



Tech Info Library

Macintosh: Sound Generator Demonstration (2 of 4)

Revised: 10/23/84
Security: Everyone

Macintosh: Sound Generator Demonstration (2 of 4)

```
=====

3420 REM
4000 REM  Build the note translation table
4010 REM
4020 PRINT: PRINT "    Building translation tables....."
4030 DATA "CDEFGAB012345678#%"
4040 DATA 0,2,4,5,7,9,11
4050 DATA -9,3,15,27,39,51,63,75,87
4060 DATA +1,-1
4070 RESTORE 4000
4080 READ NOTENAMES$
4090 DIM NOTEINDEX(LEN(NOTENAMES$))
4100 FOR I=1 TO LEN(NOTENAMES$)
4110 READ NOTEINDEX(I)
4120 NEXT I
4130 REM
4140 REM  Build duration table
4150 DURNAME$="SEQHW"
4160 DIM DURVALUE(LEN(DURNAME$))
4170 DURVALUE(1)=1
4180 FOR I=2 TO LEN(DURNAME$)
4190 DURVALUE(I)=DURVALUE(I-1)*2
4200 NEXT I
4210 REM
5000 REM  Compile the music into duration and tone lists
5010 REM
5020 PRINT: PRINT "Compiling music....."
5030 RESTORE 9000
5040 READ TEMPO
5050 TICKSPER16TH=900/TEMPO
5060 LISTPTR=0
5070 DURLIST(LISTPTR)=0  ' zero will terminate list
5080 READ DURATION$:
    IF DURATION$="X" THEN GOTO 6000
5090 DURLIST(LISTPTR)=DURVALUE(INSTR(DURNAME$,DURATION$))
5091 DURLIST(LISTPTR)=DURLIST(LISTPTR)*TICKSPER16TH
5100 FOR I=0 TO 3  ' loop to process four tones
```

```
5110 READ NOTE$: IF NOTE$<>"R" THEN GOTO 5140
5120 LPOKE!=FNCODE!(LPOKE)
5130 CALL LPOKE!(VARPTR(TONELIST(0,I,LISTPTR)), 0!): GOTO 5200
5140 INDEX=0
5150 FOR J=1 TO LEN(NOTE$)
5160 INDEX=INDEX+NOTEINDEX(INSTR(NOTENAMES$,MID$(NOTE$,J,1)))
5170 NEXT J
5180 LPEEK!=FNCODE!(LPEEK)
5190 CALL LPEEK!(VARPTR(TONELIST(0,I,LISTPTR)),VARPTR(NOTERATE(0,INDEX)))
5200 NEXT I
5210 LISTPTR=LISTPTR+1
5220 GOTO 5070
5230 REM
6000 REM Set up four-tone sound data structures
6010 REM
6020 PRINT: PRINT " Setting up sound data structures....."
6030 REM
6040 FOR BUFNUM=0 TO 1
6050 REM Set up four-tone sound Write parameter list
6060 LPOKE!=FNCODE!(LPOKE): LPEEK!=FNCODE!(LPEEK):
        IOWRITE!=FNCODE!(IOWRITE)
6070 REM IMPORTANT!NO NEW VARIABLES MAY BE USED AFTER THIS POINT
6080 REM
6090 FOR I=0 TO 49\2: PARAMLIST(I, BUFNUM)=0: NEXT I
6100 PARAMLIST(24\2, BUFNUM)=-4 ' Sound reference number
6110 CALL LPOKE!(VARPTR(PARAMLIST(0, BUFNUM))+32,VARPTR(SYNTHREC(0, BUFNUM)))
6120 CALL LPOKE!(VARPTR(PARAMLIST(0, BUFNUM))+36, 6!)
6130 REM
6140 REM Set up four-tone synthesizer record
6150 SYNTHREC(0, BUFNUM)=1
6151 REM ' Positive number (>0) indicates four-tone
6160 CALL LPOKE!(VARPTR(SYNTHREC(1, BUFNUM)),VARPTR(SOUNDREC(0,BUFNUM)))
6170 REM
6180 REM Set up four-tone sound record
6190 FOR I=0 TO 24: SOUNDREC(I, BUFNUM)=0: NEXT I
6200 CALL LPOKE!(VARPTR(SOUNDREC(17, BUFNUM)),VARPTR(WAVETHIRD(0)))
6210 CALL LPOKE!(VARPTR(SOUNDREC(19, BUFNUM)),VARPTR(WAVETRIANG(0)))
6220 CALL LPOKE!(VARPTR(SOUNDREC(21, BUFNUM)),VARPTR(WAVETRIANG(0)))
6230 CALL LPOKE!(VARPTR(SOUNDREC(23,BUFNUM)),VARPTR(WAVETRIANG(0)))
6240 NEXT BUFNUM
6250 REM
7000 REM Play the music from the lists
7010 REM
7020 IOWRITEASYNC!=FNCODE!(IOWRITEASYNC)
7030 PRINT: PRINT "Playing music."
7040 LISTPTR=0: BUFNUM=0
7050 WHILE DURLIST(LISTPTR)<>0: BUFNUM=1-BUFNUM
7060 WHILE PARAMLIST(16\2, BUFNUM)<>0: WEND
7061 REM ' Wait for next buffer available
7070 SOUNDREC(0, BUFNUM)=DURLIST(LISTPTR)
7080 FOR I=0 TO 3
7090 CALL LPEEK!(VARPTR(SOUNDREC(1+I*4,BUFNUM)),VARPTR(TONELIST(0,I,LISTPTR)))
7100 NEXT I
```

```
7110 CALL IOWRITEASYNC!(VARPTR(PARAMLIST(0, BUFNUM)))
7120 LISTPTR=LISTPTR+1: WEND
7130 WHILE PARAMLIST(16\2, BUFNUM)<>0: WEND
7131 REM ' Wait for final note to end
7140 REM
7150 PRINT: PRINT "Press a key to play again..."
7160 WHILE INKEY$="": WEND
7170 GOTO 7000
```

Apple Technical Communications

Tech Info Library Article Number:304