The machine code can be entered into your machine by typing in the source listing provided and assembling it to create a hex file that can be loaded whenever required. The look-up tables can be generated by running the following program:

```
900 REM **** CBM SOUND CALLING PROGRAM ****
910 :
915 DN=8:REM FOR CASSETTE DN=1
920 IFA=0 THENA=1:LOAD"SSOUND.HEX",DN,1
999 :
1000 REM **** SET UP DATA VALUES ****
1005 S=80        :REM NUMBER OF STEPS
1007 TB=12*4096 :REM START OF DATA AREA
1008 :
1010 REM **SINE WAVE **
1020 FOR I=0 TO S-1
1030 Y=127*SIN(X)+127
1040 POKE TB+I,Y
1045 X=X+2/S
1050 NEXT I
1060 :
1065 REM **** SAW WAVE ****
1070 Y=255:TB=TB+S
1080 FOR I=0 TO S-1
1090 POKE TB+I,Y
1100 Y=Y-255/S
1110 NEXT I
1120 :
1125 REM **** SQUARE WAVE ****
1130 Y=255:TB=TB+S
1140 FOR I=0 TO S/2-1
1150 POKE TB+I,Y
1160 NEXT I
1165 Y=0
1170 FOR I=S/2 TO S-1
1180 POKE TB+I,Y
1190 NEXT I
1999 :
2000 REM **** DISPLAY DATA TABLES ****
2005 TB=12*4096
2010 FORI=TB TO TB+3*S-1
2020 PRINTI,I-TB ,PEEK(I)
2030 NEXT
```

After running this program, type NEW and then type this sample program that demonstrates how to use the machine code, giving the SYS and POKE addresses required to interact with the machine code from BASIC. This program asks the user to enter the wave type required and then produces a tone each time a key is pressed.

```
10 REM ****  CBM 64 SOUND  ****
20 REM **** SAMPLE PROGRAM ****
30 :
40 DDR=56579:POKE DDR,255: REM ALL OUTPUT
65 CL=49392 :REM COUNTER LOBYTE LOCATION
67 TL=49412 :REM TYPE LOBYTE LOCATION
70 SOUND=49196:REM PROGRAM START ADDRESS
75 REM ** SET COUNTER VALUE **
80 NUM=80:NHI=INT(NUM/256):NLO=NUM-256*NHI
82 POKE CL,NLO:POKE CL+1,NHI
83 :
85 PRINTCHR$(147):REM CLEAR SCREEN
86 INPUT"WAVE TYPE (0)SINE (1)SAW (2)SQUARE";WT
87 POKE TL,WT*S
88 PRINT:PRINT"PRESS ANY KEY (RUN/STOP TO END)"
90 GETA$:IFA$=""THEN90 :REM WAIT FOR KEY
100 SYS SOUND:REM CALL MACHINE CODE
110 IF A$="X" THEN 85
120 GOTO 90
```

If you do not have an assembler or do not understand Assembly language then you can still use the machine code program by typing in this BASIC loader and running it. In this case, you can omit line 920 from the program that sets up the look-up table.

```
10 REM **** BASIC LOADER FOR CBM SOUND ****
20 REM ****       MACHINE CODE          ****
30 FOR I=49392 TO 49449
40 READ A:POKE I,A
50 CC=CC+A
60 NEXT I
70 READ CS:IF CC<>CS THEN PRINT"CHECKSUM ERROR":END
100 DATA120,173,240,192,141,242,192
110 DATA173,241,192,141,243,192,162,0
120 DATA189,0,192,141,1,221,232,224,80
130 DATA208,245,173,242,192,56,233,1
140 DATA141,242,192,173,243,192,233,0
150 DATA141,243,192,208,224,169,0,205
160 DATA242,192,208,217,88,96
170 DATA9115:REM*CHECKSUM*
```

FOR THE BBC

As the BBC has its own built-in assembler, the process of combining BASIC with machine code is substantially easier than on the Commodore 64.

```
  5 REM **** BBC SOUND PROGRAM ****
  8 MODE 7
 10 HIMEM=HIMEM-&0301
 20 MC%=HIMEM+1
 30 DDR=&FE62: ? DDR=255:REM ALL OUTPUT
 40 port=&FE60:REM USER PORT DATA REG
 50 steps=80  :REM NO. OF STEPS IN A WAVE
 60 table_start=MC%
 70 PROCset_up_tables
 80 PROCmachine_code
 90 PROCsample_program
999 END
1000 DEF PROCmachine_code
1010 :       -
1020 FOR opt%=1 TO 3 STEP 3
1030 P%=MC%
1060 sine=P%:   P%=P%+steps
1070 saw=P%:    P%=P%+steps
1080 square=P%:P%=P%+steps
1090 number=P%:P%=P%+2
1100 count=P%: P%=P%+2
1110 [
1120 OPT opt%
1130 \**** MAIN PROGRAM STARTS HERE ****
1150 .sound
1160        SEI
1170        LDA number
1180        STA count
1190        LDA number+1
1200        STA count+1
1220  .loop2
1230        LDX #&00
1240 .loop1
1250        LDA sine,X
1260        STA port
1270        INX
1280        CPX #steps
1290        BNE loop1
1310 \**** DECREMENT COUNT ****
1320 \
1330        LDA count
1340        SEC
1350        SBC #&01
1360        STA count
1370        LDA count+1
1380        SBC #&00
1390        STA count+1
1400        BNE loop2
1410        LDA #&00
1420        CMP count
1430        BNE loop2
1440        CLI
1450        RTS
1455 ]
1460 NEXT opt%
1480 ENDPROC
2000 DEF PROCset_up_tables
2020 REM **** SINE WAVE ****
2025 x=0
2030 FOR I=0 TO steps-1
2040 y=127*SIN(x)+127
2050 ?(table_start+I)=y
2060 x=x+2*PI/steps
2070 NEXT I
2090 REM **** SAW WAVE ****
2100 y=255:table_start=table_start+steps
2110 FOR I=0 TO steps-1
2120 ?(table_start+I)=y
2130 y=y-255/steps
2140 NEXT I
2160 REM **** SQUARE WAVE ****
2170 y=255:table_start=table_start+steps
2180 FOR I=0 TO steps/2-1
2190 ?(table_start+I)=y
2200 NEXT I
2220 y=0
2230 FOR I=steps/2 TO steps-1
2240 ?(table_start+I)=y
2250 NEXT I
2270 REM **** DISPLAY DATA TABLES ****
2280 table_start=MC%
2290 FOR I=table_start TO table_start+3*steps-1
2300 PRINT ~I,~(I-table_start),? I
2310 NEXT I
2330 ENDPROC
3000 DEF PROCsample_program
3020 counter=MC%+3*steps: REM COUNTER LOBYTE LOCATION
3030 type=loop1+1: REM TYPE LOBYTE LOCATION
3040 count_value=80
3050 count_hi=count_value DIV 256
3060 count_lo=count_value MOD 256
3070 ?counter=count_lo
3080 counter?1=count_hi
3090 CLS
3100 INPUT"WAVE TYPE (0) SINE (1) SAW (2) SQUARE":wave
3110 ?type=wave*steps
3120 REPEAT
3125 PRINT"PRESS ANY KEY (X TO EXIT)"
3130 A$=GET$
3140 CALL sound
3150 UNTIL A$="X"
3160 GOTO 3090
```