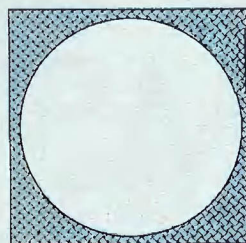


Como se han hecho
los mejores juegos.

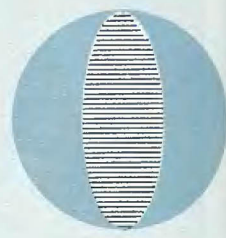
Técnica Filmation

Crear un juego en tres dimensiones es una ilusión que cualquiera puede poner en práctica siguiendo estas explicaciones sobre la Técnica Filmation.

Por: Alberto Suñer



MASCARA
SIEMPRE CON TINTA 3



GRAFICO

Dibujo 1

Hace algún tiempo apareció en el mercado un programa para ordenador, en concreto para Spectrum, aunque más tarde se pasó a otros sistemas, entre ellos a Amstrad, que revolucionó en cierta forma el mundo de la programación de juegos, por supuesto nos estamos refiriendo al programa *Knight Lore*.

Estamos seguros de que cualquier aficionado al mundo de los juegos ha oído hablar en más de una ocasión de dicho programa, ya que representó en su día una gran innovación en el mundo de los videojuegos.

Lo primero que sorprendió, en su momento, de dicho programa, fue la presentación tridimensional del escenario donde transcurre la acción, que luego ha sido trasladado a centenares de programas.

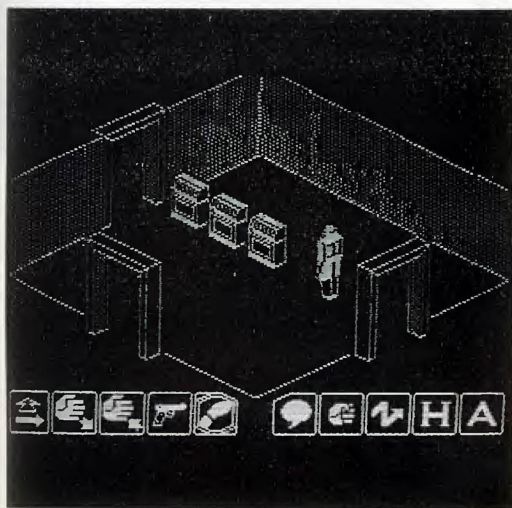
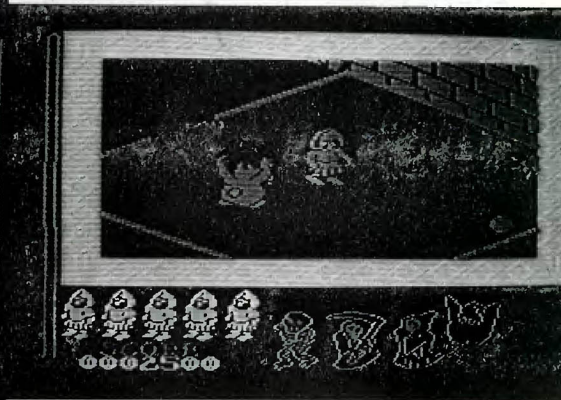
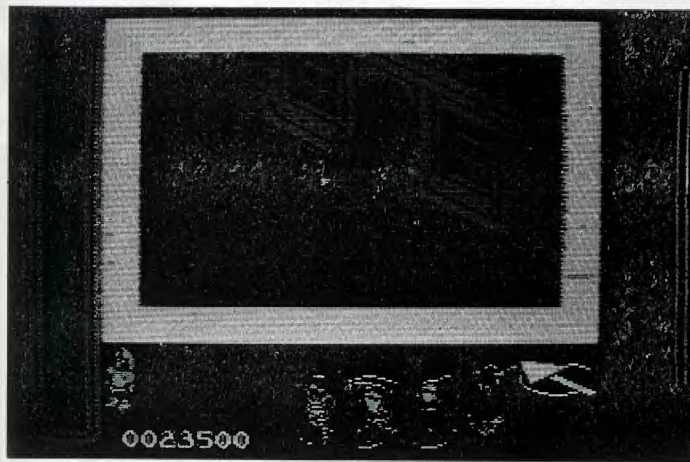
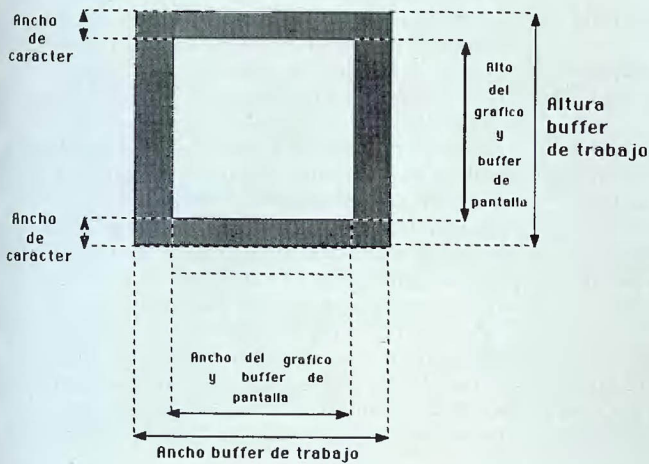
A este método de programación se le bautizó con el nombre de *Técnica Filmation*, debido a la gran perfección de sus movimientos y a la posibilidad de que cada personaje que interviene en el juego pueda aparecer en planos diferentes, dentro de una misma pantalla de juego.

Por todos estos motivos vamos hoy a intentar explicar cómo funciona la técnica mencionada anteriormente, que a partir de ahora podremos utilizar en nuestros propios programas, incluso desde Basic, gracias a la rutina que hemos preparado, la cual incluye dos nuevos comandos RSX que nos permitirán mover gráficos en pantalla con el método *Filmation*.

La primera particularidad que presenta esta técnica de impresión en pantalla, es referente a la forma en que se deben confeccionar los gráficos.

Cada uno de los personajes u objetos a los que se desee aplicar este tipo de impresión, deberá

DIBUJO 2



poscer, además de su imagen, una máscara de dicha imagen.

Cuando nosotros realizamos un gráfico para utilizarlo posteriormente en nuestro programa, realizamos un dibujo del personaje que deseamos que aparezca en pantalla. Ahora bien, la utilización de la técnica *Filmation* requiere una imagen en inverso o en negativo de dicho dibujo. Esto último es lo que se denomina máscara.

Veamos, pues, cuáles serían los pasos a seguir cuando nos pongamos a crear nuestro propio gráfico. En primer lugar, crearíamos la imagen del objeto que deseamos se imprima en pantalla y la almacenaríamos una vez acabada.

A continuación, tomando esa misma imagen o dibujo como referencia, deberemos rellenar todos los puntos vacíos que rodean dicha imagen, dejando un pixel en blanco entre el dibujo original y su relleno.

Una vez realizada esta operación, deberemos



borrar el dibujo original, con lo cual nos quedará su imagen inversa, a lo cual se denomina máscara.

Por último, la almacenaremos a continuación del dibujo original.

Debemos advertir que la máscara se realiza siempre con tinta 3, independientemente del color que posea la imagen original.

Tenemos en este momento preparado nuestro gráfico para ser imprimido en pantalla con la rutina que utiliza la técnica *Filmation* (ver dibujo 1).

Veamos ahora los pasos que deben ser realizados por la rutina de impresión, para conseguir que el gráfico que se desplaza por la pantalla, lo haga de una forma suave y sin parpadeos.

Las rutinas de impresión normales dan movimiento a los gráficos, imprimiéndolos en una posición de pantalla. Seguidamente se borra este último colocando en su lugar lo que había

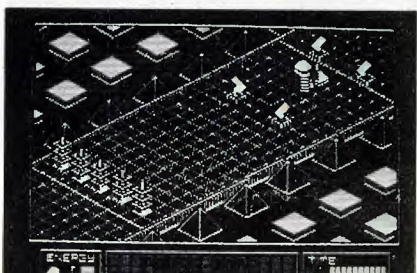
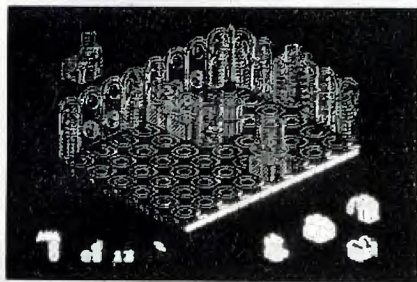
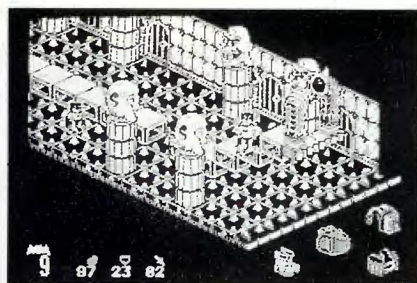
anteriormente en pantalla y se imprime es una posición contigua, con lo cual se da la sensación de movimiento.

Esto se consigue realizando las operaciones de impresión y borrado en el momento en que se produce el barrido de pantalla, por lo cual nuestros ojos no son capaces de observar dichas operaciones.

Ahora bien, esto se puede realizar con bloques gráficos relativamente pequeños, ya que, en el momento en que nuestro gráfico sea lo suficientemente grande, las operaciones de impresión y borrado superarán el tiempo en que tarda en producirse el barrido de pantalla, con lo cual aparecerá ese molesto parpadeo.

Por tanto, lo que se pretende, es intentar rebajar el tiempo de las operaciones de impresión y borrado. Se puede conseguir imprimiendo y borrando al mismo tiempo, con lo que nos quedamos con un tiempo de ejecución igual a la

```
1 REM FILMATION
2 DATOS DEL GRAFICO BOLA
10 REM *****
20 REM **** DATOS PARA CONFECCIONAR ****
30 REM **** LA PELOTA Y SU MASCARA ****
40 REM *****
50 REM
60 REM
70 FOR N=47530 TO 47650
80 READ A:SUMA=SUMA+A
90 POKE N,A
100 NEXT
110 IF SUMA<>66A73 THEN PRINT "ERROR EN DATAS"
120 DATA 255,255,255,255,255,255,255,255
130 DATA 255,136,51,255,255,255,238
140 DATA 0,0,255,255,255,136,0
150 DATA 0,51,255,255,0,0,0
160 DATA 17,255,238,0,0,0,0
170 DATA 255,238,0,0,0,0,255
180 DATA 204,0,0,0,0,119,204
190 DATA 0,0,0,0,119,136,0
200 DATA 0,0,0,51,136,0,0
210 DATA 0,0,51,136,0,0,0
220 DATA 0,51,136,0,0,0,0
230 DATA 51,136,0,0,0,0,51
240 DATA 204,0,0,0,0,119,204
250 DATA 0,0,0,0,119,238,0
260 DATA 0,0,0,255,238,0,0
270 DATA 0,0,255,255,0,0,0
280 DATA 17,255,255,136,0,0,51
290 DATA 255,255,238,0,0,255,255
300 DATA 255,255,136,51,255,255,255
310 DATA 255,255,255,255,255,255,255
320 DATA 255,255,255,255,0,0,0
330 DATA 0,0,0,0,0,0,0
340 DATA 0,0,0,0,55,200,0
350 DATA 0,0,1,59,184,0,0
360 DATA 0,7,119,220,192,0,0
370 DATA 15,119,220,224,0,0,14
380 DATA 255,238,224,0,1,14,255
390 DATA 238,240,0,1,14,255,238
400 DATA 240,0,3,29,255,255,112
410 DATA 128,3,29,255,255,112,128
420 DATA 3,29,255,255,112,128,3
430 DATA 29,255,255,112,128,3,29
440 DATA 255,255,112,128,1,14,255
450 DATA 238,240,0,1,14,255,238
460 DATA 240,0,0,14,255,238,224
470 DATA 0,0,15,119,220,224,0
480 DATA 0,7,119,220,192,0,0
490 DATA 1,59,184,0,0,0,0
500 DATA 55,200,0,0,0,0,0
510 DATA 0,0,0,0,0,0,0
520 DATA 0,0,0,0,0,0,0
530 DATA 0,0,0,0,0,0,0
```



PROGRAMA CARGADOR

1 REM FILMATION

2 REM PROGRAMA CARGADOR

10 FOR N=8000 TO 822F

20 READ A:SUMA=SUMA+A

30 POKE N,A

40 NEXT

50 IF SUMA<57408 THEN PRINT "ERROR EN DATAS"

60 DATA 1,9,128,33,26,128,195

70 DATA 209,188,17,128,195,30,128

80 DATA 195,100,128,79,206,83,80

90 DATA 82,73,84,197,0,0,0

100 DATA 0,0,221,94,0,221,96

110 DATA 2,221,110,4,221,102,5

120 DATA 34,52,130,237,83,57,130

130 DATA 237,83,61,130,221,110,6

140 DATA 221,102,7,34,54,130,221

150 DATA 78,8,221,70,10,120,50

160 DATA 50,130,121,50,49,130,4

170 DATA 4,12,12,237,67,59,130

180 DATA 65,33,0,0,17,9,0

190 DATA 25,16,253,34,63,130,195

200 DATA 157,128,221,110,0,221,102

210 DATA 2,34,57,130,237,91,61

220 DATA 130,123,189,32,4,62,2

230 DATA 24,7,56,3,175,24,2

240 DATA 62,1,50,51,130,122,188

250 DATA 32,4,62,2,24,7,56

260 DATA 4,62,1,24,1,175,50

270 DATA 56,130,34,61,130,84,93

280 DATA 195,183,128,33,65,130,237

290 DATA 75,59,130,62,2,50,51

300 DATA 130,50,56,130,237,91,57

310 DATA 130,205,201,129,205,16,129

320 DATA 201,33,65,130,237,75,59

330 DATA 130,229,213,197,205,201,129

340 DATA 205,212,128,205,18,129,205

350 DATA 69,129,193,209,225,205,110

360 DATA 129,201,33,65,130,237,91

370 DATA 63,130,58,56,130,254,2

380 DATA 40,5,254,0,40,2,25

390 DATA 25,235,58,51,130,254,2

400 DATA 40,5,254,1,40,2,19

410 DATA 19,33,113,132,58,50,130

420 DATA 203,39,203,39,203,39,71

430 DATA 197,5,0,58,49,130,75

440 DATA 237,176,19,19,130,16,242

450 DATA 201,237,91,54,130,33,65

460 DATA 130,237,75,63,130,9,35

470 DATA 221,63,113,132,58,50,130

480 DATA 203,39,203,39,203,39,71

490 DATA 197,58,49,130,71,126,221

500 DATA 119,0,26,166,119,19,35

510 DATA 221,35,16,243,35,35,193

520 DATA 15,233,201,237,91,52,130

530 DATA 33,65,130,237,75,63,130

540 DATA 9,35,58,50,130,203,39

550 DATA 203,39,203,39,71,197,58

560 DATA 49,130,71,26,182,119,19

570 DATA 35,16,249,35,35,193,16

580 DATA 239,201,221,229,205,33,130

590 DATA 235,36,44,120,203,39,203

600 DATA 39,203,39,50,47,130,121

610 DATA 50,48,130,213,235,33,176

620 DATA 191,66,22,0,29,25,17

630 DATA 80,0,25,16,253,221,225

640 DATA 58,47,130,71,24,21,124

650 DATA 230,56,254,56,40,6,124

660 DATA 198,8,103,24,8,17,60

670 DATA 0,124,238,56,103,25,197

680 DATA 229,58,48,130,71,197,221

690 DATA 126,0,119,35,221,35,193

700 DATA 16,245,225,193,16,214,221

710 DATA 225,201,221,209,235,36,44

720 DATA 120,203,39,203,39,203,39

730 DATA 50,47,130,121,50,48,130

740 DATA 213,235,33,176,191,66,22

750 DATA 0,29,25,17,86,0,25

760 DATA 16,253,221,225,58,47,130

770 DATA 71,24,21,124,239,56,254

780 DATA 56,46,6,124,198,8,103

790 DATA 24,8,17,80,0,124,258

800 DATA 56,103,25,197,229,58,43

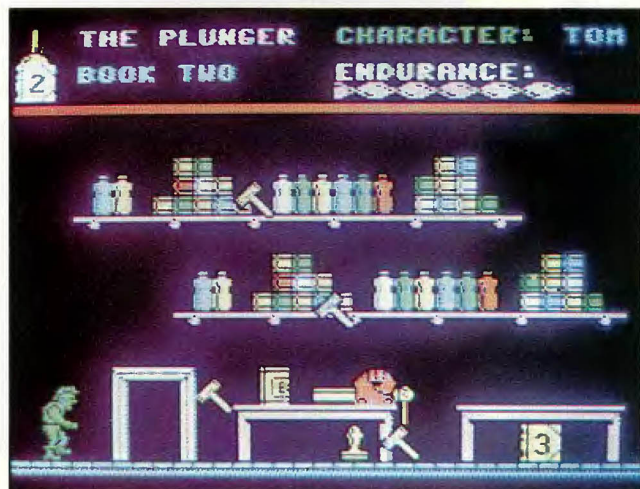
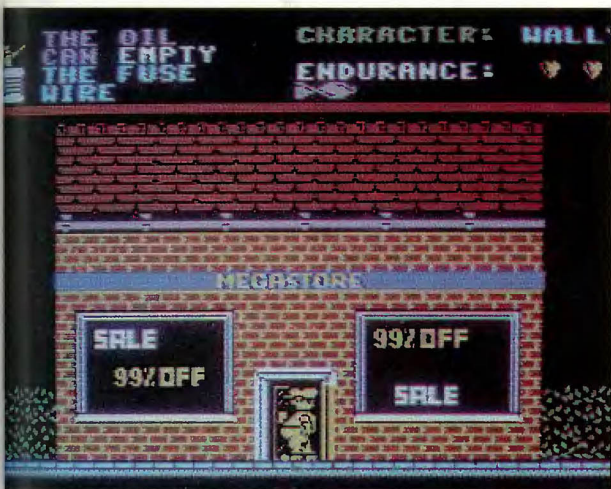
810 DATA 130,71,197,126,221,119,0

820 DATA 35,221,35,193,16,245,225

830 DATA 197,16,214,221,225,201,205

840 DATA 25,189,197,1,100,0,11

850 DATA 120,177,32,251,193,201,0



mitad del utilizado con el método anterior.

Vamos a ver, pues, qué pasos deberemos seguir para conseguir de una sola vez la impresión y el borrado del gráfico, reponiendo al mismo tiempo lo que había anteriormente en pantalla.

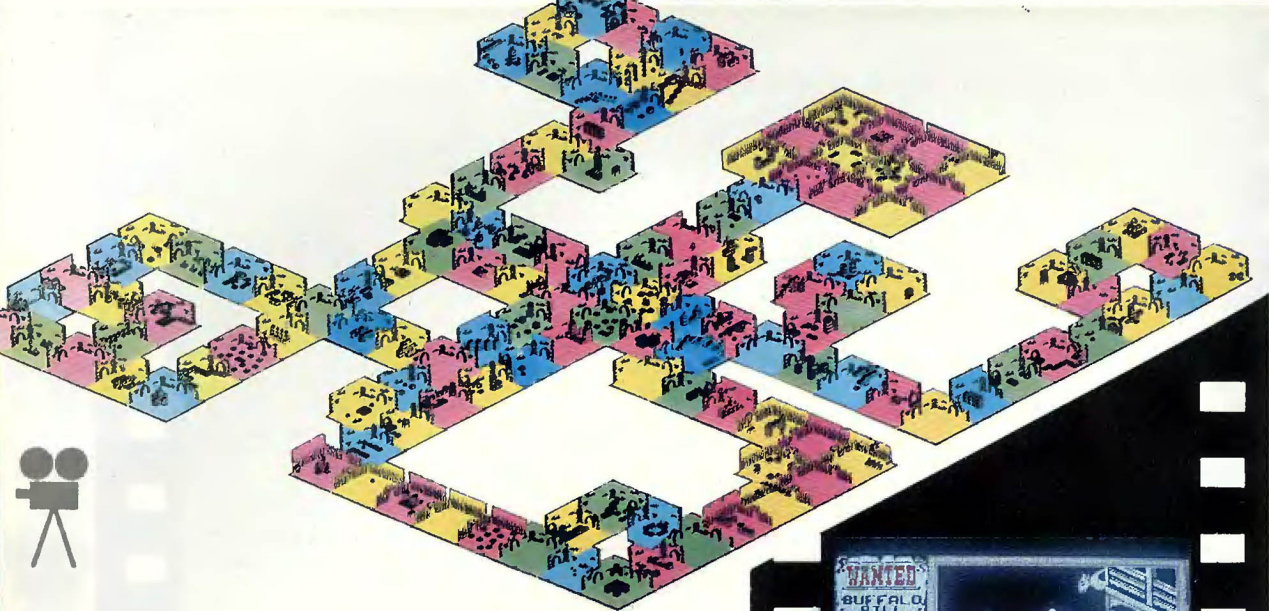
Para ello, deberemos preparar un espacio de trabajo en memoria en el cual colocaremos el trozo de pantalla donde se debe imprimir la máscara y el gráfico propiamente dicho, y otro espacio en el cual almacenaremos el trozo de pantalla que debe ser repuesto una vez el gráfico haya cambiado de posición.

Una vez preparados estos buffers de trabajo,

leeremos la porción de pantalla donde se va a imprimir el gráfico y la trasparemos a dicho buffer. Después colocaremos allí mismo la máscara del gráfico, mediante operaciones AND, con lo cual únicamente nos quedará la parte del trozo de pantalla donde el gráfico imprimirá, al mismo tiempo almacenaremos el contenido del buffer donde se va a colocar el gráfico, en otro espacio de memoria, para luego reponerlo.

Seguidamente colocaremos en ese mismo buffer el gráfico original, mediante operaciones OR, con lo que conseguiremos una copia de lo que debe aparecer en pantalla.

Ahora únicamente nos queda imprimirlo en



pantalla con la rutina correspondiente, que será una rutina de impresión normal, la cual traspasará dicho buffer de trabajo a la pantalla.

La longitud del buffer de pantalla, deberá ser la misma que la del gráfico que se desee imprimir, mientras que la dimensión del buffer de trabajo deberá ser la misma que la del gráfico, aumentando en dos la altura y la anchura (ver dibujo 2).

Veamos a continuación cuáles son los nuevos comandos RSX que nos permitirán la utilización desde Basic de la técnica *Filmation*.

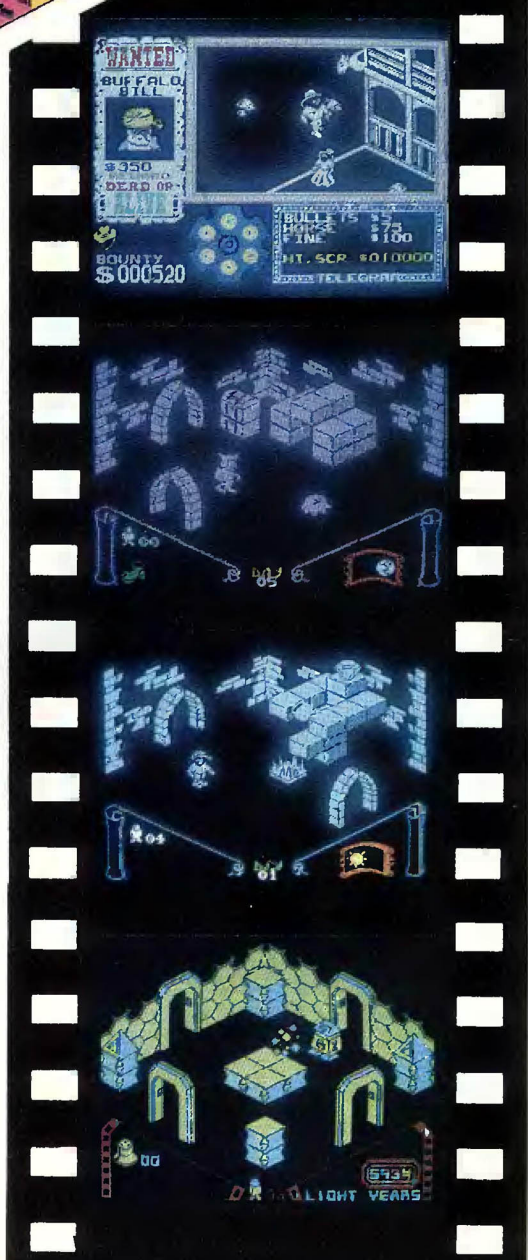
Como se ha comentado anteriormente, esta técnica se basa en la operación de impresión y borrado simultáneo del gráfico a imprimir; por tanto, los desplazamientos que se produzcan no pueden ser en ningún momento bruscos, es decir saltos de más de un carácter en cualquier sentido, ya que la rutina quedaría sin efecto al no reponer el trozo de pantalla en el que se encontraba el gráfico anteriormente.

En primer lugar tenemos el comando que nos permitirá inicializar la impresión de un gráfico,

ION,A,B,DM,DG,X,Y

donde cada uno de los parámetros corresponde:

- A Alto del gráfico
- B Ancho del gráfico
- DM Dirección de la máscara
- DG Dirección del gráfico
- X Coordenada vertical
- Y Coordenada horizontal



PROGRAMA DEMO

1 REM FILMATION

2 REM PROGRAMA DEMO

10 MODE 1:BORDER 0:INK 0,0:INK 1,24:INK 2,23:INK 3,6

20 MEMORY 9999:LOAD*pibubin*,80000

30 CALL 80000

40 GOSUB 300

50 ION,3,6,20000,20144,5,1

60 Y=5:X=1

70 Y=Y+1:X=X+1

80 IF Y=22 THEN X=X-1:Y=Y-1:GOTO 120

90 IF X=72 THEN Y=Y-1:X=X-1:GOTO 220

100 GOSUB 270

110 GOTO 70

120 Y=Y-1:X=X+1

130 IF Y=4 THEN Y=Y+1:X=X-1:GOTO 50

140 IF X=72 THEN Y=Y+1:X=X-1:GOTO 170

150 GOSUB 270

160 GOTO 120

170 X=X-1:Y=Y-1

180 IF Y=4 THEN X=X+1:Y=Y+1:GOTO 200

190 IF X=0 THEN X=X+1:Y=Y+1:GOTO 120

200 GOSUB 270

210 GOTO 170

220 X=X-1:Y=Y+1

230 IF Y=22 THEN X=X+1:Y=Y-1:GOTO 170

240 IF X=0 THEN X=X+1:Y=Y-1:GOTO 70

250 GOSUB 270

260 GOTO 220

270 !SPRITE,Y,X

280 FOR n=1 TO 60:NEXT

290 RETURN

300 P=0

310 FOR n=7 TO 15

320 FOR y=2 TO 39

330 P=P+1:IF P=4 THEN P=1

340 PEN p

350 LOCATE y,n:PRINT CHR\$(202)

360 NEXT y:NEXT n

370 RETURN



El segundo nuevo comando, que nos permitirá imprimir nuestro gráfico en pantalla es:

!SPRITE,X,Y

donde «X» e «Y» indicarán las coordenadas vertical y horizontal, respectivamente.

Para dar una idea del funcionamiento de nuestra rutina, hemos preparado un programa demostración, en el cual se podrá observar tanto el funcionamiento de los nuevos comandos, así como el movimiento que se obtiene con la utilización de la técnica *Filmation*.

Este programa de demostración utiliza un gráfico con su máscara, que deberán encontrarse

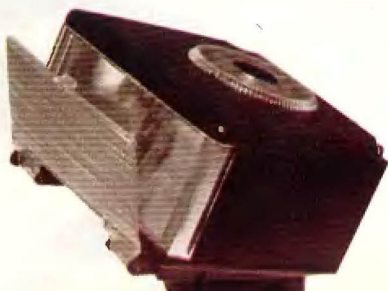


PROGRAMA ENSAMBLADOR

```

1 ;FILMATION
2 ;LISTADO ENSAMBLADORA
10     ORG #8000
20
30
40 ;***ROUTINA DE IMPRESION SISTEMA FILMATION***
50
60
70
80     LD BC,TABLA
90     LD HL,ESPACE
100    JP #BCD1
110 TABLA: DEFN NAME
120    JP INISPR
130    JP MOVSPR
140 NAME: DEFN "D"
150    DEFB "N"+#80
160    DEFM "SPRIT"
170    DEFB "E"+#80
180    DEFB 0
190 ESspace: DEFS 4
200
210
220 INISPR: LD E,(IX+0)
230         LD D,(IX+2)
240         LD L,(IX+4)
250         LD H,(IX+5)
260         LD (DIRGRA),HL
270         LD (POSANT),DE
280         LD (DIRMAS),DE
290         LD L,(IX+6)
300         LD H,(IX+7)
310         LD (DIRMAS),HL
320         LD C,(IX+8)
330         LD B,(IX+10)
340         LD A,B
350         LD (ALTO),A
360         LD A,C
370         LD (ANCHO),A
380         INC B
390         INC B
400         INC C
410         INC C
420         LD (DIMENS),BC
430         LD B,C
440         LD HL,0
450         LD DE,B
460 SUMBU: ADD HL,DE
470         DJNZ SUMBU
480         LD (LONG),HL
490         JP INIMAN
500
510
520 MOVSPR: LD L,(IX+0)
530         LD H,(IX+2)
540         LD (POSMAN),HL
550         LD DE,(POSANT)
560         LD A,E
570         CP L
580         JR NZ,NOMIS
590         LD A,2;PARADO
600         JR PONFL1
610 NOMIS: JR C,VADER
620         XOR A
630         JR PONFL1
640 VADER: LD A,I
650 PONFL1: LD (IZDER),A
660         LD A,D
670         CP H
680         JR NZ,NOMIS1
690         LD A,2
700         JR PONFL2
710 NOMIS1: JR C,VABA1
720         LD A,I
730         JR PONFL2
740 VABA1: XOR A
750 PONFL2: LD (ARRABA),A
760         LD (POSANT),HL
770         LD D,H
780         LD E,L
790         JP MOVMAN
800
810 #L+
820 ;***PINBUF***
830 #L-
840 ;INICIALIZACION
850 ;
860 ;DE-COORDENADAS DE IMPRESION
870 ;
880 INIMAN: LD HL,BUFFER
890         LD BC,(DIMENS)
900         LD A,2
910         LD (IZDER),A
920         LD (ARRABA),A
930         LD DE,(POSMAN)
940         CALL PANMEM
950         CALL PINMAS
960         RET
970 ;
980 ;MOVIMIENTOS
990 ;
1000 ;DE-COORDENADAS DE IMPRESION
1010 ;IZDER=0 IZQUIERDA
1020 ;IZDER=1 DERECHA
1030 ;IZDER=2 PARADO
1040 ;DIRGRA-DIRECCION GRAFICO
1050 ;DIRMAS-DIRECCION MASCARA
1060 ;
1070 MOVMAN: LD HL,BUFFER
1080         LD BC,(DIMENS)
1090         PUSH HL
1100         PUSH DE
1110         PUSH BC
1120         CALL PANMEM
1130         CALL TROZPAN
1140         CALL PINMAS
1150         CALL PINTB
1160         POP BC
1170         POP DE
1180         POP HL
1190         CALL PINBUF
1200         RET
1210 ;BC_COORDENADAS EN LA PANTALLA DE
1220 ; IMPRESION
1230 ;
1240 ;
1250 ;COLOCA TROZO PANTALLA EN
1260 ;EL BUFFER
1270 ;
1280 TROZPA: LD HL,BUFFER
1290         LD DE,(LONG);#ANCHO
1300         LD A,(ARRABA)
1310         CP 2
1320         JR Z,TROZE2
1330         CP 0
1340         JR Z,TROZE3
1350         ADD HL,DE
1360 TROZE2: ADD HL,DE
1370 TROZE3: EX DE,HL
1380         LD A,(IZDER)
1390         CP 2
1400         JR Z,TROZE1
1410         CP 1
1420         JR Z,TROZEM
1430         INC DE
1440 TROZE1: INC DE
1450 TROZEM: LD HL,BUFFAN
1460         LD A,(ALTO)
1470         SLA A
1480         SLA A
1490         SLA A
1500         LD B,A
1510 TRBU1: PUSH BC
1520         LD B,0
1530         LD A,(ANCHO)
1540         LD C,A
1550         LDJR
1560         INC DE
1570         INC DE
1580         POP BC
1590         DJNZ TRBU1
1600         RET
1610 ;
1620 ;PINTA MASCARA EN EL BUFFER
1630 ;Y LEE TROZPAN
1640 ;
1650 PINMAS: LD DE,(DIRMAS)
1660         LD HL,BUFFER
1670         LD BC,(LONG)
1680         ADD HL,BC
1690         INC HL
1700         LD IX,BUFFAN
1710         LD A,(ALTO)
1720         SLA A
1730         SLA A
1740         SLA A
1750         LD B,A
1760 MASBU: PUSH BC

```



en la dirección de memoria 20000.

Para cargar este gráfico, deberemos copiar el programa Basic generador del mismo y de su máscara y ejecutarlo, salvándolo a continuación en disco o cinta, teniendo en cuenta que cuando se desee utilizar el programa de demostración deberemos cargarlo en la dirección de memoria indicada antes.

1770	LD A,(ANCHD)	2390	LD HL,#C000-80	3010	DEC E
1780	LD B,A	2400	LD B,D	3020	ADD HL,DE
1790	MASB1: LD A,(HL)	2410	LD D,0	3030	LD DE,80
1800	LD (IX+0),A	2420	DEC E	3040	L_BUC: ADD HL,DE
1810	LD A,(DE)	2430	ADD HL,DE	3050	DJNZ L_BUC
1820	AND (HL)	2440	LD DE,80	3060	PDP IX
1830	LD (HL),A	2450	A_BUC: ADD HL,DE	3070	LD A,(ALT1)
1840	INC DE	2460	DJNZ A_BUC	3080	LD B,A
1850	INC HL	2470	PDP IX	3090	JR COLQ
1860	INC IX	2480	LD A,(ALT1)	3100	M_BUC: LD A,H
1870	DJNZ MASB1	2490	LD B,A	3110	AND 56
1880	INC HL	2500	JR COLDE	3120	CP 56
1890	INC HL	2510	E_BUC: LD A,H	3130	JR Z,M_PAS
1900	PDP BC	2520	AND 56	3140	LD A,H
1910	DJNZ MASB1	2530	CP 56	3150	ADD A,B
1920	RET	2540	JR Z,E_PAS	3160	LD H,A
1930 ;		2550	LD A,H	3170	JR COLQ
1940 ;PINTA FIGURA EN EL BUFFER		2560	ADD A,B	3180	M_PAS: LD DE,0080
1950 ;		2570	LD H,A	3190	LD A,H
1960 PINTB: LD DE,(DIRGRA)		2580	JR COLQ	3200	XOR 56
1970	LD HL,BUFFER	2590	E_PAS: LD DE,0080	3210	LD H,A
1980	LD BC,(LONG)	2600	LD A,H	3220	ADD HL,DE
1990	ADD HL,BC	2610	XOR 56	3230	COLQ: PUSH BC
2000	INC HL	2620	LD H,A	3240	PUSH HL
2010	LD A,(ALTO)	2630	ADD HL,DE	3250	LD A,(ANCHI)
2020	SLA A	2640	COLQ: PUSH BC	3260	LD B,A
2030	SLA A	2650	PUSH HL	3270	M_BUC1: PUSH BC
2040	SLA A	2660	LD A,(ANCHI)	3280	LD A,(HL)
2050	LD B,A	2670	LD B,A	3290	LD (IX+0),A
2060	F1GBU: PUSH BC	2680	E_BUC1: PUSH BC	3300	INC HL
2070	LD A,(ANCHD)	2690	LD A,(IX+0)	3310	INC IX
2080	LD B,A	2700	LD (HL),A	3320	PDP BC
2090	F1GB1: LD A,(DE)	2710	INC HL	3330	DJNZ M_BUC1
2100	OR (HL)	2720	INC IX	3340	PDP HL
2110	LD (HL),A	2730	PDP BC	3350	PDP BC
2120	INC DE	2740	DJNZ E_BUC1	3360	DJNZ M_BUC
2130	INC HL	2750	PDP HL	3370	PDP IX
2140	DJNZ F1GB1	2760	PDP BC	3380	RET
2150	INC HL	2770	DJNZ E_BUC	3390	FLYBAK: CALL #BC19
2160	INC HL	2780	PDP IX	3400	PUSH BC
2170	PDP BC	2790	RET	3410	LD BC,100
2180	DJNZ F1GBU	2800	:HL=DIRECCION DEL GRAFICO	3420	PAUS: DEC BC
2190	RET	2810	:DE=COORDENADAS Y*256+X	3430	LD A,B
2200	:HL=DIRECCION DEL GRAFICO	2820	:BC=DIMENSIONES	3440	OR C
2210	:DE=COORDENADAS Y*256+X	2830 ;		3450	JR NZ,PAUS
2220	:BC=DIMENSIONES	2840 ;		3460	PDP BC
2230 ;		2850	PANHEM: PUSH IX	3470	RET
2240 ;		2860	EX DE,HL	3480	
2250	PINBUF: PUSH IX	2870	INC H	3490	ALT1: DEFS 1
2260	CALL FLYBAK	2880	INC L	3500	ANCH1: DEFS 1
2270	EX DE,HL	2890	LD A,B	3510	ANGHO: DEFS 1
2280	INC H	2900	SLA A	3520	ALTO: DEFS 1
2290	INC L	2910	SLA A	3530	I2DER: DEFS 1
2300	LD A,B	2920	SLA A	3540	DIRGRA: DEFS 2
2310	SLA A	2930	LD (ALT1),A	3550	DIRMAS: DEFS 2
2320	SLA A	2940	LD A,C	3560	ARRABA: DEFS 1
2330	SLA A	2950	LD (ANCH1),A	3570	POSMA: DEFS 2
2340	LD (ALT1),A	2960	PUSH DE	3580	DIMENS: DEFS 2
2350	LD A,C	2970	EX DE,HL	3590	POSANT: DEFS 2
2360	LD (ANCH1),A	2980	LD HL,#C000-80	3600	LONG: DEFS 2
2370	PUSH DE	2990	LD B,D	3610	BUFFER: DEFS 560
2380	EX DE,HL	3000	LD D,0	3620	BUFFAM: DEFS 240



Esta figura la pondremos en disco o cinta de la siguiente forma:

SAVE" "BOLA",B,20000,288

Para poder ejecutar la rutina, deberemos en primer lugar teclear el programa cargador Basic, y ejecutarlo. En caso de que aparezca algún error, se deberán revisar las líneas DATA.

Una vez ejecutado sin errores, podremos salvarlo en disco o cinta de la siguiente forma:

SAVE" "FILMA",B,&8000.&22F

Cuando se desee utilizar, deberemos cargarlo en memoria a partir de la dirección &8000, haciendo una llamada a dicha dirección para inicializar los nuevos comandos RSX.