

# BUSCA-POKES HUMANO

por Amador MERCHÁN RIBERA



**En este artículo os ofrecemos un método rápido y eficaz para encontrar pokes; el único requisito es disponer de un «Multiface 3», de Romantic Robot.**

Lo único que tendréis que hacer es limitaros a buscar una serie de códigos o números que más adelante detallaremos. A tal fin seleccionaremos la opción «TOOL» (utilidades) de dicho interface y pondremos la pantalla en modo ventana («window») pulsando la «W». Si habéis seguido estos pasos veréis en la pantalla un rectángulo de dieciséis líneas de alto por ocho de ancho todo lleno de números. Si ahora pulsáis la «T» (texto), veréis esa pantalla llena de caracteres alfanuméricos. Es en esa pantalla donde deberéis buscar las secuencias a las que antes hicimos mención.

El formato que seguiremos es el de ofrecer el código objeto en lenguaje ensamblador (para que aquellos que tengáis ciertos conocimientos de C.M. sepáis dónde buscar), después el código objeto (que representa la secuencia de números a buscar) y después esa misma secuencia convertida en caracteres ASCII. Os recomendamos que busquéis en modo texto, pues la pantalla aparece menos recargada y es más fácil la búsqueda.

**Caso 1.** Número de vidas. En la gran mayoría de los programas (alrededor del 90-95% de los programas que hemos visto y/o destripado) la parte de código fuente encargada de tal evento reviste la siguiente forma:

```
LD A, NUMERO
LD (xxxxx), A
```

Donde número sería el número de vidas del juego y xxxxx una dirección de memoria. Por ejemplo, en la primera parte de «Rex» teníamos:

DIR. MEMORIA MNEMÓNICO	CÓDIGO OBJ
38129 LD A, 4	3E 04
38131 LD (41059),A	32 63 A0

Es decir, en la posición de memoria 38130 hay un 4. Si en vez de un 4 ponemos un 99, por ejemplo, dispondremos de ese número de vidas.

Este formato, como habéis podido intuir, sirve para inicializar cualquier otra variable, ya se trate del número de balas, del número de fase inicial, etc. Siguiendo con el ejemplo de la 1.ª parte de «Rex»:

DIR. MEM.	CÓD. FTE.	A BUSCAR	A BUSCAR ASCII
38134-35	LD A,3	3E 03	>
28186-87	LD A, 10	3E 60	+
38194-95	LD A, 96	3E 60	> #

Donde 3 es el número inicial de bombas y el 10 y el 96 las coordenadas iniciales del protagonista.

Es decir, con el «Rex» cargado en memoria iríamos buscando el grupo de caracteres de la columna de la derecha (>) y cambiaríamos el 3 de la posición 38135 por otro número. Echaos un par de partiditas (puede que no funcione a la primera) y comprobamos si el efecto causado es el deseado. Si es así, apunta la dirección, manda una carta donde ponga «Poke 38135, n = número de bombas» y únete al Club Microhobby. Si no, simplemente disfruta de un poke hallado por ti mismo.

**Caso 2.** Puntos iniciales. Si nos interesa tener puntos al comenzar la partida la secuencia a buscar suele ser:

```
LD HL, 000000      21 00 00      !..
```

En «Rex 1» las direcciones que contenían esta expresión eran la 38117, 38118 y 38119. Compruébalo y experimenta con diferentes números. La utilidad es inmediata: poner récord; pero

además hay juegos en los que cada cierto número de puntos te dan una vida. Aquí tienes otra ventaja de poder modificar la puntuación inicial.

**Caso 3.** Vidas infinitas y demás magnitudes cantorianas.

En los juegos se nos acaban las vidas con una facilidad pasmosa. Eso es un hecho irrefutable. El proceso es doble. Por un lado, cada vez que nos matan nos quitan una vida (lógico ¿no?). Eso se puede escribir así:

```
LD A (xxxxx); cargamos A con el número de vidas actual, que se halla guardado en la dirección xxxxx.
```

```
DEC A ; le restamos uno a ese número A := A - 1.
```

```
LD (xxxxx),A; guardamos en la dirección xxxxx el número de vidas actualizado.
```

Además, el programa se encarga de comprobar si el número de vidas se ha acabado, momento en el cual saltará a la subrutina que imprime el odiado «Game Over» y demás:

```
LD A, (xxxxx)
AND A ; podremos encontrar también XOR A, OR A e incluso CP 0.
```

```
JP Z, FIN ; si el número de vidas es cero, saltar a la parte del programa que se encarga del final. También puede variar y podemos encontrar RET NZ
JP FIN, por ejemplo.
```

Lógicamente, podremos encontrar estas dos subrutinas mezcladas en una sola:

```
LD A, (xxxxx)
DEC A
AND A
JR Z, FIN
LD (xxxxx), A
```

Es preferible buscar el «DEC A» puesto que suele funcionar siempre (si es el «DEC A» correcto, claro está). No ocurre así con el «AND A» o sus equivalentes, pues en algunos programas se efectúan varias comprobaciones del mismo tipo y si sólo cambiamos una de ellas es muy probable que no consigamos las vidas infinitas. En la práctica:

```
DEC A      3D      =
```

Además, es conveniente buscar en los alrededores:

```
LD A, (xxxxx) 3A num num: car car
```

```
LD (xxxxx), A 32 NUM NUM 2 car car
```

Obviamente, esto también es aplicable a los demás valores que, como el número de vidas, vayan decreciendo, como pueden ser el número de bombas (rex), balas (ikari w.), etc. Por cierto, debemos pokear la dirección del «DEC A» con un cero.

**Caso 4.** Plenilunios y número de fase. Hay juegos («Marauder», «Mad Mix», «Humphrey») que se desarrollan a lo largo de una serie de fases, completadas las cuales habremos acabado el juego.

La parte que gestiona esto y que nos interesa a nosotros es de la forma:

```
LD A, (xxxxx) ; cargamos A con el número de fase actual.
```

```
INC A ; incrementamos dicho número
```

```
A := A + 1.
```

```
LD (xxxxx), A ;lo volvemos a guardar.
CP nn ;lo comparamos con el número de fase máximo.
```

```
JP NC, LOGRADO; si hemos superado la última fase, acceder a la subrutina de juego terminado.
```

Como en el apartado anterior, puede sufrir ligeras variaciones.

Aparte de los códigos de carga (ver caso 3), debemos buscar principalmente:

```
INC A      3C      <
CP nn     FE num . car
```

Este último caso es la excepción que confirma la regla. Es mejor buscar el código «FE» que un punto, pues este último aparece en innumerables ocasiones. Debemos cambiar el número de fase máximo por el que nos interese a nosotros.

**Caso 5.** Coyunturas estructurales y subrutinas. En los programas que exhiben una programación estructurada («Wells & Fargo» es un buen ejemplo) existe un llamado bucle principal que se encarga de ir llamando a las distintas subrutinas que conforman el juego:

```
DI
LD SP, xxxxx
CALL yyyyy
CALL zzzzz
CALL ttttt
CALL vvvvv
...
```

Donde las xxxxx son distintas direcciones donde comienzan las distintas subrutinas que tratan el sonido, la comprobación de choques, actualización de marcadores, etc., etc. Si logramos anular el efecto de dichas subrutinas podemos conseguir cosas como juego sin sonido, inmunidad del protagonista, protagonista invisible y similares.

Esto puede conseguirse de dos maneras, pokeando con ceros las tres direcciones que ocupa una instrucción «CALL» o fijándonos en la dirección a la que llama e introduciendo el código 201 (C9h, código de «RET»).

Códigos a encontrar:

```
CALL xxxxx CD num num . car car
```

**Caso 6.** Paradas. Hay una instrucción cuyo mnemónico responde al nombre de «halt». Su misión consiste en detener el Z80 hasta que se produce una señal de interrupción y ésta se acepta. Hay que tratarla con cuidado pues su efecto es imprevisible. Puede proporcionar mayor velocidad al juego («Kung-Fu Master»), modificar la temporización de la impresión de sprites («Wells & Fargo») o provocar un «crash».

Para anular el efecto de «HALT», pokear la dirección donde se encuentre con un cero.

```
HALT      76      V
```

Por el momento, nada más. Creemos que tenéis material suficiente para empezar a foguearos con esto de los pokes. Esperamos vuestras cartas.