

PS1, PS2 & PS3 (-128 to 127)

PS refers to Pitch Step. At the start of the associated note, pitch is set by the SOUND command. PS1 sets the positive or negative change of pitch per step in the first section, PS2 for the second section, and PS3 for the third section. In a similar manner to SOUND, PS is set in quarter semitones.

NS1, NS2 & NS3 (0 to 255)

NS refers to Number of Steps per section; and in conjunction with PS selects the rate at which pitch changes in a section and also the duration of the whole pitch envelope. The PS and NS values for the above example are as follows:

T=1 PS1=-10 NS1=15
PS2=+10 NS2=10
PS3=-10 NS3=5

In this case, pitch is set by SOUND = 160. This results in:

ENVELOPE 11,-10,10,-10,15,10,5,0,0,0,0,0

is a price to be paid, in that resolution is halved in the direction of expansion.

A sprite can move one pixel at a time and the old position is automatically erased. Sprites can also move in and out of the normal viewing area of the screen.

Priority And Collision

When two sprites cross each other's path, one appears to pass in front of the other. If there are any holes in the sprite that is passing in front, the sprite behind will show through. Priority can be used to achieve some interesting three-dimensional effects. Each sprite is given a number from 0 to 7 and the simple rule governing priority is that lower-numbered sprites appear to move in front of higher-numbered ones. Usually, sprites appear to move in front of any normal characters on the screen, but they can be programmed to move behind as well. Again this feature can be used to give the impression of depth on the screen.

When two sprites cross each other this is signalled in a collision register. PEEKing this register can give the programmer details of which sprites have been involved. There is another similar register that signals when a sprite has been in collision with any background characters.

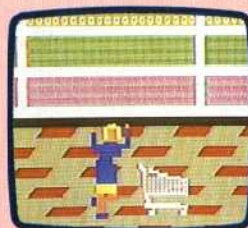
As a consequence of the availability of these features, writing programs to control fast-moving games in BASIC is now possible. Unfortunately, there are no special BASIC commands to control sprite features; everything has to be done by a succession of POKEs into the Commodore 64's memory. An alternative and easier method of creating sprites is to invest in a Simon's BASIC cartridge.

The duration of the envelope is given as $(NS1+NS2+NS3) \times T$, which in this case is $(15+10+5) \times 1 = 0.3$ seconds. Normally, the pitch envelope will automatically repeat over the duration of a note unless disabled by the timing parameter, T.

In the next instalment of the Sound And Light course we will return to the sound features of the BBC Micro and explain the operation of the volume envelope.

Simon's Basic

For approximately £50, it is possible to purchase a plug-in cartridge to extend the high resolution and sprite handling capabilities available to the BASIC programmer. The cartridge comes complete with a weighty manual detailing the 114 extra commands. These include commands to turn on high resolution mode, select background and foreground colours, and to draw circles, ellipses, rectangles and straight lines. Sprite handling instructions include: assistance with sprite design and creation, commands to switch sprites on and off, and ways of positioning them on the screen



Step Two

These lines may be added to the Supermarket program listing given on page 359. This section of the program uses two expanded, multi-coloured sprites to make up the human figure and a further expanded sprite to make up the shopping trolley. The sprite data pointers are manipulated so that the woman changes shape. This gives the effect of the figure dancing as it crosses the screen. To use the supermarket program, as a subroutine in this program, change line 3270 to read: 3270 RETURN

```

90 REM ** SPRITES 64 **
100 PRINT "J"
110 V=53248
120 REM---READ SPRITE DATA---
130 FOR J=12288 TO 12350: READ A: POKE I, A: NEXT
140 FOR J=12352 TO 12414: READ A: POKE I, A: NEXT
150 FOR J=832 TO 894: READ A: POKE I, A: NEXT
160 FOR J=896 TO 958: READ A: POKE I, A: NEXT
170 FOR J=12416 TO 12478: READ A: POKE I, A: NEXT
180 REM---EXPAND SPRITES---
190 POKE V+23,7: POKE V+29,7
200 REM---COLOR SPRITES---
210 POKE V+39,10: POKE V+40,10
220 POKE V+41,1
230 REM---MULTI COLOR---
240 POKE V+28,3: POKE V+37,7: POKE V+38,9
300 REM---MEMORY POINTERS---
310 POKE 2040,192: POKE 2041,193: POKE 2042,194
320 REM---SET Y COORDS---
330 Y0=150: Y1=Y0+42: Y2=Y0+34
340 POKE V+1, Y0: POKE V+3, Y1: POKE V+5, Y2
400 REM---TURN ON SPRITES---
410 POKE V+21,7
500 GOSUB 3000: REM OMIT IF NO SUBROUTINE
1000 X0=20
1010 POKE 2040,13: POKE 2041,14
1020 POKE V,X0: POKE V+2,X0: POKE V+4,X0+48
1030 FOR I=1 TO 500: NEXT
1040 POKE 2040,192: POKE 2041,193
1050 X0=X0+5
1060 POKE V,X0: POKE V+2,X0: POKE V+4,X0+48
1070 FOR I=1 TO 3500: NEXT
1080 X0=X0+5
1090 IF X0>200 THEN I=110
1100 GOTO 1010
1110 FOR J=1 TO 10
1120 POKE 2040,13: POKE 2041,14
1130 FOR I=1 TO 500: NEXT
1140 POKE 2040,192: POKE 2041,193
1150 FOR I=1 TO 500: NEXT
1160 NEXT
1170 GOTO 1170
9000 REM---DATA WOMAN TOP---
9010 DATA 0,0,0,21,0,0,21,0,0,22,0,0,86,0
9020 DATA 0,86,0,0,86,0,0,40,0,0,252,0
9030 DATA 15,255,0,255,255,0,255,255,0
9040 DATA 195,255,0,195,255,0,195,243,254
9050 DATA 287,243,254
9060 DATA 143,240,0,143,252,0,15,252,0
9070 DATA 15,252,0,15,252,0
9100 REM---DATA WOMAN BOTTOM---
9110 DATA 15,252,0,15,252,0,15,252,0
9120 DATA 15,252,0,5,84,0,5,84,0,5,84,0
9130 DATA 5,84,0,10,40,0,234,40,0,234,40,0
9140 DATA 234,40,0,192,40,0,192,40,0,0,40,0
9150 DATA 0,40,0,0,63,0,0,63,0,0,0,0,0,0
9160 DATA 0,0,0
9200 REM---DATA WOMAN TOP #2---
9210 DATA 0,0,0,20,32,32,85,32,32,105,48,48,105,48
9220 DATA 8,105,48,48,105,48,48,48,48,48,252,48
9230 DATA 63,255,240,63,255,240,63,255,0
9240 DATA 3,255,0,3,255,0,3,240,0
9250 DATA 15,240,0
9260 DATA 15,240,0,15,252,0,15,252,0
9270 DATA 15,252,0,15,252,0
9300 REM---DATA WOMAN BOTTOM #2---
9310 DATA 15,252,0,15,252,0,15,252,0
9320 DATA 15,252,0,5,84,0,5,84,0,5,84,0
9330 DATA 5,84,0,10,40,0,58,168,0,58,168,0
9340 DATA 58,0,0,58,0,0,10,0,0,13,0,0
9350 DATA 10,0,0,15,192,0,15,192,0,0,0,0,0,0,0
9360 DATA 0,0,0
9400 REM---TROLLEY DATA---
9410 DATA 192,0,0,224,0,0,118,0
9420 DATA 0,55,192,0,32,60,0,53
9430 DATA 87,240,32,0,15,53,85,85
9440 DATA 32,0,3,53,85,85,0,0,3
9450 DATA 21,85,85,31,255,255,24,0
9460 DATA 0,12,0,0,12,0,0,31,255
9470 DATA 240,31,255,255,1,0,2,7
9480 DATA 14,7,0,14
    
```