NIBBLE DUET

DOS 3.3 ○ 0

ProDOS ○ 0

Tired of your Apple's sound? Add two-voice sound to your programs with this short machine language routine. All you need is three POKEs and a CALL to sound each note.

by Douglas W. Jefferys, 7 Danbury St., Dundas, Ontario, Canada L9H 4P6

Jazz up your programs with two-voice sound! Nibble Duet synthesizes two-voice sound on the Apple. It brings an added dimension to your programs — sophisticated sound effects to accompany your visual effects. A machine language driver creates the notes, and a demonstration program shows you how to vary them and add sounds to your programs.

USING THE PROGRAM

Each note produced by Nibble Duet requires three parameters: depth, length, and pitch. The greater the pitch value, the lower the resulting note. Pitch values that approximate two octaves of the musical scale are listed in **Table 1**. The depth and length parameters both affect the duration of the note. In addition, the duration of the note is affected by the pitch. With the length and depth values constant, the higher the pitch of a note, the shorter the note will sound. To achieve accurate musical rhythm, you must vary the length parameter from note to note. For most purposes, a depth value of 1 should be used.

Once you BLOAD NIBBLE.DUET (**Listing 1**), sounding a note is a simple matter of three POKEs and a CALL. POKE 768 with the depth value, 769 with the length, and 770 with the pitch. Then CALL 771. For instance, to create a C, you would enter the following:

```
POKE 768,1: POKE 769,100: POKE 770,120: CALL 771
```

A common method of playing a series of notes is illustrated in the section of DUET.DEMO (**Listing 2**) that plays "The Entertainer." The depth, length and pitch for each note are read from DATA statements. Note that to produce eighth notes, the length values vary from 50 to 70 to compensate for changes in pitch.

Nibble Duet creates the illusion of two notes played simultaneously by modulating the pitch to produce harmonic sound. Therefore, it is not possible to specify two pitch values at the same time.

ENTERING THE PROGRAM

First, key in the hex code in **Listing 1** and save it to disk with the command:

```
BSAVE NIBBLE DUET,A$300,L$3C
```

Next, type in the demonstration program (**Listing 2**) and save it with:

```
SAVE DUET.DEMO
```

MAKING SOUND

Any program that synthesizes sound on the Apple accesses the speaker soft switch at memory location $C030 (−16336 decimal). References to this location (via a PEEK, POKE or LDA $C030) toggle the speaker. Toggling the speaker at high speed produces tones, and the faster the toggling, the higher the frequency of the sound, and hence the higher the pitch.

Unfortunately, BASIC is so slow it produces just a dull buzz. To make tones, we need machine language speed. Just as the BASIC statement X=PEEK (−16336) in a loop makes sound, the statement LDA $C030 in an assembly language loop also makes sound, but the difference is speed. The buzz in BASIC becomes an ear-piercing beep. A delay of about one thousandth of a second or so between toggles changes the pitch. However, there is one drawback. The sounds are just one-voice beeps.

TABLE 1: Notes and POKE Values

Note	Octave 1 POKE Values*	Octave 2 POKE Values*
C	240	120
C#	228	113
D	215	105
D#	203	99
E	190	93
F	180	86
F#	170	82
G	160	78
G#	151	74
A	142	79
A#	134	66
B	125	62

*POKE these values into location 770 to set the pitch.

EXAMPLE 1: Pulse Pattern for True Two-Voice Sound

```
Sound 1:        #         #         #         #         #         #
Sound 2:        #    #    #    #    #    #    #    #    #    #    #    #

Speaker Output: #    #    ##   #    # #  #    #  # #    #    ##   #    #
```

EXAMPLE 2: Pulse Pattern Used by Nibble Duet

```
Sound: #    #    ##    #    #    ##    #    #    ##    #    #    ##
```

TWO-VOICE SOUND

The secret of two-voice sound is to blend the two sounds (or voices) for a harmonic effect. One way to do this is to play one sound on top of the other. Sounds can be mixed using special sound generator chips, such as those in the Commodore 64, the Macintosh and Atari computers.

Let's look at the two sounds depicted in **Example 1**. Each # sign represents a toggle of the speaker. Sound 1 is a rather low-pitched sound, indicated by the wider spacing, and sound 2 has a higher pitch. Together, they create true, two-voice harmony.

WHY NOT APPLE?

The Apple has no sound generator chip, so the only available technique is toggling the speaker from a machine language program. Rather than mix the two sounds as in **Example 1**, Nibble Duet puts in an extra toggle every so often, something like **Example 2**. In fact, this routine does not produce genuine two-voice sound. Instead, it modulates a second tone on top of the first, creating a very realistic illusion of two voices. The frequency of the second tone is completely dependent on the first.

HOW IT WORKS

Lines 26-34 of **Listing 1** store the values for depth, length, and pitch in memory locations $FA-$FF. These zero page locations are unused by DOS, Applesoft or ProDOS.

From there, the program toggles the speaker and loops between WAIT1 and WAIT2 to create a delay. (Two pitch locations, PITCHA and PITCHB, are used as counters. The third, ORIGPITCH, is used to store the original value of the pitch so it can be retrieved quickly using the zero page addressing mode.)

When PITCHB reaches zero, the branch ($31D) fails and execution continues at $31F. The original pitch, ORIGPITCH, is loaded into the Accumulator, and execution jumps to TOGGLE1, where the Accumulator's contents are put back into PITCHB. The speaker is toggled, PITCHB is decremented, and the program branches to WAIT2. Now PITCHB contains the pitch (minus 1), and PITCHA contains zero. When the branch at $326 fails, the speaker is toggled again ($328). This is the extra toggle that produces the second tone.

DEPTHZ is decremented, with the program going back to WAIT1 if DEPTHZ is not yet zero. The next step is to restore DEPTHZ from its cold storage location, ORIGDEPTH, and then decrement LENGTHZ and conditionally exit to Applesoft. This process takes enough time to put the toggles out of step, but since the toggles are themselves so quick, we hear the two sounds at the same time and so interpret them as harmony.

LISTING 1: NIBBLE.DUET

```
 0                         ;
 1                         **********************
 2                         *     NIBBLE DUET    *
 3                         *   BY DOUG JEFFERYS *
 4                         * COPYRIGHT (C) 1985 *
 5                         * BY MICROSPARC, INC *
 6                         * CONCORD, MA  01742 *
 7                         **********************
 8                         *
 9                         * MICROSPARC ASSEMBLER
10                         *
11                         ORIGPITCH EQU $FA
12                         DEPTHZ    EQU $FB
13                         LENGTHZ   EQU $FC
14                         PITCHA    EQU $FD
15                         PITCHB    EQU $FE
16                         ORIGDEPTH EQU $FF
17                         ;
18                         SPKR      EQU $C030
19                         ;
20                                   ORG $300
21                         ;
22  0300  00               DEPTH     DFC 0
23  0301  00               LENGTH    DFC 0
24  0302  00               PITCH     DFC 0
25                         ;
26  0303  AD 00 03                   LDA DEPTH
27  0306  85 FB                      STA DEPTHZ
28  0308  85 FF                      STA ORIGDEPTH
29  030A  AD 01 03                   LDA LENGTH
30  030D  85 FC                      STA LENGTHZ
31  030F  AD 02 03                   LDA PITCH
32  0312  85 FA                      STA ORIGPITCH
33  0314  85 FD                      STA PITCHA
34  0316  85 FE            TOGGLE1   STA PITCHB
35  0318  8D 30 C0                   STA SPKR
36  031B  C6 FE            WAIT1     DEC PITCHB
37  031D  D0 05                      BNE WAIT2
38  031F  A5 FA                      LDA ORIGPITCH
39  0321  4C 16 03                   JMP TOGGLE1
40  0324  C6 FD            WAIT2     DEC PITCHA
41  0326  D0 F3                      BNE WAIT1
42  0328  8D 30 C0                   STA SPKR
43  032B  A5 FA                      LDA ORIGPITCH
44  032D  85 FD                      STA PITCHA
45  032F  C6 FB                      DEC DEPTHZ
46  0331  D0 E8                      BNE WAIT1
47  0333  A5 FF                      LDA ORIGDEPTH
48  0335  85 FB                      STA DEPTHZ
49  0337  C6 FC                      DEC LENGTHZ
50  0339  D0 E0                      BNE WAIT1
51  033B  60                         RTS

000  Errors

0300  Hex Start of Object
033B  Hex end of Object
003C  Hex Length of Object
7B88  Hex end of Symbols
```

END OF LISTING 1

LISTING 2: DUET.DEMO

```
10  REM ***********************
20  REM *      DUET.DEMO      *
30  REM *   BY DOUG JEFFERYS  *
40  REM * COPYRIGHT (C) 1985 *
50  REM * BY MICROSPARC, INC *
60  REM * CONCORD, MA  Ø1742 *
70  REM ***********************
80  REM
90  REM
100  PRINT  CHR$ (4);"BLOAD NIBBLE.DUET"
110  TEXT : HOME : PRINT "   TWO-VOICE SOUND,
      BY DOUG JEFFERYS": PRINT "** COPYRIGHT
     1985 BY MICROSPARC,INC **"
120  POKE 34,2
130  PRINT
140  PRINT "WELCOME TO THE WORLD OF TWO-VOICE
      SOUND.REMEMBER WHEN THIS ";: POKE 768,2
     55: POKE 769,2: POKE 770,200: CALL 771: PRINT
     "USED TO BE IMPOSSIBLE": PRINT
150  PRINT "NOT ANY MORE. YOU, TOO, CAN NOW D
     O IT,  FROM ANY PROGRAM YOU WRITE.": PRINT

160 VP =  PEEK (37): GOSUB 1Ø9Ø: VTAB VP + 1
170  PRINT "AFTER BLOADING IN THE ROUTINE, ON
     LY      THREE POKES ARE NEEDED."
180  PRINT
190  PRINT "]POKE 768,DEPTH OF WAVE"
200  PRINT "]POKE 769,LENGTH OF TONE"
210  PRINT "]POKE 77Ø,OVERALL PITCH"
220  PRINT : PRINT "AND": PRINT
230  PRINT "]CALL 771"
240  PRINT
250  PRINT "WILL EXECUTE THE ROUTINE
260  GOSUB 1Ø8Ø: REM  CONTINUE
270  HOME
280  PRINT "FOR EXAMPLE, THE TONE YOU HEARD A
     T THE  START OF THIS PROGRAM WAS DONE LI
     KE THIS"
290  PRINT "]POKE 768,255:POKE 769,2:POKE 77Ø
     ,2ØØ"
300  PRINT "]CALL 771"
310 VP =  PEEK (37): GOSUB 1Ø9Ø: VTAB VP + 1
320  PRINT "AND THE RESULT WAS THIS!": CALL 7
     71
330 VP =  PEEK (37): GOSUB 1Ø9Ø: VTAB VP + 1
340  PRINT : PRINT "THE HIGHER THE PITCH, THE
      SHORTER THE   SOUND.  THIS WAS DONE CHA
     NGING ONLY THE PITCH.": POKE 77Ø,1ØØ: CALL
     771: FOR I = 1 TO 3ØØØ: NEXT I
350  PRINT : PRINT "ALSO NOTE THAT INCREASING
      THE PITCH     NUMBER LOWERS THE PITCH."
     : PRINT "FOR EXAMPLE:  PITCH=255": POKE
     77Ø,255: CALL 771: PRINT "
     PITCH=15Ø": POKE 77Ø,15Ø: CALL 771: PRINT

360  GOSUB 1Ø8Ø
370  HOME
380  PRINT "LOW DEPTH VALUES GREATLY SHORTEN
     TONES.": POKE 768,2: POKE 769,5Ø: CALL 7
     71: PRINT "THAT WAS A LOW DEPTH VALUE."
390  PRINT "(THE DEPTH WAS '2', AS COMPARED T
     O THE  '255' YOU HAVE BEEN USED TO, AND
     JUST TOMAKE IT AUDIBLE, THE LENGTH WAS
            INCREASED TO '5Ø', RATHER THAN TH
     E USUAL'2'."
400  GOSUB 1Ø8Ø
410  HOME
420  PRINT "CHANGING THE PITCH IN A FOR-NEXT
     LOOP   CAN BE INTERESTING.  FOR EXAMPLE:
     ": PRINT
430  PRINT "]POKE 768,1ØØ:POKE 769,1"
440  PRINT "]FOR I=1ØØ TO 1Ø STEP-1:POKE 77Ø,
     I:CALL 771:NEXT I"
450  PRINT : PRINT "YIELDS THIS"
460 VP =  PEEK (37): GOSUB 1Ø9Ø: VTAB VP + 1

470  POKE 768,1ØØ: POKE 769,1
480  FOR I = 1ØØ TO 1Ø STEP  - 1: POKE 77Ø,I:
      CALL 771: NEXT
490  PRINT : PRINT "OF COURSE, IT CAN GO THE
     OTHER WAY..."
500  FOR I = 1Ø TO 1ØØ: POKE 77Ø,I: CALL 771:
      NEXT I
510  GOSUB 1Ø8Ø
520  HOME
530  PRINT "EVER THOUGHT THAT THE OLD BEEP OF
      A      CONTROL-G WAS DULL?"
540  PRINT  CHR$ (7): REM  DULL CONTROL-G
550  FOR I = 1 TO 2Ø
560 X =  INT ( RND (1) * 2)
570  IF X = 1 THEN  PRINT  CHR$ (7);
580  NEXT I
590  PRINT "JUST THINK HOW MUCH MORE LIVELY T
     HIS     SOUNDS!": PRINT
600  POKE 768,25: POKE 769,1: POKE 77Ø,175
610  FOR I = 1 TO 2Ø
620 X =  INT ( RND (1) * 2)
630  IF X = 1 THEN  CALL 771
640  NEXT I
650 VP =  PEEK (37): GOSUB 1Ø9Ø: VTAB VP + 1
660  PRINT "HOW ABOUT A SHORT,SWEET ROCKET LA
     UNCH   FOR YOUR NEXT VIDEO GAME?"
670  GOSUB 1Ø8Ø
680  VTAB 12
690  PRINT "]POKE 768,5:POKE 769,5"
700  PRINT "]FOR I=5Ø TO 2 STEP-1:POKE 77Ø,I:
     CALL   771:NEXT I"
710  POKE 768,5: POKE 769,5
720  FOR I = 5Ø TO 2 STEP  - 1: POKE 77Ø,I: CALL
     771: NEXT I
730  GOSUB 1Ø8Ø
740  HOME
750  PRINT "THE SYNTHESIZER CAN ALSO BE USED
     FOR    MUSICAL NOTES."
760  PRINT : PRINT "NOTE   OCTAVE 1 (LOW)   O
     CTAVE 2 (HIGH) "
770  PRINT " C            24Ø              12Ø"
780  PRINT " C#           228              113"
790  PRINT " D            215              1Ø5"
800  PRINT " D#           2Ø3               99"
810  PRINT " E            19Ø               93"
820  PRINT " F            18Ø               86"
830  PRINT " F#           17Ø               82"
840  PRINT " G            16Ø               78"
850  PRINT " G#           151               74"
860  PRINT " A            142               7Ø"
870  PRINT " A#           134               66"
880  PRINT " B            125               62"
890  GOSUB 1Ø8Ø
900  HOME
910  PRINT "FOR MOST MUSIC, A VALUE OF '1' IS
            SUFFICIENT FOR THE DEPTH OF THE W
     AVE."
920  PRINT : PRINT "PICK A VALUE FOR A SHORT
     NOTE, LIKE AN  EIGHTH NOTE, AND USE IT A
     S A BASE FOR   THE LONGER NOTES. (QUARTE
     RS, ETC...)"
930  PRINT : PRINT "USE THE PREVIOUSLY PRESEN
     TED TABLE TO   ENTER THE NOTE VALUES THE
     MSELVES."
940  PRINT : PRINT "YOU MAY THEN ENTER THE NO
     TES IN THE FORMOF 'DATA' STATEMENTS."
950  PRINT : PRINT "USE A FOR-NEXT LOOP TO RE
     AD THE VALUES, POKE THEM IN, AND CALL 77
     1."
960  PRINT : PRINT "THE DATA FOR 'THE ENTERTA
     INER' HAS      ALREADY BEEN TYPED IN.  H
     ERE IS THE CODEUSED TO PLAY IT."
970  GOSUB 1Ø8Ø
980  HOME
990  PRINT : PRINT
1000  PRINT : PRINT "]FOR I=1 TO 79:READ D,L,
     P:POKE 768,D:     POKE 769,L:POKE 77Ø,P:C
     ALL 771:NEXT I"
1010  FOR I = 1 TO 79: READ D,L,P: POKE 768,D
     : POKE 769,L: POKE 77Ø,P: CALL 771: NEXT
     I
1020  PRINT : PRINT : PRINT "ENTERTAINED?"
1030  GOSUB 1Ø8Ø
1040  HOME
1050  PRINT "... AND SO ENDS THIS DEMONSTRATI
     ON OF   TWO-VOICE SOUND ON THE APPLE COM
     PUTER."
1060  PRINT : PRINT : PRINT : PRINT "BYE FOR
     NOW!"
1070  TEXT : VTAB 23: END
1080  REM  CONTINUE
1090  POKE  - 16368,Ø
1100  VTAB 23: PRINT "PRESS <RETURN> TO CONTI
     NUE"
1110  IF  PEEK ( - 16384) < 128 THEN 111Ø
1120  POKE  - 16368,Ø
1130  RETURN
1140  REM  DATA FOR 'THE ENTERTAINER'
1150  DATA  1,5Ø,215,1,5Ø,2Ø3,1,5Ø,19Ø,1,12Ø,
     12Ø,1,5Ø,19Ø,1,12Ø,12Ø,1,5Ø,19Ø,1,18Ø,12
     Ø
1160  DATA  1,7Ø,113,1,7Ø,1Ø5,1,7Ø,99,1,7Ø,93
     ,1,7Ø,12Ø,1,12Ø,1Ø5,1,7Ø,93,1,7Ø,12Ø,1,1
     2Ø,1Ø5,1,18Ø,12Ø
1170  DATA  1,5Ø,215,1,5Ø,2Ø3,1,5Ø,19Ø,1,12Ø,
     12Ø,1,5Ø,19Ø,1,12Ø,12Ø,1,5Ø,19Ø,1,18Ø,12
     Ø
1180  DATA  1,5Ø,142,1,5Ø,16Ø,1,5Ø,17Ø,1,5Ø,1
     42,1,7Ø,12Ø,1,14Ø,93,1,7Ø,12Ø,1,65,125,1
     ,6Ø,142,1,18Ø,1Ø5
1190  DATA  1,5Ø,215,1,5Ø,2Ø3,1,5Ø,19Ø,1,12Ø,
     12Ø,1,5Ø,19Ø,1,12Ø,12Ø,1,5Ø,19Ø,1,18Ø,12
     Ø
1200  DATA  1,7Ø,113,1,7Ø,1Ø5,1,7Ø,99,1,7Ø,93
     ,1,7Ø,12Ø,1,12Ø,1Ø5,1,7Ø,93,1,7Ø,125,1,1
     2Ø,1Ø5,1,18Ø,12Ø
1210  DATA  1,6Ø,12Ø,1,6Ø,1Ø5,1,7Ø,93,1,6Ø,12
     Ø,1, 6Ø,1Ø5,1,14Ø,93,1,6Ø,1Ø5,1,6Ø,12Ø,1
     ,6Ø,1Ø5
1220  DATA  1,7Ø,93,1,6Ø,12Ø,1, 6Ø,1Ø5,1,14Ø,
     93,1,6Ø,1Ø5,1,6Ø,12Ø,1,6Ø,1Ø5
1230  DATA  1,7Ø,93,1,6Ø,12Ø,1,12Ø,1Ø5,1,7Ø,9
     3,1,6Ø,125,1,12Ø,1Ø5,1,12Ø,12Ø,1,1ØØ,16Ø
     ,1,12Ø,12Ø
```

END OF LISTING 2

```
              KEY PERFECT 5.Ø
                  RUN ON
                 DUET.DEMO
  ========================================
    CODE-5.Ø    LINE# - LINE#    CODE-4.Ø
    --------    -------------    --------
    FBCE793C      1Ø -   1ØØ       5D3D
    C33D4AB6     11Ø -   2ØØ       B68A
    2E253FØ2     21Ø -   3ØØ       6AC5
    9B17182E     31Ø -   4ØØ     Ø1ØDE2
    E777C22A     41Ø -   5ØØ       8CFØ
    1AEBF8FØ     51Ø -   6ØØ       5BF3
    99AØE73F     61Ø -   7ØØ       6322
    8AD5F653     71Ø -   8ØØ       5A8D
    56ØCBE8Ø     81Ø -   9ØØ       3498
    2858CC64     91Ø -  1ØØØ     Ø1Ø86F
    68ACDB45    1Ø1Ø -  11ØØ       7643
    DØ53D2EB    111Ø -  12ØØ     Ø118A7
    74CAF737    121Ø -  123Ø       74FE
    Ø7E117C3 = PROGRAM TOTAL =     ØFEF
```