decimal; otherwise, they are in hexadecimal. Fear not, however; you don't need to understand hex numbers to enter data. You do need to know that any number from 0 to 9 is the same in hex and decimal; thus, you don't need to use ! in front of them.

The data most commonly used with ASOUND is a three number group; look at the first group on line 1000 above. The first number must be from 1 to 254; it specifies duration in sixtieths of a second. It's a 5, meaning 5/60 of a second. The second number specifies frequency (1 to 255). It is 121, or middle C. (See reference manual for complete table.) The third number is a combination of two values (each ranging from 0 to 15) representing distortion and volume. To get the third number, multiply distortion by 16 and add the volume. Remember that a distortion of 10 produces a pure note. 168 equals 10*16+8, and thus gives a pure tone at about half volume. Since 15 produces maximum volume, 8 gives about half volume.

Now look at the second group of three numbers on line 1000. The 7 means that this sound will play for 7/60 of a second. The 243 means it is a low C. The 170 (10*16+10) means it is a pure tone of slightly over half volume.

Notice that the last group on line 1000 starts with a zero. Since it does not make any sense to play a note for 0/60 second, this zero is a signal that what follows is a special command. There are three of these: 0,FF tells the system to stop the sound on that channel, 0,FE stops all channels, and 0,FD is explained later in this chapter. Back to line 1000 again; 0,FF turns off the sound after the 7/60 second low C.

If you need to know where you are in the sound sequence, give an RTIME at the start, and use the TIME function to determine how many sixtieths of a second have elapsed.

The SCONTROL on line 20 starts the sound. The WAIT 60% on line 30 prevents the program from reaching the END before the tune has been played. Note that all voices are turned off by the END command.

It is easy to play a "tune" just once, as in the previous program. But how do you play a tune a limited number of times or repeat it continuously? And how do you put attack and decay into notes? You can use the tools described above, but it will be rather painful. Advan BASIC solves these problems with another special command. Consider the following example:

```
1000 CODE"FF,#2000,FF,#2000,FF,#2000,0,FF"
2000 CODE"5,!121,!168,7,!170,0,FD"
```

Line 1000 has three special commands which act somewhat like a GOSUB. Each command starts with FF, followed by a comma, a # symbol, and a linenumber. # means that the next number is a linenumber. FF is the hexadecimal form of 255; we use it because it's shorter. When the system detects a group starting with FF, it switches to the line specified by the number after # (in this case, line 2000). The data may lie on just one line or extend over several successive lines. 0,FD (at the end of line 2000) acts like a RETURN and causes the system to return to line 1000 and continue on that line, where it will again be sent to line 2000. Thus, the three special commands in line 1000 cause the sound pattern in line 2000 to be played three times. Here is another example: