produce a machine code program that will enable us to control both frequency and volume.

The best method of reducing the waveform amplitude, while retaining the overall shape of the wave, is to divide each value in the waveform table by a constant. This can be done in two ways: after each value is loaded into the accumulator but before the value is placed in the data register, or prior to entering the main program loop. The first method will increase the amount of time required to execute each cycle of the main loop, and as this factor limits the maximum frequency obtainable we should opt for the second method. A second table is produced from the original waveform table by dividing each value taken from the original table by a constant, then placing the result in the new table. The new table is then used for waveform data by the main program loop. The division method used is crude. An amplitude constant dictates the number of times that the table value is shifted right. As each shift right is an integer division by two, the effect of using an amplitude factor of 'n' is to divide each table by 2n.

A rather clever method is employed when the amplitude factor is zero. In this case we use the original table and the program modifies itself to specify the base address of the original table, rather than that of the table of divided values.

Execution of the main loop of the program can be delayed by inserting a small piece of code that does nothing except take time to execute. Normally this is done by decrementing either an eight-bit number in one of the index registers, or a 16-bit number in memory, from a set delay value to zero. We find that if we calculate the maximum delays produced by these two methods, decrementing an eight-bit number will provide sufficient delay.

The main problem is not providing enough delay — i.e. producing the lowest frequency — but providing the minimum delay; in other words, producing the maximum frequency. On page 732 we used a waveform divided into 80 steps. The extra code required for the delay slows down execution time so much that it is no longer practical to have this number of steps. The code making up the main loop is shown below.

MAIN		Time in machine cycles
	LDX #$00	2
NEXVAL		
	LDA AMPTAB,X	4
	LDY DELAY	4
MORDEL		
	DEY	2
	BNE MORDEL	3 (2 if loop fails)
	STA PORT	4
	INX	2
	CPX # STEPS	2
	BNE NEXVAL	3 (2 if loop fails)

The total number of machine cycles required is given by: $2+(4+4+(2+3)\times delay-1+4+2+2+3) \times steps -1 = 1+(18+5\times delay)\times steps$: and a minimum delay value of 1 yields the maximum frequency as follows: max freq $= 1000000 \div (1+23\times steps)$.

For a maximum frequency of around 3,000 Hz, this formula gives the number of steps as 15. This is the number of samples of the waveform we should use to produce a reasonable maximum frequency. Using 15 steps, the original formula can be written as: no of machine cycles $= 271+75\times delay$.

If we require a minimum frequency of, say, 128 Hz (about two octaves below middle C) then the delay value will be 101. This value can be held and decremented by an index register.

The final problem produced by an altered frequency is that, for a set number of repeats of the loop delay, given the duration of the tone produced will decrease as the frequency increases. This is because as the frequency increases, the main loop takes less time to execute. To balance this we must
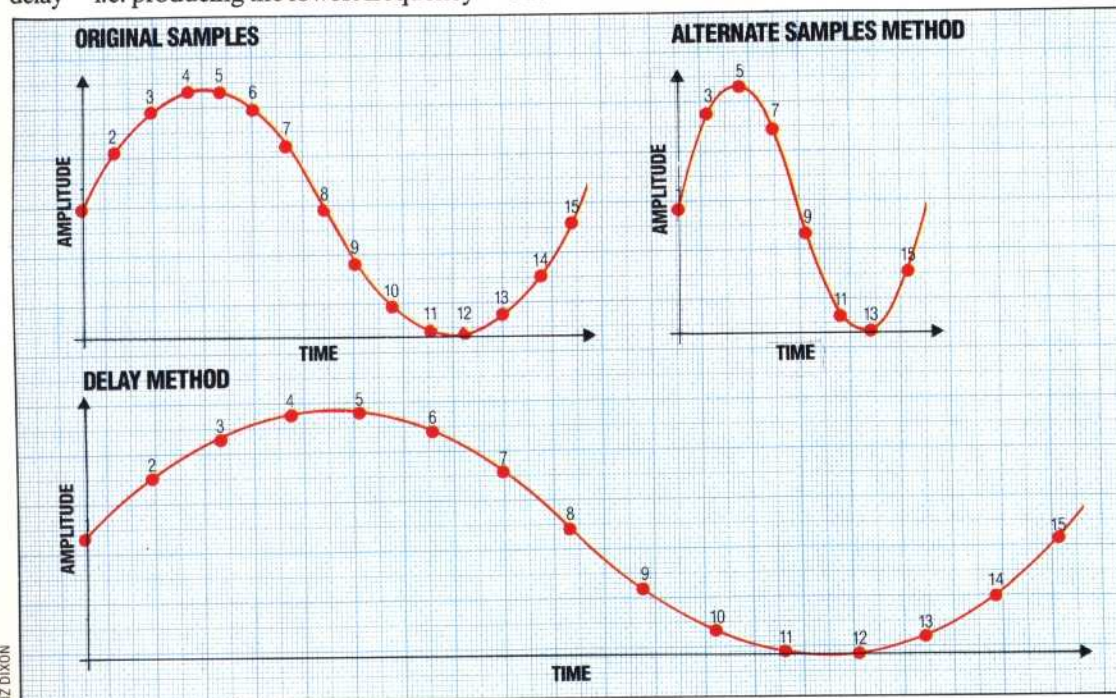
Stretching The Snake
The frequency of a digitally sampled wave can be altered by taking fewer samples or by inserting a delay between each sample value. If the original look-up table for the wave contains 15 samples of the wave, then the frequency of the output wave can be doubled by taking every other sample only. Alternatively all 15 values can be output inserting a delay to double the time taken to output the entire wave, halving the frequency. The first method allows many samples to be used at lower frequencies but affords only crude frequency control. The second method allows much finer frequency control but means that fewer samples must be used

ORIGINAL SAMPLES

ALTERNATE SAMPLES METHOD

DELAY METHOD