

# QL

MAGAZINE

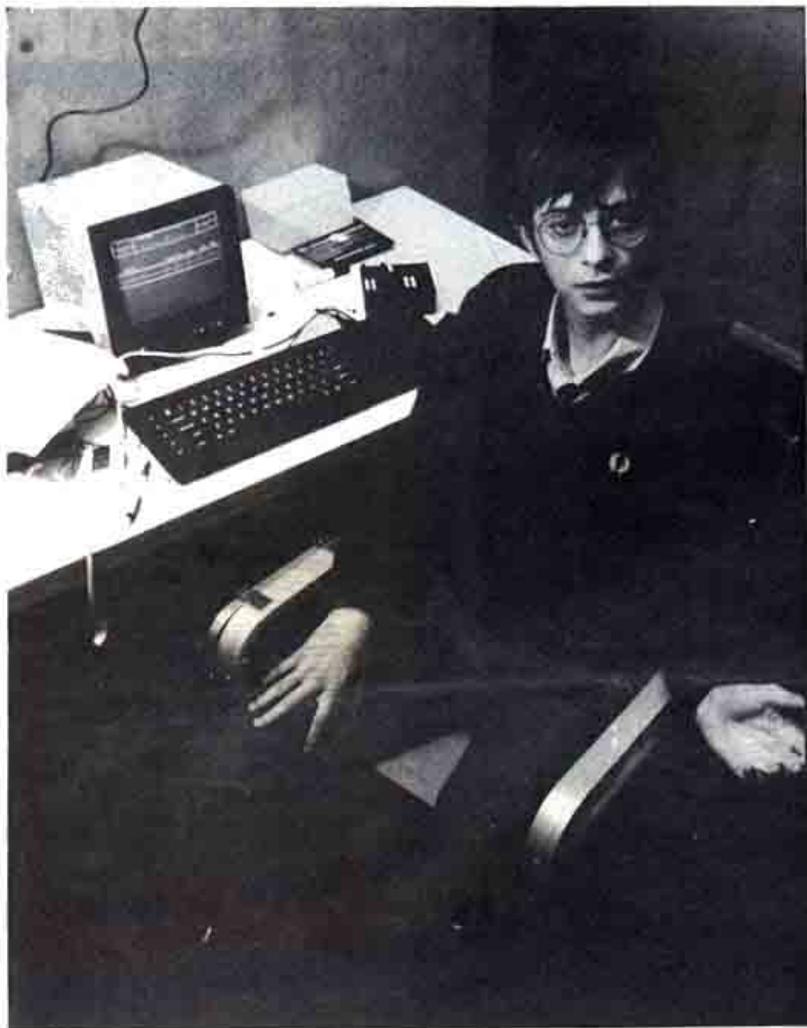
Suplemento especial Julio 1985

# QUILL

Procesador de textos



**LISTADO:**  
Como programar  
las teclas de función



## Programas a medida

Un ordenador, por muy bueno que sea, necesita buen *software* que lo acompañe. Los programas de **Psion** son muy completos, pero no suficientes. Conscientes de ello **Sinclair Store** ha puesto al frente del QL a **Juan Solsona**. A pesar de su juventud (17 años), lleva ya cinco años con los ordenadores y es un enamorado del QL.

En los últimos meses **Sinclair Store** ha ampliado sus operaciones a otros ordenadores, principalmente MSX, sin olvidar a la familia Sinclair ni al QL; como nos decía Juan: «Lle-

vamos cinco meses con el QL, principalmente con la contabilidad, a través de la base de datos del **Archive**. También estamos trabajando con el **Easel**, ya que los gráficos sólo se obtienen con impresoras compatibles **Epson**. También realizamos programas personalizados para uso concreto, como la topografía.»

En el plano personal, nos confiesa sus preferencias por el QL: «Eso lo tengo muy claro. Me compraré un QL y modificaré los programas de **Psion** para adaptarlos a mis necesidades.»

## ¿Para cuándo el QL?

Parece que Sinclair haya dado un cursillo de marketing a sus distribuidores en el extranjero. La comercialización de la versión «local» de la máquina, con una versión mejorada del QDOS, mensajes en castellano y teclado con acceso a los acentos y caracteres especiales, fue anunciada en el **Informat**, a mediados del mes de abril.

Algo parecido hizo Sinclair al anunciar la nueva máquina casi seis meses antes de tener una versión provisional en la calle. Según todos los analistas, la combinación de un retraso en las expectativas de los compradores con una primera versión comercial defectuosa han sido responsables del mal resultado inicial de esta máquina, que sólo mejora ahora, dos años después del primer anuncio.

Casi dos meses han pasado desde el **Informat**, aún no se han visto en España los nuevos ordenadores y el anuncio de su llegada definitiva se retrasa de semana en semana. Esperamos que, en el momento de leer estas líneas, el fallo se haya enmendado. También esperamos que no cunda el ejemplo y estas técnicas de marketing tan audaces no se sigan aplicando.

## Gestión Integrada

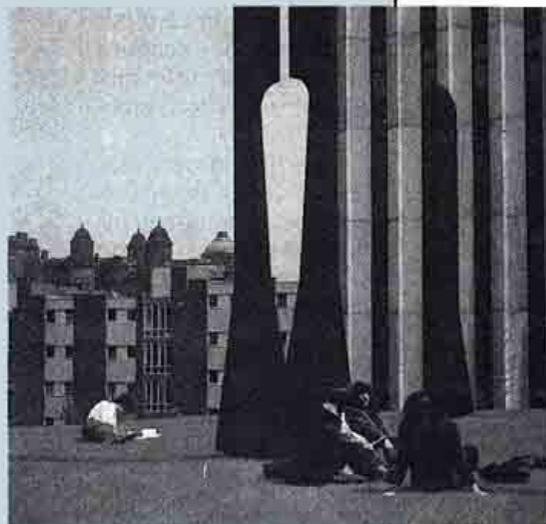
**Alsi Comercial**, que ya ha desarrollado programas de aplicación para Spectrum, anuncia su programa de gestión integrada para QL. Su programa permite el manejo de ficheros de clientes y artículos y facilita la realización de facturas, pedidos, ofertas y albaranes. Basta señalar el código de un artículo para que nombre y precio sean añadidos automáticamente a nuestro documento. Un programa que facilita el uso del QL en pequeñas empresas y comercios, sin necesidad de realizar programas específicos mediante ARCHIVE.

Esperamos presentar en un próximo número el análisis completo de este programa, y también que otras casas sigan el ejemplo de **Alsi** y desarrollen otros productos para el usuario profesional.

## QL en la Universidad de Strathclyde (Glasgow)

La Universidad de Strathclyde ha adquirido 7.000 QLs para la consecución de un ambicioso plan que viene a ser algo así como «Un estudiante... un QL». Los ordenadores se conectarán a un miniordenador VAX. Esta Universidad se caracteriza por sus avanzados estudios en materia de inteligencia artificial.

Lógicamente, Sinclair se



muestra particularmente orgulloso por la decisión y no menos generoso: 250.000 libras es la cantidad destinada a apoyar el proyecto (aproximadamente, 50 millones de pesetas).

## Comunicaciones vía QCOM

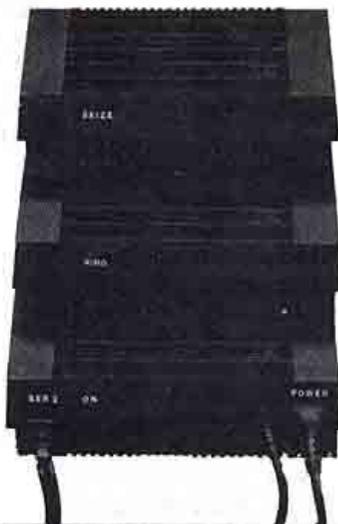
Con el QL todo parecen ser siglas más o menos inteligibles. QCOM es la denominación que reciben los tres periféricos que permiten la conexión del QL con el exterior, ya sea una base de datos o distintos ordenadores: QL-Interface, QL-Call y QL-Modem.

QL-Interface permite la conexión con «grandes ordenado-

res», pudiendo transmitir y recibir a una velocidad entre 75 y 9.600 baudios.

QL-Call une el ordenador a la línea telefónica, convirtiendo a su ordenador en una «pequeña central telefónica».

Finalmente, QL-Modem es un auténtico modem. Trabaja con una velocidad de transmisión 75/1.200 ó 1.200/1.200.



NOTICIAS

**S**in embargo, hoy en día los procesadores de texto son quizá los programas más utilizados, por su carácter general: ¿Quién no necesita preparar un informe, escribir una carta, etcétera?

Con los procesadores de texto desaparecen los borradores interminables, así como el proceso final de «pasar a limpio»; todo el trabajo se realiza sobre la pantalla, borrar no es ningún problema, se pierde el «miedo al error» ganando en velocidad y creatividad. La posibilidad de realizar cambios en cualquier estadio del desarrollo de un documento añade otra componente de libertad a nuestro trabajo.

Existen dos grandes grupos de procesadores de texto: los que trabajan en un formato adaptado al ordenador y después «dan forma» al documento de cara a la impresora, y los que muestran, dentro de lo posible, el texto como aparecerá en su forma final.

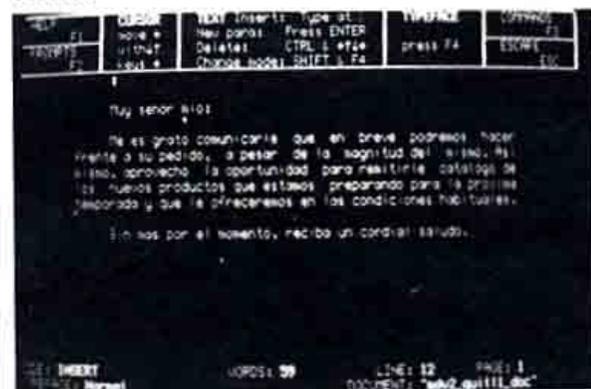
### QUILL, escribir sobre la pantalla

El procesador de texto del QL, pertenece a este segundo tipo. Los distintos tipos de letra y la marginación de las líneas se presentan en su forma final, de manera que la edición es más fácil, ya que los cambios introducidos se visualizan inmediatamente en la pantalla. La presentación se realiza con 40, 64 ó 80 caracteres por línea, debiendo elegir una u

# QUILL:

## Máquina de escribir inteligente

En la prehistoria de los ordenadores, allá por los 60, ningún profesional de la informática habría considerado interesante «desperdiciar» la potencia de un ordenador en una tarea tan poco importante como redactar o editar un texto.



otra según las características de nuestra televisión; con monitor se trabaja bien a 80 caracteres. Con televisión la máquina comienza por defecto a 64, pero si la calidad deja que desear se puede utilizar el *display* de 40 caracteres. El programa admite líneas de hasta 160 caracteres, más que suficiente para cualquier aplicación.

Si nuestro documento es más ancho que la pantalla, el programa nos mostrará una «ventana», que cubre

la mayor zona posible del documento, siempre alrededor del cursor, aunque, según introducimos texto, el programa lo lista en pantalla con el ancho disponible; al pulsar alguna tecla de movimiento de cursor se realiza el ensamblado definitivo del texto. Esto reduce el desagradable *scroll* lateral (que realizará automáticamente el programa cada vez que salgamos de la ventana visible) al mínimo, al permitirnos ver en la pantalla

el texto que estemos tecleando.

La ventana superior, de manera análoga a los demás programas de **Psion**, muestra las acciones que ejecutarán las teclas de función, así como información general sobre el funcionamiento del programa, desapareciendo si pulsamos F2, con lo que nos deja una presentación con 21 líneas útiles en la pantalla. En el formato inicial son 17 las líneas de texto que se nos enseñan.

Esta información se complementa, como los otros programas, con dos líneas de información en la parte inferior de la pantalla. Los comandos se introducen por elección en un menú, pulsando la tecla correspondiente a la inicial del comando, tras F3.

### Lentitud excesiva en las primeras versiones

La tecla F1 sirve para obtener ayuda relativa a los comandos y modos de trabajo del programa, y es uno de los casos en que el programa carga información del Microdrive. En las primeras versiones de QUILL el código no está optimizado, por lo que el programa resulta lento, debido al constante acceso a Microdrive, tanto para almacenar el documento como para cargar overlays, cada vez que pedimos la ejecución de un comando no residente en memoria.

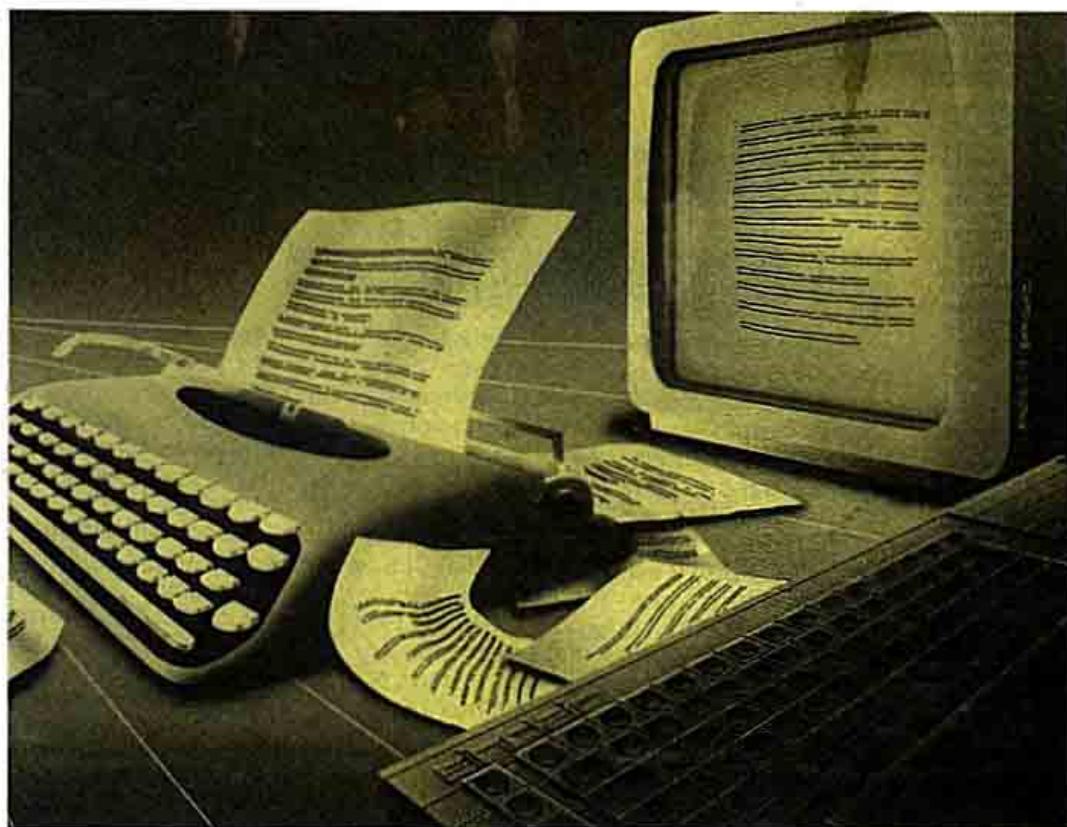
En la versión 2 este problema ha sido arreglado, ya que el programa ha sido optimi-

zado, quedando sitio suficiente para todos los comandos. Incluso el almacenamiento del documento en *micro-drive* sólo es necesario si el tamaño de nuestro documento sobrepasa unas mil quinientas palabras con 128K, pudiéndose trabajar en memoria en caso de ser menor.

### Trabajar en castellano sin problemas

Otra característica interesante de la última versión es la disposición de un programa BASIC que permite cambiar la correspondencia de caracteres de cara a la impresora; unido a la posibilidad de utilizar los caracteres extendidos del QL, hace esta versión útil para el usuario español, ya que las vocales acentuadas y la Ñ no plantearán más problemas con este programa. El programa permite también modificar los valores por defecto que tomará el programa para imprimir o almacenar: imprescindible para almacenamiento en *diskette* o impresión mediante *interface Centronics*.

Los usuarios de versiones anteriores se encontrarán en la imposibilidad de utilizar *diskettes* como medio de almacenamiento, además de verse obligados a imprimir en una impresora conectada al puerto serie. La configuración del programa puede modificarse para imprimir subíndices, superíndices, negrita y subrayado, faltando sólo la



cursiva para que el programa cubra las posibilidades habituales de las impresoras matriciales. Los tipos de letra citados se pueden insertar en cualquier texto con facilidad, y mezclarse libremente en la misma línea.

### Tan sencillo como escribir a máquina

Tras este breve repaso a las características generales del programa, vemos su funcionamiento. Una vez cargado podemos comenzar a teclear texto. El programa tiene dos modos de trabajo: el modo de INSERCIÓN, en el que se trabaja habitualmente, inserta el texto introducido en la posición del cursor. El modo de SOBREIMPRESIÓN cambia los caracteres presentes en

la posición del cursor por los que vayamos introduciendo. Este segundo modo es más lento que el anterior, y se recomienda no utilizarlo excepto en pequeñas correcciones. El texto se introduce tecleando en la posición deseada.

Si queremos introducir un comando debemos pulsar F3 y a continuación elegir en el menú de comandos. Los comandos se reparten en tres menús, un menú principal, el alternativo al que se accede pulsando O (otros), y un menú de ficheros (F en el menú alternativo). El conjunto de comandos es satisfactorio, aunque su modo de operación no sea siempre el más cómodo. Por ejemplo, el comando de borrado o copia de bloques no

permite retroceder una vez se ha comenzado a señalar el párrafo. Así, si nos «pasamos», no habrá más opción que pulsar ESCAPE y volver a comenzar. El comando SAVE nos pregunta al acabar si queremos seguir con el mismo documento, y si la respuesta es positiva, nos envía al comienzo, cuando lo normal sería que nos dejara en la posición en que decidimos salvar nuestro documento. Varios defectos como el indicado, además de una sospechosa tendencia a «colgarse», empañan las primeras versiones de este programa.

En la versión 2 se han corregido muchos de estos errores, y el programa resulta definitivamente «profesional».

APLICACION

# Teclas de función programables

Algunos ordenadores, como el QL, disponen de una serie de teclas llamadas «de función». Estas teclas suelen servir para simplificar el manejo de los programas comerciales. Pero, y eso es importante, también suele existir la posibilidad de programar una cadena de caracteres para que sea ejecutada por el ordenador cada vez que pulsemos esa tecla.

Las teclas de función programables harán su aparición en cuanto introduzcáis el programa que os presentamos. Escrito en código máquina, y con su propio cargador en BASIC, no hace falta ensamblador para introducirlo en la máquina.

El QDOS, sistema operativo del QL, permite extender con facilidad el BASIC de la máquina siempre desde Código máquina. Nuestro programa le añade a la lista de procedimientos residentes en la máquina uno llamado «tecla», con dos argumentos: un número y una cadena de caracteres, separados por una coma. Si, por ejemplo pulsamos

TECLA 1,  
"CLS"&CHR\$(10)

el ordenador borrará la pantalla cada vez que pulsemos F1. La concatenación de CHR\$(10) es necesaria: en caso contrario, el ordenador escribirá en la parte inferior de la pantalla CLS y se quedará esperando que pulsemos ENTER. Esta otra manera de utilizar el nuevo procedimiento es muy útil. Por ejemplo: TECLA 2, "LOAD MDV1\_" nos dejará

preparado el ordenador para que, pulsando F2 y el nombre de nuestro programa, podamos cargar cualquier programa en memoria.

Recomendamos salvar el programa en microdrive después de teclearlo y antes de intentar ejecutar. El programa irá imprimiendo en pantalla cada línea de DATA que «pokee» correctamente. Si aparecie-

ra algún error hay que detener el programa (CNTRL y SPACE) y repasar la línea incorrecta. Si no da ningún error, se puede proceder a probarlo. El programa BASIC 2 proporciona algunos ejemplos de su uso, que esperamos os sea útil.

Para acabar, queremos señalar que esta función se puede inhibir mediante la pulsación simultánea de CNTRL y CAPS LOCK, lo que evita problemas si cargamos programas que utilicen las teclas de función después de haberlas definido me-

## Programa BASIC 1

```

100 a=RESPR(706):RESTORE
110 FOR i=0 TO 299 STEP 20
120 READ a$
130 prueba=0
140 FOR j=1 TO 40 STEP 2
150 temp=conv(a$(j TO j+1))
160 POKE a+i+INT(j/2),temp
170 prueba=prueba+temp
180 END FOR j
190 READ prueba1:IF prueba<>prueba1 THEN
200 PRINT "Falla el 'checksum' en la línea";330+i/2
210 ELSE
220 PRINT 'Línea ';330+i/2;' correcta'
230 END IF
240 END FOR i
250 FOR i=300 TO 705
260 POKE a+i,0
270 END FOR i
280 CALL a
290 INPUT 'Quieres salvar el código máquina como bytes?';a$
300 IF a$(1)='s' OR a$(1)='S' THEN
310 SBYTES mdv1_pfprog_cde,a,706
320 PRINT "El programa se carga con a=respr(706):
lbytes mdv1_pfprog_cde,a:call a"
330 DATA "43FA001A347801104E9241FA002043FA00242149", 1562
340 DATA "0004701C4E414E750001008C055445434C410000", 989
350 DATA "000000000000000000000000000000000024790002804C226A", 503
360 DATA "0008204AD1FC00000010B3C86604226A00045389", 1440
370 DATA "47FA00C241FA00D2121167260C0100E2660E12BC", 1777

```



```

380 DATA "00E1740141FA00A8B5504E7543FA00A04A51660A", 2025
390 DATA "B2136708588BB1CB66F64E757600362B000249FA", 1998
400 DATA "009CD9C37800381C53446BEA121C307800E04E90", 1924
410 DATA "4A8066DE51CCFFF24E75347801164E92665E70F1", 2471
420 DATA "0C43000266567200740034369800123698020401", 988
430 DATA "00300C0100016D400C0100056E3A530152420242", 721
440 DATA "FFFD3C2343698020C4200506E26363C0052C2C3", 2065
450 DATA "41FA0036D1C130C24A42671252420242FFFE5342", 2148
460 DATA "30F6980454B951CAFFF870004E750000EB000000", 1996
470 DATA "EC000052F00000A4F40000F6F800014800000000", 1533
480 DEFine FuNction conv(a$)
490 LOCAL n1
500 n1=CODE(a$(LEN(a$)))-48
510 IF n1>9 : n1=n1-7
520 IF LEN(a$)=1 THEN
530 RETURN n1
540 ELSE
550 RETURN n1+16*conv(a$(1 TO LEN(a$)-1))
560 END IF
570 END DEFine conv

```

diente nuestro programa. Una nueva pulsación volverá a dejar las cosas como estaban. Si queremos eliminar alguna de las teclas, basta con asignarle una cadena vacía ("").

**NOTA TECNICA:** El programa trabaja encadenando una rutina en código máquina al programa que lee el teclado a cada interrupción. Cuando detecta la aparición de alguno de los caracteres correspondientes, inserta el comando programado. El programa inicial se encarga asimismo de ampliar el BASIC con un procedimiento (TECLA) que almacena la cadena de caracteres que se le pasa como segundo parámetro en el espacio asignado a la tecla indicada por el primer parámetro.

## Programa BASIC 2

```

290 TECLA 1, 'cls#0:cls#1:cls#2'&CHR$(10)
300 TECLA 2, 'load mdv1_'
310 TECLA 3, 'list'&CHR$(10)
320 TECLA 4, 'mode 4'&CHR$(10)
330 TECLA 5, 'mode 8'&CHR$(10)

```

PROGRAMAS