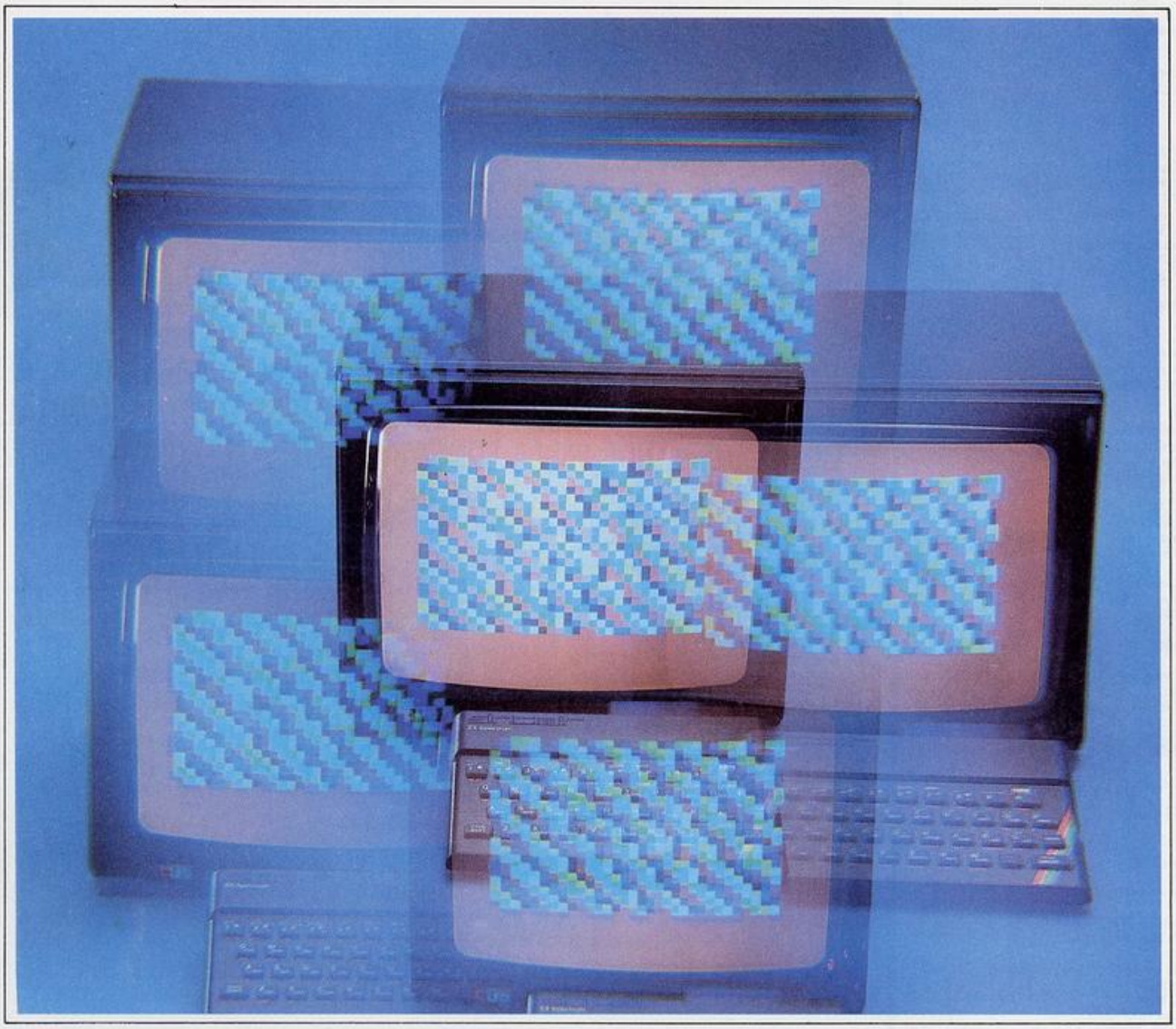
34
150pts.

RUN

Enciclopedia Práctica del Spectrum



Nueva Lente/Ingelek

LA MEMORIA DEL SPECTRUM

HASTA ahora nos hemos ocupado de adquirir la mayor cantidad de conocimientos en lo referente al lenguaje de programación de nuestro Spectrum: el BASIC.

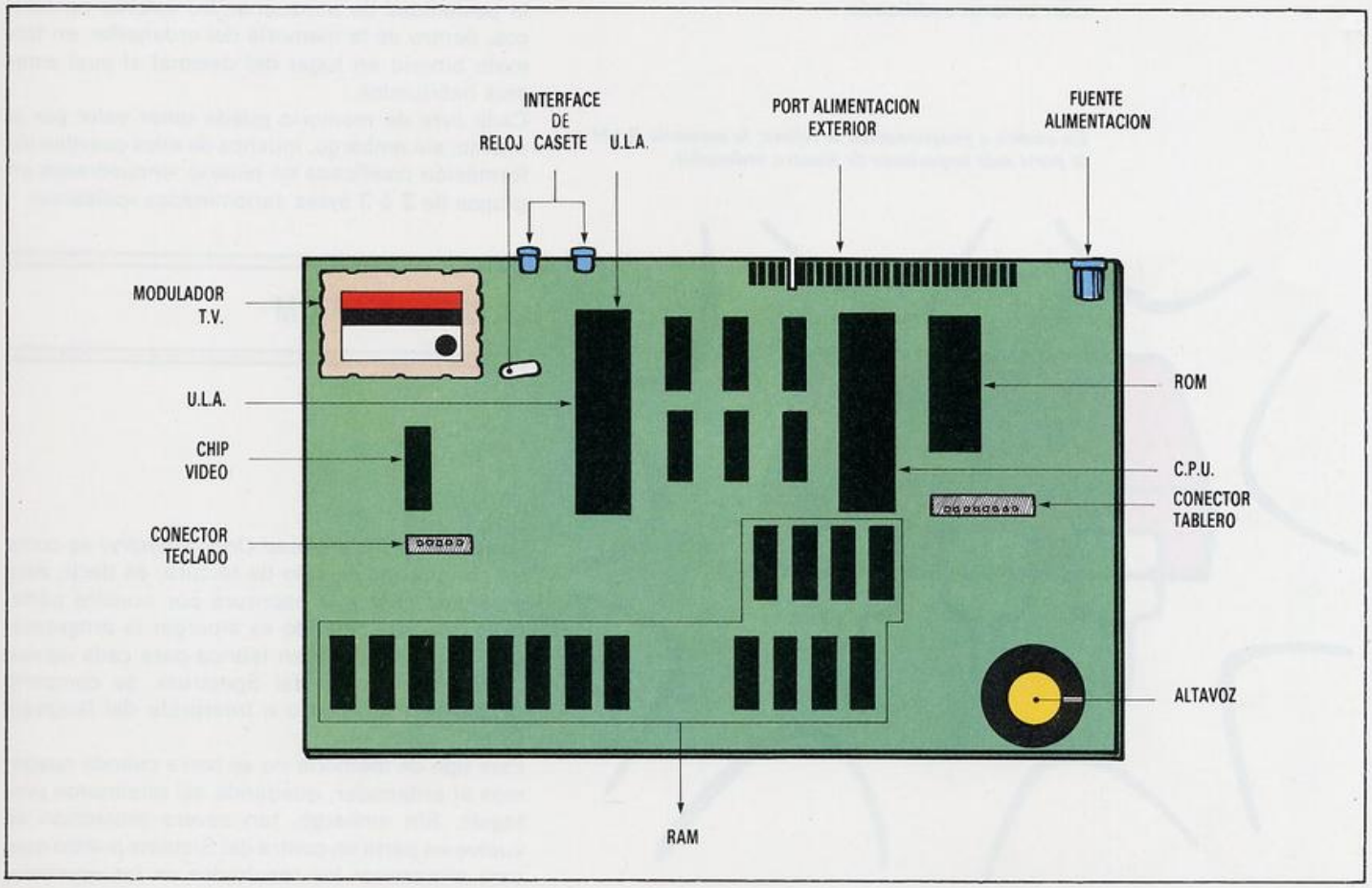
Por lo que hemos podido ver, el repertorio de sentencias y comandos BASIC es tan amplio que nos permite realizar casi cualquier cosa que se nos ocurra en cuanto a programación se refiere. Esto es debido a que el BASIC, como el resto de los lenguajes de alto nivel, está concebido de forma que permita resolver la codificación de un programa adaptado específicamente a nuestras necesidades.

Sin embargo, no está de más el conocer más a fondo la constitución interna del Spectrum, pues existe un grupo de sentencias del repertorio BASIC que nos dan opción de actuar directamente sobre los contenidos de la propia memoria del ordenador.

El interior de nuestro Spectrum está dividido en 12 zonas fundamentales: modulador de T.V., altavoz, fuente de alimentación, memoria RAM, conectores del teclado, reloj, *port* de comunicación exterior, chips lógicos, interface para casete, U.L.A., C.P.U. (unidad central de proceso) y memoria ROM.

Desde el aspecto de la programación, la memoria RAM es la parte más importante de nuestro

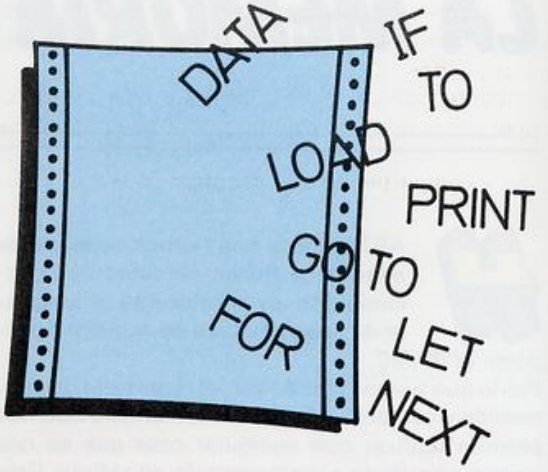
El interior de nuestro Spectrum está dividido en doce zonas.



ordenador. Esto es debido a que se trata precisamente de la zona empleada para almacenar nuestros programas y las variables utilizadas por los mismos. Por otra parte, la memoria ROM puede suministrarnos información de interés utilizable en nuestros programas, con el empleo de las sentencias BASIC que comentaremos en este capítulo.

TIPOS DE MEMORIA

Existen en el ordenador dos tipos de memorias: la ROM y la RAM. Como característica común, podemos decir que ambas están formadas por una sucesión de *bytes* o porciones elementales de memoria, compuestas a su vez por la combinación de 8 *bits* (valores 0 ó 1 lógicos exclusivamente). Así, cada una de estas posiciones puede dar cobijo a un carácter ASCII (numérico o alfabético), del mismo modo que contener información binaria codificada.

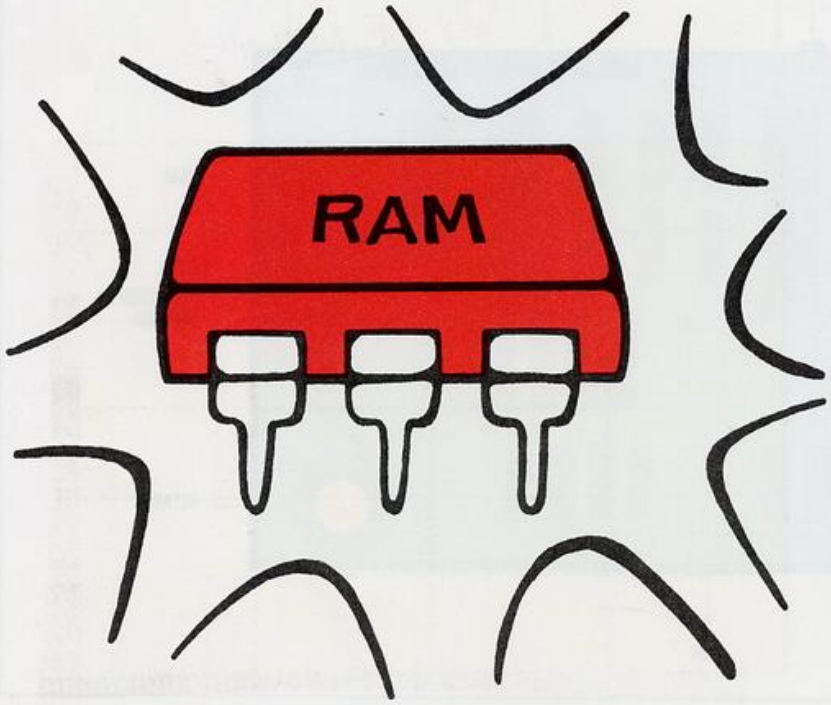


El BASIC está concebido de forma que permita la codificación de un programa adaptado específicamente a nuestras necesidades.

En una posición de memoria podemos almacenar un número desde 0 a 255, en dos desde 0 a 65535 y así sucesivamente. Todo esto, gracias a la posibilidad de almacenar los valores numéricos, dentro de la memoria del ordenador, en formato binario en lugar del decimal al cual estamos habituados.

Cada *byte* de memoria puede tener valor por sí mismo; sin embargo, muchos de ellos guardan información codificada en binario, encuadrados en grupos de 2 ó 3 *bytes* denominados «palabras».

En cuanto a programación se refiere, la memoria RAM es la parte más importante de nuestro ordenador.



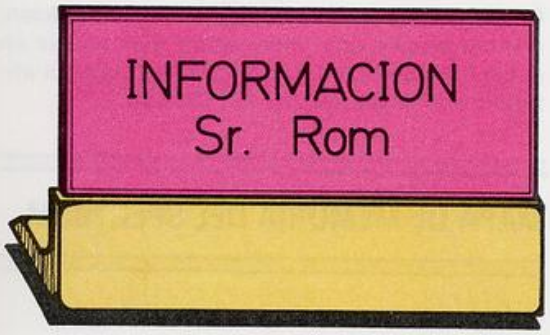
LA MEMORIA ROM

La memoria ROM *(Read Only Memory)* es como su nombre indica sólo de lectura, es decir, está protegida contra la escritura por nuestra parte, dado que su cometido es albergar la programación inicial incluida en fábrica para cada equipo y que, en el caso del Spectrum, se compone de Sistema Operativo e Intérprete del lenguaje BASIC.

Este tipo de memoria no se borra cuando apagamos el ordenador, quedando así totalmente protegida. Sin embargo, tan severa protección se vuelve en parte en contra del Sistema puesto que, para almacenar los resultados de las operacio-

nes necesarias para la interpretación del BASIC y los manejos del Sistema Operativo, la ROM debe hacer uso de una pequeña parte de la memoria RAM.
El contenido de la ROM está formado en exclusiva por la programación del fabricante del equipo *(firmware),* siendo de uso reservado al Sistema y compartiendo con nosotros parte de la memoria RAM.

La memoria ROM suministra información de interés, utilizable en nuestros programas.



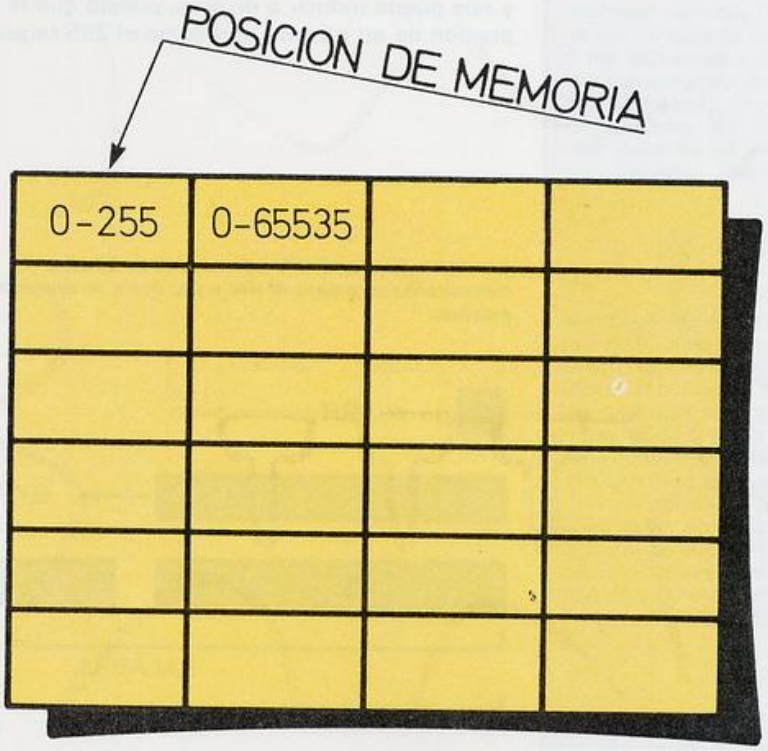
LA MEMORIA RAM

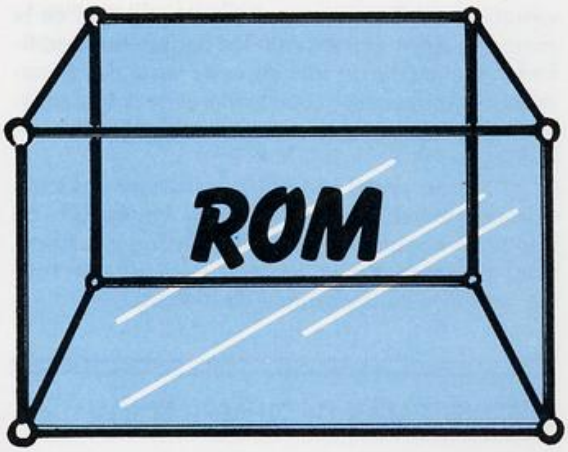
La memoria RAM *(Random Access Memory)* permite la lectura y escritura, siendo sin duda la más importante para nosotros por servir de soporte al texto BASIC y sus variables en cada uno de nuestros programas. Al contrario de lo que sucede con la ROM, el contenido de la memoria RAM se borra cuando apagamos el ordenador. Por ello, el Spectrum dispone de medios para conservar la información contenida en este espacio sobre soportes de mayor durabilidad, como lo son la cinta de casete y la unidad MICRODRIVE.
La memoria RAM no es patrimonio exclusivo del usuario. De hecho, existen dos zonas de especial importancia dentro de ella masivamente utilizadas por las rutinas contenidas en la ROM: el área de variables del Sistema y la propia memoria de pantalla y atributos.
De la segunda ya conocemos su estructura y composición desde los capítulos dedicados al color y al manejo de la alta resolución. El área de variables del Sistema es una zona especial de la memoria RAM que alberga los contenidos codificados en binario de una serie de variables denominadas «punteros», cuyos valores son actualizados automáticamente por rutinas contenidas en la ROM.
Estas variables del sistema guardan información que podríamos considerar como un «mapa» de dónde se encuentra cada cosa y en qué estado. Son, por tanto, fundamentales para el buen funcionamiento de las rutinas de la ROM.

LOS SISTEMAS DE NUMERACION

Además del sistema decimal, existen otros dos que debemos conocer dada su aplicación masiva en el ámbito de la programación de todo tipo de ordenadores: el binario y el hexadecimal.
El sistema binario, o de base 2, emplea única-

En una posición de memoria podemos almacenar un número desde 0 a 255, y en dos, de 0 a 65535.





La memoria ROM está protegida contra la escritura por nuestra parte.

¡!

El sistema binario o de base 2 emplea solamente los guarismos 0 y 1, siendo el único sistema de numeración válido en el interior del ordenador, por ser inteligible directamente por el microprocesador.

*

El sistema hexadecimal se basa en los dígitos decimales del 0 al 9 completando éstos hasta el total de 15 con las primeras letras del alfabeto, desde la A hasta la F.

*

La CPU del Spectrum está dotada de un microprocesador capaz de direccionar (acceder para lectura o escritura) 65536 posiciones de memoria (64 *Kbytes*); para recuperar o enviar a cualquiera de estas posiciones un *byte* cuyo valor hexadecimal oscila entre 00 y FF (0 a 255 decimal).

mente los guarismos 0 y 1. Realmente, éste es el único sistema de numeración válido en el interior del ordenador, por ser inteligible directamente por el microprocesador. Es además el más simple, pues permite la expresión de cualquier número con el empleo de dos únicos guarismos que denominamos estados lógicos, de forma que el 1 binario se corresponde con el nivel alto o verdadero lógico y el 0 con el nivel bajo o falso lógico.

Sin embargo, su utilización se nos hace pesada y nos puede inducir a errores, puesto que la expresión de un número que como el 255 requiere de tres dígitos decimales, necesita 8 en binario (11111111).

Debido a esta necesidad de expresión sencilla del contenido de cada unidad elemental de la memoria del ordenador (el *byte*), así como de los conjuntos de *bytes* a los que denominamos «palabras», se pensó en lo idóneo del sistema hexadecimal (base 16) para esta representación.

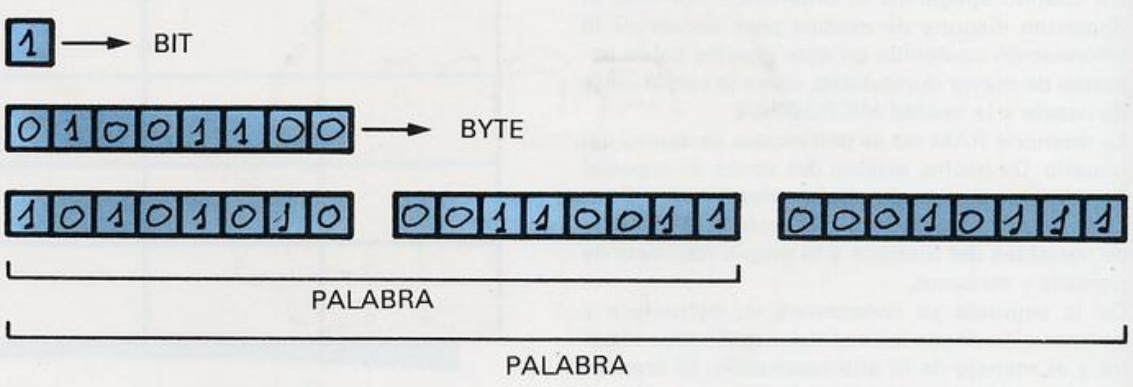
El sistema hexadecimal se basa en los dígitos decimales del 0 al 9 y completando éstos hasta el total de 15 con las primeras letras del alfabeto, desde la A hasta la F, correspondiendo el dígito A al valor 10 y el dígito F al valor 15. Remitiéndonos al caso del ejemplo anterior, la representación del número decimal 255 sería FF en hexadecimal.

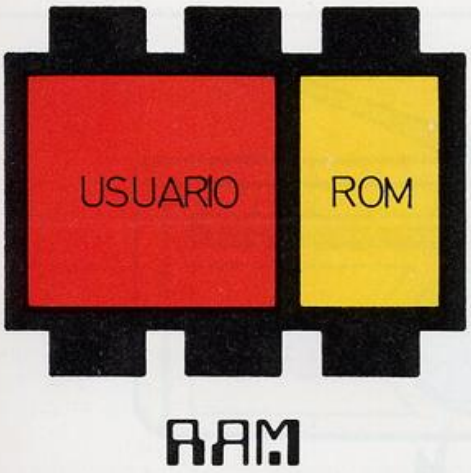
En todo caso, sobre los sistemas de numeración, encontraremos una información más amplia en el capítulo dedicado a este tema en páginas anteriores de la sección TU SPECTRUM.

MAPA DE MEMORIA DEL SPECTRUM

La CPU del Spectrum está dotado de un microprocesador capaz de direccionar (acceder para lectura o escritura) 65536 posiciones de memoria (64 *Kbytes*); para recuperar o enviar a cualquiera de estas posiciones un *byte* cuyo valor hexadecimal oscila entre 00 y FF (0 a 255 decimal). Debido a esta circunstancia, los dos modelos de Spectrum (16 y 48 K) disponen de la misma capacidad nominal de 64 K. Tanto en uno como en otro la memoria ROM ocupa un total de 16 K, por lo que quedan disponibles para memoria RAM un máximo de 48 K. La diferencia real entre un modelo y otro, es que el de 48 K dispone de una memoria RAM adicional de 32 K que el modelo de 16 K no posee.

Los bytes que guardan la información en binario, encuadrados en grupos de dos o tres bytes, se denominan palabras.





La memoria RAM no es patrimonio exclusivo del usuario: el área de Variables del Sistema y el de pantalla son utilizados frecuentemente por rutinas de la ROM.

En las dos versiones de Spectrum, las posiciones $0000 a $3FFF (significaremos precedidos de $ los valores en hexadecimal) están ocupadas por la memoria ROM, correspondiendo las posiciones $4000 a $7FFF a las primeras 16 K de RAM comunes y las posiciones $8000 a $FFFF a la extensión de 32 K RAM del modelo de 48 *Kbytes* El archivo de imagen del Spectrum, que abarca a su vez la memoria de pantalla y la de atributos, ocupa las posiciones decimales que van desde la 16384 a la 22527.

A continuación se encuentra el *buffer* de impresora (23296 a 23551), las variables del Sistema (23552 a 23734) y los mapas de MICRODRIVE e información sobre los canales.

A partir de este punto, comienza en realidad la zona de texto BASIC y seguidamente las variables del programa. Más adelante se encuentra el espacio de trabajo del intérprete BASIC así como para edición de las líneas de programa y, por último, la pila o *stack* conteniendo las direcciones de retorno del Sistema y las del propio programa BASIC del usuario.

Justo al final de la memoria e inmediatamente anterior al *P-RAMT* (final físico de la memoria RAM), se encuentran los 21 *U.D.G.* (gráficos definidos por el usuario), ocupando un total de 168 *bytes.*

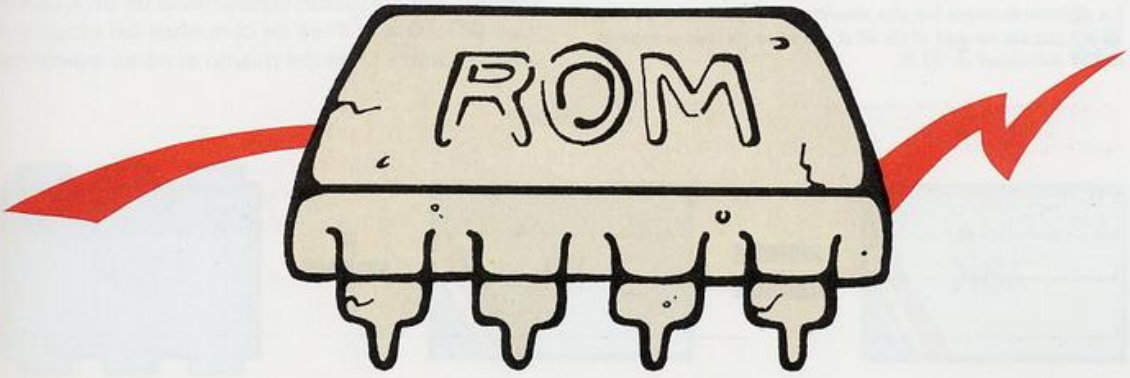
Dentro de la memoria RAM existen zonas que mantienen un emplazamiento fijo desde que encendemos el equipo y otras que van modificando su posición relativa según las necesidades. Como partes fijas podemos destacar la memoria de pantalla, *buffer* de impresora y variables del Sistema, así como la zona de gráficos definidos situada al final de la RAM.

El resto de la memoria, compuesto por el texto BASIC así como sus variables y el *stack,* tiene la propiedad de crecer y decrecer de acuerdo a nuestras necesidades. De esta forma, cada vez que nosotros incluimos una nueva sentencia dentro del programa, el Sistema desplaza automáticamente la zona de variables del programa y el *stack* hacia posiciones más altas.

Del mismo modo, cuando suprimimos líneas de programas, se comprime el texto BASIC y todo lo que viene a continuación pasa a ocupar lugares más bajos en la RAM.

Esta situación itinerante podría resultar una verdadera complicación para el ordenador. Para solucionarla, se emplean las variables del Sistema, gracias a cuyos contenidos la CPU puede saber en todo momento dónde comienzan las diferentes zonas de la RAM y en qué punto de actualización se encuentran las cosas.

La memoria ROM no se borra cuando apagamos el ordenador.



¡!

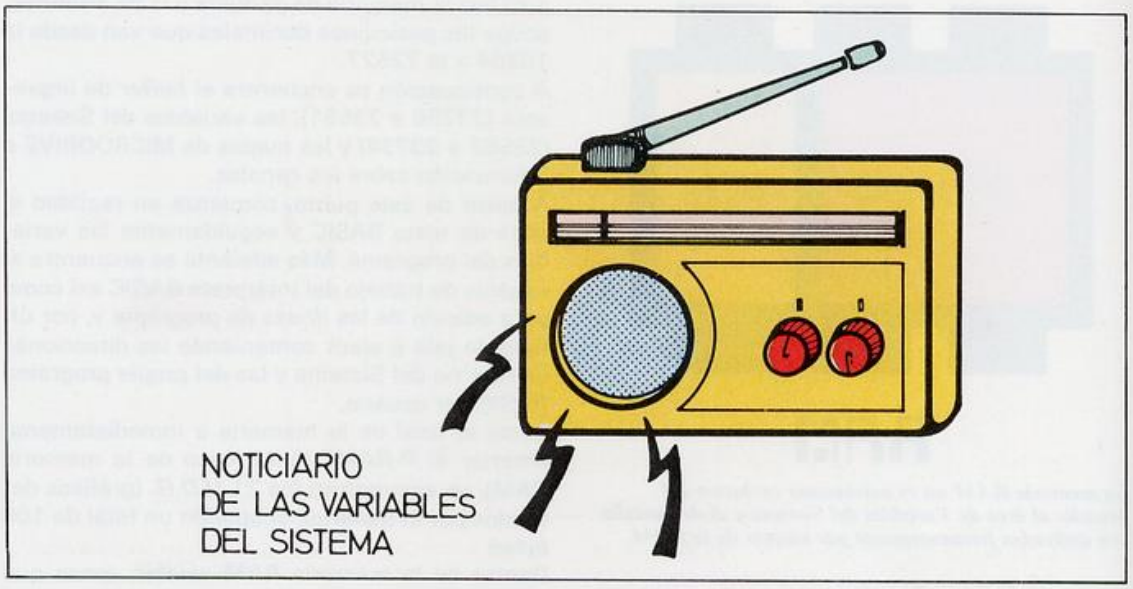
Justo al final de la memoria e inmediatamente anterior al *P-RAMT* (final físico de la memoria RAM), se encuentran los 21 *U.D.G.* (gráficos definidos por el usuario), ocupando un total de 168 *bytes.*

*

En las dos versiones de Spectrum las posiciones $0000 a $3FFF hexadecimal están ocupadas por la memoria ROM, correspondiendo las posiciones $4000 a $7FFF a las primeras 16 K de RAM comunes y las posiciones $8000 a $FFFF a la extensión de 32 K RAM del modelo de 48 *Kbytes.*

*

Dentro de la memoria RAM existen zonas que mantienen un emplazamiento fijo desde que encendemos el equipo y otras que van modificando su posición relativa según las necesidades.



¡!

CLEAR borra las variables, la pantalla **(CLS)**, reajusta el puntero de la sentencia **PLOT**, ejecuta un **RESTORE**, borra el *stack* de los **GO SUB** y asigna nuevo límite superior a la memoria RAM *(RAMTOP)*, manteniendo intacta la zona de texto BASIC.

*

La diferencia entre los dos modelos de Spectrum es que el de 48 K dispone de una memoria RAM adicional de 32 K.

*

El archivo de imagen del Spectrum, que abarca a su vez la memoria de pantalla y la de atributos, ocupa las posiciones decimales que van desde la 16384 a la 22527.

Las Variables del Sistema guardan información sobre determinadas condiciones y estados.

LA SENTENCIA CLEAR

La sentencia **CLEAR** tiene como misión asignar un nuevo *RAMTOP* (puntero que señala el último byte de RAM que el ordenador debe considerar disponible para el BASIC). Su formato general es:

CLEAR nuevo-valor

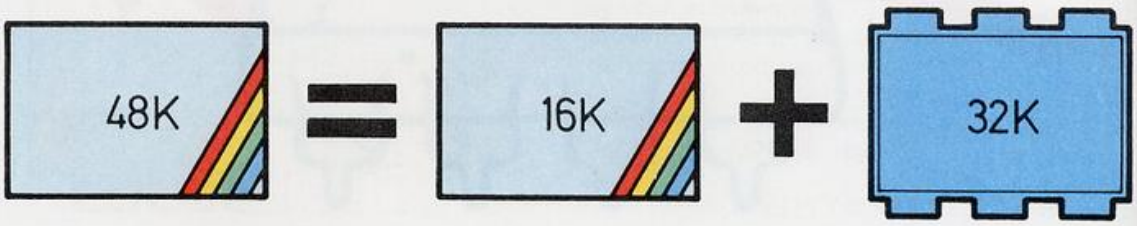
La diferencia entre los dos modelos de Spectrum (16 K y 48 K) estriba en que el de 48 K dispone de una memoria RAM adicional de 32 K.

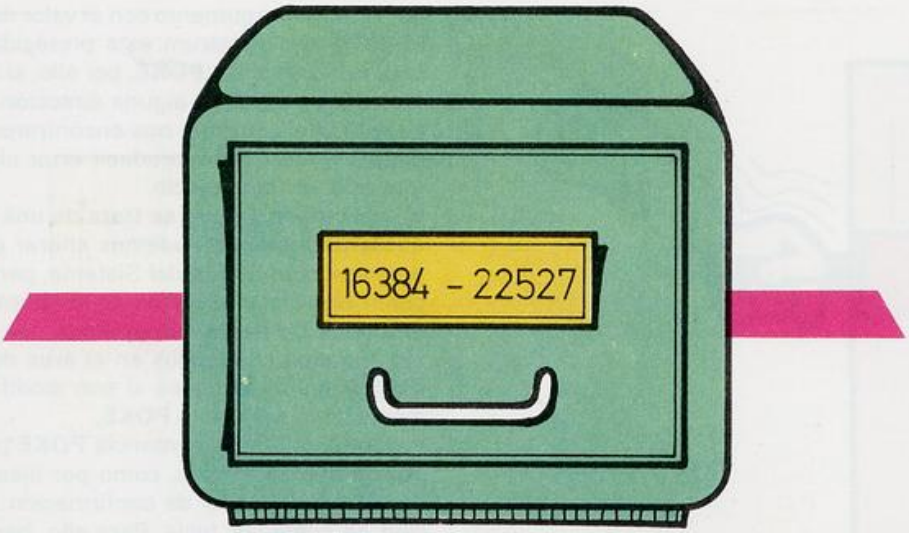
Esta sentencia borra todas las variables BASIC, el contenido de la pantalla **(CLS)**, reajusta el puntero de la sentencia **PLOT** al origen de la posición del extremo inferior izquierdo de la pantalla, ejecuta un **RESTORE** al comienzo de las **DATA**, borra el *stack* de los **GO SUB** (cancela cualquier retorno pendiente) y asigna un nuevo límite superior a la memoria RAM *(RAMTOP)*, manteniendo intacta la zona de texto BASIC.

La nueva asignación puede efectuarse tanto por arriba, recubriendo en este caso la zona de gráficos definidos por el usuario y consiguiendo de esta forma más espacio para los programas BASIC; como por debajo, limitando la longitud del BASIC para liberar una porción de la RAM alta en la cual colocar rutinas escritas en código máquina, o información codificada en binario manejables desde nuestro programa BASIC.

En cualquier caso, puede optarse por emplear **CLEAR** sin asignación de nuevo *RAMTOP*, con lo cual se conseguirán los efectos antes descritos manteniéndose el límite superior de la memoria en el mismo sitio.

Existen otras dos sentencias BASIC vinculadas a **CLEAR: RUN** y **NEW**. La sentencia **RUN** es equivalente a la ejecución consecutiva de un **CLEAR** y un **GO TO** a la línea de comienzo del programa o a la primera línea del mismo si no se especifica





El archivo de imagen del Spectrum ocupa las posiciones decimales comprendidas entre 16384 y 22527.

parámetro en la sentencia **RUN. NEW,** por su parte, no es más que una ampliación de las facultades de **CLEAR,** ya que ésta última borra todo el contenido de la memoria incluyendo el programa BASIC y sus variables, para asignar un nuevo *RAMTOP* en la zona más alta de la memoria posible, dependiendo de la cantidad total de RAM de que disponga el ordenador.
Otra aplicación de la sentencia **CLEAR** puede ser la de simular en un modelo de 48 K que nos encontramos trabajando con uno de 16 K. Para ello, bastaría con escribir:

CLEAR 32767

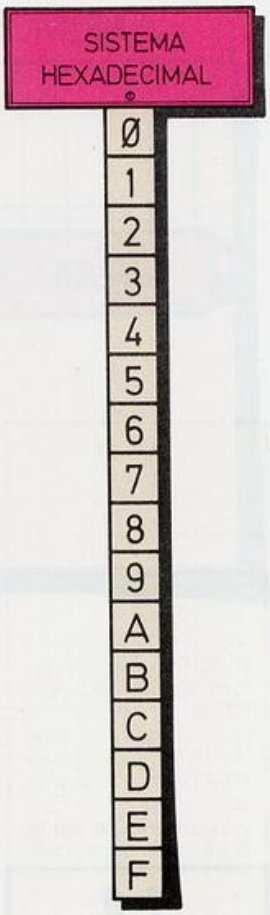
LA SENTENCIA PEEK

El formato general de la sentencia es:

PEEK dirección

La función devuelve como resultado el valor decimal del *byte* contenido en la dirección especificada, cuyo valor debe oscilar entre 0 y 65535. **PEEK** nos permite investigar cualquier posición tanto de la memoria RAM como ROM, puesto que la protección de que dispone la memoria ROM es contra escritura y la función **PEEK** no modifica con su ejecución, en modo alguno, el contenido de la posición encuestada.
La siguiente línea nos muestra el contenido de las 40 primeras posiciones de la memoria ROM:

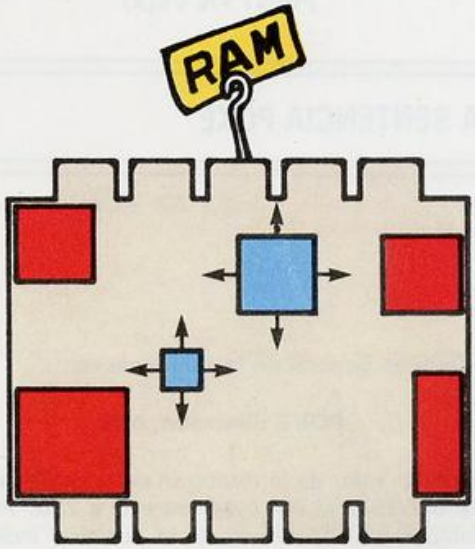
```
FOR I=0 TO 39:PRINT PEEK I,;:NEXT I
```

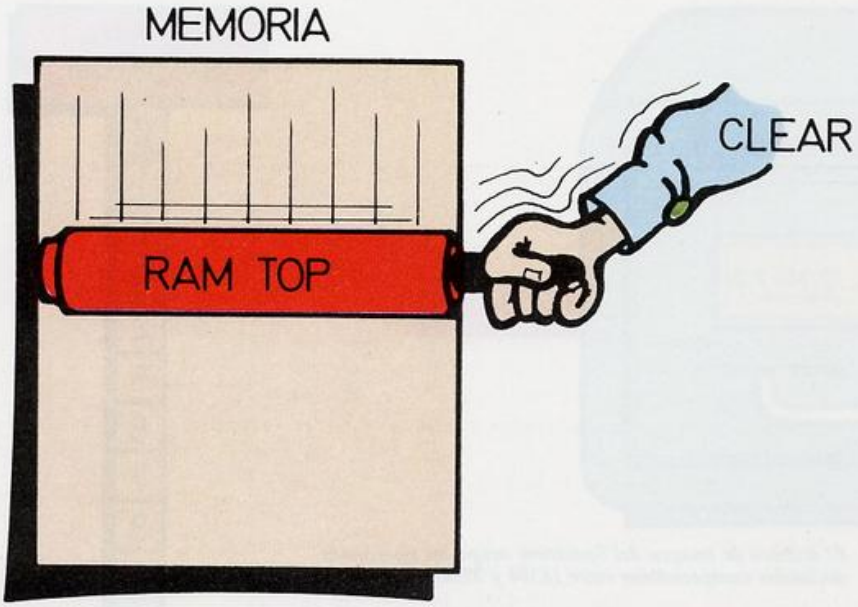


Otra posibilidad que nos brinda esta sentencia es la de averiguar el valor de una palabra, es decir, el valor codificado en binario en dos posiciones de memoria RAM consecutivas, como es el caso de las variables del Sistema.
El valor codificado en binario se calcula como la

El sistema hexadecimal se basa en los dígitos decimales del 0 al 9, completando éstos hasta el total de 15 con las letras de la A a la F.

Dentro de la memoria RAM existen zonas que mantienen un emplazamiento fijo, y otras que se van desplazando según las necesidades.





La sentencia **CLEAR** *tiene por misión asignar un nuevo* **RAMTOP.**

suma del contenido del primer *byte* más el segundo multiplicado por 256. Podemos definir una función de usuario que cumpla este cometido de la forma:

DEF FN V$(X)=PEEK X+256*PEEK (X+1)

Siendo **X** la posición de comienzo de la variable binaria, obtendremos el valor de cualquier palabra con:

PRINT FN V$(X)

¡!

El valor codificado en binario de una «palabra» se calcula como la suma del contenido del primer *byte* más el segundo multiplicado por 256.

*

Una aplicación de la sentencia **CLEAR** puede ser la de simular en un modelo de 48 K que nos encontramos trabajando con uno de 16 K.

LA SENTENCIA POKE

El formato general de la sentencia es:

POKE dirección, *byte*

Donde el valor de la dirección debe oscilar entre 0 y 65535 y el del *byte* entre 0 y 255. **POKE** reemplaza el contenido de la posición indicada por el primer argumento con el valor del segundo. La ROM del Spectrum está protegida contra el uso de la sentencia **POKE**, por ello, si intentamos efectuar un **POKE** a alguna dirección por debajo de 4000 hexadecimal nos encontraremos que el BASIC la ejecuta sin producir error alguno, aunque ésta no tiene efecto.

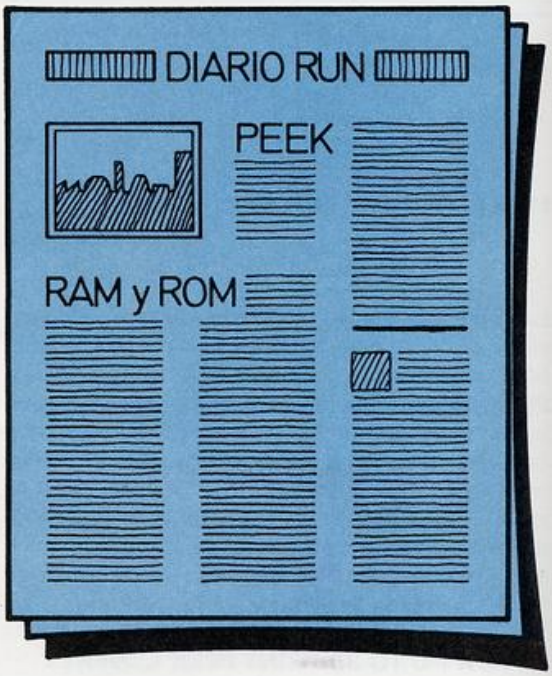
No cabe duda de que se trata de una protección eficaz para que no podamos alterar por error la programación básica del Sistema, pero debemos poner especial cuidado en no modificar tampoco, al menos de forma incontrolada, los valores de los punteros contenidos en el área de Variables del Sistema, los cuales sí son modificables por medio de la sentencia **POKE**.

Podemos utilizar la sentencia **POKE** para conseguir múltiples efectos, como por ejemplo la obtención de un «clic» de confirmación a la pulsación de cualquier tecla. Para ello, basta con escribir:

POKE 23609,n

Donde n oscila entre 0 (sin sonido) a 255 especificando la duración deseada de éste. Los valores muy altos retardan bastante la ejecución de la entrada de datos, por ello es recomendable utilizar un valor no superior a 16, que es perfectamente perceptible.

PEEK *permite investigar cualquier posición de la memoria RAM o ROM.*



EN ALTA VOZ

CUANDO se trata de mejorar los resultados de un determinado programa, normalmente se le incorporan efectos sonoros destinados a realzar las posibilidades gráficas de éste, o servir de guía a cada una de sus opciones.

El estallido de una bomba, el potente rugido del motor de un fórmula 1, una melodía popular, los gritos del público en un gran estadio, son todos ellos buenos ejemplos de las composiciones sonoras realizadas para nuestro micro. Incluso programas más serios, amenizan los tiempos de espera necesarios para los cálculos, procesando simultáneamente con éstos subrutinas sonoras.

Pero de todos es conocida una de las debilidades más acusadas del Spectrum: su deficiente capacidad para la emisión de sonidos. El pequeño minialtavoz cumple a duras penas con su cometido, siendo insuficiente a todos los efectos cuando disponemos de un programa, el cual haga alarde de unas buenas cualidades sonoras.

Ante ello, caben varias posibilidades para conseguir aumentar las bajas prestaciones del *buzzer* (tipo de altavoz que incorpora el Spectrum), que podemos clasificar en los siguientes tres grupos:

1. Manejo de la señal MIC por un amplificador de audio.
2. Conexión de un microperiférico amplificador especial para Spectrum.
3. Incorporar al Sistema un adaptador de sonido para televisor.

La primera es la opción más barata, y está al alcance de cualquiera el poder incorporarla a su micro. Consiste en acoplar la salida del Spectrum señalada con MIC a un amplificador de audio convencional, dotado éste último de salida para altavoz exterior, como se muestra en la figura.

De esta manera, actuando sobre los controles de volumen y tono de amplificador, podemos conseguir el nivel sonoro adecuado a cada programa, sin necesidad de «pegar el oído» al teclado de nuestro ordenador.

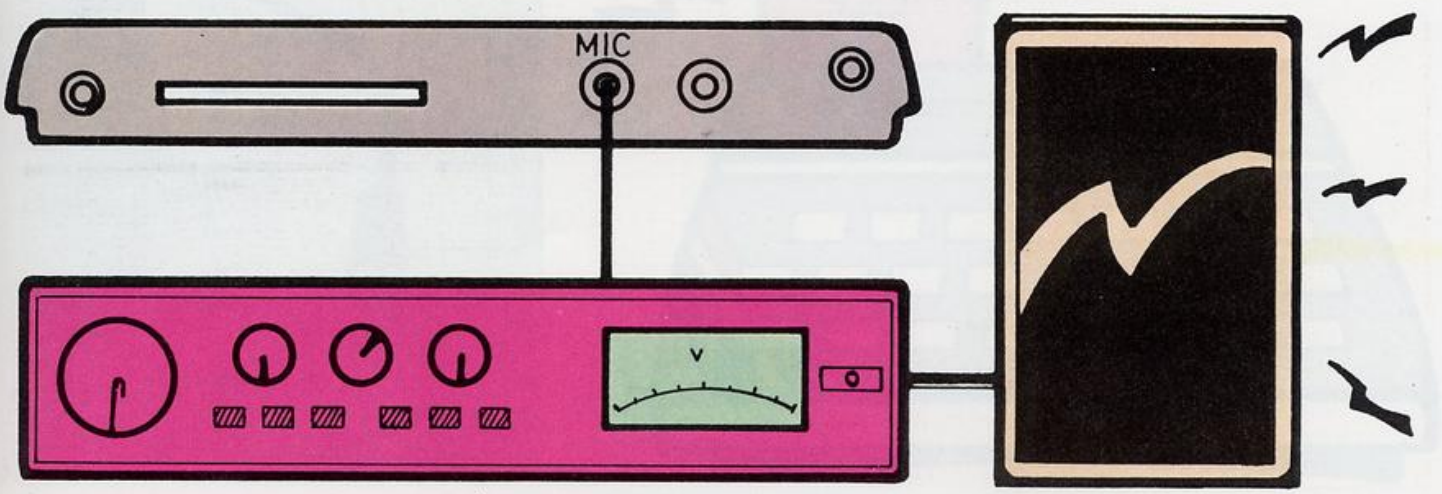
Tampoco es necesaria la utilización de un completísimo amplificador de alta fidelidad (sería desaprovechar las virtudes de éste, además de un abuso de decibelios). Quizá baste con el manejo del casete que normalmente empleamos para cargar y grabar programas.

Para ello, generalmente, será suficiente pulsar la tecla **RECORD** o **REC** de la grabadora para poder acceder a su amplificador, teniendo la precaución de actuar sobre la guía de protecc6n contra borrado, para no forzarla y provocar una siempre molesta avería.

La calidad del sonido reproducido por este sistema es bastante buena, siempre que éste se produzca de forma continua o sin grandes pausas entre impulsos sonoros, pues de otra manera, cuan-

¡!

A la hora de escoger un determinado modelo de amplificador de sonido hay que verificar si tiene duplicada la tarjeta de expansión y si dispone de mando regulador de volumen.



Para conseguir aumentar las bajas prestaciones del** buzzer, **podemos utilizar la señal MIC junto con un amplificador de audio.

¡!

Entre los sintetizadores de voz actuales hay dos grupos bien diferenciados: síntesis por software y síntesis por hardware.

*

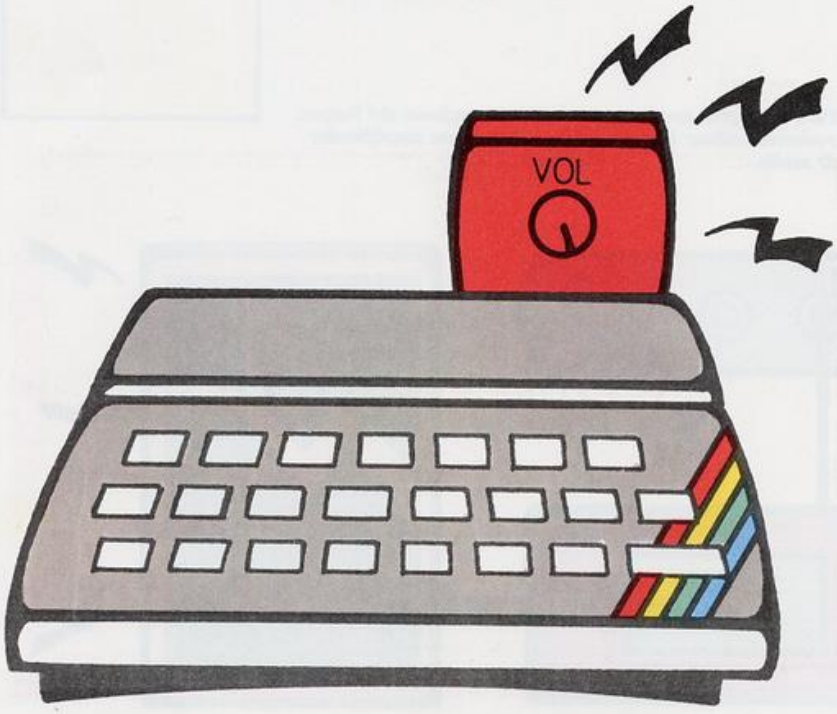
Para mejorar la calidad de un programa se le incorporan efectos sonoros, destinados a realzar las posibilidades gráficas de éste.

do el Spectrum no está emitiendo, la señal procedente de la salida MIC, no estará exenta de ciertas interferencias, y al estar dotados estos equipos portátiles de nivel automático de volumen en la grabación, pueden llegar a convertirse en molestas.

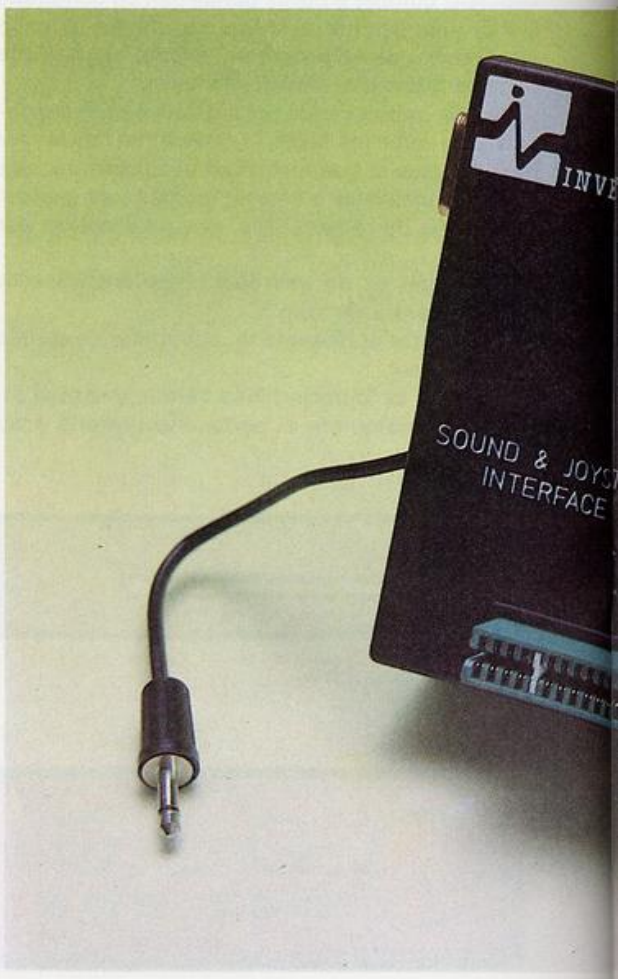
MICROAMPLIFICADORES

La segunda posibilidad consiste en adquirir en el mercado un amplificador de sonido especialmente diseñado para el Spectrum. Periféricos de este tipo, existen para todos los gustos y precios.
Por lo general, se conectan en el port de expansión trasero del cual recogen la alimentación, recibiendo la señal sonora a través de una clavija, la cual se aloja en la salida MIC. Aunque los hay que exclusivamente funcionan como amplificadores, lo normal es que vayan asociados a interfaces para joystick (Investrónica Sound & Joystick Interface, por ejemplo) o formando parte de teclado, como es el caso de los Multifunción de Indescomp.
Sea cual sea el sistema escogido, debemos sopesar ciertos factores antes de decidirnos por uno u otro modelo. Primeramente, conviene verificar si el periférico lleva duplicada o no la tarjeta de expansión, pues en caso de no ser así, impide la conexión de nuevos interfaces, limitando la capacidad de ampliación de nuestro Spectrum. También es importante considerar si dispone de mando para regular el volumen, con objeto de inhabilitar el sonido siempre que se desee, además de permitir su ajuste.
La calidad obtenida es aceptable y el nivel de ruidos por interferencias disminuye bastante, pues

Si queremos sacar más provecho del sonido de nuestro Spectrum, podemos hacer una conexión con un microperiférico amplificador.



Interface para sonido y joystick de Investrónica.



por lo general, estos aparatos incorporan filtros para minimizarlos.

A TRAVES DEL TELEVISOR

Un sistema completamente diferente a los dos anteriores, consiste en hacernos con un adaptador de sonido para televisor. Como vimos en el capítulo dedicado a los monitores y pantallas de visualización, el modulador del Spectrum sólo genera la señal necesaria para producir la imagen de vídeo, y no conjuntamente ésta con la sonora, como ocurre en la mayoría de los microordenadores de otras marcas.

Estos aparatos generan la subportadora de sonido mezclándola con la de vídeo, de manera que los sonidos emitidos por nuestro micro sean reproducidos por el altavoz del televisor. Obviamente, podremos actuar sobre los controles de tono y volumen de este último, para ajustarlo al nivel óptimo deseado.

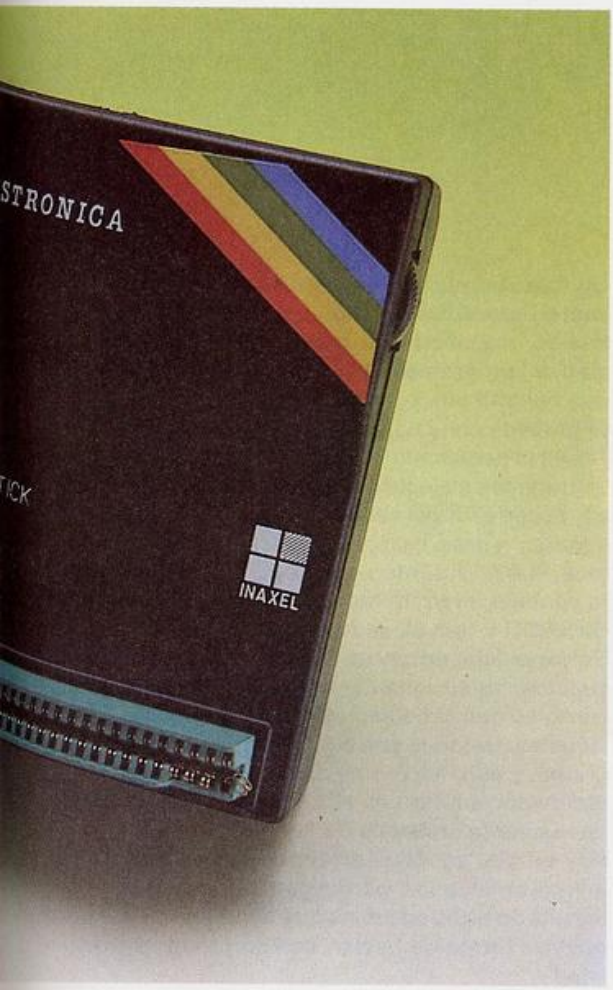
Las interferencias sonoras desaparecen prácticamente en su totalidad, aunque la mezcla de ambas señales suele provocar unas ligerísimas perturbaciones en la imagen de televisión, siendo más acusadas en un aparato de color que en uno en blanco y negro.

Buen ejemplo de ello, es el comercializado por la firma SYNTER. Se conecta en la ranura de expansión, de donde recoge la corriente necesaria para su funcionamiento. Mediante un corto cable que se acopla en la salida MIC del Spectrum, toma las señales sonoras de éste, y una vez generada la subportadora de sonido, la envía a través del pin 15B para su posterior emisión hacia el televisor por la salida de antena.

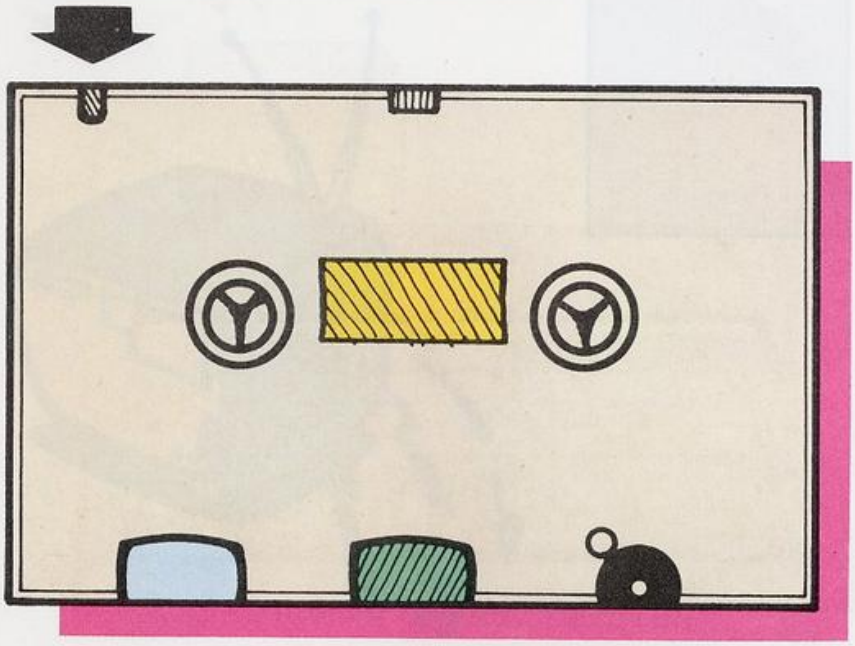
La tarjeta está duplicada en la parte trasera de este interface, por lo cual las posibilidades de expansión al conectarlo, no quedan limitadas a un único periférico.

¡!

Otra opción para mejorar el sonido de nuestro ordenador consiste en adquirir un amplificador especial para Spectrum, el cual se conecta al *port* de expansión trasero.



Para utilizar el casete como reproductor de sonido basta con pulsar la tecla **RECORD**, *teniendo la precaución de actuar sobre la guía de protección contra el borrado.*



¡!

Podemos utilizar el televisor como reproductor de sonido, gracias a un adaptador a tal fin.

*

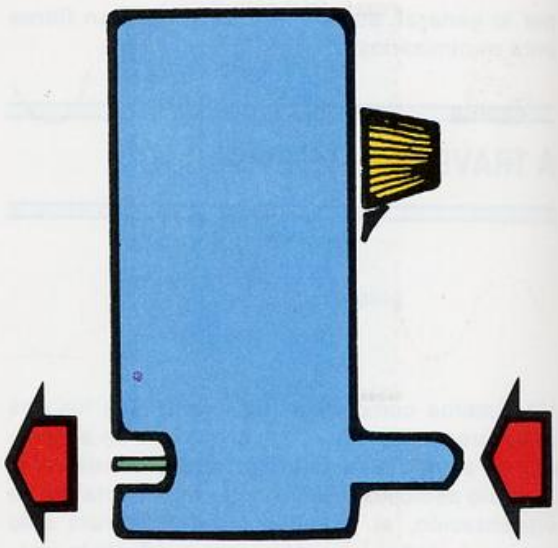
Con los sintetizadores de voz podemos escuchar la «voz» de nuestro Spectrum.

SINTESIS DE VOZ

Hasta ahora, hemos comentado las posibles soluciones encaminadas a incrementar la deficiente emisión del sonido por parte de nuestro micro. Pero entre la amplia gama de periféricos existentes para el Spectrum, podemos encontrar otros capaces de sorprendernos grátamente: se trata de los sintetizadores de voz.

Efectivamente, parece realmente increíble poder escuchar la voz de nuestro ordenador, pero es posible. Y lo que es más importante: se le entiende.

Desde la aparición del Spectrum en nuestro país, uno de los programas dentro del extenso campo del software que por entonces comenzaba a desarrollarse era el SPEAKEASY. Sorprendentemente, una vez cargado en la memoria, aparecía ante nosotros un menú de opciones entre las cuales llamaban poderosamente la atención dos de ellas: oír y hablar.



Uno de los factores que influyen en la elección de un determinado modelo de amplificador es si tiene duplicada o no la tarjeta de expansión.

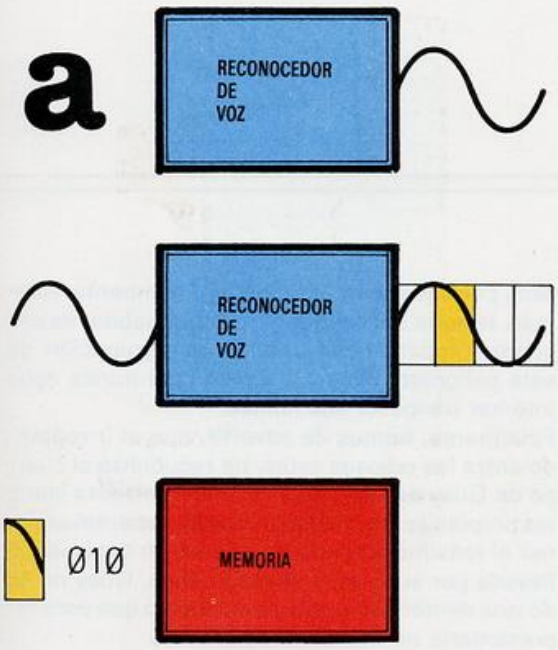
Podemos incorporar un adaptador de sonido para T.V., proporcionando un mayor realismo a nuestros juegos.

Un tanto perplejos y desconfiados ante aquel «invento», procedíamos según las instrucciones, las cuales indicaban que préviamente, a la utilización del programa deberíamos grabar una casete con nuestra voz, o simplemente emplear una cinta grabada con cualquier melodía.

Ahora conectábamos Spectrum y grabadora como si fuéramos a ejecutar un comando **LOAD**, es decir, salida EAR del casete con entrada EAR del ordenador y tras escoger la opción OIR, pulsábamos PLAY. Durante unos segundos aparecía en la pantalla intermitentemente, el mensaje ESCUCHANDO y tras él, se retornaba al menú.

Sólo quedaba entonces comprobar la eficacia del sistema, seleccionando la opción HABLAR. Y lo cierto es que si habíamos utilizado palabras individuales, tratando con cierta benevolencia al programa, y echándole una dosis de imaginación, el ordenador emitía por el minialtavoz los sonidos previamente grabados en la cinta.

Hoy en día, aquellos resultados están completamente anticuados, pero significaron el punto de partida de otros sistemas que actualmente imitan nuestra forma de hablar, con sorprendente fidelidad.



Esquema del proceso de conversión analógico/digital.

métodos, radica en la distinta localización del programa de control encargado de analizar los códigos binarios (señal digital) correspondientes a determinado sonido y posteriormente generarlo (señal analógica), así como en quién es el encargado de sintetizarlo.

Normalmente, en los primeros, este programa se carga desde una cinta en la memoria RAM del ordenador, donde tras el análisis, se procesan rutinas que activan el *buzzer,* de tal manera que podamos escuchar a través de él las voces seleccionadas.

En los sintetizadores por hardware, el programa de control se halla habitualmente grabado en memorias ROM o EPROM, incluidas dentro de un interface directamente conectable al Spectrum. Es en el interioir de este periférico donde se aloja un chip sintetizador de voz, del cual parte la señal sonora hacia un altavoz.

Al proceso seguido tanto en uno como en otro tipo de sintetizador se le denomina conversión analógica/digital. En la figura adjunta se representan básicamente las distintas etapas que debe recorrer la información hasta conseguir que el ordenador la convierta en una señal sonora.

¡!

Las diferencias principales entre la síntesis de voz por soft y hard, radican en la localización del programa de control y quién es el encargado de la sintetización.

*

Existen varias posibilidades para mejorar la calidad de sonido del Spectrum. Una de ellas puede ser acoplar a la salida MIC de nuestro ordenador un amplificador de AUDIO con salida a altavoz exterior.

CONVERSORES D/A Y A/D

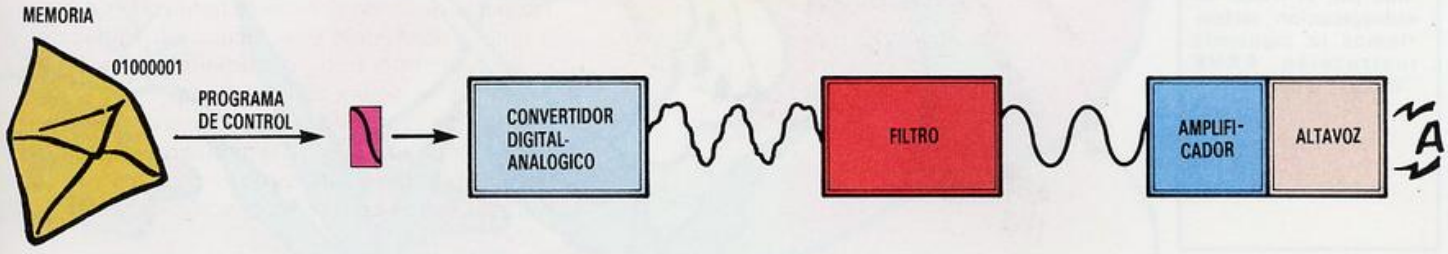
Entre los sintetizadores actuales cabe distinguir dos grupos bien diferenciados, según la forma de realizar su trabajo: síntesis de voz por software y síntesis por hardware. La diferencia entre ambos

CONTINUARA...

En el próximo capítulo, analizaremos dos sintetizadores, cada uno perteneciente a uno de los grupos anteriores: el comercializado por PIN SOFT y el CURRAH μSPEECH de CECOMSA.

Además, el proceso inverso también es posible, es decir, mediante un interface especial, nuestro Spectrum es capaz de reconocer un grupo de palabras si nosotros previamente le instruimos para identificarlas. Para ello, comentaremos las posibilidades del MICRO COMMAND, un dispositivo capaz de actuar como conversor analógico/digital.

Esquema del proceso de conversión digital/analógico.



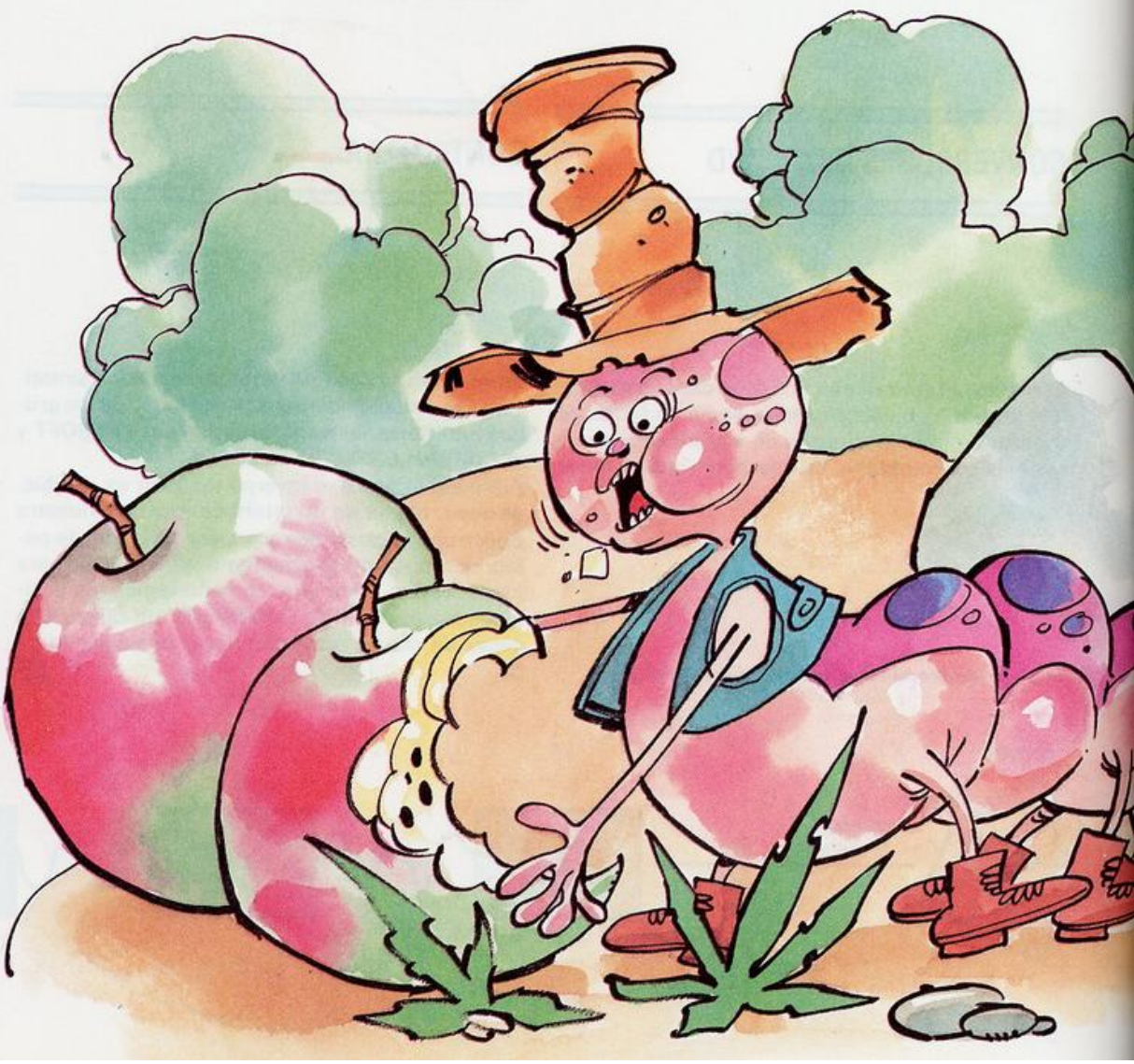
GUSY

GUSY es una gusanita muy especial: le encanta comer caramelos de menta vitaminados. Cada vez que encuentra uno de estos dulces, aumenta el número de sus anillos y en consecuencia, su tamaño.

El simpático animalillo es un poco torpe a la hora de buscar su golosina preferida, por lo cual necesita que alguien más hábil que ella le ayude a descubrir su alimento. Nosotros seremos su guía. Pero... ¡cuidado! el terreno del bosque en el cual vive Gusy es muy prolífico en setas venenosas.

Si nuestro gusanito comiese solamente una de esas plantas sucumbiría instantáneamente. Además, la zona del bosque en la cual habita, se encuentra rodeada por una densa agrupación de este peligroso alimento, por lo cual nunca debe intentar traspasar sus límites.

Finalmente, hemos de advertir, que el ir reptando entre las odiosas setas, ha recubierto el cuerpo de Gusy de una sustancia tan mortífera como las propias setas, debido a lo cual, deberemos poner el máximo empeño en que en un descuido, o llevada por su apetito pantagruélico, Gusy no se dé una dentellada a sí misma, puesto que esto representaría un irremediable suicidio.



¡!

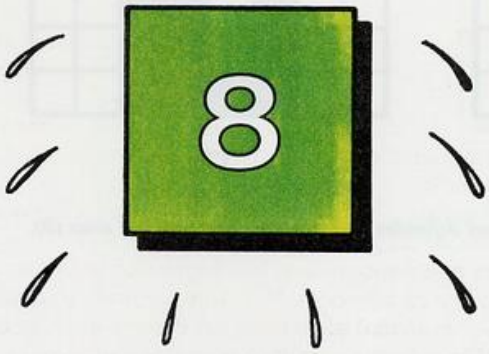
El tamaño de Gusy aumentará dependiendo del número de vitaminas que tenga el caramelo que ingiera.

*

A la hora de grabar el programa lo haremos utilizando el siguiente comando: **SAVE "GUSY"**. Si optásemos por el modo de autoejecución teclearíamos la siguiente instrucción: **SAVE "GUSY" LINE 1.**

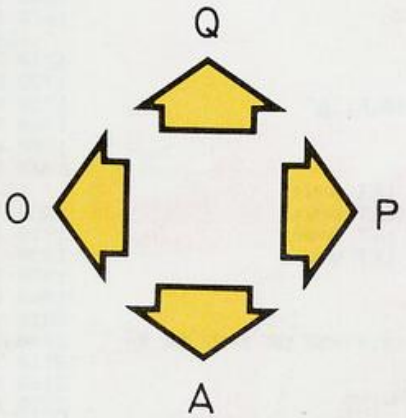
EL PROGRAMA

Una vez el programa ha sido correctamente introducido y ejecutado, aparecerá en la pantalla el protagonista de la aventura: Gusy. Junto con ella surgirá una posición de carácter intermitente de color verde (el caramelo vitaminado), en cuyo interior se observa un número. Dicho número, variable de 0 a 9, indica la cantidad de anillos que Gusy crecerá si lo come.



En la pantalla aparecerá un carácter de color verde intermitente, en cuyo interior se observa un número del 0 al 9 que indica la cantidad de anillos que Gusy crecerá.

Las setas venenosas surgirán cada vez que Gusy ingiera un caramelo o cada cuatro cambios de sentido (pulsaciones de tecla).



Estas son las teclas que permiten a Gusy cambiar el sentido de su marcha.

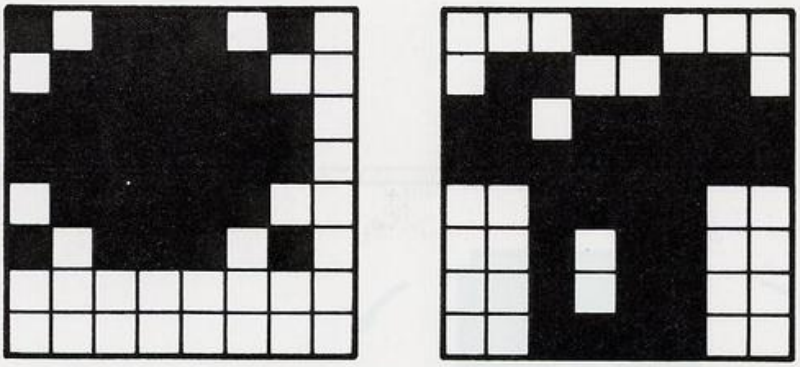
¡!

Los controles para la dirección de nuestra gusanita son los siguientes:

Q ..... Arriba.
A ..... Abajo.
O ..... Izquierda.
P ..... Derecha.

*

Gusy morirá si durante su camino come alguna seta venenosa.



Gráficos definidos para el juego: anillos (A) y setas (B).

Así pues, si el caramelo vitaminado indica el número 1, nuestra gusanita aumentará un anillo en su longitud. Si el dulce señalase el 2, Gusy crecería dos anillos más. Y así sucesivamente.

En contra de lo que se pudiese esperar, si Gusy comiese un caramelo marcado con el número 0, aumentaría 20 unidades, y en consecuencia la dificultad del juego crecería considerablemente.
Las setas venenosas surgirán cada vez que Gusy ingiera un caramelo vitaminado, o cada cuatro cambios de sentido, por lo cual nosotros deberemos tener las máximas precauciones a la hora de dirigir a nuestra protegida.
El juego irá aumentando de dificultad a medida que Gusy vaya creciendo y proliferen las setas venenosas. Sólo con habilidad, paciencia y en ocasiones un poco de sangre fría, lograremos que Gusy no sucumba entre los peligros del bosque.
Las teclas que permiten cambiar de dirección a Gusy son las siguientes:

Q ..... Arriba.
A ..... Abajo.
O ..... Izquierda.
P ..... Derecha.

Para la adopción del listado, seguiremos los criterios habituales en lo referente a representación de gráficos definidos, etc.

```
 10 REM ************************
 20 REM * J.M.MAYORAL SERRANO *
 30 REM ************************
 40 REM * Prg. GUSY  (C) 1985 *
 50 REM ************************
 60 BORDER 1
 70 BRIGHT 0
 80 PAPER 7
 90 CLS
100 LET Z=0
110 POKE 23658,8
160 LET W=0
200 RESTORE 210
210 DATA 186,124,254,254,254,124,186,0
220 DATA 24,102,223,255,60,44,44,60
230 LET Q$="AB"
240 FOR N=1 TO LEN Q$
260 FOR F=0 TO 7: READ A
263 POKE USR Q$(N)+F,A
265 NEXT F
270 NEXT N
310 LET G$="P": LET O=0: CLS
360 LET X=10: LET A$=""
365 LET Y=19
410 FOR F=10 TO 19
420 LET A$=A$+"10"+STR$ F
430 PRINT  BRIGHT 1; INK 0;AT 10,F;"A"
440 NEXT F
460 GO SUB 1560
510 LET z$=INKEY$
560 IF Z$="A" THEN  LET G$=Z$: LET W=W+1
610 IF Z$="Q" THEN  LET G$=Z$: LET W=W+1
660 IF Z$="O" THEN  LET G$=Z$: LET W=W+1
710 IF Z$="P" THEN  LET G$=Z$: LET W=W+1
760 LET X=X+(G$="A")-(G$="Q")
770 LET Y=Y+(G$="P")-(G$="O")
810 IF W>3 THEN  GO SUB 2010
860 IF ATTR (X,Y)=120 OR ATTR (X,Y)=58 OR X=22 OR Y=
32 THEN  GO TO 1610
910 GO SUB 1410
960 LET S$=STR$ X
970 IF X<10 THEN  LET S$=" "+STR$ X
1010 LET D$=STR$ Y
1020 IF Y<10 THEN  LET D$=" "+STR$ Y
1060 LET X1=VAL A$( TO 2)
1070 LET Y1=VAL A$(3 TO 4)
1080 IF ATTR (X1,Y1)<>120 THEN  GO TO 1160
1110 PRINT AT X1,Y1;" "
1160 PRINT  BRIGHT 1; INK 0;AT X,Y;"A"
1210 BEEP .001,40
1260 LET A$=A$+S$+D$
1270 IF O=0 THEN  LET A$=A$(5 TO )
1310 LET O=O-(O>0)
1360 GO TO 510
1410 LET F$=SCREEN$ (X,Y)
1412 IF F$>"9" OR F$<"0" THEN  RETURN
1415 IF F$="0" THEN  GO SUB 2010: LET W=6: LET O=O+20
1420 LET Z=1
1460 LET O=O+VAL F$
1510 BEEP .01,0: BEEP .01,10: BEEP .01,20
1520 GO SUB 2010
1560 PRINT  FLASH 1; BRIGHT 1; PAPER 4;AT (RND*22)-1,
(RND*32)-1;INT (RND*10): RETURN
1610 LET X=X-(X=22)+(X=-1)
1620 LET Y=Y-(Y=32)+(Y=-1)
1660 PRINT AT X,Y; FLASH 1; PAPER 2; INK 6;"A"
1670 PRINT  PAPER 2; INK 6;AT 10,5;"    PUNTOS  =
 ";(LEN A$/4)-10;" "
1710 FOR F=20 TO 40
1720 BEEP .02,F
1730 NEXT F
1760 FOR F=1 TO 100
1780 NEXT F
1810 PRINT  PAPER 0; INK 6;AT 12,3;" OTRA PARTIDITA ?
   (S/N) "
1860 IF INKEY$="" THEN  GO TO 1860
1910 IF INKEY$="S" THEN  RUN
1920 IF INKEY$="N" THEN  GO TO 10000
1930 BEEP .01,40
1960 GO TO 1860
2010 LET Q=INT (RND*22)
2060 LET A=INT (RND*32)
2110 IF ATTR (Q,A)<>56 THEN  RETURN
2160 PRINT  INK 2;AT Q,A;"B"
2210 BEEP .05,40
2260 IF Z=0 THEN  LET W=0: RETURN
2270 LET Z=0
2310 RETURN
```