

44
185 pts.
IVA INCLUIDO

PULN

Enciclopedia Práctica del Spectrum



Nueva Lente/Ingelek



HOJAS ELECTRONICAS



Las «hojas electrónicas» son la solución informática de muchos de los problemas que se plantean en gestión. De hecho, podemos decir que no hay trabajo de cálculo que no pueda simplificarse enormemente con su ayuda. El cálculo de una nómina, una factura de venta o la confección de un presupuesto, son ejemplos claros de aplicación de este sistema.

Básicamente una hoja electrónica es una matriz de determinado número de filas y columnas que permite, en la mayoría de los casos, asignar un título extenso para cada fila y otro más reducido a las columnas.

Cada uno de los elementos de la matriz (casilla o celda) es referenciado por su fila y columna, correspondiendo generalmente caracteres alfabéticos a las filas y numéricos a las columnas.

Dentro de estas celdas pueden albergarse tres tipos de datos: constantes alfanuméricas, numéricas y variables calculadas a partir de los conte-

nidos de otras celdas empleando determinada fórmula.

Es el empleo de fórmulas lo que permite que, alterando los valores de determinadas celdas, pueda obtenerse de forma automática el nuevo contenido de otras. Estas fórmulas pueden relacionar cualquier conjunto de celdas por medio de operadores matemáticos, generalmente los elementales (+, -, * y /), aunque algunas hojas electrónicas permiten el empleo de cualquier función BASIC convencional.

En cuanto al número de filas y columnas, así como el ancho de estas últimas y su capacidad de almacenamiento, poco se puede decir de for-

Las hojas electrónicas admiten la entrada tanto de textos como de fórmulas, y por supuesto de los datos básicos para las fórmulas.

i!

VU-CALC es un programa apto para los modelos de 16 y 48 Kbytes RAM resultando una poderosísima herramienta en análisis financieros, presupuestos, cálculos de ingeniería o de tablas científicas, análisis estadísticos, etc...

*

Otra característica de las hojas de cálculo es la posibilidad de obtener subtotales y totales, tanto de filas como de columnas.

*

La base de datos de MINI OFFICE permite el mantenimiento de un fichero de datos constituido por cierto número de registros (fichas), cuya información se encuentra subdividida en campos (datos elementales).

| PULSE # para comandos, " para | | textos, datos y formulas directos | |
|-------------------------------|----------------------------|-----------------------------------|-------|
| 01 | 02 | 03 | 04 |
| A | GALLETAS LTD. PRESUEPEST | | |
| B | Ene. | | |
| C | ===== | | |
| D | VENTAS | | 11000 |
| E | PRECIO de ventas | | 1980 |
| F | ----- | | |
| G | BENEFICIO BRUTO | | 9020 |
| H | menos | | |
| I | Sueldo | 2010 | |
| J | Renta | 1931 | |
| K | Varios | 895 | |
| L | ----- | | |
| M | TOTAL GASTOS | | 4836 |
| N | ----- | | |
| O | Beneficio Comercial | | 4184 |
| P | Menos Interes | | 3723 |
| Q | ----- | | |
| R | BENEFICIO GANANCIA LIQUIDA | | |

i!

El procesador de textos de MINI OFFICE muestra el tiempo transcurrido desde el comienzo de la sesión, el número de palabras escrito y el de caracteres que restan antes que la memoria esté completa.

*

El empleo de fórmulas permite la obtención de forma automática del nuevo contenido de otras.

ma general, puesto que cada programa está dotado de características específicas.

Por último, diremos que otra de las características común a las hojas de cálculo suele ser la posibilidad de obtener subtotales y totales, tanto de filas como de columnas; aunque esta opción no es más que el caso restringido de la aplicación de una fórmula.

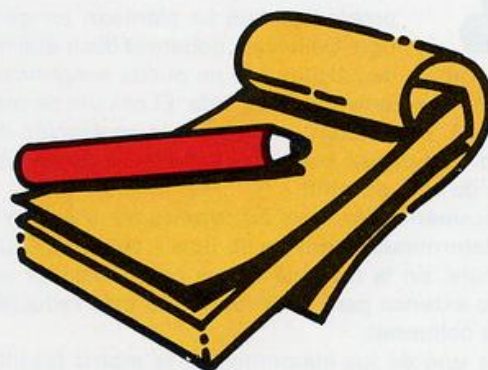
VU-CALC (PSION LTD.)

VU-CALC es un programa apto para los modelos de 16 y 48 Kbytes RAM resultando una poderosísima herramienta en análisis financieros, presupuestos, cálculos de ingeniería o de tablas científicas, análisis estadísticos, etc...

La pantalla VU-CALC está dividida en tres zonas: dos líneas de comando en la parte superior, un área en blanco en el centro y una línea para entrada de datos en la parte inferior. La zona central es una «ventana» de la tabla.

Las filas están representadas alfabéticamente y las que se encuentran en la ventana actual, pueden verse a lo largo del margen izquierdo de la pantalla. Las columnas se identifican por números sucesivos, comenzando por 01, y las que pueden verse en la pantalla actual se encuentran justo encima de la ventana de representación de la tabla.

La hoja electrónica puede venir en nuestra ayuda para la resolución de multitud de problemas de cálculo y administrativos.



Los problemas de administración son sin duda muy frecuentes.

Un rectángulo rojo denominado cursor identifica en todo momento la celda en que se está trabajando. Guiados por este cursor pueden leerse datos e introducir textos, fórmulas o cantidades, haciendo uso de las teclas normales de desplazamiento del cursor (flechas de la fila superior del teclado). Al rebasar el movimiento del cursor los límites físicos de la ventana, ésta se desplaza horizontal o verticalmente, en la dirección indicada, para mostrar nuevas zonas de la tabla, ocultas hasta este momento.

Para introducir un texto debe posicionarse el cursor sobre la celda o casilla donde se desea situarlo para, a continuación, pulsar comillas e introducir el texto en cuestión, finalizando la entrada con la pulsación de **ENTER**. La introducción de un valor numérico sigue las mismas normas que el caso anterior, no debiendo incluir las comillas.

La obtención de un resultado en determinada casilla implica la introducción de una fórmula, relacionando cualquier conjunto de casillas. Esta fórmula puede ser comprobada observando la línea de entrada de datos, y si resulta correcta, validarse definitivamente pulsando **ENTER**. Automáticamente, el resultado calculado pasa a ocupar la celda elegida.

Una fórmula puede ser aplicada en diferentes casillas por medio del comando **REPETIR**. Por medio también de esta tecla, se accede al modo co-

mando, apareciendo los disponibles en la zona superior de la pantalla, seleccionándose por la inicial.

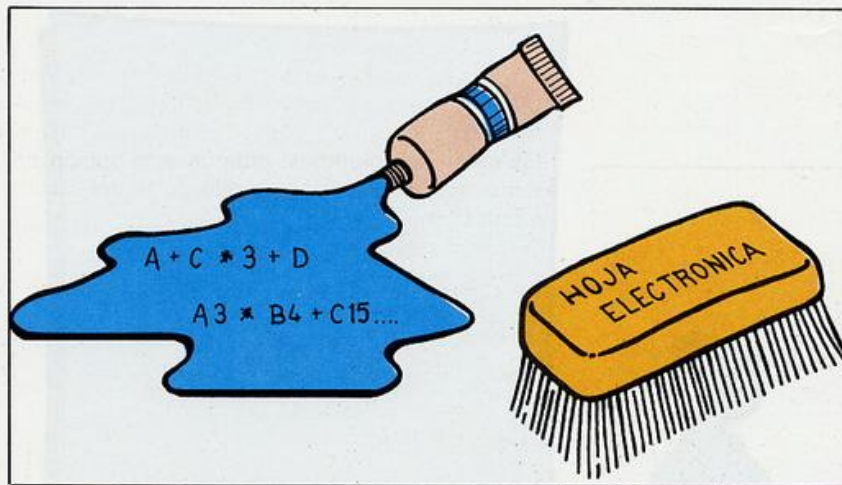
La sintaxis de una fórmula permite incluir constantes, referencias a otras casillas y operadores aritméticos simples: +, -, * y /. Una fórmula puede aplicarse a una celda en concreto o bien repetirse a lo largo de una fila o columna, o por todo un conjunto de ellas.

Las fórmulas se aplican siempre de una forma relativa. Si afecta a una secuencia de celdas de una fila, la columna de referencia se va incrementando siempre, automáticamente, a medida que se aplique la fórmula a lo largo de la fila. Así mismo, el concepto de relatividad es aplicable a columnas o bloques en los que las letras correspondientes a las filas se incrementarán secuencialmente.

En caso de que se desee designar de forma absoluta una celda de referencia en una fórmula, y para que ésta no varíe con el comando repetir, se debe anteponer el prefijo \$ a la celda en cuestión en el momento de escribir la fórmula.

Existe una gran variedad de comandos a los que podemos acceder anteponiendo el símbolo # como primer carácter, seguido de la inicial del comando elegido.

B vacía el contenido de la celda indicada por el cursor. # C recalcula los valores de todas las celdas, en base a las fórmulas correspondientes. # E permite sustituir una fórmula por otra nueva. # F,c,f,j especifica el formato de representación



Las hojas de cálculo admiten fórmulas con las cuales definir nuestro problema.

de los números en una columna de acuerdo con los parámetros c, f y j; el primero indica la columna donde aplicar el formato (A cuando el formato deba afectar a todas las columnas), el segundo el tipo de formato (enteros o reales con dos decimales) y el tercero indica si desea la justificación a la derecha (R) o izquierda (L).

G,r,c permite desplazar el cursor a la celda indicada a continuación. # L limpia la pantalla y carga el fichero que se especifique. # P efectúa el hard copy de la pantalla en impresora. # Q borra el contenido de la hoja dando opción de finalizar el programa.

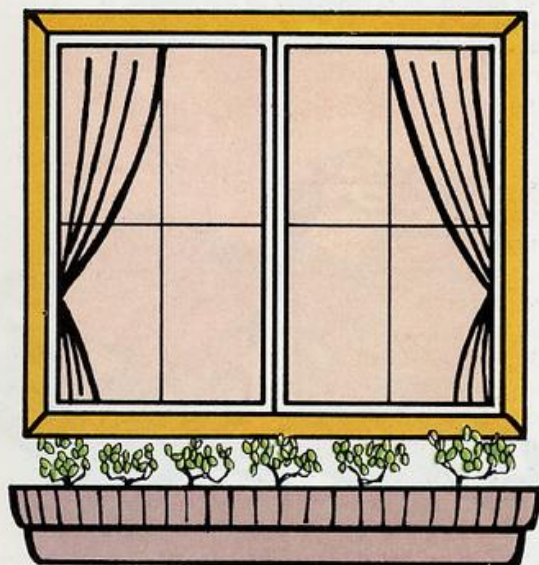
R,rc,f:l provoca la repetición de la fórmula contenida en la celda rc a lo largo del rango indicado. f y l indican las celdas de referencia, donde f corresponde a la celda situada más a la izquierda y arriba dentro del bloque y l es la situada más abajo y a la derecha; principio y final del bloque, respectivamente.

S limpia la pantalla y almacena el contenido de la tabla en casete. # T,r o c,r o c desplaza o copia el contenido de una fila o columna especificada por el primer parámetro a la fila o columna indicada por el segundo, no permitiéndose referir filas a columnas ni viceversa.

El sumatorio es semejante a una fórmula, permitiendo acumular automáticamente parte de una fila, columna o conjunto de celdas. Para ello basta con posicionarse en la celda en que se desea aparezca el resultado de la suma, introduciendo la fórmula con el formato: & f:l pulsando ENTER; donde f y l indican la primera y última celda del bloque, respectivamente.

Si el programa se ve interrumpido por causa de un error en los datos, puede continuarse la ejecución tecleando GO TO 9000 y ENTER.

La pantalla es una ventana a la hoja de cálculo.

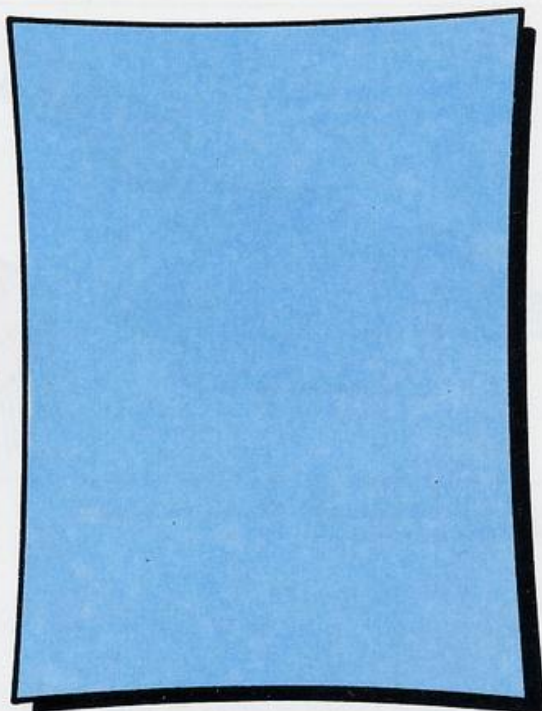


i!

MINI OFFICE está preparado para su uso con la impresora ZX PRINTER, o bien con interfaces de impresora que no ocupen memoria RAM, como la impresora SEIKOSHA GP50.

*

La pantalla VU-CALC está dividida en tres zonas: dos líneas de comando, un área en blanco en el centro y una línea para entrada de datos en la parte inferior.



Las dimensiones máximas de una hoja de cálculo suelen ser bastante superiores a las de cualquier hoja normal.

OMNI CALC

OMNI CALC es otra de las posibilidades que se presentan para el manejo de información en forma de tabla. Su característica más relevante es la de poder definir, dentro de la lógica limitación de la cantidad de memoria RAM disponible, el número de filas y columnas que se desean.

Este hecho redundará en un mejor aprovechamiento de la memoria, ya que en determinadas aplicaciones puede mantenerse un gran número de filas a costa de reducir el de columnas, como es el caso del mantenimiento de una ficha de cuenta corriente con datos de fecha, movimiento, Debe, Haber y Saldo.

Al cargar el programa en memoria se ofrece la opción de definir una nueva «hoja de trabajo», o de leer una ya creada almacenada en casete. En el primer caso, el programa solicita número de filas y columnas, efectuando la verificación de que el formato indicado tiene cabida en la memoria

del ordenador, volviendo a la petición de los datos en caso contrario. Por otro lado, en el caso de seleccionarse la lectura de un fichero ya existente, el programa pide el nombre de éste y lo carga en memoria.

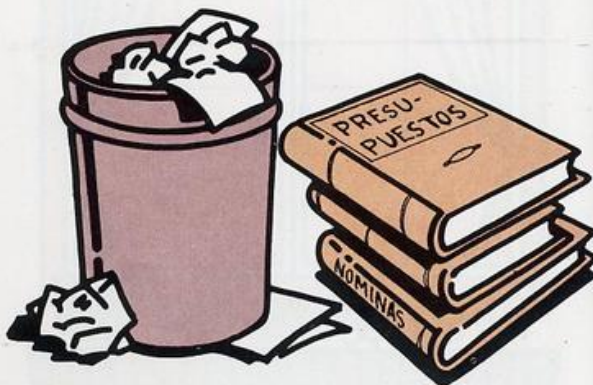
Como es norma general en las «hojas electrónicas» se tiene acceso permanente a una ventana por la cual podemos desplazar un cursor por la pulsación de las teclas **5** a **8**. Con el uso de las mismas teclas en combinación con **CAPS SHIFT**, se consigue el desplazamiento de la ventana a lo largo y ancho de la tabla.

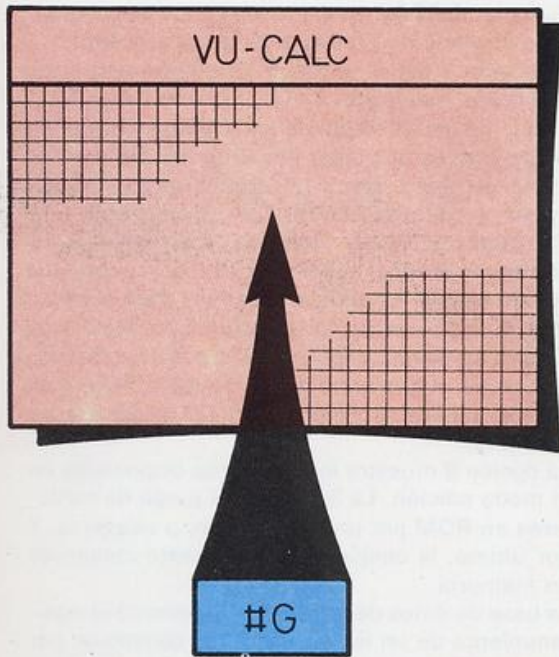
En cada una de las pantallas, además de la porción de datos que podemos ver (bloque de casillas), aparecen permanentemente unos títulos asociados a cada una de las filas, así como los correspondientes títulos de cada columna que se muestra en pantalla.

En cualquier momento, pulsando **H** (*Help*) aparece en pantalla una lista de los comandos que se pueden emplear, los cuales son invocados por su inicial.

C (*Calculate*) recalcula la hoja. **D** (*Decode*) muestra la fórmula asociada a una celda. **G** (*Goto*) desplaza el cursor a la celda indicada. **I** (*Input*) permite introducir textos, valores y fórmulas. **L** (*Load/init*) carga o inicializa la tabla. **M** (*Mode*) permite seleccionar entre la representación de valores en formato entero o dos decimales. **P** (*Print*) copia el contenido de la pantalla o la tabla entera en impresora. **R** (*Repeat*) generaliza una fórmula para un conjunto de celdas. **S** (*Save*) graba en casete el contenido de la tabla bajo el nombre que se indique. **T** (*Total*) calcula subtotales o totales de cualquier fila o columna.

Por ejemplo, en la confección de una nómina, el programa de hoja electrónica puede realizar desde su cálculo hasta su impresión.





El comando #G del programa VU CALC nos permite posicionarnos directamente en cualquier celdilla de la hoja.

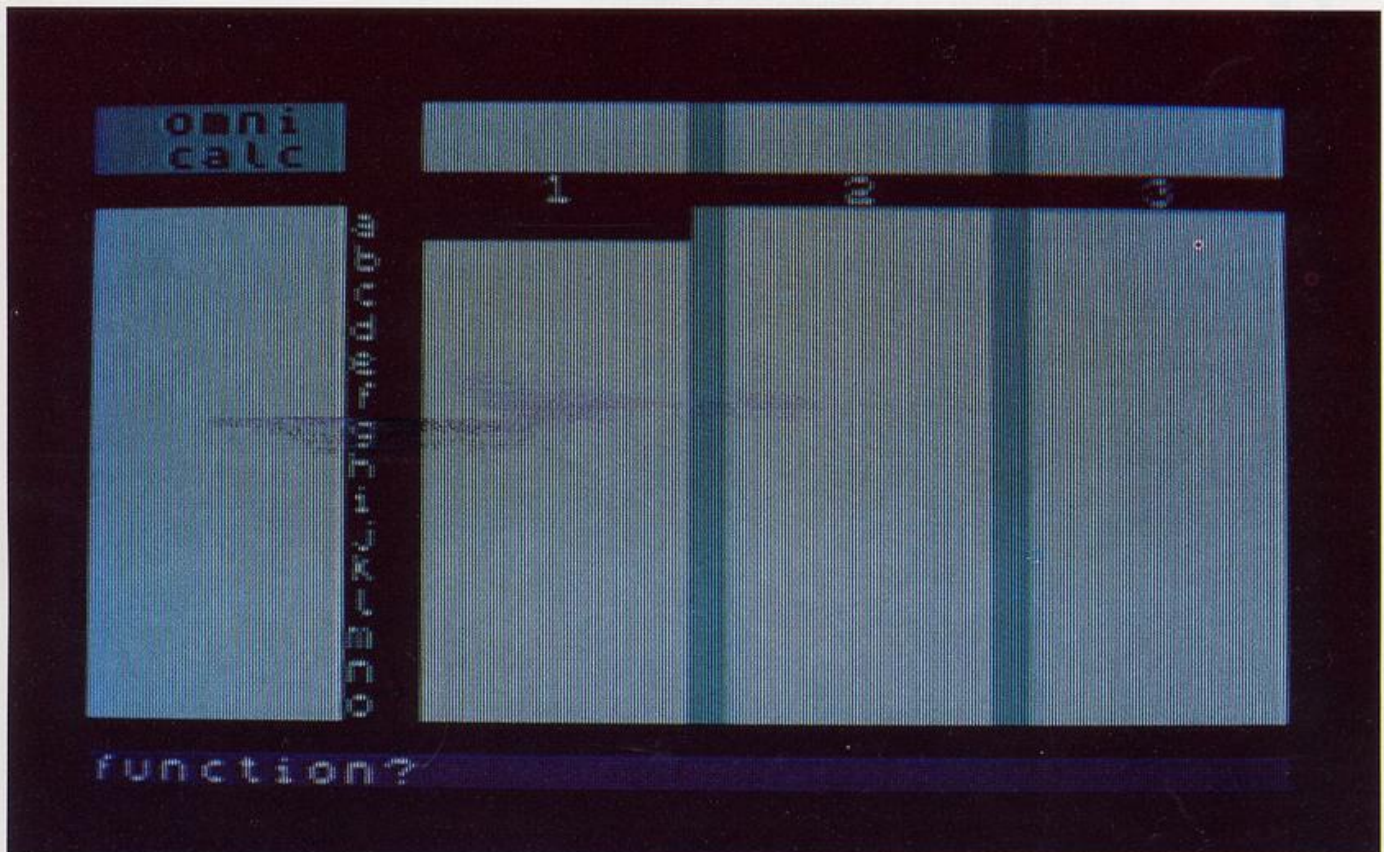
La casilla en que nos encontramos viene señalada por un CURSOR; una franja en el caso de OMNICALC.

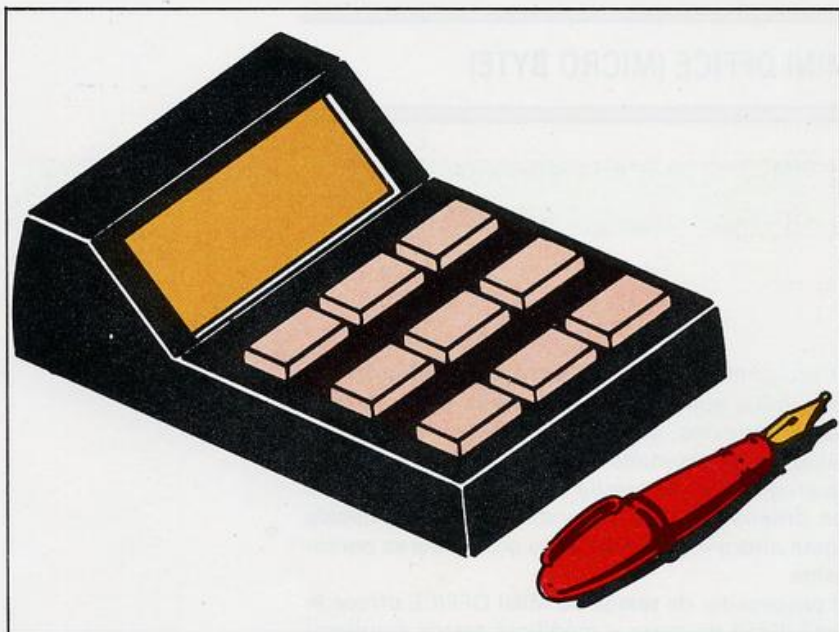
MINI OFFICE (MICRO BYTE)

El programa MINI OFFICE trata de aglutinar en una única aplicación el desarrollo de un procesador de textos, una base de datos, una hoja de cálculo y un diseñador de gráficos. Este conjunto de programas, de amplia difusión en el campo de los ordenadores profesionales, ha permanecido hasta ahora inaccesible a los ordenadores personales.

El procesador de textos de MINI OFFICE ofrece la posibilidad de crear y modificar textos empleando 16 ó 32 caracteres por línea, consiguiendo una gran calidad. En la parte superior de la pantalla se muestra el tiempo transcurrido desde el comienzo de la sesión, el número de palabras escrito hasta el momento y el número de caracteres que restan antes que la memoria esté completa.

Al comenzar con un nuevo texto aparece en su





La hoja electrónica nos evita el engorroso trabajo a base de calculadora y lápiz.

i!

Las «hojas electrónicas» son la solución informática de muchos de los problemas que se plantean en gestión.

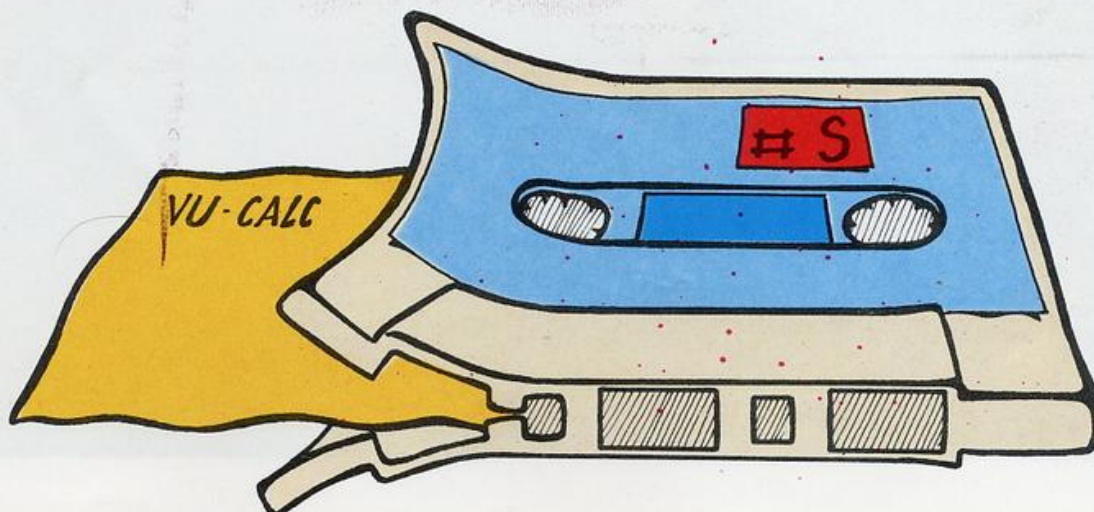


El cálculo de una nómina, una factura de venta o la confección de un presupuesto, son ejemplos claros de aplicación de las «hojas electrónicas».

cabecera las palabras COMIENZO y FIN, y entre estos dos límites se va produciendo la entrada y edición del texto.

Por medio de las teclas de control del cursor, se puede acceder a cualquier punto del texto para su modificación o inserción, pudiendo hacer uso de la tecla **DELETE** para eliminar caracteres.

El comando #S almacena el contenido de la hoja VU CALC en casete.



Pulsando **BREAK** en el modo edición se tiene acceso al menú de 10 opciones del procesador.

La opción 1 activa el juego de caracteres de ancho doble y la opción 2 el de tamaño normal.

Con la opción 3 se puede seleccionar el número de caracteres por línea (5 a 32), así como el tamaño de éstos, tanto horizontal como verticalmente, sobre una ZX PRINTER o compatible.

Las tabulaciones se fijan con la opción 4. La 5 permite la grabación del texto con el nombre que se indique (10 caracteres máximo). Para verificar la grabación se emplea la opción 6. La lectura de fichero se lleva a cabo con la opción 7, teniendo en cuenta que la lectura de un nuevo fichero en casete implica la destrucción del residente en memoria.

La opción 8 muestra las funciones disponibles en el modo edición. La 9 cambia el juego de caracteres en ROM por uno alternativo, o viceversa. Y por último, la opción 0 borra el texto contenido en memoria.

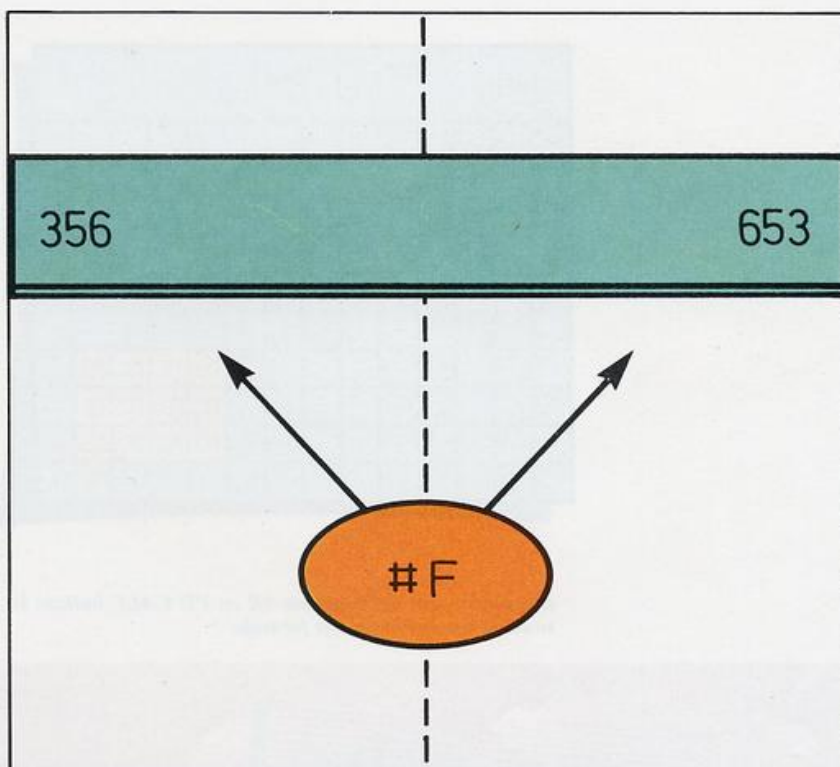
La base de datos de MINI OFFICE permite el mantenimiento de un fichero de datos constituido por cierto número de registros (fichas), cuya información se encuentra subdividida en campos (datos elementales) numéricos o alfanuméricos, permitiendo la clasificación del fichero por cualquiera de sus campos.

La primera información que se suministra al programa es el número de campos de que se compone el registro. El siguiente paso es dar nombre a cada uno de los campos especificando al mismo tiempo la clase de datos que albergarán (numéricos o alfanuméricos), así como el número máximo de caracteres por campo, indicando además, en los numéricos, el número de cifras decimales.

Una vez definido el formato del fichero pueden comenzarse a introducir datos (opción A), listar registros por pantalla o impresora (opción F), efectuar búsquedas de determinado dato en un campo (opción G), reemplazar contenidos de campos por uno especificado (opción H), ordenación por uno o varios campos (opción I), suma de valores de un campo (opción D) o grabación del fichero (opción J).

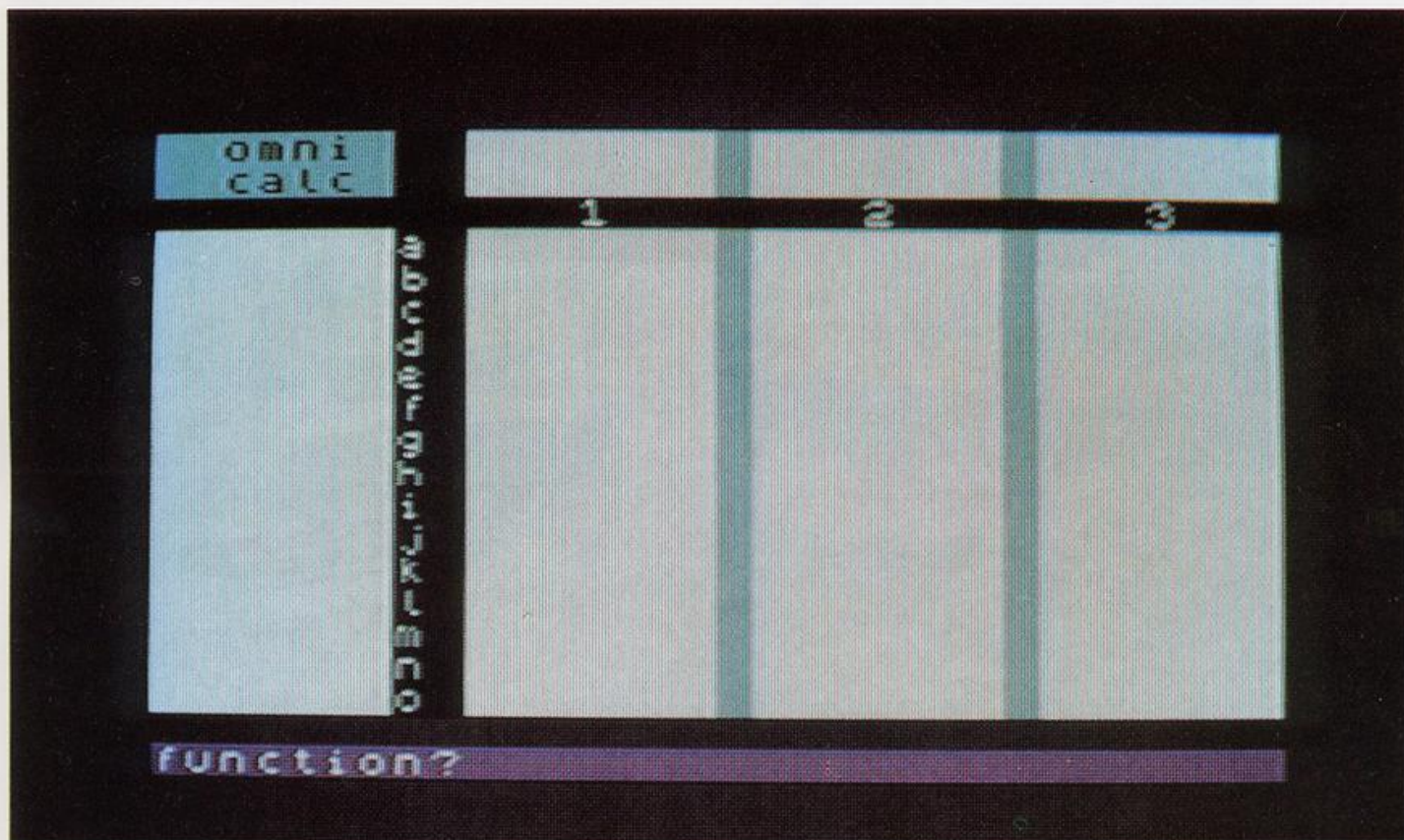


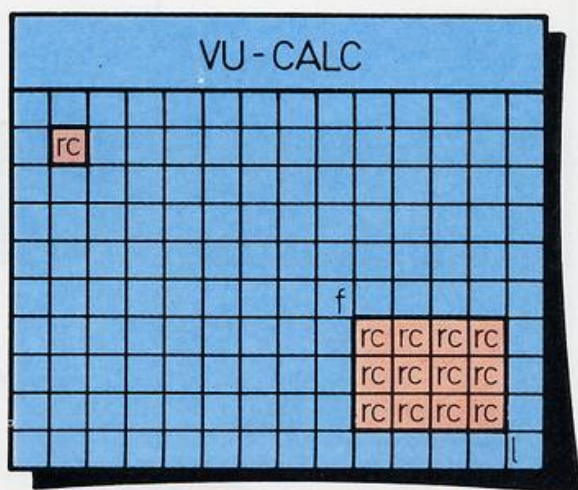
La opción F permite además examinar los registros del fichero, indicando qué registros deben visualizarse o imprimirse. Pulsando N es posible acceder al registro siguiente, pulsando D eliminar el registro en pantalla, pulsando A modificar su contenido, regresando al Menú pulsando M. La opción de búsqueda por campo permite localizar cualquier registro rápidamente sin necesidad de visualizar la totalidad del fichero. Debe señalarse qué campo se desea buscar. Las opciones disponibles a partir de entonces dependen de si se trata de un campo alfanumérico o numérico. En el primer caso puede escogerse entre la búsqueda de un campo determinado o parte de un campo, buscar un valor determinado, valores mayores que uno dado, menores, o entre límites. La hoja de cálculo permite incluir encabezamientos para las columnas (opción C), copiar fórmulas de una casilla a otra (opción D), mostrar la fórmula contenida en una celda (opción E), introducir una nueva fórmula (opción F), cargar el programa de gráficos (opción G), colocar el cursor al principio de la hoja (opción H), listar las fórmulas de la hoja (opción L), restaurar número anterior



La realización de totalizaciones es una de las características más empleadas de las hojas electrónicas.

Para mejorar la salida impresa de VU CALC, los números pueden aparecer justificados a izquierda o derecha con el comando #F.





Los parámetros del comando #R en VU CALC indican la zona de repetición de una fórmula.

(opción O), imprimir (opción P), inicializar (opción Q), introducir nombre de fila (opción R), grabar fichero (opción S), introducir título hoja (opción T), recalcular (opción U), borrar una fórmula (opción W) y borrar los valores de la hoja (opción Z).

El programa de gráficos requiere la organización previa de una archivo con la hoja de cálculo. Hecho esto, se puede obtener la interpretación de los datos contenidos en la hoja en forma analógica.

MINI OFFICE está preparado para su uso con la impresora ZX PRINTER, o bien con interfaces de impresora que no ocupen memoria RAM, como la impresora SEIKOSHA GP50.



El programa MINI OFFICE constituye una gestión integrada que incorpora procesador de textos, base de datos, hoja electrónica y generador de gráficos.





LOS INDICADORES



asta este capítulo los nuevos conceptos se acumularon uno tras otro, sin una explicación que clarificara su aparición. Como ya sabemos no era capricho: todo lo contrario.

Sin ellos sería imposible obtener provecho de todo lo siguiente. Es más, tengamos en cuenta que el significado de muchas instrucciones está relacionado con varios de estos conceptos, y sería complicado además de repetitivo establecer sin su conocimiento previo, un método que expusiera claramente qué es lo que hace determinada orden.

Por estas razones, nuestro esfuerzo inicial, como en adelante comprobaremos, quedará sobradamente recompensado y no nos cansaremos de recordar, aun a costa de resultar redundantes, la importancia fundamental de la lectura atenta y detallada de los capítulos anteriores antes de seguir adelante.

Como observaremos a partir de ahora, además de discutir el objetivo de una determinada instrucción y su funcionamiento, encontraremos otros datos de interés como son los referentes al tiempo de operación y las posibles inferencias de ésta con el registro F.

Aunque estos temas fueron tratados también anteriormente, no está de más profundizar sobre

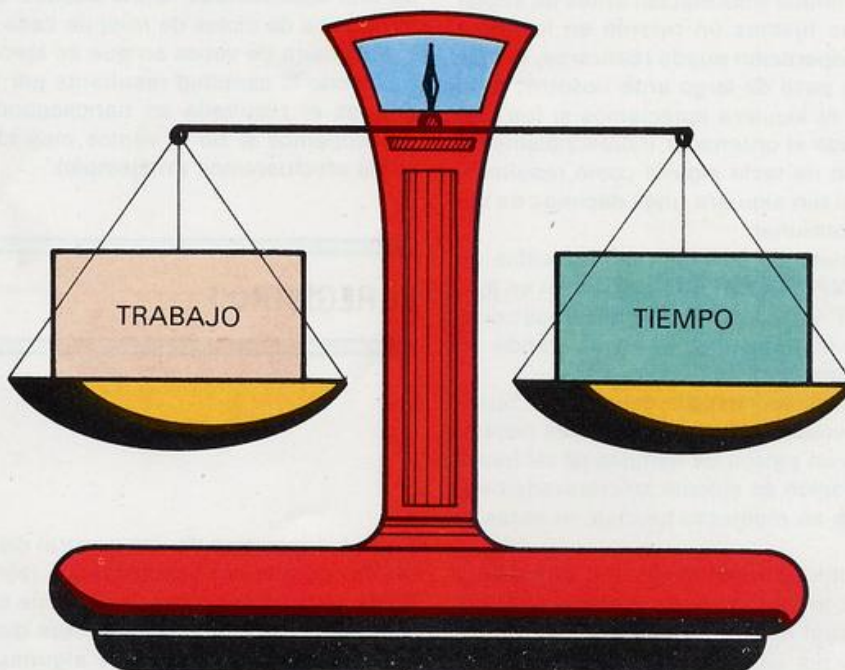
ellos con tal de familiarizarnos con su significado, pues su participación en cualquier rutina en C/M es de importancia capital.

CUESTION DE TIEMPO

El factor «tiempo», cuando se trata de programas escritos en lenguaje máquina, es fundamental. Por dos razones: la primera justifica si nuestra inversión de trabajo al construir una rutina en concreto ha sido acertada, es decir, si esta es lo suficientemente rápida como para emplearla en lugar de una idéntica codificada en BASIC.

Si el proceso seguido es correcto, la considera-

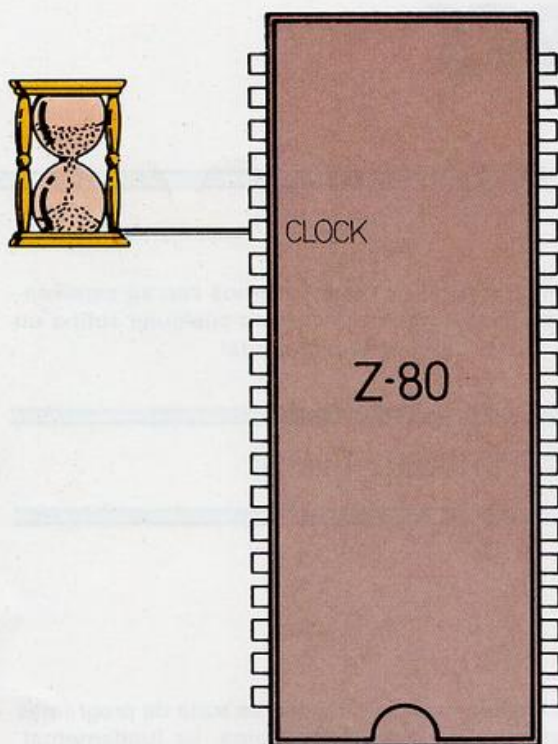
Antes de decidirse por la ejecución de una rutina en C/M es importante sopesar si el trabajo está en equilibrio con el beneficio obtenido.



i!

El factor tiempo, cuando se trata de programas escritos en C/M es fundamental.

Al utilizar el cargador hexadecimal emplea el modo mayúsculas para teclear los datos en la sentencia DATA.



A través del pin número 6 recibe el A 80 la señal de reloj.

i!

El registro F tiene como misión revelarnos determinados acontecimientos que se producen tras la ejecución de una instrucción.

Si tras ejecutar una instrucción, el bit de CERO (Z) está a cero, quiere decir que el resultado de tal operación NO fue cero.

El Z80-A está capacitado para funcionar con una frecuencia de hasta 4MHz.

Este cronometraje lo efectúa un oscilador de cuarzo, el cual entrega para nuestro micro tres millones y medio de señales cada segundo (técnicamente se dice que está regulado a 3,5 MHz), denominándose cada una ciclo de reloj o estado, recibidas en el Z80-A a través de su pin número 6 (CLOCK, reloj).

En realidad el Z80-A está capacitado para funcionar hasta con una frecuencia de 4MHz, pero los fabricantes eligieron la anteriormente señalada al considerar que si se llevaba al límite podría constituir una posible fuente de errores.

Una sencilla división nos muestra el tiempo empleado por un ciclo de reloj:

$$\frac{1}{3500000} = 2,85 \times 10^{-7}$$

es decir, 285 nanosegundos (segundos divididos por 10^9).

Se conoce como ciclo de instrucción al tiempo utilizado por el microprocesador en ejecutarla completamente. Dicha operación está compuesta por otras más elementales, denominándose al tiempo empleado en cada una de ellas ciclo de máquina.

Ha de quedar claro para todos que estos tiempos son diferentes según la instrucción particular o la operación elemental en cuestión. Así, por ejemplo, el ciclo de máquina durante el cual el microprocesador busca el código de operación de la instrucción toma 4 ciclos de reloj, mientras que las operaciones de lectura o escritura en la memoria (ciclos de memoria) ocupan 3 de reloj.

Así pues, para determinar el tiempo de ejecución de una determinada rutina bastará con conocer el número de ciclos de reloj de cada instrucción y el número de veces en que es ejecutada. Multiplicando la cantidad resultante por 285, obtendremos el resultado en nanosegundos (no nos preocupemos si no lo vemos muy claro. En seguida efectuaremos un ejemplo).

ción anterior se cumplirá en la inmensa mayoría de los casos, pero a veces, no siempre constituirá una ventaja, sino tal vez, todo lo contrario. Imaginemos un programa que necesite recibir del teclado determinada información antes de seguir adelante. Si no fijamos un retardo en la rutina para que esta operación pueda realizarse, lo más seguro es que pase de largo ante nosotros a tal velocidad que ni siquiera apreciemos si fue ejecutada, tomando el ordenador indefectiblemente la no pulsación de tecla alguna como resultado, al no darnos ni tan siquiera unas décimas de segundo para reaccionar.

Recordemos que instrucciones tan sencillas en BASIC como INPUT o PAUSE, no existen en lenguaje máquina, y para simularlas debemos construir la rutina correspondiente considerando en ella el tiempo que la necesitamos activa.

Para que el microprocesador de nuestro Spectrum pueda ejecutar todas sus funciones necesita disponer de un patrón de tiempos de tal forma que cada operación se ejecute sincronizadamente, es decir, en su momento preciso; ni antes ni después.

Es algo así como el director de una orquesta y bajo su batuta, todos los componentes realizan la labor para la cual fueron contruidos sin dar lugar a errores y por supuesto, sin desafinar.

EL REGISTRO F

Entre los registros de uso general disponibles en la CPU de nuestro Spectrum el F (abreviatura de Flags, indicadores o banderas) tiene como misión revelarnos determinados sucesos que se producen durante la ejecución de algunas instruccio-

nes, las cuales aparentemente, no tienen ninguna relación con él.

Se trata de un registro de ocho bits y como podéis comprobar en la figura cada uno de sus bits individuales tiene un significado especial. Analicémoslos con detenimiento:

— Indicador de arrastre o acarreo C (abreviatura del inglés *Carry*): su contenido varía en operaciones de suma y resta cuando se ha producido acarreo en el bit más significativo del acumulador. Por ejemplo, en la aritmética de 8 bits, se comporta como un testigo colocado al lado de un cuentakilómetros que sólo puede marcar de 0 a 255. ¿Y el 256? En este imaginario cuentakilómetros volvería a ser 0, pero el indicador C estaría encendido señalando que le hemos dado la vuelta.

— Indicador de cero Z (del inglés *Zero*): son muchas las instrucciones que afectan a su valor, por ejemplo, las que modifiquen el contenido del acumulador, las operaciones sobre bits individuales, las comparaciones, etc.

Su forma de trabajar resulta un tanto confusa si consideramos que cuando encontramos en su lugar correspondiente del registro F un 0, quiere decir que el resultado de la operación efectuada NO FUE CERO.

— Indicador de signo S (*Sign*, signo en inglés): el propósito de este bit es señalar si el resultado de una operación que involucre números en complemento a dos es positiva (bandera a 0) o negativa (bandera a 1).

Por tanto, no es otra cosa que una copia del bit más significativo de un número expresado en complemento a dos.

— Indicador de paridad/desbordamiento P/V (*Parity/overflow*): a este bit le están encomendadas dos misiones diferentes. La primera se encarga de señalar si, por ejemplo, tras una instrucción de rotación, desplazamiento o lógica, el nú-

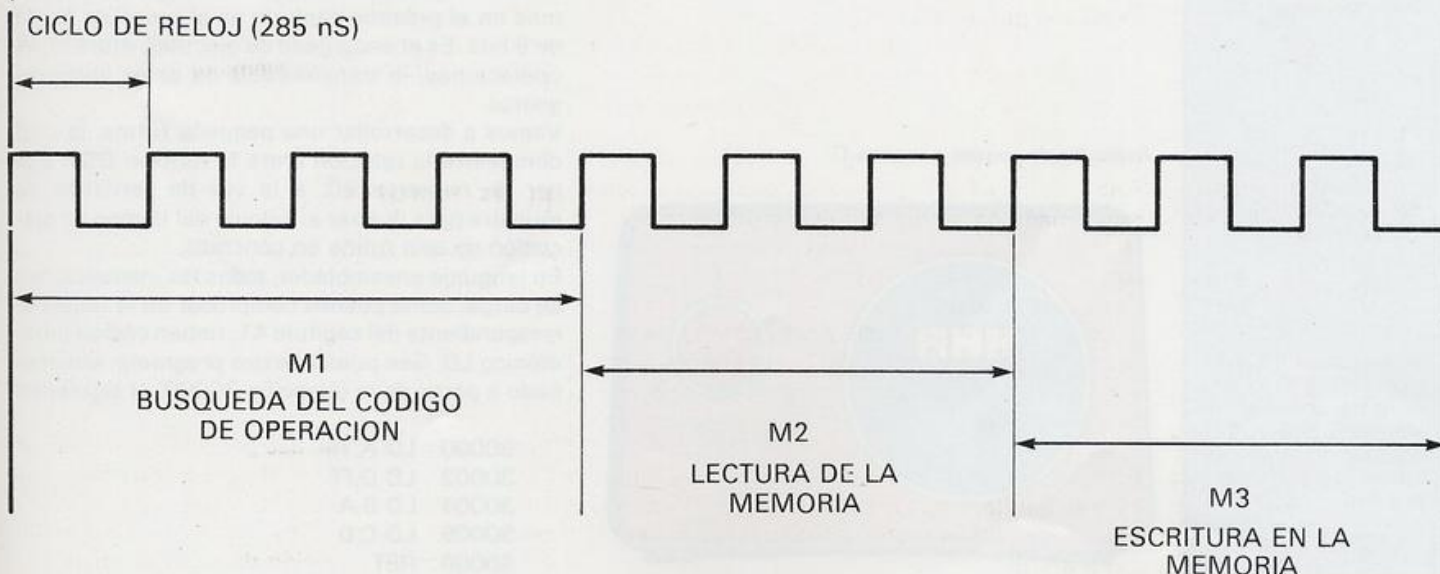


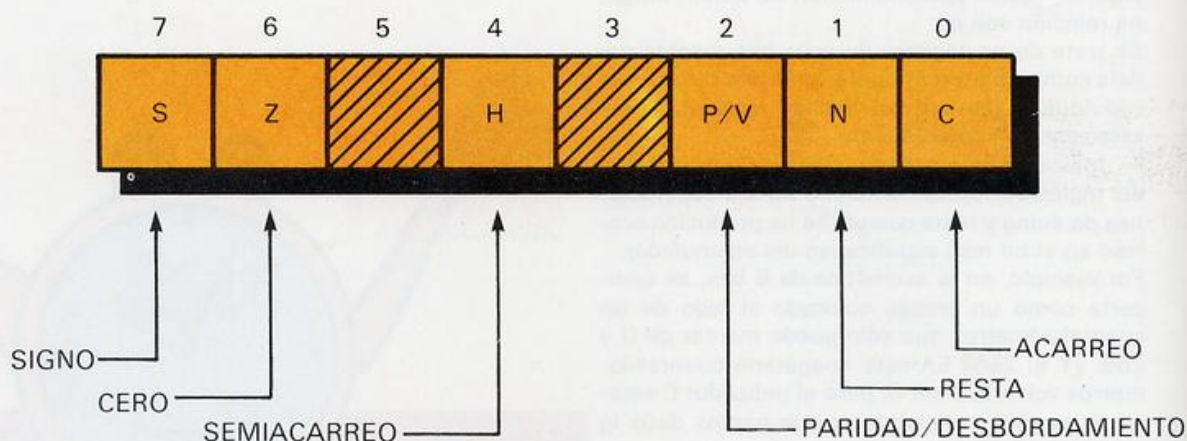
El reloj es el director de la orquesta informática interior al Spectrum.

mero de bits que están en 1 lógico es par (indicador a 1) o impar (indicador a 0).

La segunda consiste en revelar si se ha producido desbordamiento en operaciones aritméticas de números expresados en complemento a dos. Tal es el caso que se presenta cuando la suma de

Cronograma de los ciclos de máquina típicos en el Z80-A.





El registro F.

dos números positivos es negativa o la de dos negativos resulta positiva.

En estas circunstancias, el indicador V se coloca a 1, y en caso de no existir sobrepasamiento el valor encontrado en la posición correspondiente del registro F será 0.

— Indicador de semiacarreo H (*Half carry*): se utiliza solamente en la aritmética codificada en decimal (BCD), la cual será comentada en su momento. Su funcionamiento es similar a como lo hacía el de arrastre, pero con la salvedad de que nuestro cuentakilómetros en este caso tendrá solamente 4 bits a diferencia con el anterior, el cual disponía de 8.

— Indicador de resta N: al igual que el anterior,

solamente se emplea en la aritmética BCD, colocándose a 1 cuando la última operación efectuada fue una resta.

Como observaremos, sólo 6 de los 8 bits que componen el registro F tienen significado para nosotros. Los otros dos no tienen ningún interés para el programador.

No quiere esto decir que su valor no sea modificado durante la ejecución de una rutina determinada. Al contrario, la CPU los utiliza como apoyo para completar determinadas labores internas. Antes de adentrarnos definitivamente en los diferentes grupos de instrucciones, hemos de recalcar la importancia que como nexo de unión entre BASIC y C/M tienen el par de registros BC, dentro del sistema operativo de nuestro Spectrum.

Es precisamente su contenido en el momento de retornar desde una rutina en código máquina, lo que obtenemos tras ejecutar la función **USR**. Comprobémoslo con algunos ejemplos.

El primer grupo de instrucciones que analizaremos en el próximo capítulo es el grupo de carga de 8 bits. Es el encargado de ejecutar, entre otras operaciones, la transferencia de datos entre registros.

Vamos a desarrollar una pequeña rutina, la cual demuestre la relación entre la función **USR** y el par de registros BC, a la vez de servirnos de muestra para ilustrar el cálculo del tiempo de ejecución de una rutina en concreto.

En lenguaje ensamblador, todas las instrucciones de carga, como puedes comprobar en la tabla correspondiente del capítulo 41, tienen código mnemónico LD. Sea pues nuestro programa, almacenado a partir de la dirección 30000, el siguiente:

```
30000 LD A,7F
30002 LD D,FF
30004 LD B,A
30005 LD C,D
30006 RET
```

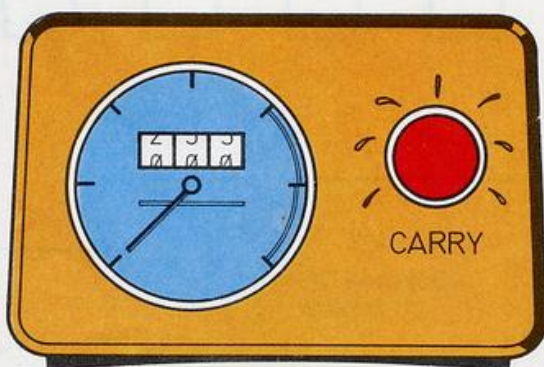
i!

Sólo 6 de los 8 bits del registro F tienen utilidad práctica desde el punto de vista de la programación.

*

No confundir los indicadores C y V. El primero se utiliza en la aritmética binaria, y el segundo en la de números en complemento a dos.

Indicador de arrastre o acarreo C.



¿Qué es lo que hace? Las dos primeras instrucciones almacenan en los registros A y D los valores hexadecimales 7F y FF respectivamente. A continuación, estos valores son transferidos a los registros B y C. Finalmente, mediante la instrucción RET se devuelve el control desde el C/M al BASIC.

Para almacenarla en la memoria del ordenador puedes utilizar el cargador de código máquina aparecido anteriormente en esta misma sección. Carga éste y añade la siguiente línea:

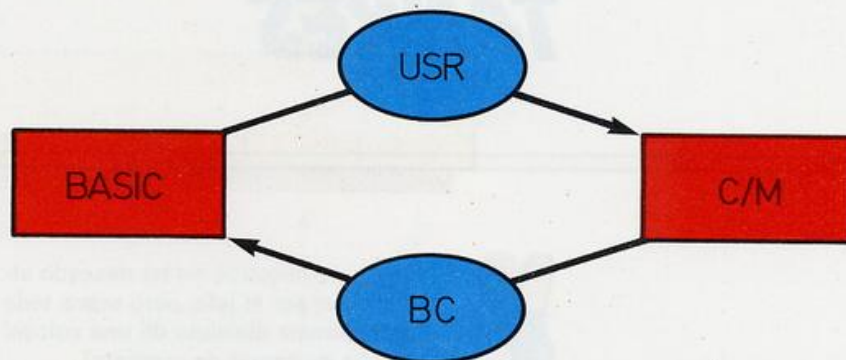
10 DATA "3E 7F 16 FF 47 4A C9 ***",812

Ejecuta el programa mediante **RUN** contestando a la pregunta sobre dirección de ubicación con 30000. Si el cargador indica algún error, repasa los datos anteriores, hasta que todo vaya bien. Tecla entonces, como comando directo **CLS: PRINT USR 30000**.

Tras pulsar **ENTER** aparecerá sobre la pantalla 32767, el cual es efectivamente, la versión decimal del número 7FFF hexadecimal, almacenado en el par BC.

Para terminar, calculemos el tiempo empleado en ejecutarse esta pequeña rutina. En las dos primeras instrucciones, el microprocesador debe acceder dos veces a la memoria del ordenador, para leer en primer lugar el código de operación (4 ciclos de reloj), y posteriormente, el dato a ser introducido en el registro correspondiente (3 ciclos).

En las dos siguientes le basta con el código de operación para ejecutarlas, es decir, cada una de



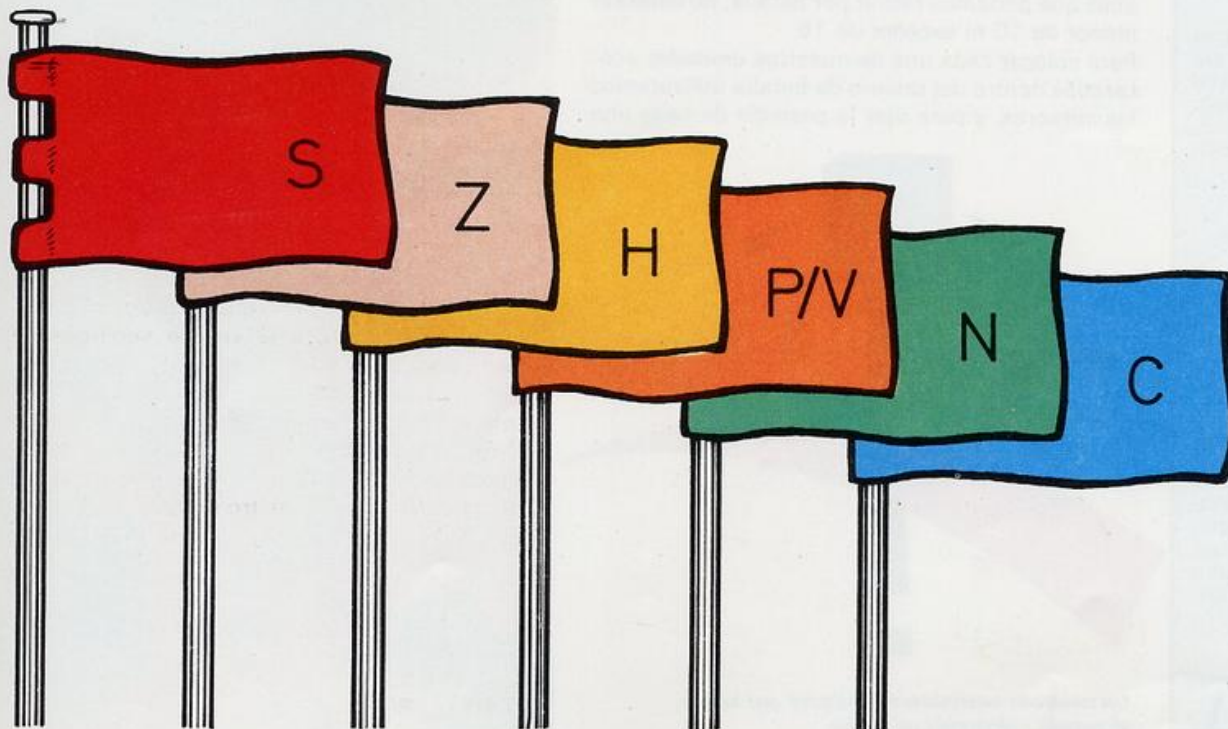
El par de registros BC constituye el nexo de unión entre BASIC y C/M.

ellas empleará 4 estados. La instrucción RET, la cual será discutida más extensamente cuando tratemos las instrucciones de retorno, toma 10 ciclos en su ejecución.

En definitiva $2 \times 7 + 2 \times 4 + 10 = 32$ ciclos de reloj, o lo que es lo mismo, 9120 nanosegundos. ¡Menos de 10 millonésimas de segundo en efectuar todo lo anterior!



El registro F está formado por 6 banderas de interés para nosotros.



TANKES

i!

Para la grabación utilizaremos el comando **SAVE "TANQUES" LINE 10**. La autoejecución en la línea 10 propicia la carga automática del generador de caracteres.

*

El juego de caracteres utilizado para este programa aparece bajo el nombre de **ROTULACION** en nuestro anterior programa **IM-PRENTA**.

*

El juego de caracteres **ROTULACION** deberá ubicarse a continuación del programa principal en la misma cinta. Para su grabación deberemos seguir las instrucciones que aparecen en nuestro anterior programa **IM-PRENTA**.



quien de nosotros no ha deseado alguna vez ser el jefe, pero sobre todo protagonista absoluto de una colosal batalla de carros de combate?

Con este programa podremos, sin lugar a dudas, emprender una superaventura bélica, teniendo por enemigo la escuadra blindada del Spectrum.

EL PROGRAMA

Una vez el listado del programa ha sido correctamente introducido en la memoria de nuestro ordenador y ejecutado, surgirán en nuestra pantalla un par de mensajes de captación de datos. El primero de ellos nos pedirá que introduzcamos nuestras iniciales; con ésto, nuestro Spectrum se hace una ligera idea de con quien tiene que enfrentarse. El segundo mensaje de captación nos requiere el número de carros de combate que queremos entren en batalla. El número de tanques que podemos reunir por batalla, no debe ser menor de 10 ni exceder de 16.

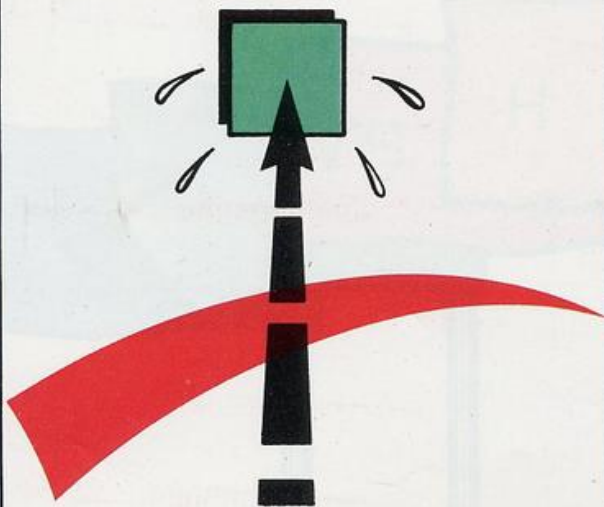
Para colocar cada una de nuestras unidades acorazadas dentro del tablero de batalla utilizaremos los cursores, y para fijar la posición de cada uno

pulsaremos la tecla donde está representado el número cero.

Una vez hemos posicionado todos los tanques, nuestro serio oponente Spectrum hará lo propio con los suyos y la batalla comenzará.

A la hora de efectuar un disparo, debemos seguir las mismas reglas de pulsación de teclas que cuando situamos nuestros carros de combate. Con las teclas 6 y 7 elegiremos nuestro campo de tiro. Con el 5 y 8 daremos más alcance a nuestro disparo o corregiremos, y con el 0 efectuamos el cañonazo.

Un cuadrado intermitente significará que hemos alcanzado un blanco y, como consecuencia, des-



Un cuadrado intermitente significará que hemos alcanzado y destruido un blanco.

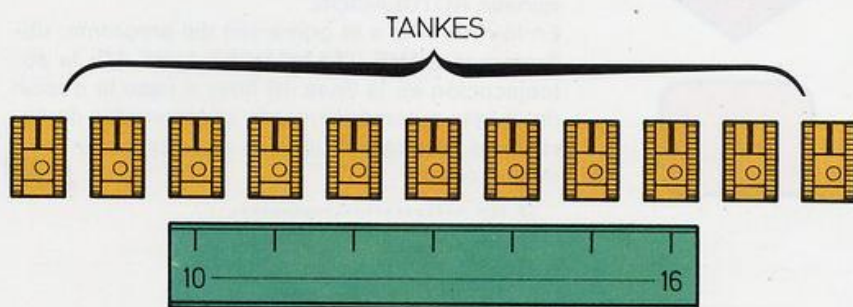




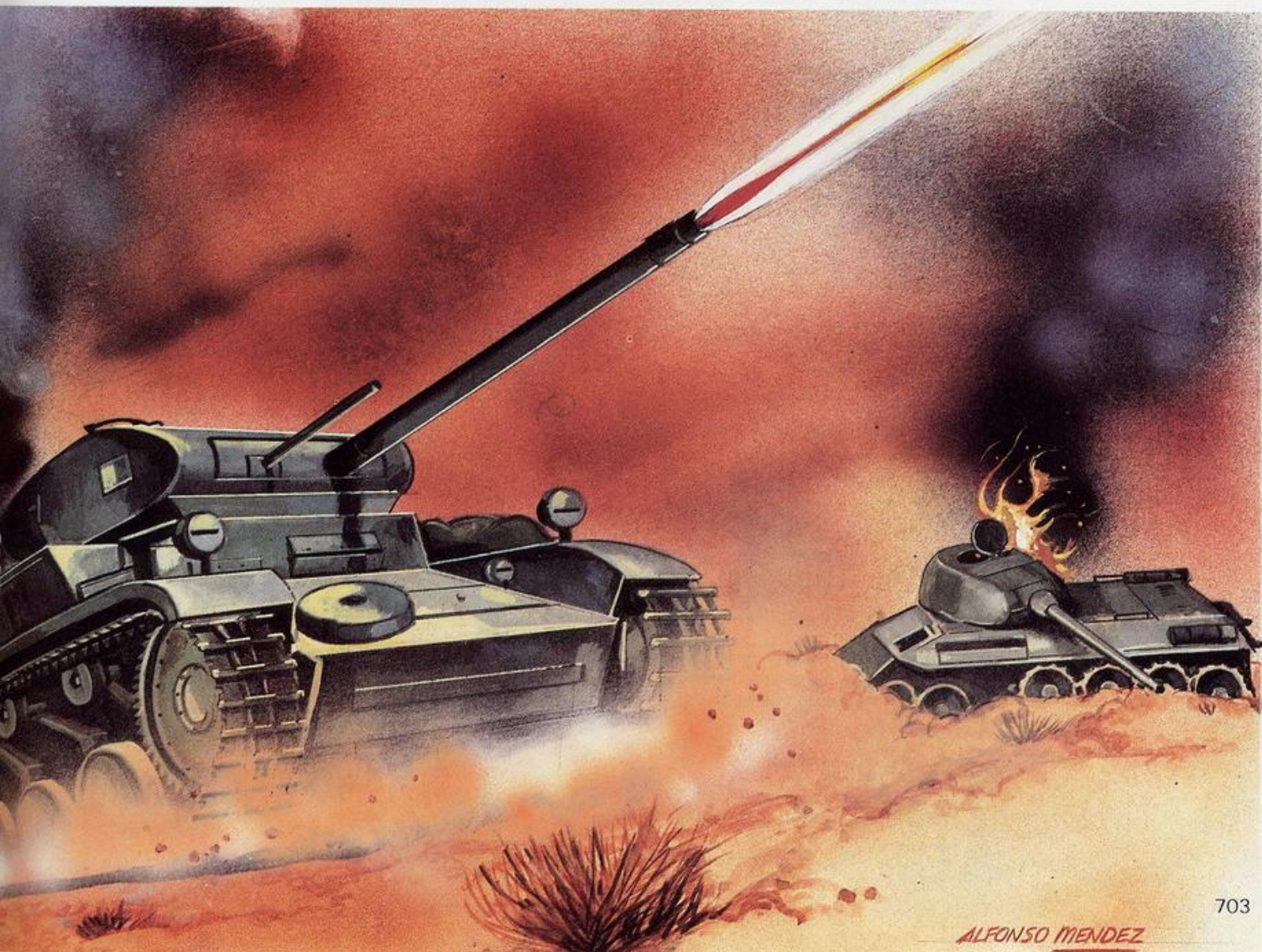
truido. Si nuestro disparo no hubiese encontrado el objetivo deseado, nuestro ordenador lo marcaría en la pantalla de una forma diferente: delimitación de un cuadrado.

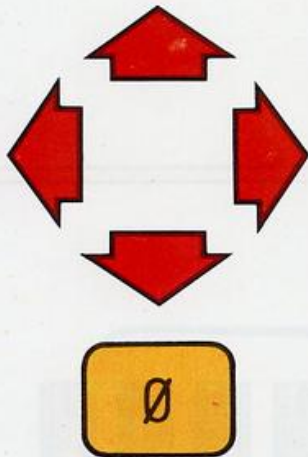
La batalla finaliza cuando la totalidad de los tanques de alguno de los dos bandos ha sido destruido. En caso de que seamos los perdedores, nuestro enemigo, no por ello menos galante, nos mostrará aquellos blancos que no pudimos destruir.

El programa utiliza un nuevo generador de caracteres para dar un poco más de ambientación al juego. Los **POKEs** que efectúa el Spectrum para acceder a este generador, son los siguientes:



El número de tanques que podemos reunir por batalla, no debe ser menor que diez ni exceder de dieciséis.





POKE 23606,102 y POKE 23607,216. Así pues, la carga del nuevo generador de caracteres se debe efectuar en 55654 (102+256*217=55654). Por tanto, el programa no es compatible con el modelo de 16K.

Para la adopción del generador de caracteres, procederemos por el sistema explicado en esta misma sección al tratar el programa IMPRENTA. Hemos de emplear el juego de caracteres denominado ROTULACION.

En lo referente a la grabación del programa, utilizaremos **SAVE "TANQUES" LINE 10**; la autoejecución en la línea 10 lleva a cabo la misión de cargar automáticamente el generador de caracteres, que estará ubicado a continuación en la misma cinta.

```

10 REM *****
20 REM * J.M.MAYORAL SERRANO *
30 REM *****
40 REM * TANQUES (C) 1985 *
50 REM *****
60 INK 9: BORDER 1: PAPER 1
70 CLS
80 PRINT AT 10,4: "UN MOMENTO, POR FAVOR"
90 PRINT AT 12,6: FLASH 1: "GRAFICOS USUARIO"
100 GO SUB 1320: CLS
110 LET SWC=0
120 LET CP=1: LET V1=0: LET V2=0
130 DIM G$(2,5): LET G$(2,5)="YO!"
140 LET JK=0: LET G=0
150 PRINT AT 11,3: "INTRODUCE TUS INICIALES"
160 INPUT LINE G$(1)
170 PRINT AT 11,3:
180 PRINT AT 14,0: "CUANTOS TANKES QUIERES ? (10-16)"
190 INPUT N
200 IF N<10 OR N>16 THEN GO TO 190
210 LET BDM=0
220 CLS
230 GO SUB 1230
240 DIM P(2)
250 GO SUB 1090
260 DIM A(64)
270 DIM B(64)
280 FOR K=2 TO 16 STEP 2
290 FOR W=4 TO 18 STEP 2
300 PRINT AT K,W: INK 5: "G"
310 NEXT W
320 NEXT K
330 GO SUB 1210
340 PRINT AT 13,23: "PULSE "AT 14,23: "5-6-7-8 "AT
15,23: "TANKE:"
350 FOR K=1 TO N
360 LET POS=2
370 PRINT INK 6: AT POS,2: "A": INK 7: AT 2,20: "C"
380 PRINT AT 16,23: BRIGHT 1: FLASH 1: K
390 LET M$=INKEY$
400 IF M$<>"5" AND M$<>"6" AND M$<>"7" AND M$<>"8" T
HEN GO TO 390
410 LET M=VAL M$: IF M=8 OR M=5 THEN GO TO 480
420 LET PZ=POS
430 LET POS=POS+(2 AND M=6)-(2 AND M=7): IF POS<2 TH
EN LET POS=4
440 IF POS>16 THEN LET POS=14
450 PRINT INK 6: AT POS,2: "A": AT PZ,2:
460 BEEP .06,15
470 GO TO 390
480 PRINT INK 4: BRIGHT 1: AT POS,4: "B": LET PZ=4
490 LET M$=INKEY$: IF M$="0" THEN GO TO 580+(220 AN
D BDM=1)
500 IF M$<>"5" AND M$<>"8" THEN GO TO 490
510 LET PT=PZ
520 LET PZ=PZ+(2 AND M$="8")-(2 AND M$="5"): IF PZ<4
THEN LET PZ=6
530 IF PZ>18 THEN LET PZ=16
540 GO SUB 1100
550 PRINT INK 4: BRIGHT 1: AT POS,PZ: "B"
560 LET PY=PY-(M$="8")+(M$="5"): GO SUB 1040
570 GO TO 490
580 GO SUB 1100
590 IF A(PY)=1 THEN GO TO 490
600 LET SWC=1
610 LET A(PY)=1
620 GO SUB 1040
630 BEEP .12,6: BEEP .12,0
640 PRINT AT POS,2:
650 NEXT K
660 LET POS=2
670 PRINT AT 19,1: FLASH 1: BRIGHT 1: "UN MOMENTO, PO
R FAVOR"
680 FOR K=1 TO N
690 LET NC=INT (RND*64)+1: IF A(NC)=1 OR A(NC)=2 THE
N GO TO 690

```

```

700 LET A(NC)=2: LET B(NC)=1: NEXT K
710 PRINT PAPER 2: AT 19,1:
720 FOR K=13 TO 19: PRINT AT K,23:
: PRINT AT 14,23: "TURNO DE"
730 PRINT INK 6: AT POS,2: "A"
740 LET G=G+1: IF G=3 THEN LET G=1
750 PRINT AT 15,23: BRIGHT 1: G$(G)
760 PRINT INK 6: AT POS,2: "AT 2,2: "A"
770 IF G=2 THEN GO TO 900
780 LET BDM=1
790 GO TO 390
800 LET SWC=1
810 GO SUB 1100
820 IF A(PY)=0 THEN LET A(PY)=5: BEEP .3,8: BEEP .5
,-30: GO SUB 1040: GO TO 740
830 IF A(PY)<>2 THEN GO TO 740
840 LET P(1)=P(1)+1
850 GO SUB 1090
860 LET JK=1: LET A(PY)=4: GO SUB 1040
870 FOR Z=1 TO 3: BEEP .3,7: BEEP .13,4: NEXT Z: BEE
P .5,0
880 LET SWC=1: GO SUB 1040: IF P(1)=N THEN GO TO 11
10
890 GO TO 740
900 PAUSE 50
910 LET T=INT (RND*64)+1: IF B(T)=1 THEN GO TO 910
920 LET SP=INT (T/8): IF SP<8 THEN LET SP=SP+1
930 LET POS=SP*2: FOR X=2 TO POS STEP 2: PRINT AT X,
20: "C": AT X-1-(X<>2),20: "BEEP .2,15: BEEP .3,0: N
EXT X"
940 LET ED=0: LET CONT=SP*8-T+1
950 FOR X=18 TO 4 STEP -2: PRINT AT POS,X: INK 4: BR
IGHT 1: "B": BEEP .2,30
960 GO SUB 1220: LET PT=X: GO SUB 1040: LET CONT=CON
T-1: IF CONT<0 THEN GO TO 980
970 LET ED=ED+1: NEXT X
980 LET B(T)=1: IF A(T)=1 THEN GO TO 1010
990 BEEP .2,0: BEEP .3,3
1000 PRINT AT POS,20: "AT 2,20: "C": LET POS=2: GO T
O 740
1010 LET JK=1: LET P(2)=P(2)+1: GO SUB 1090: GO SUB 1
040: FOR W=1 TO 3: BEEP .2,-3: BEEP .4,3: NEXT W
1020 BEEP .4,-35: LET A(T)=3: GO SUB 1040: IF P(2)=N
THEN GO TO 1170
1030 GO TO 1000
1040 BRIGHT (JK=1): FLASH (JK=1)
1050 PRINT AT POS,(PT AND SWC=0)+(PZ AND SWC=1): INK
5: ("B" AND A(PY)=0): INK 6: ("D" AND A(PY)=1): INK 5: (
"B" AND A(PY)=2): INK 3: BRIGHT 1: ("X" AND A(PY)=3):
INK 2: BRIGHT 1: FLASH 1: ("E" AND A(PY)=4): FLASH 0:
INK 6: ("E" AND A(PY)=5)
1060 BEEP .02,-3+(6 AND BDM=1)
1070 FLASH 0: BRIGHT 0
1080 LET JK=0: LET SWC=0: RETURN
1090 PRINT AT 6,29: V1: AT 7,29: V2: AT 10,29: P(1): AT 11,
29: P(2): RETURN
1100 LET PY=((POS/2-1)*7+POS/2)+(PZ/2-2): RETURN
1110 GO SUB 1090: GO SUB 1210: PRINT AT 14,23: INK 2:
FLASH 1: "TU "AT 15,23: "GANAS ": PAUSE 100
1120 LET V1=V1+1: LET CP=CP+1
1130 PRINT INK 6: AT 21,2: "DESEAS SEGUIR JUGANDO ?(S/
N)": INK 0
1140 IF INKEY$="S" THEN GO TO 210
1150 IF INKEY$="N" THEN GO TO 10000
1160 GO TO 1140
1170 GO SUB 1090: GO SUB 1210: PRINT AT 14,23: FLASH
1: "GANO YO"
1180 LET X=0: FOR K=2 TO 16 STEP 2: FOR W=4 TO 18 STE
P 2: LET X=X+1: IF A(X)=2 THEN PRINT AT K,W: FLASH 1
: BRIGHT 1: "D": BEEP .3,5: BEEP .3,-40
1190 NEXT W: NEXT K
1200 LET V2=V2+1: LET CP=CP+1: GO TO 1130
1210 FOR K=13 TO 19: PRINT AT K,23: BRIGHT 1:
: NEXT K: RETURN
1220 LET PY=POS*4-ED: RETURN
1230 CLS
1240 FOR K=0 TO 31: PRINT AT 0,K: "B": AT 18,K: "B": AT 2
0,K: "B": NEXT K
1250 FOR K=0 TO 20: PRINT AT K,0: "B": AT K,22: "B": AT K
,31: "B": NEXT K
1260 PRINT AT 10,23: "AT 8,23: "33333333": AT
12,23: "33333333": AT 4,23: "88888888"
1270 PRINT AT 0,5: INVERSE 1: "EL ALAMEIN ": INVERSE
0: AT 5,23: "TRIUNFOS": AT 6,23: G$(1): AT 7,23: "COMP."
1280 PRINT AT 9,23: "IMPACTOS": AT 10,23: G$(1): AT 11,23
: "SPECT": AT 2,24: BRIGHT 1: "TANKES"
1290 FOR K=27 TO 155 STEP 16: PLOT K,36: DRAW 0,128,0
: NEXT K
1300 FOR K=36 TO 164 STEP 16: PLOT 27,K: DRAW 128,0,0
: NEXT K
1310 RETURN
1320 DATA "A",255,195,40,47,40,195,255,0,"B",0,0,60,6
0,60,60,0,0,"C",0,255,195,20,244,20,195,255,"D",0,128
,76,60,44,255,171,84
1330 DATA "E",255,129,189,189,189,189,129,255,"F",255
,129,129,129,129,129,129,255
1340 FOR K=1 TO 6: READ C$: FOR W=0 TO 7: READ C: POK
E USR C$+W,C: NEXT W: NEXT K: RETURN
1350 CLEAR 55650: LOAD "CODE"
1360 POKE 23606,102
1370 POKE 23607,216
1380 RUN

```