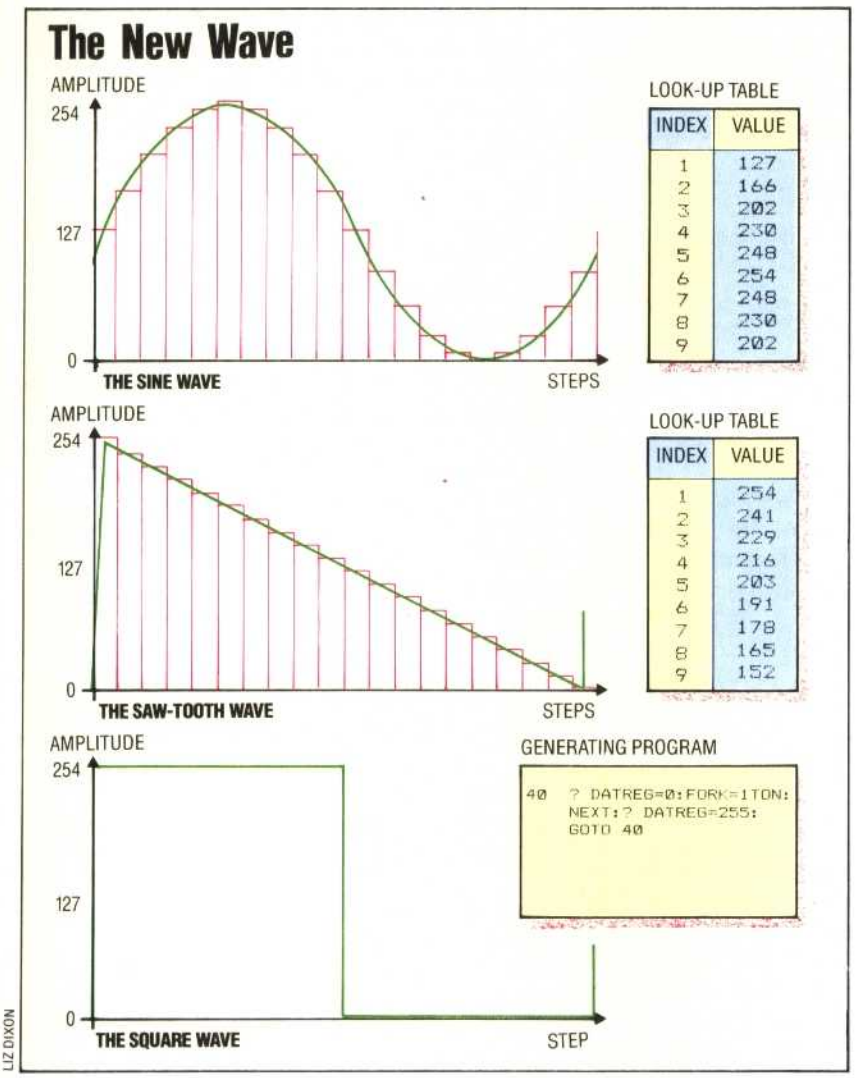

Form Filling

The sine wave and the saw-tooth waveforms are created by first deciding how many sample steps are to constitute one cycle of the wave. These samples of the wave's amplitude are then calculated and stored in a look-up table. The values in this table can then be copied in sequence to the user-port data register and thus to the digital-to-analogue converter where they become voltage levels. The advantage of the look-up table is that it allows the time-consuming calculations to be done in advance; actually generating the waveform therefore takes little time, and this makes a frequency range of several octaves possible. Without the look-up table the range would be restricted to two octaves.

The square wave can be generated by a BASIC program because the process is so simple. BASIC's slowness, however, restricts the frequency range considerably

should be pitched just a few notes higher than middle C.

It can be seen that the number of sample steps in which we choose to divide our waveform has a direct effect on the frequency of the final note. Doubling the number of sample steps would halve the final note frequency. Obviously, the more samples we make of the note the nearer we are likely to get to the tone quality we are synthesising, but this must always be weighed against the final maximum frequency that can be achieved for a given number of steps.

One waveform cycle is unlikely to be long enough to be audible, so we must also include code to repeat the waveform-generating section of code a set number of times. The number of repeats can be determined by setting a counter value and decrementing it to zero. To give a large range of counter values, a 16-bit number stored in two adjacent locations has been used. In addition to this code, interrupts are disabled at the start of the program by SE1 and re-enabled by CL1 at the end. If interrupts were to occur during execution, then this would make the timing of the program inaccurate. However, we can disable some interrupts only; non-maskable interrupts, if they occur during program execution, can still be the cause of some timing errors.

The waveform data must be set up in memory as a look-up table, with each waveform type taking 80 consecutive locations. In the Commodore version the sine wave look-up table is located in memory starting at $C000; the saw-tooth table is at $C050 and the square wave table starts at $C0A0. The machine code program is designed to default to load data from the sine wave table using indexed addressing, but we can switch to another table by modifying the program directly with a POKE from BASIC. The LDA part of LDA SINE,X is in location $C103. The start address of the table of data to be loaded has its LO–byte in location $C104 and its HI-byte in location $C105. To modify the start address of the data to be loaded all we have to do is to change the number held in $C104. Normally, for a sine wave, this location will hold 0; if we wish to change to a saw-tooth wave then all we need to do is change the contents of $C104 to 80. Changing the contents of this location to 160 will change the waveform to a square wave.

The BBC version is also designed so that the look-up tables start at the beginning of a new page in memory. Because the tables start at a new page, the HI-byte of each of the table start addresses is the same, and we need modify the LO-byte only. On a normally configured BBC Model B in mode 7, HIMEM is &7C00. By lowering the top of memory by three pages we allocate more than enough space for the look-up tables and machine code program.

FOR THE COMMODORE

```
                ;++++++++++++++++++++
                ;++++++++++++++++++++
                ;++                ++
                ;++ CBM 64  SOUND  ++
                ;++   GENERATOR    ++
                ;++                ++
                ;++++++++++++++++++++
                ;++++++++++++++++++++
                ;
                PORT   = 56577      ;DATA REGISTER ADDRESS
                STEPS  = 80         ;NO. OF STEPS IN WAVE
                *=$C000
                ;
                ;++++ SET UP DATA TABLE AREA ++++
                SINE   *=*+STEPS
                SAW    *=*+STEPS
                SQUARE *=*+STEPS
                NUMBER *=*+2
                COUNT  *=*+2
                ;
                ;++++ MAIN PROGRAM ++++
                ;
78              SEI
AD F0 C0        LDA NUMBER
8D F2 C0        STA COUNT       ;SET COUNT VALUE
AD F1 C0        LDA NUMBER+1
8D F3 C0        STA COUNT+1
                ;
                LOOP2
A2 00           LDX #$00
                LOOP1
BD 00 C0        LDA SINE,X      ;GET DATA
8D 01 DD        STA PORT        ;PUSH TO USER PORT
E8              INX
E0 50           CPX #STEPS
D0 F5           BNE LOOP1       ;END OF ONE CYCLE
                ;
                ;++++ DECREMENT COUNT ++++
AD F2 C0        LDA COUNT
38              SEC
E9 01           SBC #$01
8D F2 C0        STA COUNT
AD F3 C0        LDA COUNT+1
E9 00           SBC #$00
8D F3 C0        STA COUNT+1
D0 E0           BNE LOOP2       ;IF HIBYTE>0
A9 00           LDA #$00
CD F2 C0        CMP COUNT
D0 D9           BNE LOOP2       ;IF LOBYTE>0
58              CLI
60              RTS
```