6900 - 694F
 2B60 6950 - 699F
 2B77 69A0 - 69EF
 1C2D 69F0 - 6A3F
 28AD 6A40 - 6A8F
 285C 6A90 - 6ADF
 1C0A 6AE0 - 6B2F
 2A64 6B30 - 6B7F
 2587 6B80 - 6BCF
 130C 6BD0 - 6C1F
 294A 6C20 - 6C6F
 00 6C70 - 6CBF
 069F 6CC0 - 6D0F
 00 6D10 - 6D5F
 00 6D60 - 6DAF
 00 6DB0 - 6DFF
 2B08 6E00 - 6E4F
 2143 6E50 - 6E9F
 23BD 6EA0 - 6EEF
 240F 6EF0 - 6F3F
 23C2 6F40 - 6F8F
 2096 6F90 - 6FDF
 2719 6FE0 - 702F
 28CF 7030 - 707F
 285E 7080 - 70CF
 2EA5 70D0 - 711F
 25CF 7120 - 716F
 24C1 7170 - 71BF
 2189 71C0 - 71FF
 PROGRAM CHECK IS : 0E00

CHECK CODE 3.0

ON: SOUNDS
TYPE: B

LENGTH: 0E00
CHECKSUM: 93