

BUFFER DE TECLADO

Cuando el Spectrum necesita leer un carácter por teclado, chequea en ese preciso momento si hay alguna tecla pulsada, y si la hay la utiliza. La base de este programa consiste en chequear constantemente el teclado para almacenar en el buffer las teclas que se vayan pulsando; así, cuando el ordenador requiera la pulsación de una tecla, lo primero que hará será comprobar si hay alguna tecla en el buffer para utilizarla.

Otra parte importante del programa es la que se encarga de sacar del buffer las teclas cuando son necesarias. Como sabéis, el Spectrum está basado en una estructura de canales en la cual para cada uno de ellos se usa una rutina para entrada y otra para salida de datos. Además existen una serie de corrientes asignadas a los canales que en principio son:

—#0, #1: impresión en la parte baja de la pantalla y lectura del teclado.

—#2: impresión en el resto de la pantalla.

—#3: impresora.

Pues bien, esta parte de la rutina es asignada a la parte de lectura del canal cero, es decir, a la lectura del teclado, con lo cual antes de leer una tecla se puede comprobar si hay alguna en el buffer.

La rutina funciona por interrupciones para conseguir que en cada instante se compruebe el estado del teclado, y está basada en las rutinas de la ROM que lo gestionan. La capacidad del buffer es de 15 teclas, y cuando esta cantidad es superada el ordenador emite un sonido para que el usuario sepa que está lleno y que por tanto lo que pulse a partir de ese momento no se

Una de las aplicaciones más interesantes que aportan los ordenadores personales del tipo compatible PC, —tan de moda últimamente—, es la de disponer de una memoria temporal, buffer, donde almacenar las teclas que pulsen. Para que el Spectrum no sea menos aquí tenéis un programa que posibilita esta opción.

Antonio BERMÚDEZ

almacenará.

Para activar la rutina hay que teclear: RANDOMIZE USR 65345

y para desactivarla:

RANDOMIZE USR 65365

Una vez activada, todas las teclas que se pulsen serán almacenadas para posteriormente ser utilizadas por el editor de la ROM. También sirve dentro de programas BASIC que usen la orden INPUT. Para INKEY\$ la utilización de la rutina es algo especial, ya que esta orden está diseñada para leer del teclado en el momento en que se ejecuta, pero esto se puede solucionar direccionando la lectura hacia el canal cero mediante INKEY\$#0, con lo cual en lugar de che

quear el teclado leerá del buffer.

Para vaciar el buffer desde un programa se debe volver a activar la rutina, es decir, hacer un RANDOMIZE USR 65345. Para vaciarlo cuando se está en el editor de líneas de la ROM basta con pulsar BREAK.

Para los conocedores de lenguaje máquina que deseen usar las interrupciones conjuntamente con el buffer de teclado, se ha incluido en el programa una llamada a una rutina externa, por tanto, para poder usarla se debe poner en las posiciones 65112 y 65113 de memoria la dirección de la rutina que se quiere ejecutar por interrupciones.

Para que podáis comprobar cómo funciona la rutina hemos incluido un programa de demostración cuyo uso es bastante sencillo si se siguen las instrucciones que aparecen en pantalla. Además podéis probar a pulsar, en modo «K», la tecla «T» (comando RANDOMIZE), sin soltarla hasta que oigáis el sonido que indica que el buffer está lleno, soltadla en ese momento y comprobaréis como el ordenador sigue imprimiendo RANDOMIZE durante un tiempo.

Como podréis comprobar con la demostración, la utilidad de esta rutina radica en que se pueden teclear líneas de programas, instrucciones, o bien introducir textos para un INPUT, para INKEY\$#0, etc., mientras el ordenador está realizando otra tarea.



UTILIDADES

LISTADO 1

```
10 REM Buffer de teclado
Antonio Bermudez. Granada
1990
20 CLEAR 64847: LOAD ""CODE 64
848,520
30 RANDOMIZE USR 65345
```

LISTADO 2

```
1 FDCB025EC41D11A7FDCB 1417
2 474EC2EAFDFDCB074E28 1411
3 08FDCB01DEFDCB01D6FD 1611
4 CB026E2811CD6E0DFDCB 1256
5 074E2008FDCB0196AF32 957
6 415C2AFBF5E233FEFBE 1339
7 7B280C46F04E07DFD5601 923
8 E5CD3303E132065CFDCB 1319
9 4786CD26FF22FBFECDC34 1499
10 FF2004FDCB47CE3A085C 1182
11 FE20021B11FE10302FFE 1159
12 06300A47E6014F781FC6 794
13 12182C2009216A5C3E08 428
14 AE771810FE0ED060021 1077
15 415CBE20043600180177 581
16 FDCB02DEBFC947E6074F 1459
17 3E10C85820013CFD71D3 1039
18 1104FEFDCB47D6C31311 1247
19 3A005C1150FDFDCB4796 1190
20 C3131100000000000000 231
21 000000F5E52A785C2322 797
22 785C7CB52003FD3440C5 1118
23 D5DDE52A4F5C23231150 1043
24 FDFDCB475628031104FE 1184
25 732372CDS7FFC065FEDD 1592
26 E101C1E1F1FBC9CD8E02 1894
27 C021005CCB7E20072335 773
28 2B200236FF7D21045C8D 829
29 20EECD1E030021005CBE 1031
30 CA16FFEB21045CBECA16 1257
31 FFCB7E2004EBCB7EC85F 1479
32 7233605233A095C7723 561
33 48C5FD4E07DFD5601E5C 1384
34 3303E177C179FDCB478E 1381
35 FE2020074F78FE272821 690
36 79FDCB474620382AFDCE 1355
37 72370CD26FF22FDFDCB 1510
38 34FF2004FDCB47CF6DCB 1524
```

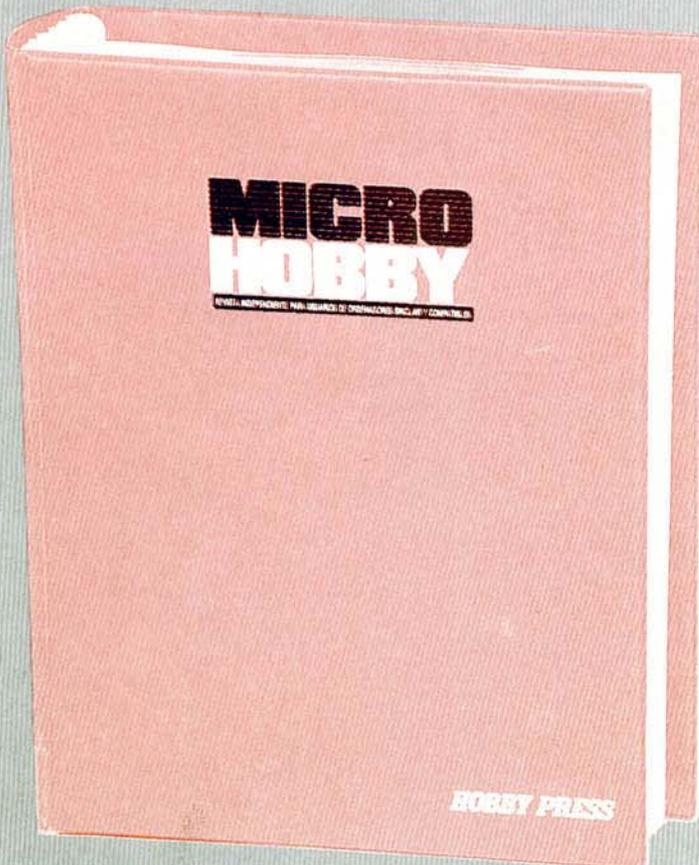
```
41 01EEC92111FE22FBFE22 1317
42 FDFEFD0CB47CEFD0CB4786 1901
43 FDCB01AEC307FE11FE11 1375
44 FE2FFE3A485C0F0F0F06 828
45 4BD3FEC5060010FEC1EE 1444
46 1010F4C92336052335C0 851
47 3A0A5C77237E06FE188E 866
48 23012FFE78BCC079BDC0 1339
49 2111FEC92AFBFEEED58FD 1633
50 FE7CBAC07DBBC92111FE 1573
51 22FBFE22DFEFD0364702 1460
52 3EFED047E05CEDE56C9 1650
53 00000000000000000000 0
```

DUMP: 40.000
N.º DE BYTES: 39

DEMO

```
10 REM Programa Demostracion
Antonio Bermudez. Granada
1990
20 CLEAR 64847: IF PEEK 64848<
>253 THEN LOAD ""CODE 64848,520
30 PAPER 7: BORDER 7: INK 9: C
LS
40 PRINT PAPER 2, BRIGHT 1: " D
emostracion Buffer de teclado "
50 PRINT "TAB 3: "Para comproba
r que las teclas que se pulsas so
n almacenadas y usadas posterior
mente en INPUTS teclasa cualquier
texto mientras se mantiene la c
uenta atras."
60 GO SUB 2000
70 PRINT AT 8,0: " ( Recuerda qu
e si pulsas mas de 15 teclas se
emitira un pitido y no se atas
cenaran mas teclas) ": GO SUB 100
```

```
0
80 INPUT AT 0,0:CHR$ 19+CHR$ 1
+"PULSA ENTER",, LINE A$
90 FOR A=2 TO 10: PRINT AT A,0
: NEXT A
100 PRINT AT 2,2: "A continuacio
n repite la misma operacion que a
nteriormente, con la diferencia
de que esta vez servira para de
mostrar como funciona el buffer
con el comando INKEY$#0."
110 GO SUB 2000
120 PRINT AT 9,0: " GO SUB 1000
130 PRINT AT 11,0: "Este es el t
exto que has escrito recuperado a
traves del buffer: (PULSA ENTER
)": PRINT
140 LET A$=INKEY$#0: IF A$="" T
HEN GO TO 140
150 IF A$<>CHR$ 13 THEN PRINT A
$: GO TO 140
160 FOR A=2 TO 15: PRINT AT A,0
: NEXT A
170 PRINT AT 2,2: "Por ultimo va
mos a ver como se pueden introduc
ir lineas BASIC aunque el orden
ador este ejecu- tando un progra
ma. Para lo cual debes fijarte b
ien en las teclas que tienes que
pulsar, que son: "
180 PRINT "TAB 3: "1EDEMO<ENTER>
200,0<ENTER>"
190 PRINT "TAB 2: "Con lo que mi
rando a la parte inferior de la
pantalla veras " como se introdu
cen las lineas: "
200 PRINT "1 REM DEMO""2 PLOT
0,0"
210 PRINT ""Pulsa una tecla pa
ra comenzar a teclear la serie i
ndicada."
220 GO SUB 2010
230 PRINT AT 19,0: "
240 POKE 23658,6: GO SUB 1000
250 STOP
1000 FOR N=999 TO 0 STEP -1: PRI
NT AT 21,0: N,CHR$ 32:CHR$ 32
1010 IF N=100 THEN PRINT AT 21,4
: "No pulsas mas teclas"
1020 NEXT N
1030 PRINT AT 21,0: " RETURN
2000 PRINT "" Pulsa una tecla pa
ra comenzar."
2010 RANDOMIZE USR 65345
2020 IF INKEY$#0="" THEN GO TO 2
020
2030 RETURN
```



COLECCIONA MICROHOBBY

**850
ptas.**

Rellena el cupón que aparece en la solapa de esta revista o bien llámanos al tel. (91) 734 65 00

