

GRAN BIBLIOTECA AMSTRAD



GENERACIÓN DE SONIDO

EL ORDENADOR

GRAN BIBLIOTECA
AMSTRAD

19
GENERACIÓN DE SONIDO

Director editor:

Antonio M.^o Ferrer Abelló

Director de producción:

Vicente Robles

Director de la obra:

Fernando López Martínez

Redactor técnico:

Rafael de la Ossa Villacañas

Colaboradores:

L-H Servicios Informáticos

Pilar Manzanera Amaro

Diseño:

Bravo/Lofish

Maquetación:

Carlos González Amezúa

Dibujos:

José Ochoa

Fotografía:

Grupo Gálata

© Ediciones Ingelek, S. A.

Todos los derechos reservados. Este libro no puede ser, en parte o totalmente, reproducido, memorizado en sistemas de archivo, o transmitido en cualquier forma o medio, electrónico, mecánico, fotocopia o cualquier otro sin la previa autorización del editor.

ISBN del tomo: 84-7708-060-7

ISBN de la obra: 84-7708-004-6.

Fotocomposición: Andueza, S. A.

Imprime: Eurosur, S. A.

Depósito legal: M-20970-1987

Precio en Canarias, Ceuta y Melilla, 435 ptas.

Octubre 1987.

GENERACION DE SONIDO

Introducción	5
El sonido es diferente	7
Leyendo música	17
Componiendo música	33
Armonía y estéreo	49
Envoltentes de sonido	69
El ruido	89
Apéndice I	103
Apéndice II	109

INTRODUCCION



inco son los sentidos que el ser humano posee: gusto, tacto, olfato, vista y oído. Evidentemente, un ordenador no lo podemos degustar, pero no escucharlo sería renunciar a uno de nuestros sentidos.

El sonido ocupa un lugar preponderante dentro de la programación. Todo programa, sea educativo, de juego, cálculo, negocios, contabilidad, etc., posee el sonido como uno de sus componentes esenciales. Si lo nuestro son los videojuegos sabremos perfectamente cuánta espectacularidad y adicción pierde un juego sin un buen acompañamiento musical.

En otros programas, el sonido se emplea para enviar mensajes, pues recordemos que la música es un lenguaje, distinto al convencional, pero universal. Nuestro mismo ordenador nos avisa con un pequeño pitido cuando se produce algún error.

Además, los poseedores de un AMSTRAD CPC tenemos la suerte de disponer de un ordenador con auténtica potencia en este campo. Sólo tiene un pequeño defecto: su altavoz interno no reproduce las notas todo lo nítidas que desearíamos para nuestros oídos, aunque este obstáculo puede ser salvado conectando nuestro equipo a un sistema estéreo; en cuanto a la forma de conseguir esta conexión, la estudiaremos brevemente en el segundo apéndice de este volumen.

Aprenderemos qué es realmente el sonido, cómo se forma y de qué está compuesto. Pasaremos revista a la historia de las escalas musicales, pues para escribir y componer música debemos aprender primero a leerla y a transcribirla a un papel, y de aquí, a nuestro ordenador.

Encontraremos posteriormente fórmulas sencillas para componer música, acompañarla, producir armonía y efecto estereofónico, creando acordes. Todo ello, unido a la explicación concienzuda de todos los comandos BASIC referentes al sonido que utiliza nuestro AMS-TRAD, aprendiendo a utilizar las envolventes de sonido que producen efectos extraordinarios.

Finalizaremos con una forma de sonido muy especial, pero sonido al fin y al cabo: el ruido, muy fácilmente producido por el ordenador, y con el cual se pueden llegar a conseguir todo tipo de efectos especiales.

Aprenderemos, en resumen, a crear sonido en nuestro ordenador, y a conocer mejor ese apasionante mundo sin fronteras para la imaginación: el mundo de la música.

EL SONIDO ES DIFERENTE



Para empezar hemos de resaltar lo diferente que es la creación y manipulación del sonido, respecto a cualquier otra función que pueda realizar nuestro ordenador. Imaginemos por un momento un programa de gráficos. Observando unos instantes el listado del mismo, podremos llegar a tener cierta idea de lo que va a aparecer en la pantalla una vez haya sido ejecutado. Con el sonido no ocurre lo mismo, y a pesar que tengamos una larga experiencia, es muy difícil quedarnos mirando un programa y adivinar la melodía que vamos a escuchar, o al menos el tipo de sonido que vamos a percibir.

Además, describir un sonido no es algo sencillo. Quizá podamos calcular en un papel aquello que va a aparecer en la pantalla de nuestro ordenador cuando tratamos un programa gráfico, o los números que serán presentados en la pantalla si ejecutamos un programa de contabilidad. Hablar de sonido es diferente. Para entender y adivinar lo que va a emitir el pequeño altavoz de nuestro Amstrad, es preciso un conocimiento más que respetable de programación en BASIC en materia de sonido.

Cualquier enciclopedia puede decirnos que el sonido es la «transmisión, a través de un cuerpo elástico, de las vibraciones producidas por un cuerpo (foco sonoro) en vibración; también se suele designar

como sonido al efecto producido en nuestro oído por las vibraciones elásticas». Pero empezemos desde el principio.

La mayor parte del sonido que nosotros podemos escuchar se transmite por medio del aire, así que vamos a tratar algún tiempo sobre él. El medio llamado aire es un material elástico, cuyo volumen es posible alterar por medio de la compresión o expansión del mismo.

Si tomamos la bomba de una bicicleta y empujamos el mango hacia dentro, al tiempo que tapamos con un dedo el orificio de salida de aire, estamos comprimiéndolo, con lo cual disminuimos su volumen (ocupa menos espacio dentro de la bomba). Una jeringuilla llevaría a cabo el mismo trabajo en sentido inverso, es decir, aumentaría el volumen del aire: lo expandiría.

Tomemos ahora la bomba llena de aire comprimido y soltemos el dedo del orificio de salida. El aire saldrá de allí en todas las direcciones formando una onda de aire comprimido. Algo muy parecido son las ondas de sonido. Que nosotros escuchemos un sonido o no, depende ante todo de la vibración del aire que llega hasta nuestros oídos.

Cuando el ordenador produce un sonido, una onda sonora se transmite al aire por el altavoz. Un émbolo que éste posee comprime y libera el aire sucesivamente produciendo la onda que llegará hasta nuestros oídos. Dicha onda encontrará el tímpano, haciendo vibrar los huesecillos que se encuentran en el órgano, con lo cual una señal eléctrica será transmitida a los nervios sensitivos que se encargarán de que nosotros escuchemos un sonido.

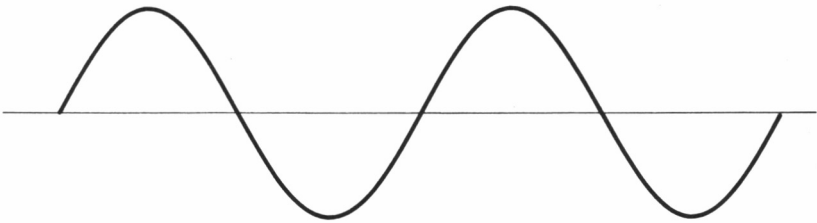
Todo aquello que escuchamos son ondas que viajan de un lado a otro del medio (aire). Una cuerda no puede producir mucho movimiento de aire, pero quizás unido a una caja de resonancia consigamos que al tocar la cuerda, sea la caja quien se encargue de mover el aire por ella. Actualmente, la caja de una guitarra puede incorporar un aparato que convierta las ondas en señales eléctricas, que sean convertidas de nuevo en ondas sonoras amplificadas por un altavoz. Son muchos los instrumentos que mueven el aire directamente, como el clarinete, la trompeta, la flauta, el trombón, la corneta, y otros muchos instrumentos musicales de viento.

CÓMO ES UNA ONDA

En la orilla de un lago de lisas y tranquilas aguas podemos observar al arrojar una piedra ondas muy bien definidas. Sin embargo, las ondas sonoras son invisibles, a no ser que tengamos a nuestra disposi-

ción un equipo electrónico que se encargue de estudiar la forma de las mismas. El aparato que necesitamos es el OSCILOSCOPIO, el cual conectado a un micrófono visualiza cualquier onda sonora que llegue hasta el mismo.

La onda más simple es aquella que únicamente sube y baja, en un movimiento parecido al de una serpiente reptando. Este tipo de onda se llama técnicamente «onda del seno», pues reproduce exactamente dicha función trigonométrica. Sin embargo, las ondas de los instrumentos musicales son mucho más complicadas, y su forma depende sobre todo del instrumento que estemos utilizando, y como esté siendo empleado. Una flauta suena de manera distinta que un piano. Para nuestro cerebro, esto no serán más que distintos tipos de ondas, y rápidamente nos indicará que provienen de instrumentos diferentes (19.1.1).



ONDA DEL SENO

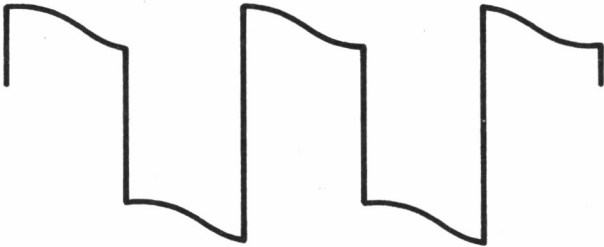
■ 19.1.1.

Un sintetizador de sonido es un instrumento electrónico que puede imitar el sonido de otros instrumentos, produciendo ondas de forma muy similar a las del original. Nuestro Amstrad no posee esas amplias posibilidades, además de producir unas ondas bastas de forma cuadrada. La causa es que este es el tipo de onda más fácil que el ordenador puede generar. Las ondas cuadradas se producen por una conexión y desconexión del voltaje dentro de una zona del ordenador. De todas formas, un sintetizador, además de costar más que un

ordenador, no puede calcularnos la Declaración de la Renta o hacernos vibrar con un apasionante juego de marceanos en una larga tarde de invierno... (19.1.2).



ONDA CUADRADA



ONDA QUE EMITE EL ALTAVOZ

■ 19.1.2.

Existe otra razón por la cual el ordenador utiliza este tipo de onda: es más parecida a la de los instrumentos musicales que las ondas de seno. Así, sucede que los sintetizadores de sonido utilizan este tipo de ondas, produciéndolas eléctricamente, y después las transforman para originar los diferentes sonidos. Esto requiere el uso de unos filtros cuya misión es cambiar la forma de una onda cuadrada.

Con práctica sabremos cómo alterar la forma de las ondas que produce el altavoz de nuestro Amstrad. Otra de sus particularidades es la generación de ruido, cualidad que no posee un violín o un piano, a no ser que se encuentre en manos de, por ejemplo, el redactor de este libro. El ruido lo forman ondas que no tienen una forma ni diseño prefijado. Es el equivalente musical a una serie de números aleatorios, por lo cual el ordenador genera este ruido empleando señales que son controladas por el generador de números aleatorios del computador.

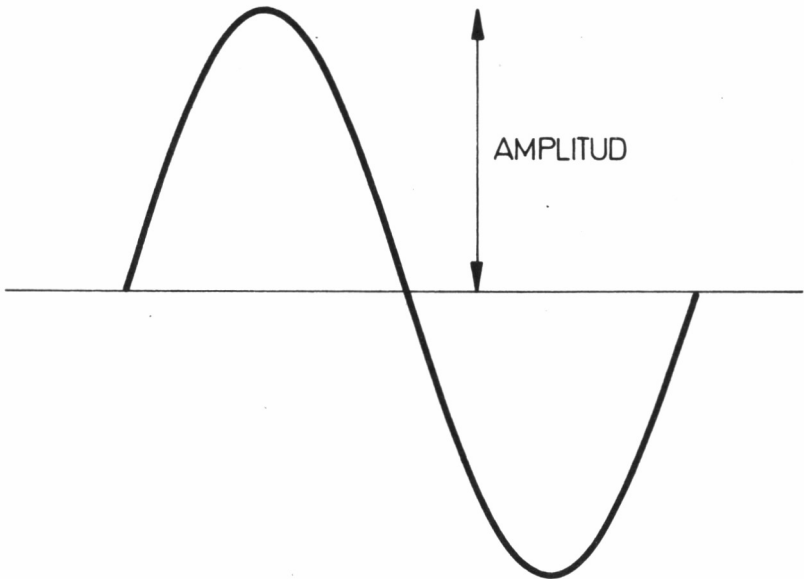
Quizá todavía no lleguemos a comprender la importancia que puede tener este ruido en el sonido de muchos de nuestros programas; disparos, trenes en marcha, estadios enfervorecidos, todos ellos se

consiguen gracias al ruido de nuestro ordenador. Si vivimos en una gran ciudad, acerquémonos a la ventana. ¿Qué escuchamos? Sólo ruido. Gracias a la combinación de ondas cuadradas y de ruido podremos obtener efectos a todas luces impresionantes.

AMPLIEMOS LA FRECUENCIA DE NUESTRO CONOCIMIENTO

Describir un sonido puede ser harto difícil, a no ser que seamos técnicos o aficionados a la alta fidelidad, o a la síntesis musical. Esto es debido a que las palabras que son capaces de describir una onda no son precisamente de uso frecuente para nosotros, a menos que hayamos trabajado con osciloscopios. Seguro que a muchos de nosotros ni siquiera nos suenan las palabras amplitud y frecuencia.

La amplitud de una onda equivale a su altura. Pensemos en una onda flotante en el agua. La amplitud distaría desde el nivel normal del agua hasta la parte más alta de la onda. En el diagrama podemos observar perfectamente cuál es la amplitud de una onda y podemos hacer equivaler este término con el de volumen (19.1.3).



Cuanto mayor sea la amplitud de una onda, más alto será el sonido que nosotros escuchemos. De todas formas no es tan fácil como parece, pues si escuchamos varias notas cuyas ondas poseen la misma amplitud, pronto podremos apreciar que no las escuchamos de la misma manera. Esto es debido a que el oído humano no posee la misma sensibilidad para todos los sonidos. El oído es más sensible para todos aquellos sonidos que se encuentran dentro del conjunto de los que nosotros podemos producir y menos sensible para los que se encuentran fuera de este conjunto.

Existe otro rasgo que poseen las ondas que influye mucho sobre el sonido. Volvamos al lago y miremos de nuevo las ondas producidas por una piedra al caer sobre el agua. Antes de tirar la primera piedra, coloquemos una barra hundida en el fondo del lago de la que sólo asoma una pequeña punta. Ahora tiremos un pequeño cascote a un par de metros de ella y contemos las ondas que llegan a tocar la barra en un minuto.

Esperemos un par de días a que el lago vuelva a la calma y lancemos un buen canto rodado de un par de kilos de peso. El número de ondas que pasa por la barra es mucho mayor, por lo cual podemos decir que las ondas pasan más frecuentemente por allí que en el anterior lanzamiento. La última onda era pues la de mayor frecuencia. Lo mismo ocurre con el sonido, pues al número de ondas que pasan por un punto cada segundo se le llama frecuencia.

El segundo es elegido como medida del tiempo, porque un minuto resultaría demasiado largo. Normalmente, cientos de ondas pasan por nuestros oídos. La frecuencia de una onda se corresponde con lo que nosotros venimos a llamar el tono de un sonido. A un tono bajo le corresponde una baja frecuencia y viceversa.

Las notas de más bajo tono de un órgano pueden llegar a tener tan sólo unas 30 ondas por segundo, y las de mayor tono de una pequeña flauta, pueden alcanzar las 4.000 ondas por segundo. Una frecuencia de una onda por segundo se denomina por el nombre universal de Hertz (Hercio), cuyo nombre homenajea al descubridor de las ondas de radio: Heinrich Hertz.

La frecuencia de las notas que antes escuchábamos en el órgano son de 30 Hz, y las de la flauta de 4.000 Hz. Para abreviar el escribir tantos ceros, como en 4.000 Hz, se suele utilizar la unidad kHz, equivalente a 1.000 Hz, con lo cual los 4.000 Hz equivaldrán a 4 kHz.

El oído humano, en perfectas condiciones, puede detectar sonidos que se encuentren entre los 30 Hz. y los 18 kHz. A medida que el órgano auditivo envejece, pierde la sensibilidad con respecto a las notas de más alto tono, y tan sólo puede escuchar sonidos de no más

de 7 kHz, o incluso menos. De todas formas, la nota de mayor tono que puede generar un instrumento musical es de 4 kHz. Un sonido cercano a los 18 kHz. nos produciría un pérdida casi total o parcial de la sensibilidad auditiva.

La razón por la cual una onda tiene una forma desigual es porque ésta contienen mayores frecuencias que la frecuencia principal. Esto quiere decir que si un instrumento genera notas de 4 kHz, entonces esa nota también posee frecuencias de 8 kHz, 12 kHz, 16 kHz, y de aquí en adelante. Estas frecuencias son una vez, o dos, o tres, o más veces, mayores que la frecuencia de la nota principal.

Veamos lo que queremos decir. Entonando con nuestra voz la nota FA, otra persona puede entonar otra nota FA de mayor o menor tono (de mayor o menor frecuencia) que armonice perfectamente con nuestra nota. Quizá la frecuencia de nuestro FA era de 4 kHz y la de el otro FA que armonizaba (sonaba bien) con el nuestro poseía una frecuencia de 12 kHz.

El sonido posee otro rasgo que es casi tan importante como la amplitud o la frecuencia, pero mucho más difícil de medir. Nosotros podemos definir la amplitud con una medida de longitud, y la frecuencia con un número de Hertz determinado, pero no será tan fácil hacer una medida de la ENVOLVENTE de una nota musical.

Como podemos comprobar, los instrumentos musicales no tocan una música continuada, sin cambios, sino que cada vez que ha de salir al aire un sonido diferente, hace su aparición una nota distinta. Esto por el momento parece obvio. Cada nota es interpretada por separado, por lo que la onda comienza, se mantiene por un momento y desaparece.

Por la forma en la cual los instrumentos musicales han sido contruidos, la amplitud de las ondas no es constante durante el tiempo de una nota. Al principio de la nota, la amplitud es baja, por lo cual el sonido es bajo, entonces la amplitud aumenta hasta su cima, se mantiene durante un momento y baja hasta morir.

Nosotros podemos observar la amplitud de una onda fácilmente con un osciloscopio, y seguramente observaremos algo parecido a la figura: una envolvente. Esta no es la forma de una onda, sino el aspecto que tiene ésta cuando la amplitud está cambiando. Esta envolvente interpreta un gran papel en la forma en la cual llega el sonido hasta nuestros oídos.

Los instrumentos de percusión poseen envolventes que crecen y mueren rápidamente. Muchos instrumentos musicales producen sonidos cuyas envolventes crecen muy deprisa y mueren en dos fases: despacio al principio y rápido al final.

Si deseamos crear estupendas melodías y efectos musicales, debemos ser capaces de controlar las envolventes con precisión. Además de esta variación de amplitud mientras suena una nota, también podemos cambiar el tono de la misma forma. Pero, ¡tengamos paciencia!, todo llegará.

EL SONIDO ESTÉREO

Seguro que todos alguna vez hemos observado la considerable diferencia que existe entre escuchar un sonido en mono y oírlo en estéreo. Se trata de escuchar el sonido que proviene de dos altavoces por los que suenan las mismas notas. El efecto de utilizar dos altavoces como éstos es una pérdida de la ilusión del espacio, algo como «un sonido que proviene de todas partes». El oído humano puede perfectamente, empleando sus dos órganos en perfectas condiciones, apreciar la diferencia entre estéreo y mono.

Sin embargo, la mayor de las diferencias entre mono y estéreo no se basan en el número de altavoces de que dispongamos, sino en el sonido que emerge de los altavoces. Si cada altavoz transmite al aire una versión levemente diferente de el sonido, correspondiente al lugar desde el cual fue grabado, entonces obtendremos efecto de sonido estéreo. Esto significa que podremos más o menos saber donde estuvieron colocados los instrumentos en el momento de la grabación.

Pero además tendremos una sensación de que el sonido proviene de todas partes, y no de un punto determinado. Si no poseemos dos altavoces podemos utilizar los cascos estereofónicos. Cuando los estamos usando, podemos sentir mejor las diferencias existentes entre los dos focos de sonido que cuando escuchamos dos altavoces.

Naturalmente, nuestro ordenador Amstrad es capaz de producir sonidos en estéreo. Esto se debe a que poseemos tres canales de sonido a nuestra entera disposición, y tan sólo tomando dos de ellos podremos producirlo. Si un programa produce distintas señales para cada canal, el resultado será el efecto estéreo. La sensación de estéreo puede ser que no sea muy buena para la música, pero para efectos de sonidos las consecuencias pueden ser fantásticas.

Sin embargo, queramos o no, el sonido sólo proviene de un punto de nuestro ordenador, por lo cual el efecto estéreo puede resultar bastante pobre, aunque conectando nuestro ordenador a unos buenos altavoces o a unos cascos estereofónicos jamás imaginaremos lo que podemos llegar a escuchar.

Conectándolo a una cadena musical podremos controlar el volumen, el tono, e incluso podremos grabar los sonidos para escucharlos cuando lo deseemos. Aunque hemos de avisar que producir sonido con un ordenador necesita, antes de nada, buenas dosis de práctica y paciencia. Algo muy importante es escuchar melodías y sonidos de ejemplo, como de los que está plagado todo este libro.

NADIE ES PERFECTO

En efecto, nadie ni nada es perfecto, y los sistemas de sonido no iban a ser una excepción. La causa es que el ordenador se alimenta con una fuente principal de electricidad. Esta fuente principal posee una onda eléctrica cuya frecuencia es de 50 Hz, y cuando es alterada en los DC que nuestro ordenador necesita para operar, se genera una frecuencia de 100 Hz.

Esta señal particular no inquieta a los circuitos del computador, porque estos trabajan a velocidades muy altas e ignoran algo tan lento. Los circuitos de sonido, toman esta señal y la amplifican. Estos suenan bastante mal porque su frecuencia es bastante baja y los sonidos no se escuchan muy claros en los cascos estereofónicos ni en el altavoz interno. Esto es debido a que los pequeños altavoces no reproducen muy bien frecuencias muy bajas. Si conectamos nuestro ordenador a una cadena de alta fidelidad, podremos escuchar el sonido con mayor calidad.

Los sonidos que provengan de un Amstrad poseerán una frecuencia que distará entre 30 Hz. a 125 kHz. Sin embargo, 30 Hz es demasiado poco para unos amplificadores, cuando la mayoría de los altavoces «retransmiten» a más de 80 Hz. Debemos, pues, evitar este tipo de frecuencia a menos que conectemos nuestro micro a un Hi-Fi. Pero también deberíamos, por otro lado, evitar frecuencias por encima de los 15 kHz.

De nuevo, buenos altavoces podrían reproducir frecuencias más altas, pero nuestro pequeño Amstrad, no. Pensemos que las frecuencias por encima de los 20 kHz. son sólo escuchadas por perros, gatos y murciélagos y de poco pueden servirnos a nosotros.

Nosotros no podemos grabar esas frecuencias, y a menos que no dispongamos de un osciloscopio jamás podremos detectarlas. Para todas nuestras melodías y efectos de sonido, lo mejor es utilizar una frecuencia cuyo rango se encuentre entre 100 Hz. y 4 kHz, que aunque el sonido que escuchemos sea más o menos musical, más o menos molesto, al menos podremos escucharlo, y ya es bastante.

LEYENDO MÚSICA



Una típica frase de todos los tiempos dice que toda música está formada por sonidos, pero no todos los sonidos son música. En la antigüedad, una de las más avanzadas civilizaciones, cultivadora de todas las artes, los griegos, pensaron que la música era parte de las matemáticas, y que toda melodía podría resumirse en una fórmula matemática.

En primer lugar, debemos asumir que no sabemos absolutamente nada de música, todo esto para empezar por el principio. Sin embargo, si realmente sabemos o somos entendidos, gran parte de este capítulo podrá resultarnos un tanto elemental. Si no es así, esta parte del libro posee gran cantidad de programas ejemplo, simplemente para escucharlos, que nos permitirán realizar un primer esbozo de lo que llegaremos a ser capaces de hacer cuando finalicemos la lectura y el estudio de este volumen. Escuchar es importante, pues es la única manera de empezar a asociar los comandos a los sonidos que producen.

La historia de la música es, sin ninguna duda, mucho más amplia que la de los ordenadores. Los compositores desarrollan su propio lenguaje musical, y la manera de interpretarlo. Seguramente, todos sabemos algo sobre el sistema musical, es decir, sobre lo que es un pentagrama, cuáles son las notas musicales... De todas formas, vea-

mos el primer programa (19.2.1): este toca una nota, hace una pausa, toca otra nota, para de nuevo, y luego toca las dos juntas.

```
10 SOUND 1,478,200
20 FOR I=1 TO 3000:NEXT
30 SOUND 1,239,200
40 FOR I=1 TO 3000:NEXT
50 SOUND 1,478,300
```

■ 19.2.1.

Estas dos notas pueden parecernos ahora muy semejantes, pero no es así. La segunda posee un tono más alto que la primera, y suena más alto, aunque cuando escuchamos ambas al mismo tiempo, percibimos que armonizan perfectamente. Esto es debido a que la frecuencia de la primera es de 261.5 Hz. aproximadamente, y la segunda de 523 Hz, esto es, el doble que la primera. Lo importante es que la segunda frecuencia es exactamente el doble que la primera, decimal más, decimal menos. Ambas notas concuerdan armónicamente.

En otros tiempos no se podían medir frecuencias, pero si viajáramos hasta la Antigua Grecia, tan sólo para escuchar estas dos notas sonando al mismo tiempo, comprobaríamos que los antiguos también se percataban que existía alguna relación entre esa pareja sonora. Los compositores y músicos, con el tiempo, estuvieron de acuerdo en apuntar, acertadamente creemos, que la segunda nota era una octava más alta que la primera.

Resultó que comprobaron que si tocaban una tercera nota, que más tarde se supo poseía una frecuencia de la mitad de la primera, esta también armonizaba perfectamente con las otras dos, y no tuvieron más remedio que confesar que se trataba de la misma nota, pero una octava más baja.

Para que nosotros podamos estructurar, crear, llegar hasta la escala musical moderna, seguiremos un método bastante efectivo, consistente en ordenar las notas por frecuencias. Este sistema comienza muy bien con las tres primeras notas, como podemos observar en el programa, pero a partir de aquí, los sonidos que llegan hasta nuestros oídos no son realmente lo que andábamos buscando. Con estas notas, eso sí, se pueden producir muy buenas armonías. Armonía, como casi la propia palabra indica, es un suave y bello sonido producido por la interpretación de varias notas determinadas al mismo tiempo (19.2.2).

Pero volvamos a la escala. Si las notas no son exactamente éstas,

```

10 FOR I=1 TO 8
20 READ NOTA
30 SOUND 1,NOTA,200
40 FOR A=1 TO 800:NEXT
50 NEXT
60 DATA 478,418,372,335,304,279,257,239

```

■ 19.2.2.

¿qué podemos hacer? Los músicos de la antigüedad no se rompieron demasiado la cabeza pensando, y decidieron que la mejor forma de construir una buena escala de música era ajustar simplemente algunas de las notas de la anterior. A algunas de ellas se les redujo el tono y a otras se les aumentó.

Como buen ejemplo de lo que ellos hicieron, probemos a teclear el siguiente programa (19.2.3). Este comienza con la nota principal y termina una octava más arriba, como en la anterior escala (fijémonos en la línea DATA final), pero las notas de enmedio han sido cambiadas para obtener una escala ajustada. Realmente suena más familiar a nuestros oídos. Además, estas notas nos permiten crear mejores melodías y armonías.

```

10 FOR I=1 TO 8
20 READ NOTA
30 SOUND 1,NOTA,200
40 FOR A=1 TO 800:NEXT
50 NEXT
60 DATA 478,426,379,358,319,284,253,239

```

■ 19.2.3.

Esta escala particular se llama Mayor. La nota principal se denomina C, pero volveremos sobre ello más tarde. Esta escala sirve, más que nada, para distinguir las otras escalas. Se trata, pues, de un punto de referencia. Naturalmente, podemos emplear otras notas como escala, pero ya no será la escala Mayor.

¿Qué es lo que tiene esta escala Mayor que no tengan las demás? La respuesta es más o menos sencilla. Observemos la línea DATA donde se encuentran las notas. Si medimos la diferencia que existe

entre las notas 3 y 4 o las notas 7 y 8, observaremos que esa diferencia es mucho menor que la que pueda existir entre la 1 y 2, o la 6 y 7. A los grandes intervalos o diferencias se les conoce como tonos, y los pequeños se denominan semitonos.

Lo que convierte a una escala en una Mayor, es precisamente que sus semitonos se encuentran entre las notas 3 y 4 y 7 y 8, con el resto de los intervalos como tonos. Nosotros podemos comenzar una escala en cualquier otra nota, que llamaremos principal, pero para que se trate de una escala Mayor deberemos colocar los semitonos en los mismos lugares. Cualquiera que sea la nota principal, una escala Mayor nos permitirá crear unas melodías estupendas.

SIN EMBARGO...

La escala Mayor nos será muy útil, pero no toda la música que nosotros queremos hacer es alegre y dicharachera, rasgos habituales de esta escala; decididamente, necesitamos otras escalas que contengan notas tristes o lúgubres. Hablamos así, porque a veces la música parece tener sus propios sentimientos, que además pueden afectar a los nuestros, y en ocasiones, la única manera de describir una melodía es empleando términos que evoquen sentimientos, como los anteriores.

Desde el mismo momento en que nosotros deseamos convertir gran cantidad de sentimientos de todo tipo en música, necesitamos otras escalas que son llamadas Menores. Pero los problemas empiezan cuando descubrimos que mientras que sólo existe un tipo de escala Mayor, tenemos a nuestra disposición gran cantidad de escalas menores. Muchas de éstas sólo son de interés para los profesionales de la música, así que nos vamos a fijar preferentemente en una: la escala Natural.

En el siguiente programa (19.2.4) podemos escuchar cómo suena esta escala. Posee los semitonos entre las notas 2 y 3 y la 5 y la 6. Lo mejor que podemos hacer es escucharla y compararla con la escala mayor. Podremos apreciar que las notas suenan bastante diferentes, lo cual sería muy difícil de predecir simplemente mirando los números de las líneas DATA.

Podemos intentar teclear el siguiente programa (19.2.5) que interpreta otra escala menor, llamada la Armónica, un tanto más tenebrosa que la anterior. Sin embargo, no todas las escalas poseen ocho notas, una octava, porque no toda la música proviene de Europa, allí

```

10 FOR I=1 TO 8
20 READ NOTA
30 SOUND 1,NOTA,200
40 FOR A=1 TO 800:NEXT
50 NEXT
60 DATA 478,426,402,358,319,301,268,239

```

■ 19.2.4.

```

10 FOR I=1 TO 8
20 READ NOTA
30 SOUND 1,NOTA,200
40 FOR A=1 TO 800:NEXT
50 NEXT
60 DATA 478,426,402,358,319,301,268,239

```

■ 19.2.5.

donde fue creado este sistema de música. Podemos encontrar escalas diferentes, por ejemplo, en China. Incluso algunas canciones populares inglesas están basadas en la escala Africana, la cual fue transformada por la música de Jazz para convertirse en la escala de Blues, o The Blues Escala, como la conocen los británicos y americanos.

En el siguiente programa (19.2.6) podremos disfrutar de este tipo de escala que sólo emplea siete notas en vez de ocho, cuyas diferencias de período entre notas es bastante diferente a las de las notas de una escala Mayor o Menor. Cuando empecemos a crear nuestra propia música en el ordenador, primero habremos de decidarnos sobre cuál vamos a trabajar.

```

10 FOR I=1 TO 7
20 READ NOTA
30 SOUND 1,NOTA,200
40 FOR A=1 TO 800:NEXT
50 NEXT
60 DATA 478,402,358,338,319,268,239

```

■ 19.2.6.

Para crear una melodía en nuestro ordenador, el primer paso, aunque no absolutamente necesario pero sí conveniente, es disponer en música escrita, en un pentagrama, aquella melodía que deseamos interpretar informáticamente. El transformar música escrita en música computada es una técnica o una acción llamada trasposición, y para llevarla a cabo de buena manera es necesario poseer un excelente oído musical, para así diferenciar el sonido de diferentes escalas.

Ahora, para terminar con las escalas definitivamente, volquemos nuestra atención sobre la que reproduce el programa listado a continuación (19.2.7). Se trata de una escala Cromática, y posee todas las notas que hemos utilizado en las anteriores. Esta escala suena desde la nota principal hasta otra, también llamada principal, pero una octava más alta. Poseemos entonces una escala de trece notas, aunque se suele dejar de incluir la número trece, que es la nota principal alta; no obstante, algunos músicos si la utilizan, de manera que, antes de perder la cabeza, cambiemos el tercio y pasemos a aprender algo más de las notas que componen cada una de las escalas que hemos podido escuchar.

```
10 FOR I=1 TO 13
20 READ NOTA
30 SOUND 1,NOTA,200
40 FOR A=1 TO 800:NEXT
50 NEXT
60 DATA 478,451,426,402,379,358,338,319,301,284,268,253,239
```

■ 19.2.7.

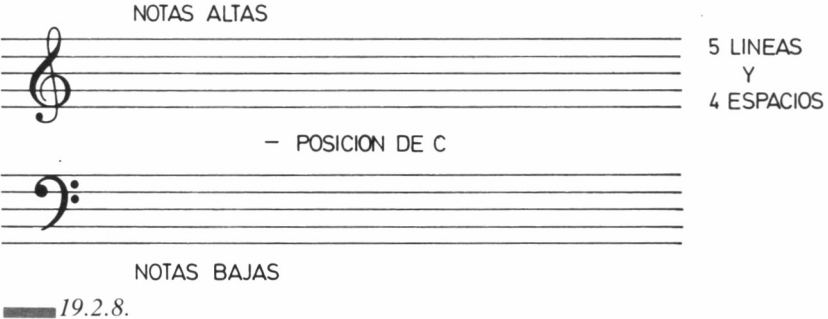
EL NOMBRE DE LA NOTA

Transformar una melodía escrita en una melodía computerizada puede resultarnos relativamente fácil; se trata de transformar notas en números que puedan ser comprendidos por el ordenador. Sin embargo, transformar una melodía escrita para otro ordenador al nuestro suele resultar mucho más difícil. A veces, el camino más fácil es convertir la música del otro ordenador en melodía escrita en pentagrama, y de aquí a la forma de sonido Amstrad.

Puede que sea una técnica un tanto «chapucera», pero no hay du-

da que es el camino más sencillo. Todo esto significa que nosotros hemos de ser capaces de leer la música escrita convencional. No tendremos que leer tan deprisa como un músico o un compositor, pero sí deberemos saber al menos cuál es cada nota.

Empecemos con ello pues. La música se escribe en un espacio limitado de cinco líneas, denominado pentagrama. El tono de una nota se indica según la situación que ocupe la nota entre dichas líneas. Usualmente, podremos ver dos pentagramas, uno debajo del otro: el superior para las notas altas, y el inferior para las bajas. Estos dos pentagramas están separados entre sí por un espacio (19.2.8).



En este espacio no existe ninguna línea, pues sería bastante difícil distinguir en cuál de los dos pentagramas nos encontramos. En este lugar se halla la nota DO, a la que conocemos en otra notación como C. Esta es la nota principal de la escala. Cuando una nota se coloca en esta posición, una pequeña línea la atraviesa, para informar que realmente pertenece a esa situación.

El sistema que se emplea para nombrar las notas es sencillamente las letras del alfabeto de la A a la G. En la figura podemos observar cómo se designan todas las notas. Sin embargo, tras estudiar este sistema, podemos pensar que en él existe una imperfección: colocando entre las barras las notas jamás sabemos dónde se encuentran los tonos y los semitonos (19.2.9).

De todos modos, esta forma de escribir música es la única que nos permite utilizar cualquier tipo de escala. Al mismo tiempo, está hecha a la medida de la escala Mayor, pues ésta puede ser escrita tranquila-



19.2.9.

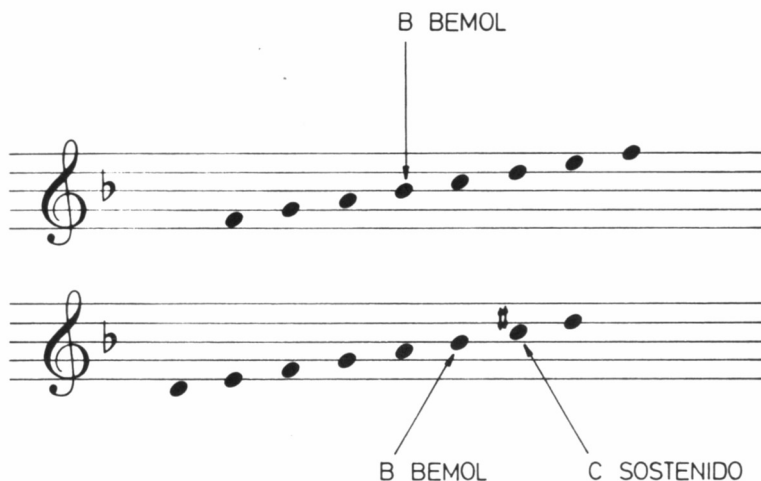
mente sobre los espacios sin ningún tipo de transformación. Para alguna otra escala, debemos utilizar signos y marcas especiales, para indicar notas que colocadas en ese lugar en la escala Mayor no sonarían del mismo modo. Estos signos se llaman sostenidos y bemoles. El primero tiene la forma del símbolo *hush* (#) y el segundo forma de B minúscula (*b*).

Estos signos se colocan al principio del pentagrama y afectan a cada nota que se encuentra en línea con ellos, es decir, tanto los sostenidos como los bemoles tan sólo se sitúan en un lugar, línea o espacio, del pentagrama. También puede ocurrir que observemos una nota con uno de estos signos junto a ella, lo cual significará que sólo ella ha de ser transformada.

El signo del sostenido (#) significa que la nota sobre la cual actúa, debe ser interpretada con un semitono mayor que el que tendría esa nota en la escala Mayor. Ahora bien, si colocamos un signo de sostenido en lo alto de el pentagrama de notas altas, significará que esta nota siempre habrá de ser un semitono más alta. Fijémonos en la figura. La nota normal es la F, y asignándole el signo de sostenido se convierte en F#.

El otro signo es el bemol, el cual cuando se coloca sobre la nota, la hace sonar con un semitono menor que el que tendría en la escala Mayor. En la siguiente figura (19.2.10) podemos observar las notas de esta escala, y gracias al siguiente programa (19.2.11) apreciaremos el sonido producido por ambas escalas. Suenan de manera muy diferente: una mayor y otra menor, aun utilizando el mismo pentagrama y la misma parte del mismo. A veces, podremos observar sobre el pentagrama notas que pertenecen a diferentes escalas, pero que han podido ser distribuidas sobre las mismas cinco líneas, para ahorrar espacio, utilizando los signos sostenido y bemol.

En el pentagrama, si queremos colocar una nota que se sale por



19.2.10.

encima o por debajo de las líneas, es decir, tenemos una nota muy alta o muy baja, podemos añadir líneas por abajo o por arriba. Estas no recorren todo el pentagrama, y tan sólo cruzan aquellas notas que así lo requieren. El sistema es por tanto parecido al que seguimos con la nota DO (la C), que se atraviesa con una barra para evitar confusiones.

Sin embargo, por mucho que nos hagamos los distraídos, las notas tienen una duración y un volumen, y estas dos características, al menos por el momento, no están siendo reflejadas sobre el pentagrama. Necesitamos, pues, algún método para hacerlo.

```

10 FOR I=1 TO 2
20 FOR A=1 TO 8
30 READ NOTA
40 SOUND I,NOTA,100
50 FOR X=1 TO 500:NEXT
60 NEXT
70 FOR X=1 TO 3000:NEXT
80 NEXT
90 DATA 358,319,284,268,239,213,190,179,426,379,368,319,284,268,239,213

```

19.2.11.

Aquí también, como la historia de la música es larga, los músicos jamás se han puesto de acuerdo en la manera de indicar estos valores, con lo cual recogemos la herencia de dos caminos a seguir, o dos formas de señalar estos valores.

La primera técnica se basa en que existe una nota que posee una duración prefijada, y el tiempo de las demás se mide con respecto a esta nota unidad, denominada Negra. En el cuadro, podemos verlas todas. Las negras son círculos o circunferencias rellenas. Si una nota posee una duración igual al doble de una negra se denomina Blanca (19.2.12).



19.2.12.

La mayor de las notas, la Redonda, es cuatro veces más larga que la Negra. Por otro lado, si la nota posee la mitad de duración que la nota base se llama Corchea. La mitad de una corchea, todos se felicitaron por asignarle tan original nombre, se llama Semicorchea; y la mitad de una semicorchea: Fusa. Por último, la mitad de una fusa (de nuevo vítores y felicitaciones) Semifusa.

Existe además un método tradicional, por el cual dividimos con barras el pentagrama. Entre dos barras de cualquier parte del pentagrama siempre transcurrirá una misma cantidad de tiempo. Con esto, lo que se consigue es adivinar si la música la queremos rápida, lenta o pausada.

Hacia el año 1823 se inventó un aparato llamado metrónomo. Este artificio utiliza un péndulo que se desplaza de un lado a otro cada cierto tiempo, como el péndulo de un reloj al que se le ha dado la vuelta. El tiempo es seleccionable, y dicho ajuste se realiza siempre en número de negras por minuto.

El metrónomo fue utilizado por importantes compositores, pero empezó a caer en desuso, al menos respecto a la misión para la que fue concebido: señalar el tiempo de la música. Sin embargo, ¿qué tal si empezamos ya a pensar en transformar la música en números que nuestro ordenador pueda interpretar?

Una negra debe durar al menos la mitad de un segundo. Esto, en un metrónomo, como dato informativo, es un número de 120. El sonido que produce nuestro ordenador posee un tiempo por defecto de una quinta parte de segundo para cada nota, que en un metrónomo corresponde a 300. Quizás esto sea demasiado rápido. Con el siguiente programa (19.2.13) podremos escuchar dieciséis notas con un tiempo de duración de medio segundo para cada una.

```
10 FOR I=1 TO 16
20 READ NOTA
30 SOUND 1,NOTA,50
40 NEXT
50 DATA 478,426,379,478
60 DATA 358,478,319,478
70 DATA 284,478,319,358
80 DATA 379,426,478,478
```

■ 19.2.13.

Si suprimimos la última parte de la instrucción SOUND que se encuentra en la línea 30, oiremos los sonidos con el tiempo por defecto del ordenador. Estaremos escuchando una forma de música muy famosa, llamada Allegro.

MÚSICA ELÁSTICA Y FLEXIBLE

Si hay algo que los músicos, compositores, canta-autores o cualquier persona relacionada profesionalmente con la música, pueda odiar con más fuerza, es precisamente estar atado a unas normas, unas reglas, unas medidas de algo, tanto si es el tiempo como si es el tono. Por lo tanto, podemos asignarle al tiempo de una negra un poquito de flexibilidad, que siempre es bien agradecida, y no depender de una sola medida de tiempo para cada nota.

Para aquellos que no estemos ligados a la música como profesionales, esto puede resultar un fastidio al pasar de música escrita a números para el comando SOUND ya dijimos que el sonido es diferente, y ésta es una de sus diferencias. Nosotros podemos, con un poco de práctica, aprender pronto a transferir música escrita a códigos, sin embargo, de si el tiempo asignado a cada nota es correcto o no, tan sólo nos podrá informar nuestro propio oído.

Con experiencia, mirando un programa podremos saber de qué notas se tratan, pero para controlar el tiempo necesitaremos más bien ejercitar nuestro propio aparato auditivo. En todo caso, lo más aconsejable cuando creemos música en nuestro ordenador, es construirla con un poquito de flexibilidad.

Una técnica encaminada a este fin es emplear notas punteadas. Imaginemos que necesitamos una nota que bien mirada parece una negra, pero en realidad no deseamos que posea la longitud de tiempo estándar que acredita. Pues bien, simplemente colocamos un punto junto a ella. Esto significa que es un medio más larga de lo que ella misma mide (a ver, leamos esta última frase de nuevo). Por ejemplo, si le asignamos a una negra un tiempo de uno, esta misma nota acompañada por un pequeño puntito poseerá una duración de uno y medio (19.2.14).

The image shows a grand staff with two staves. The top staff has a treble clef and the bottom staff has a bass clef. Four musical notes are placed on the staves, each with a small dot next to it to indicate it is a dotted note. From left to right, they are: a dotted quarter note on the first line of the bass staff, a dotted half note on the second line of the bass staff, a dotted whole note on the second space of the bass staff, and a dotted eighth note on the first space of the treble staff.

NEGRA	BLANCA	REDONDA	CORCHEA
PUNTEADA	PUNTEADA	PUNTEADA	PUNTEADA

■ 19.2.14.

Si el tiempo estándar de una blanca es de dos, la blanca punteada durará tres y si la estándar redonda posee una duración de cuatro, con punto su longitud será de seis. La técnica de asignar un punto a una nota se extiende análogamente y por igual a las demás pequeñas notas.

Otra forma de indicar la duración de una nota diferente a lo normal, es por medio de cuerdas que ligan las notas. Si dibujamos una línea que une dos notas exactamente iguales, esto significa que estas notas han de ser interpretadas como una sola, pero con la duración de una más la otra. Todo ello podemos observarlo en la siguiente figura.

Es también muy importante en música la duración de las pausas dentro de una melodía. En el siguiente esquema podemos apreciar la forma de los silencios dentro de una composición musical escrita en un pentagrama. También podemos emplear puntos y lazos con estos signos, como si de notas se trataran, para flexibilizar el tiempo de cada uno de ellos. Cada uno posee debajo el nombre de una figura musical, y el tiempo que representan es precisamente el estándar de esa figura (19.2.15).



19.2.15.

Ya hace algún tiempo, los músicos empezaron a designar o describir una melodía mediante unas palabras de la jerga musical como allegro o andante. De esta manera, se puede prever cómo de largos van a ser los silencios incluidos dentro de la composición.

La guía principal para un músico de cuál ha de ser la duración de una negra, se llama el Tempo. Esto, retorciéndonos a pensar, significa tiempo en italiano, y las palabras que se utilizan para designarlo son también de procedencia italiana, por lo que no nos costará demasiado descifrar su significado. En el siguiente esquema podemos observar todas estas palabras, que señalan cuál va a ser el tiempo predominante de la melodía. Los números son aproximados, porque los músicos no se ciñen herméticamente a ellos.

Término	Metronomo	Duración en Amstrad
Largo	60/70	100/85
Lento	60/70	100/85
Grave	60/70	100/85
Andante	120/75	50/80
Alegreto	150/120	40/50
Alegro	300/150	20/40
Vivace	300/150	20/40
Con brío	300/150	20/40
Presto	600/300	10/20
Prestísimo	1.200/400	5/15

Las palabras de la lista, y sus medidas pueden llegar a ser para nosotros un buen punto de partida y de referencia, y las podemos utilizar como guía. Esto es, si es nuestra intención crear una música alegre, podemos mirar la tabla del Tempo, fijar nuestra atención en el Allegro, y utilizar un tiempo para cada nota de 20/40, como se indica. En el Apéndice I encontraremos el significado de otros muchos términos italianos de uso frecuente en la música.

Sin embargo, seguro que todos hemos notado que aquí todavía no se ha hablado absolutamente nada de volumen, la amplitud del sonido. Una vez más, a los profesionales de la música les gusta disponer de un poco de libertad o flexibilidad en esta parte de la escritura musical.

La música, como todos sabemos, puede ser interpretada por una gran cantidad y variedad de instrumentos, algunos de los cuales llegan a alcanzar un mayor volumen que otros. El método tradicional para su medida es emplear la letra *f*, que viene del italiano «forte», cuyo significado es aumento de volumen, y la letra *p*, del italiano «piano», que significa disminución de volumen.

La nota más fuerte de un instrumento musical puede ser denotada mediante *fff*, y la más débil con *ppp*. Cuando los compositores deseaban que la música subiera en volumen, todos rumoreaban por la sala que se iba a producir un «crescendo», lo cual se indicaba en el pentagrama. Si el compositor y la sala empezaban a sentir cierto malestar en los oídos, se debilitaba el sonido mediante el «diminuendo». Ambas marcas las podemos observar en el pentagrama.

Pero todavía nos hemos dejado algo en el tintero. Existe una característica de la música que es casi indescriptible con palabras o signos: el ritmo. Estamos escuchando un ritmo cuando las notas son

interpretadas por grupos, y ciertas notas se enfatizan de alguna manera.

Una nota puede señalarse tocándola un poco más alta, o más larga, que las demás. En el pentagrama, observamos a veces una cantidad indeterminada de notas agrupadas entre dos líneas. Llamaremos compás al espacio que existe entre dos líneas. Si tenemos cuatro negras en un compás, podríamos cambiarlas por una redonda, o dos blancas, o una blanca punteada y una negra, siempre sin alterar el tiempo absoluto, que era cuatro.

Para conseguir un ritmo, podemos enfatizar una de las notas de cada compás, frecuentemente la primera o la segunda. Sin embargo, esto puede llegar a ser monótono, aunque la enfatización se puede reservar a cualquier otra nota, incluso aumentar unas de volumen y bajar el de otras.

Esto es todo en lo referente al nombre y la situación de las notas. Es absolutamente necesario aprenderlo si deseamos crear nuestras propias melodías en un futuro muy próximo, es decir, en los próximos capítulos. Pero no olvidemos que éstos son sólo los pasos iniciales, no hemos hecho más que tomar la batuta. Pronto aprenderemos a moverla para que nuestra orquesta ejecute una gran variedad de increíbles melodías y efectos de sonido.

COMPONRIENDO MÚSICA

N

osotros jamás podremos producir con nuestro ordenador el sonido que pueda salir de una flauta dulce, un piano, un órgano o una guitarra, pero por otra parte, seremos capaces de generar sonidos que ninguno de estos instrumentos, excepto un sintetizador, sería capaz de producir.

Aun así, la música que pueda sonar en nuestro Amstrad puede resultarnos, en la gran mayoría de las ocasiones, agradable para los oídos, sobre todo si conectamos éste a un amplificador, casete, o cadena de alta fidelidad, lo cual nos será posible ojeando el apéndice final.

A partir de ahora, aprenderemos cómo se produce el sonido en nuestro ordenador, tanto el musical, como aquel que podemos escuchar en cualquier juego de marcianos. El chip de sonido produce un sonido por el altavoz cuando llega hasta él una señal eléctrica.

Si el voltaje sube, el aire se presiona fuera del altavoz, y de manera análoga, si el voltaje baja, el aire se presiona hacia adentro. Si el émbolo que empuja y se contrae realiza el movimiento de expulsión y toma de aire, alternativamente, unas cien veces por segundo el resultado es de una onda de sonido cuya frecuencia es de 100 Hz.

Cuanto mayor sea la onda eléctrica, mayor será la onda de sonido

que producirá el émbolo del altavoz, es decir, la hará lo más parecida a la onda eléctrica que recibe. Las señales que recibe el altavoz son enormemente rápidas, y su nombre técnico exacto es «clock pulses».

EL PRIMER SONIDO QUE ESCUCHAMOS EN NUESTRO AMSTRAD

Se trata del sonido más simple que podemos producir en BASIC, empleado frecuentemente por el ordenador para llamar nuestra atención sobre algo. La frecuencia de este sonido es de aproximadamente 1.389 kHz, y para obtenerlo nos valemos de la instrucción BASIC PRINT CHR\$(7). Esta envía una señal que produce la ejecución de este sonido. Así pues, simplemente tecleando

```
PRINT CHR$(7)
```

obtendremos el primer sonido, que con casi total seguridad, hemos escuchado todos en nuestro Amstrad. Quizá no encontremos ahora ninguna utilidad a este pitido, pero frecuentemente se utiliza en algunos programas para llamar la atención del usuario sobre algo, avisar que una acción ha sido realizada o ha de ser llevada a cabo etc. Para este fin, la mayor parte de las veces, no merece la pena que nos compliquemos la existencia utilizando la instrucción SOUND. Gracias a PRINT CHR\$(7) disponemos de un pitido siempre listo, sin necesidad de preocuparnos, de frecuencias, volúmenes, etc...

AMPLIANDO FRONTERAS

El sistema de sonido de los ordenadores de la serie CPC está montado alrededor de una unidad llamada Generador Programable de Sonido (PSG), el chip de sonido para los amigos, aunque su verdadero nombre sea realmente AY-3-8912, y su lugar de nacimiento los Estados Unidos de América.

Lo único que puede producir este chip son ondas, las cuales serán traducidas en sonido por el altavoz del ordenador, aunque no las reproducirá perfectamente, sino que las transformará un poco. Si conectamos nuestro CPC a una cadena de alta fidelidad, escucharemos un sonido muy distinto, no hay duda, aunque tampoco podrá reproducir ningún baffle con perfección, por potente que sea, las ondas eléctricas que llegan del chip de sonido.

El ordenador genera tres canales de sonido, dos de los cuales pueden ser empleados para el sonido estéreo, sumándose el del tercer canal a los otros dos. Existe también un generador de ruido, que puede enviarse por alguno, algunos o todos los canales.

Seguro que hemos comprobado ya, o lo comprobaremos, que cuando un programa de sonido se ejecuta, el mensaje READY aparece antes de que la música haya dejado de tocar. Esto es debido a que todas las instrucciones de sonido se introducen en la memoria, utilizando un sistema llamado FIFO, que proviene del inglés *First in - first out* (Primero en entrar - primero en salir).

Esta memoria puede guardar hasta cinco instrucciones de sonido por cada canal, con lo cual si construimos un programa que sólo posea cinco notas, el proceso acaba rápidamente. De esta manera, mientras que las notas están sonando, el ordenador puede dedicarse a realizar otras operaciones totalmente diferentes a esta. A priori esto puede parecer interesante, pero no lo es tanto si queremos que el sonido suene acompasado con una imagen de la pantalla.

SOUND es una orden, instrucción o palabra clave que puede ser utilizada de una manera más o menos complicada, según la complejidad del sonido a generar. La forma más simple de emplear la instrucción SOUND es haciendo uso tan sólo de dos parámetros.

El primero de éstos es el número de canal. Mientras estemos trabajando con la forma más simple de la instrucción, los números de canal serán 1, 2 y 4. También podemos designarlos como A, B y C, pero jamás encontraremos un canal G, cuyo número sería el seis, sencillamente porque no existe. Aun así, podemos usar el canal cero, cuyo resultado es simplemente la no producción de sonido alguno.

Por lo tanto, disponemos de tres canales totalmente independientes. Esto trae como consecuencia que podemos tocar tres melodías totalmente diferentes, simultáneamente, por cada uno de ellos. Es decir, los canales pueden actuar por separado o al mismo tiempo.

Pero, ¿por qué existe un canal 4, y no el 3? Es muy sencillo. Si deseamos que actúen tan sólo los canales A y C habremos de indicar el número $1+4=5$. Pero para que el sonido tan sólo se produzca por los canales A y B sumaremos $1+2=3$, con lo cual si existiera un canal 3, sería éste el que sonaría, y no los dos primeros. Para que los tres canales funcionen simultáneamente, utilizaremos el número $1+2+4=7$ como canal.

El segundo número o parámetro de la instrucción SOUND es el número de tono. Quizá no sea muy acertado llamarlo así, pues un número alto provoca un tono bajo y viceversa. El manual se refiere a él como el número del período del tono, así pues nos quedaremos con

esta denominación. El valor que puede utilizarse oscila entre 0 y 4095. Si empleamos cero no habrá sonido, lo cual puede resultar muy útil para crear un silencio, que necesitaremos en ocasiones. Debemos dejar claro que el número de período de tono determina la nota que será tocada.

Los números «ultras», es decir, los muy altos y los muy bajos, apenas tienen ninguna utilidad. Algunos tonos bastante bajos no pueden ser reproducidos por el diminuto altavoz que lleva incorporado nuestro Amstrad, aunque la cosa puede cambiar si poseemos una cadena de alta fidelidad. De todas formas, es recomendable no utilizar valores por encima del 1000. Realmente se puede obtener sonidos con valores más altos, pero no son precisamente la clase de sonido que estamos buscando o que esperamos escuchar. Con unos cascos estéreo-fónicos no podremos escuchar todos estos sonidos, a menos que sean muy sensibles. Aquellos números por debajo de 16 son casi inaudibles y muy difíciles de reproducir por el altavoz de el ordenador.

Existe una fórmula:

$$\text{PERIODO DEL TONO} = 125000/\text{FRECUENCIA}$$

con la cual podemos calcular la frecuencia de cada sonido. Por un período de tono de 1000, la frecuencia es de 125 Hz. Si empleamos un período de 995, cinco por debajo, la frecuencia es de 125.63 Hz, lo cual es una leve diferencia. Pero tomemos ahora un período de 20, cuya frecuencia sería de 6250 Hz. Tomando un período menor en cinco puntos, 15, obtenemos una frecuencia de 8333 Hz. Esta no es la misma diferencia que obtuvimos anteriormente, sino exageradamente mayor. Por tanto, en tonos bajos la diferencia de frecuencia es mínima, mientras que es enorme en los altos (19.3.1).

```
10 FOR I=1000 TO 10 STEP -5
20 SOUND 1,N
30 NEXT
```

■ 19.3.1.

LA DURACION DE LA NOTA

Si decidimos emplear sólo un canal y un único período de tono, cada nota sonará una quinta parte de segundo. Si está de acuerdo con esta duración, tan sólo habremos de especificar en la instrucción SOUND dos números. Pero usando un tercer dígito, separado en la orden por una coma, podremos especificar el tiempo de duración de una nota, valor que oscila entre 1 y 500. Podemos utilizar el cero, y valores negativos superiores a 32767, aunque por el momento sólo nos quedaremos con los primeros 500 valores.

La unidad de tiempo es la centésima de segundo, por lo cual, el número 400 significará $400/100=4$ segundos de duración. El número 1 producirá una duración de tan sólo una centésima de segundo. El siguiente programa, utilizando un bucle que va aumentando la frecuencia de una nota, utiliza una duración de una centésima de segundo para cada una. El efecto es el de una sirena de policía sonando en diez ocasiones. Para aumentar el tiempo de sonido tan sólo habría que cambiar el tercer parámetro de la sentencia SOUND, pero el efecto perdería muchísimo. No obstante, ¡animémonos a experimentar! (19.3.2).

```
10 TONO=125000
20 FOR I=1 TO 5
30 FOR A=1000 TO 2000 STEP 10
40 SOUND 1,TONO/A,1
50 NEXT:NEXT
```

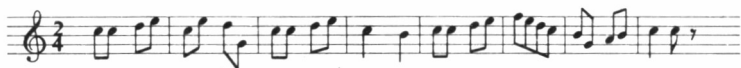
19.3.2.

Usamos la variable PERIODO, para posteriormente obtener la frecuencia por medio de la división por el número de tono. Se utiliza la variable y no directamente el número 125000 para acelerar la ejecución del programa. Podemos probar a cambiar el número de canal, intercambiándolo por 3, 5 y 7, para comprobar los efectos producidos.

CREANDO MÚSICA

Es tiempo de aprender las posibilidades musicales de nuestro Amstrad. Empezaremos con una simple melodía, partiendo del pen-

tagrama donde se encuentra la primera parte de ésta. Todo lo que necesitamos hacer es transformar esta pieza de música escrita en un programa de ordenador. La manera más usual de llevar a cabo esto es por medio de una línea DATA (19.3.3).



C' C' D'E' C'E' D'G C'C' D'E' C' B C'C' D'E' F'E'D'C' B'G A'B C' C'

19.3.3.

Como podemos observar, la totalidad de las notas son negras. Tenemos 31 notas, por lo cual necesitaremos un bucle FOR... NEXT que cuente de 1 a 31. El período de tono será leído de una línea DATA que ya pondremos más tarde. Por el momento, el programa queda de la siguiente manera:

```
10 FOR I=1 TO 31:READ A
20 SOUND 1,A:NEXT
```

Una vez hecho esto, necesitamos una línea DATA para cada barra, y como cada una de ellas posee cuatro notas, cada línea DATA tendrá cuatro dígitos. Esto es aconsejable para facilitar la corrección, en caso que al ejecutar el programa encontremos una nota que no nos suene demasiado bien. Cambiar una nota defectuosa o errónea es fácil siguiendo este sistema. Para el tiempo, vamos a utilizar el que adopta el ordenador por defecto, es decir una quinta parte de segundo.

Si hemos leído frecuentemente música de un pentagrama, no nos será difícil cambiar las notas del papel, en números que pueda entender el ordenador. Si no, la mejor manera de llevarlo a cabo es cumpliendo dos fases. En primer lugar, debemos escribir el nombre de las notas debajo de cada una de ellas.

La melodía comienza con C' pues usamos la comilla para denominar aquellas notas que están una octava superior a la normal. C es el DO internacional, C' es Do alto, o una octava más alto. Ahora, des-

pués de colocar cada uno de los nombres, debemos mirar el cuadro de transformación y observar qué período corresponde a cada una, para introducirlo en cada una de las líneas DATA. Este es el cuadro de transformación:

CUADRO DE TRANSFORMACION

NOTA		FRECUENCIA	PERIODO	
C	DO	1046.502	119	
C#	DO#	1108.731	113	
D	RE	1174.659	106	
D#	RE#	1244.508	100	
E	MI	1318.510	95	
F	FA	1396.913	89	
F#	FA#	1479.978	84	Octava 2
G	SOL	1567.982	80	
G#	SOL#	1661.219	75	
A	LA	1760.000	71	
A#	LA#	1864.655	67	
B	SI	1975.533	63	

NOTA		FRECUENCIA	PERIODO	
C	DO	2093.004	60	
C#	DO#	2217.461	56	
D	RE	2349.318	53	
D#	RE#	2489.016	50	
E	MI	2637.021	47	
F	FA	2793.826	45	Octava 3
F#	FA#	2959.955	42	
G	SOL	3135.963	40	
G#	SOL#	3322.438	38	
A	LA	3520.000	36	
A#	LA#	3729.310	34	
B	SI	3951.066	32	

NOTA		FRECUENCIA	PERIODO	
C	DO	4186.009	30	
C#	DO#	4434.922	28	
D	RE	4698.636	27	
D#	RE#	4978.032	25	
E	MI	5274.041	24	
F	FA	5587.652	22	Octava 4
F#	FA#	5919.911	21	
G	SOL	6271.927	20	
G#	SOL#	6644.875	19	
A	LA	7040.000	18	
A#	LA#	7458.621	17	
B	SI	7902.133	16	

NOTA		FRECUENCIA	PERIODO	
C	DO	32.703	3822	
C#	DO#	34.648	3608	
D	RE	36.708	3405	
D#	RE#	38.891	3214	
E	MI	41.203	3034	
F	FA	43.654	2863	Octava 3
F#	FA#	46.249	2703	
G	SOL	48.999	2551	
G#	SOL#	51.913	2408	
A	LA	55.000	2273	
A#	LA#	58.270	2145	
B	SI	61.735	2025	

NOTA		FRECUENCIA	PERIODO	
C	DO	65.406	1911	
C#	DO#	69.296	1804	
D	RE	73.416	1703	
D#	RE#	77.782	1607	
E	MI	82.407	1517	
F	FA	87.307	1432	Octava 2

NOTA		FRECUENCIA	PERIODO
F#	FA#	92.499	1351
G	SOL	97.999	1276
G#	SOL#	103.826	1204
A	LA	110.000	1136
A#	LA#	116.541	1073
B	SI	123.471	1012

NOTA		FRECUENCIA	PERIODO
C	DO	130.813	956
C#	DO#	138.591	902
D	RE	146.832	851
D#	RE#	155.564	804
E	MI	164.814	758
F	FA	174.614	716
F#	FA#	184.997	676
G	SOL	195.998	638
G#	SOL#	207.652	602
A	LA	220.000	568
A#	LA#	233.082	536
B	SI	246.942	506

Octava -1

NOTA		FRECUENCIA	PERIODO
C	DO	261.626	478
C#	DO#	277.183	451
D	RE	293.665	426
D#	RE#	311.127	402
E	MI	329.628	379
F	FA	349.228	358
F#	FA#	369.994	338
G	SOL	391.995	319
G#	SOL#	415.305	301
A	LA	440.000	284
A#	LA#	466.164	268
B	SI	493.883	253

Octava 0

NOTA		FRECUENCIA	PERIODO	
C	DO	523.251	239	
C#	DO#	554.365	225	
D	RE	587.330	213	
D#	RE#	622.254	201	
E	MI	659.255	190	
F	FA	698.457	179	Octava 1
F#	FA#	739.989	169	
G	SOL	783.991	159	
G#	SOL#	830.609	150	
A	LA	880.000	142	
A#	LA#	932.328	134	
B	SI	987.767	127	

Con cuatro datos por línea necesitaremos ocho líneas. La última línea finaliza con un cero, pero no será en realidad leída por el programa, pues este sólo tomara 31 valores, y no 32. El programa quedará de esta manera (19.3.4):

```

10 FOR I=1 TO 31:READ NOTA
20 SOUND 1,NOTA:NEXT
30 DATA 239,239,213,190
40 DATA 239,190,213,319
50 DATA 239,239,213,190
60 DATA 239,239,253,253
70 DATA 239,239,213,190
80 DATA 179,190,213,239
90 DATA 253,319,284,253
100 DATA 239,239,239,0

```

■ 19.3.4.

```

10 FOR I=1 TO 31:READ NOTA
20 SOUND 1,NOTA
30 SOUND 1,0,2
40 NEXT
50 DATA 239,239,213,190
60 DATA 239,190,213,319
70 DATA 239,239,213,190
80 DATA 239,239,253,253
90 DATA 239,239,213,190
100 DATA 179,190,213,239
110 DATA 253,319,284,253
120 DATA 239,239,239,0

```

■ 19.3.5.

Su nombre, técnica y musicalmente hablando, es Staccato. El problema que causa es que produce la separación de notas que deberían ser tocadas en conjunto, o «de una tirada». Sin embargo, si nos encontramos con un problema como ese, podemos solucionarlo fácilmente de la forma que indica el siguiente programa (19.3.6):

```

10 FOR I=1 TO 31:READ NOTA
20 SOUND 1,NOTA
30 IF (I<13)OR(I>16 AND I<29) OR (I>29) THEN SOUND 1,0,2
40 NEXT
50 DATA 239,239,213,190
60 DATA 239,190,213,319
70 DATA 239,239,213,190
80 DATA 239,239,253,253
90 DATA 239,239,213,190
100 DATA 179,190,213,239
110 DATA 253,319,284,253
120 DATA 239,239,239,0

```

■ 19.3.6.

Actúa en aquellos lugares donde necesitamos el staccato, que son en los que I posee valores de 13, 14, 15, 16 y 29. Este sistema puede ser de gran utilidad para pequeñas melodías, pues sólo se coloca un silencio en aquellos lugares donde sea requerido.

Pero veamos otra melodía ejemplo, comenzando por el pentagrama con las notas. Podremos observar que esta melodía posee, además de negras, blancas y corcheas, por lo que habrá que andar con más cuidado en lo que se respecta al tiempo de las notas (19.3.7).



■ 19.3.7.

El problema consiste en hallar el tiempo relativo correctamente y entonces calcular el tiempo absoluto. Para empezar, de todas formas, coloquemos los nombres de las notas debajo de cada una de ellas. Casi todas ellas están una octava por encima de la escala normal por lo cual haremos uso de la comilla para designarlas. Observemos que hay dos que sobrepasan el pentagrama por encima, se trata de E'' y C'', dos octavas por encima de la escala habitual (19.3.8).

```

10 NOTA=1
20 WHILE NOTA<>0
30 READ NOTA, TIME
40 SOUND 1,NOTA,8*TIME
50 SOUND 1,0,2
60 WEND
70 DATA 190,4,239,3,190,1
80 DATA 179,4,213,3,179,1
90 DATA 190,4,239,3,190,1
100 DATA 213,3,253,1,319,4
110 DATA 190,4,239,3,190,1
120 DATA 179,4,213,3,179,1
130 DATA 190,3,239,1,213,3,253,1
140 DATA 239,8
150 DATA 239,3,190,1,159,3,119,1
160 DATA 142,4,159,3,179,1
170 DATA 190,4,239,3,190,1
180 DATA 213,3,253,1,319,4
190 DATA 239,3,190,1,159,3,119,1
200 DATA 142,4,159,3,179,1
210 DATA 190,3,239,1,213,3,253,1

```

■/9.3.8.

El tiempo debe ser incluido, junto con cada nota, en las líneas DATA. Para cada nota leeremos TONO, que es el período, y TIME, que es su tiempo de duración. Utilizamos una duración de 4 para una negra, 8 para una blanca, 3 para una corchea, y 1 para una semicorchea.

Pero ahora debemos calcular el tiempo absoluto, para lo cual multiplicamos TIME por 8. Si acortamos este número conseguiremos que la melodía transcurra más deprisa, y por contra, valores más grandes producirán la ralentización de la misma. Esto es mucho más fácil que andar cambiando los valores de tiempo de las líneas DATA, y es una técnica que emplearemos siempre que nos sea posible.

UN TRUCO NUNCA VIENE MAL

La peor parte, sin duda, de convertir música escrita en pentagrama a números que pueda entender el ordenador es buscar en el cuadro de transformación el período que corresponde a cada nota. Ade-

más, la mayoría de las veces, por culpa de la rutina o el cansancio de los ojos, se suelen cometer errores que pueden ser más o menos fáciles de subsanar.

La frecuencia de cada nota de la escala puede ser calculada, y este valor es muy aproximado al que nosotros buscamos para asignar a cada nota. Sólo un oído de oro podría detectar la diferencia. Todo se resume en que si la frecuencia y el período de tono pueden ser calculados, ¿por qué no los averigua el ordenador por nosotros? (19.3.9).

```
10 CLS:A=125000:ZONE 10
20 PRINT TAB(15)"BUSCADOR DE NOTAS"
30 LOCATE 1,3:PRINT"ESCRIBE CADENA MUSICAL"
40 PRINT"USA 0 PARA OCTAVA (1 A 8)":PRINT"C COMIENZA EN LA OCTAVA CUARTA"
50 PRINT"DE A HASTA G PARA LAS NOTAS. USA # PARA BEMOL"
60 PRINT" Y ! PARA SOSTENIDO. EJEMPLO:"
70 PRINT"01BCD#CGA"
80 INPUT MUSICA$:B=LEN(MUSICA$)
90 IF B>80 THEN PRINT"DEMASIADO LARGA, MAXIMO 80":B=80
100 CLS:PRINT TAB(15)"TUS NOTAS..."
110 LOCATE 1,2:A2=4
120 FOR I=1 TO B:C$=MID$(MUSICA$,I,1)
130 IF C$="0" THEN GOSUB 1000:GOTO 170
140 IF ASC(C$)>64 AND ASC(C$)<72 THEN GOSUB 2000:GOTO 170
150 PRINT"ERROR EN LA CADENA MUSICAL. REVISALA"
160 PRINT MUSIC$:END
170 NEXT:END
1000 I=I+1:C$=MID$(MUSICA$,I,1)
1010 A2=VAL(C$):IF A2<1 OR A2>8 THEN GOTO 150
1020 A2=A2-4
1030 RETURN
2000 IF C$="C" THEN D=1
2010 IF C$="D" THEN D=3
2020 IF C$="E" THEN D=5
2030 IF C$="F" THEN D=6
2040 IF C$="G" THEN D=8
2050 IF C$="A" THEN D=10
2060 IF C$="B" THEN D=12
2070 E$=MID$(MUSICA$,I+1,1)
2080 IF E$="#" THEN D=D+1:I=I+1
2090 IF E$="!" THEN D=D-1:I=I-1
2100 GOSUB 3000
2110 RETURN
3000 G=440*(2^(A2+(D-10)/12))
3010 H=ROUND(A/G)
3020 SOUND 1,H,10
3030 PRINT H,
3040 FOR Z=1 TO 400:NEXT
3050 RETURN
```

19.3.9.

Sería bastante fácil que tecléáramos el nombre de cada nota y el ordenador se encargará de generar un número de período. Esto no es posible, porque para nombrar las notas sólo empleamos letras de la A a la G, y también necesitaríamos especificar la octava en la cual está situada la nota.

Si observamos la tabla de transformación, se asigna un 0 a la octava en la escala normal, un -1 para la inferior y un 1 para la superior. En nuestro programa utilizaremos 1 para la octava más baja, 4 para la normal, y 8 para la alta. Tendremos que introducir cadenas como ésta: 03B04CDEF#B y cada nota quedará convertida en su período de tono. El programa leerá cada letra de la cadena, y el ordenador decidirá si se trata de una nota o un comando de identificación de octava.

El programa comienza borrando la pantalla y asignando a GBA el número 125000, además de estructurar la pantalla colocando una zona de diez caracteres entre cada mensaje que sea representado. Esto se consigue mediante el uso del comando ZONE, lo cual nos permitirá tener cuatro períodos de tono por cada línea DATA.

La longitud de la cadena se ha limitado a ochenta para evitar cálculos interminables, además de otra causa que luego explicaremos. La cadena se introduce en la línea 90 y se mide su longitud en la 100. A partir de la línea 110 se limpia la pantalla, analizan e imprimen los datos.

ARMONÍA Y ESTÉREO

La armonía en música supone que varias notas adecuadas se tocan al mismo tiempo. Una secuencia de estas notas se llama acorde. Pero ¿qué quiere decir que sean adecuadas? Una vez más no existe una respuesta precisa a esta pregunta, pero se trata de las notas que suenan bien para nuestro oído cuando son interpretadas al mismo tiempo.

A pesar que la historia de la música ha sido larga, los compositores han estado inventando nuevos acordes, nuevas armonías, hasta este siglo. Si queremos una armonía que suene alegre, al correspondiente acorde lo llamaremos Mayor. Una vez más, la palabra Mayor se emplea para denotar alegría, y como es normal esperar, las notas de un acorde mayor son de esta manera, y tomadas de una escala Mayor. Veamos en el siguiente programa cómo funcionan estos acordes (19.4.1).

```
10 ALV=50
20 FOR I=1 TO 7
30 READ A$,A1,A2,A3
40 PRINT A$;" ACORDES MAYORES"
50 SOUND 1,A1,ALV
60 GOSUB 1000
70 SOUND 1,A1,ALV:SOUND 2,A2,ALV
80 GOSUB 1000
90 SOUND 1,A1,ALV:SOUND 4,A3,ALV
100 GOSUB 1000
```

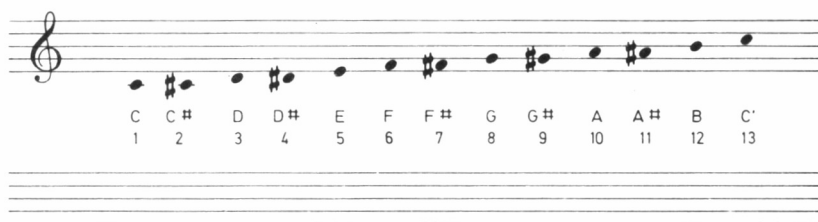
```

110 SOUND 1,A1,ALV:SOUND 2,A2,ALV:SOUND 4,A3,ALV
120 GOSUB 1000
130 PRINT
140 NEXT
150 END
160 DATA C,478,379,319
170 DATA D,426,338,284
180 DATA E,379,301,253
190 DATA F,358,284,239
200 DATA G,319,253,213
210 DATA A,284,225,190
220 DATA B,253,201,169
1000 COM=SEC
1010 WHILE SEC < COM + 300
1020 WEND
1030 RETURN

```

19.4.1.

Mirando este tipo de programas nos podemos dar una idea de cómo son producidos los acordes. Primero hemos de elegir una nota, llamada principal. Este es el centro neurálgico del acorde. Veamos un pentagrama con una escala musical numerada. Las notas principales del acorde son la 1, la 5, y la 8. La primera nota es la principal, la segunda se llama Tercera, y la octava Quinta (19.4.2).



19.4.2.

Si seleccionamos la nota F como nota principal, la llamaremos 1. Ahora contamos notas hasta la 5, que es la A, y será la Tercera. Contamos hasta ocho, C', que es la Quinta. Las notas F, A y C' son las tres notas de un acorde Mayor. Podemos hacer sonar tan solo a F

y A o a F y C', o un acorde con las tres notas. Esto es en realidad lo que lleva a cabo el programa.

En él, el nombre de la nota principal se lee de una línea DATA. Esta nota es presentada en la pantalla y emitida por el canal 1. Más tarde, sonarán la principal y la Tercera, la principal y la Quinta, y finalmente, las tres al tiempo. En todos los casos excepto en el primero, sonará un acorde. Podremos escuchar como suenan estos acordes tomando todas las notas de la escala Mayor.

Podemos también utilizar otro tipo de notas y cambiar los acordes, buscando efectos interesantes. En el siguiente programa podemos escuchar un acorde menor, y observaremos el profundo cambio existente entre éste y el anterior, pasando de la alegría de uno, a la tristeza del otro (29.4.3).

```
10 T=50
20 PRINT"ACORDE MAYOR"
30 SOUND 1,478,T:SOUND 2,379,T:SOUND 4,319,T
40 GOSUB 1000
50 PRINT"ACORDE MENOR"
60 SOUND 1,478,T:SOUND 2,402,T:SOUND 4,319,T
70 GOSUB 1000
80 GOTO 20
1000 COM=SEC
1010 WHILE SEC < COM + 300
1020 WEND
1030 RETURN
```

■ 19.4.3.

Ahora intentemos otro tipo de experimento. Escucharemos un acorde mayor seguido de un acorde que va aumentando poco a poco. Este último posee una Quinta agudizada, es decir, tiene un tono aumentado. El efecto lo comprobaremos en el siguiente programa (19.4.4).

Este otro programa (19.4.5.) nos enseñará una nueva técnica o «chacucilla» que puede llevarse a cabo con los acordes. Esta vez modificamos los tonos de la Tercera y de la Quinta, produciendo un efecto más triste.

```

10 T=50
20 PRINT"ACORDE MAYOR"
30 SOUND 1,478,T:SOUND 2,379,T:SOUND 4,319,T
40 GOSUB 100
50 PRINT"ACORDE AUMENTADO"
60 SOUND 1,478,T:SOUND 2,379,T:SOUND 4,301,T
70 GOSUB 100
80 GOTO 20
90 INICIO=SEG
100 WHILE SEG<INICIO+300:WEND
110 RETURN

```

■ 19.4.4.

```

10 T=50
20 PRINT"ACORDE MAYOR"
30 SOUND 1,478,T:SOUND 2,379,T:SOUND 4,319,T
40 GOSUB 1000
50 PRINT" ACORDE DISMINUIDO"
60 SOUND 1,478,T:SOUND 2,402,T:SOUND 4,338,T
70 GOSUB 1000
80 GOTO 20
1000 INICIO=SEG
1010 WHILE SEG<INICIO+300
1020 WEND
1030 RETURN

```

■ 19.4.5.

Todas estas armonías han sido interpretadas por los tres canales de nuestro ordenador, por lo cual hemos tenido que emplear los números 1, 2 y 4 en las instrucciones de sonido. El siguiente paso es aprender cómo se construyen algunas melodías simples con armonía, utilizando estos tres canales. Por el momento, lo primero que debemos aprender es a tener dominio de la música, trabajando con piezas escritas por otros compositores.

La música escrita para piano, o algunos tipos de pequeños órganos electrónicos es la más útil para nosotros por el momento. Sería muy difícil trabajar con una melodía escrita para una trompeta o un trombón, pues los sonidos que escuchamos no corresponden con las notas que vemos escritas, lo cual puede causar algunos estragos en el resultado final transformado en nuestro ordenador.

La forma en que trabajan estos instrumentos (casi todos los de viento) puede ser muy útil para los maestros compositores orquestales, pero se transforman en algo oscuro y retorcido para aquellos que lo único que quieren es entender un pentagrama escrito para estos instrumentos. Si nos reservamos a la música de un piano o un órgano, jamás tendremos ningún problema.

En el siguiente programa (19.4.6) podremos escuchar una melodía, que usa la armonía de una manera muy simple y sencilla. Utiliza blancas para prácticamente todas las notas, excepto las últimas, que son redondas. Podemos tocar las redondas como si de dos blancas se trataran, pero observaremos que no empleamos en el programa un bucle FOR... NEXT, con lo cual no podremos interpretarlas completamente unidas.

```
10 B1=1:T=70:A2=650
20 WHILE B1>0
30 READ B1,B2,B3
40 SOUND 1,B1,TIEMPO:SOUND 2,B2,TIEMPO:SOUND 4,B3,TIEMPO
50 FOR I=1 TO A2:NEXT
60 WEND
70 DATA 426,319,506
80 DATA 426,284,568
90 DATA 426,253,638
100 DATA 426,284,676
110 DATA 451,319,568
120 DATA 426,338,568
130 DATA 451,379,568
140 DATA 676,426,851
150 DATA 506,319,758
160 DATA 426,284,676
170 DATA 426,253,638
180 DATA 319,239,638
190 DATA 426,253,638
200 DATA 478,284,676
210 DATA 506,319,638
220 DATA 506,319,638
230 DATA 0,0,0
```

■ 19.4.6.

El programa hace uso de un tipo de instrucciones que ya deben ser familiares para nosotros: un bucle WHILE... WEND, para leer los datos hasta que llegamos a uno cuyo valor sea cero. Las notas han sido cambiadas a dígitos por medio de la tabla de conversión o transformación. El tiempo ha sido fijado en 70 para que la melodía sea

lenta, asignándose este valor a la variable TIME. Existe un bucle que produce un silencio que nos permitirá escuchar cada acorde separadamente, éste va desde 1 hasta I=650, pero es casi más aconsejable agrandararlo.

En el programa podremos observar que justo cuando termina un acorde comienza el siguiente, por lo cual, si deseamos cambiar el valor de la longitud de la nota LON, habremos también de alterar el valor I del bucle de silencio.

Ahora supongamos que vamos a enfatizar la melodía un pòquito. La melodía se halla en el canal 2, donde se utilizan los números NOT; podríamos enfatizarlos todos subiéndoles un poco el volumen. Ahora, gracias al simple uso que se puede hacer de la instrucción SOUND, podemos especificar el volumen deseado, añadiendo otro número al de duración de la nota.

Es posible elegir entre un rango que oscila de 0 a 7, aunque luego veremos que podemos aumentar este rango hasta 15. Si no especificamos ningún número de volumen, se adoptará por defecto el 4. Supongamos, por ejemplo, que queremos alterar el volumen de 4 a 7; sólo habría que tomar la línea 40 del programa y teclear:

```
40 SOUND 1,NOT1,TIME,:SOUND 2,NOT2,TIME,  
7:SOUND 4,NOT3,TIME
```

Con el cual produciríamos el correspondiente cambio de volumen tan sólo en el canal medio, en el 2, que es el que toca la melodía. Recordemos que los demás canales lo que hacen es producir acordes con estas notas. Si ahora escuchamos esta melodía, encontraremos que suena mucho más clara, y en conjunto es mejor que la anterior.

Podemos hacer otro experimento, cambiando el volumen de los bajos, pero el efecto sólo puede ser escuchado con nitidez si poseemos una cadena de alta fidelidad conectada a nuestro ordenador, o con unos cascos estereofónicos de cierta calidad. Si colocamos un volumen de cinco para las notas NOT3, la instrucción SOUND producirá una sensible enfatización de los bajos. Si usamos unos cascos podremos apreciar que las notas suenan separadas, y que los bajos se aprecian principalmente en la parte derecha. Es bastante importante que si escribimos una música en estéreo, la melodía siempre sea situada en el canal medio.

Ahora debemos perfeccionar aún más este sonido que hemos escuchado. Todavía podemos observar que las notas suenan de la misma manera, como si de un aburrido orador se tratara. Aquí lo que hace falta es ritmo. La solución es rápida enfatizamos la primera nota

de cada compás, lo cual podemos conseguirlo elevando el volumen o alargándola.

Supongamos que vamos a subir su volumen. Lo primero que deberíamos hacer es numerar de alguna manera las notas con 1, 2, 3, etc... y nada mejor para ello que volver al viejo sistema del bucle FOR... NEXT para leer los datos, y asignar a cada nota el valor que le corresponde en este bucle por su situación. Ahora decidimos que todas las notas impares sean de mayor volumen que las pares. Entonces, todos aquellos que no puedan ser divididos por dos exactamente son impares, y se les subirá el volumen. ¿Cómo llevamos a cabo esto?

El método para imprimir un poquito de ritmo en una melodía queda reflejado en el siguiente programa. El bucle FOR... NEXT se emplea para leer todas las notas de las líneas DATA, y también para conseguir la enfatización de alguna de ellas. El sistema puede ser nuevo para nosotros, además que MOD no es una orden BASIC, sino un operador, como pueda ser el signo de sumar, multiplicar o dividir.

MOD mantiene una relación entre dos números. Esta palabra se sitúa entre dos valores, como si de un signo de sumar se tratara, pero debe haber siempre un espacio a un lado y al otro de la palabra. Este operador puede emplearse para detectar números impares; en nuestro caso, MOD actúa así:

1 MOD 3 da 1	4 MOD 3 da 1 de nuevo
2 MOD 3 da 2	5 MOD 3 da 2 de nuevo
3 MOD 3 da 0	6 MOD 3 da 0 de nuevo

Así pues, como decíamos, podemos emplear MOD para detectar los números impares. Para ello, dividimos cada valor del bucle por 2, mediante MOD, y si obtenemos 0 es que hemos topado con un número par. Si un 1, es que el valor es impar (19.4.7).

```
10 T=70:A1=650
20 FOR I=1 TO 16
30 IF (I MOD 2)=0 THEN A2=5 ELSE A2=7
40 READ B1,B2,B3
50 SOUND 1,B1,T:SOUND 2,B2,T,A2:SOUND 4,B3,T,5
60 IF I<16 THEN FOR I=1 TO A1:NEXT
70 NEXT
80 DATA 426,319,506
90 DATA 426,284,568
100 DATA 426,253,638
110 DATA 426,284,676
120 DATA 451,319,568
```

```

130 DATA 426,338,568
140 DATA 451,379,568
150 DATA 676,426,851
160 DATA 506,319,758
170 DATA 426,284,676
180 DATA 426,253,638
190 DATA 319,239,638
200 DATA 426,253,638
210 DATA 478,284,676
220 DATA 506,319,638
230 DATA 506,319,638

```

19.4.7.

En el programa la investigación de esta circunstancia se ha llevado a cabo en la línea 30. Si el valor de I es impar, entonces el valor del volumen es 5, pero si es par, se aumenta el volumen a 7. Esto viene a dar un fuerte énfasis a la primera nota de cada compás. La línea 60 nos permite combinar las últimas dos notas, dos blancas, en una única redonda.

A continuación, y para terminar con la armonía, podemos observar y disfrutar del siguiente programa (19.4.8). Como vemos, el sistema es el mismo que en el anterior programa, pero lo que ahora llevamos a cabo es un alargamiento de la primera nota de cada compás. El nombre técnico de este tipo de composición es «Rubato», que significa robo, pues hemos robado a la segunda nota de cada compás una parte de su tiempo para cedérselo a la primera.

```

10 T=70:A1=650:A2=5
20 FOR I=1 TO 16
30 IF (I MOD 2)=0 THEN T=50 ELSE T=70
40 READ B1,B2,B3
50 SOUND 1,B1,T:SOUND 2,B2,T,A2:SOUND 4,B3,T,5
60 IF I<16 THEN FOR I=1 TO A1:NEXT
70 NEXT
80 DATA 426,319,506
90 DATA 426,284,568
100 DATA 426,253,638
110 DATA 426,284,676
120 DATA 451,319,568
130 DATA 426,338,568
140 DATA 451,379,568
150 DATA 676,426,851
160 DATA 506,319,758

```

170 DATA 426,284,676
 180 DATA 426,253,638
 190 DATA 319,239,638
 200 DATA 426,253,638
 210 DATA 478,284,676
 220 DATA 506,319,638
 230 DATA 506,319,638

■ 19.4.8.

SINCRONÍZATE

Imaginemos que tenemos una pieza musical tal como la de la figura (19.4.9) que podemos observar a continuación. El pentagrama en donde se encuentran los altos usa negras y corcheas, pero los bajos tienen negras y blancas. En cada compás, una blanca baja suena en el mismo tiempo que dos negras altas. Lo importante es ahora sincronizarlas.



■ 19.4.9.

Estamos utilizando una melodía extraordinariamente corta, y no necesitaremos esforzarnos demasiado. Podemos seguir un método que ya hemos empleado antes: colocar el número de canal como parte de las líneas DATA. Vamos a ver el primer compás de la melodía que usa una negra y dos corcheas como altas y una blanca en los bajos.

Comprobando que el tiempo es correcto, es decir, que es el mismo en ambos compases, esto debe sonar al menos aceptable. Los datos de la redonda son leídos y la nota es interpretada por el canal A; después se leen las tres notas altas y se interpretan por el canal B, de manera que apenas se puede apreciar la diferencia existente entre el principio de ejecución de una y otra melodía, y suena un acorde.

Para unas pocas notas, e incluso para unos pocos compases, este sistema puede resultar aceptable para la sincronización de las notas, pero después de algún tiempo dicha sincronización se pierde. Esto es debido a que el ordenador no toma los datos y los interpreta, tanto la

de los altos como los bajos, de manera absolutamente simultánea. Existe una muy pequeña diferencia que poco a poco se va acumulando, hasta que cada nota empieza a vivir su propia vida.

Cuando una melodía se compone de una cantidad determinada de notas, que no poseen silencios entre ellas, podemos emplear un efecto de «no separación» a simple oído entre nota y nota. Para probar esto, nada mejor que un ejemplo, como el que sigue a continuación (19.4.10).

```
10 A=40:B=1
20 WHILE B<>0
30 READ B,C,T
40 IF C=2 THEN A2=7 ELSE A2=5
50 SOUND B,C,T*A,A2
60 WEND
70 DATA 1,478,4
80 DATA 2,190,2
90 DATA 2,179,1
100 DATA 2,190,1
110 DATA 1,506,4
120 DATA 2,213,2
130 DATA 2,159,2
140 DATA 1,568,4
150 DATA 2,239,2
160 DATA 2,213,1
170 DATA 2,190,2
180 DATA 1,638,4
190 DATA 2,190,2
200 DATA 2,213,2
210 DATA 1,506,2
220 DATA 2,179,2
230 DATA 1,638,2
240 DATA 2,159,1
250 DATA 2,179,1
260 DATA 1,478,2
270 DATA 2,190,2
280 DATA 1,716,2
290 DATA 2,213,1
300 DATA 2,239,1
310 DATA 1,638,2
320 DATA 2,213,3
330 DATA 1,1276,2
340 DATA 2,239,1
350 DATA 1,956,4
360 DATA 2,239,4
370 DATA 0,0,0
```

■ 19.4.10.

Ha sido programado con líneas de datos que contienen el número de canal, el período y la duración. La instrucción SOUND emplea estos valores, con un número que multiplica la duración para obtener el tiempo absoluto. La música original es Andante, lo cual significa que no es ni lenta ni rápida. La melodía está un poco enfatizada su- biendo el volumen del canal 2. El sistema para conseguir este efecto ha sido aumentar el volumen si se lee un 2 como número de canal, y dejarlo en 5 si no es así.

Cuando tenemos de esta manera estructurado el programa, cambiar el volumen es bastante fácil, como hemos visto. Podríamos incluso cambiar el volumen de cada nota, pero es muy extraño que se dé este caso. Un método sería leer los datos mediante un bucle FOR ...NEXT, y cambiar el volumen chequeando el valor en cada momento de la variable del bucle.

Esta pequeña melodía, así suena de manera aceptable y sin problemas de sincronización, todo ello teniendo en cuenta que no hemos dejado silencios entre notas. Pero cuando deseamos que existan espacios entre nota y nota, tan sólo empleando un pequeño bucle que produzca un mínimo silencio, los efectos pueden ser auténticamente devastadores.

Otro sistema podría ser utilizar alguna instrucción que detectara aquellas notas que salen por el canal dos, para colocar mediante la instrucción SOUND un silencio detrás de ellas. No hace falta recordar que las notas del canal 2 son las de la melodía. Tomemos la línea 55 y añadamos:

```
IF CANAL=2 THEN SOUND 2,0,5,0
```

Producirá un corto silencio entre nota y nota. Ahora tenemos las notas separadas unas de otras, pero el acompañamiento sigue viviendo su vida por separado. Tecleemos el siguiente programa (19.4.11) y pongamos atención a lo que escuchamos. Empleamos, nada más y nada menos, que los códigos de sincronización de la instrucción SOUND.

```
10 A=40:B=1:I=500
20 WHILE B<>0
30 READ B,C,T
40 IF C=2 THEN A2=7 ELSE A2=5
50 SOUND B,C,T*A,A2
60 IF B=2 THEN SOUND 2,0,5,0
70 WEND
```

80 DATA 17,478,4
90 DATA 10,190,2
100 DATA 2,179,1
110 DATA 2,190,1
120 DATA 17,506,4
130 DATA 10,213,2
140 DATA 2,159,2
150 DATA 1,568,4
160 DATA 2,239,2
170 DATA 2,213,1
180 DATA 2,190,2
190 DATA 17,638,4
200 DATA 10,190,2
210 DATA 2,213,2
220 DATA 1,506,2
230 DATA 2,179,2
240 DATA 1,638,2
250 DATA 2,159,1
260 DATA 2,179,1
270 DATA 17,478,2
280 DATA 10,190,2
290 DATA 1,716,2
300 DATA 2,213,1
310 DATA 2,239,1
320 DATA 1,638,2
330 DATA 2,213,3
340 DATA 1,1276,2
350 DATA 2,239,1
360 DATA 17,956,4
370 DATA 10,239,4
380 DATA 0,0,0

■ 19.4.11.

Podremos observar que alguno de los números de canal han sido cambiados, y cuando escuchamos la melodía apreciaremos que la sincronización es mucho mejor. Lo que hemos hecho es forzar los canales a que se sincronicen al principio de cada compás. Esto todavía no es perfecto, pero las imperfecciones son casi inapreciables.

SINCRONICEMOS NUESTRAS MELODÍAS

Para sincronizar la melodía se utilizan valores que se añaden a los ya de por sí números de canal.

Número	Efecto
8	SINCRONIZACION CON NOTA EN CANAL A
16	SINCRONIZACION CON NOTA EN CANAL B
32	SINCRONIZACION CON NOTA EN CANAL C

Estos números deben ser añadidos al del canal que debe ser sincronizado. En nuestra melodía, queremos que la primera nota del canal B se sincronice con la primera nota del canal A. La primera línea de datos contiene como número de canal el 1, que para hacerlo sincronizar con el dos, debemos añadir 16 al número uno, obteniendo 17.

Sin embargo, esto no tiene ningún efecto si no efectuamos la misma operación con el canal B, añadiéndole 8 para que sincronice con el A, obteniendo el valor diez como número de canal de sonido. El programa ha sido transformado alterando las primeras notas de cada barra en los dos canales.

Estas notas tienen alterados sus números de canal cambiando cada 1 por un 17 y cada 2 por un 10. Esto permite que los canales se sincronicen al principio de cada compás, es decir, cada cierto tiempo. Si lo deseáramos, podríamos sincronizar la melodía de otra manera, pero es mucho más útil llevar a cabo este trabajo de esta manera.

Sin embargo, hay que dejar algo muy claro, y es que en todas aquellas melodías que escribamos que no posean ningún tipo de silencio entre notas, jamás deberemos preocuparnos de la sincronización. Sólo la emplearemos en aquellas piezas, normalmente largas, plagadas de silencios dedicados a separar las notas o para transmitir un cierto ritmo a la melodía.

El mejor camino es utilizar siempre el canal B como canal de la melodía, y realizar la sincronización con él en el principio de cada compás del pentagrama, frecuentemente cada cuatro notas o números en el ordenador.

La sincronización comienza a ser absolutamente imprescindible cuando empezamos a hacer uso de las envolventes, para crear el sonido de otros instrumentos musicales. De las envolventes hablaremos el próximo capítulo, pero a modo de introducción podemos reseñar que gracias a una envolvente, podemos modificar el volumen de una nota durante el tiempo en que está sonando. Esto nos permitirá escuchar notas separadas unas de otras sin el uso de molestos silencios, aunque el control de la melodía empezará a ser un tanto complicado.

La sincronización en el uso de envolventes es francamente difícil. En ocasiones, tendremos que sincronizar más de una vez por compás,

pero las fronteras de la programación musical se nos irán ensanchando poco a poco, cada vez más.

AMSTRAD ESTÉREO

El sonido estéreo en nuestro ordenador se consigue simplemente empleando debidamente los canales. Si tenemos a mano unos cascos estereofónicos, podremos comprobar que el sonido producido por el canal A, de código 1, es el que sale al aire por el casco izquierdo. Por ejemplo:

SOUND 1,239,5000,7

y podremos escuchar la nota con el oído izquierdo; pero si probamos

SOUND 4,200,5000,7

la escucharemos por el derecho. Y ahora

SOUND 2,300,5000,7

nos permitirá escuchar la nota por las dos salidas de los cascos. Esto es por lo que siempre hemos utilizado hasta ahora el canal B para la melodía. El volumen de ésta, cuando la estamos escuchando por medio de unos cascos, sólo puede ser variado por medio del número de volumen. Sin embargo, si tenemos conectado nuestro ordenador a una cadena de alta fidelidad, podremos conseguir maravillas con las señales provenientes de nuestro Amstrad.

Veamos el siguiente programa (19.4.12). La melodía se «toca» por el canal izquierdo, y el acompañamiento por el canal derecho. Con unos buenos cascos o una cadena Hi-Fi, el efecto es perfectamente reconocible y apreciable por nuestro oído, si bien apenas podremos distinguirlo por el altavoz de nuestro Amstrad (19.4.13).

El efecto escuchado dista bastante del que deseamos conseguir. En el siguiente programa, podremos oír la misma melodía, pero con algunas alteraciones en las líneas DATA. Hemos colocado, como ya es de uso y costumbre, la melodía en el canal B, pero el acompañamiento será llevado a cabo por los canales A y C (19.4.14).

La melodía es un vals, y cada línea de bajos es precisamente el del característico «ram pam pam» de este tipo de música. La primera de estas tres notas, que normalmente posee el tono más bajo, serán emitidas por el canal C, y las otras dos por el A. Cuando este acompaña-

miento consta de una sola nota, la haremos sonar por los dos canales simultáneamente.

```
10 AS=20
20 FOR I=1 TO 31
30 READ A,B,C
40 IF A=31 THEN A2=7 ELSE A2=5
50 SOUND A,B,A#C,A2
60 SOUND 1,0,2,0:SOUND 4,0,2,0
70 NEXT
80 DATA 1,284,2,1,239,2,1,213,2
90 DATA 33,190,6,12,568,2,4,478,2,4,478,2
100 DATA 12,379,6,33,190,2,1,213,2,1,239,2
110 DATA 33,213,6,12,506,2,4,426,2,4,426,2
120 DATA 12,379,6,33,253,2,1,213,2,1,190,2
130 DATA 33,179,6,12,506,2,4,426,2,4,426,2
140 DATA 12,379,6,33,179,2,1,190,2,1,213,2
150 DATA 33,239,6,12,568,2,4,478,2,4,478,2
```

■ 19.4.12.

■ 19.4.13.

La sincronización, sin embargo, es bastante complicada. No la necesitaremos en la primera línea de datos, porque estas notas no llevan acompañamiento. Pero el asunto se complica en la siguiente línea DATA. La primera nota que encontramos, una blanca que es reproducida por el canal B, debe ser sincronizada con una negra que será reproducida por el canal C. Como el número de sincronización con C

es 32, al añadir éste al código del canal B, que es 2, obtenemos 34 como número de canal.

La siguiente nota de la línea de datos es la primera línea del acompañamiento. Esta sale por el canal C, de código 4, y debe ser sincronizada con la melodía del canal B, para lo cual utilizamos el número de sincronización con B (16) y obtenemos el correspondiente número de canal, que es 20.

```
10 AS=20
20 FOR I=1 TO 38
30 READ A,B,C
40 IF A=31 THEN A2=7 ELSE A2=5
50 SOUND A,B,AS*C,A2
60 SOUND 1,0,2,0:SOUND 4,0,2,0:SOUND 2,0,2,0
70 NEXT
100 DATA 2,284,2,2,239,2,2,213,2
110 DATA 34,190,6,20,568,2,12,0,4,33,478,2,1,478,2
120 DATA 17,379,6,20,379,6,42,190,2,2,213,2,2,239,2
130 DATA 34,213,6,20,506,2,12,0,4,33,426,2,1,426,2
140 DATA 17,379,6,20,379,6,42,253,2,2,213,2,2,190,2
150 DATA 34,179,6,20,506,2,12,0,4,33,426,2,1,426,2
160 DATA 17,379,6,20,379,6,42,179,2,2,190,2,2,213,2
170 DATA 34,239,6,20,568,2,12,0,4,33,478,2,1,478,2
```

19.4.14.

Pero la dificultad aumenta cada vez más. Las siguientes dos notas del acompañamiento tienen que ser tocadas en el canal A (recorremos que corresponden al PAM PAM). Si las interpretamos sin el uso de la sincronización, estas dos notas serán tocadas al mismo tiempo que otras dos de la melodía. Para remediar este problema, creamos un silencio en el canal C.

Ahora el canal C se sincroniza con A ($4+8=12$), y utilizamos un período de cero para el silencio, con una duración de 4. Pero observemos la línea 120. Todo empieza con la primera nota de acompañamiento, emitida por el canal A y sincronizada con el canal B, y número de canal, pues, 17. En código 20 se emplea para tocar los bajos también en el canal C, sincronizado con B.

Ahora viene la primera nota de la melodía, en el canal B naturalmente, y sincronizada con los canales A y C. Deberemos tomar el

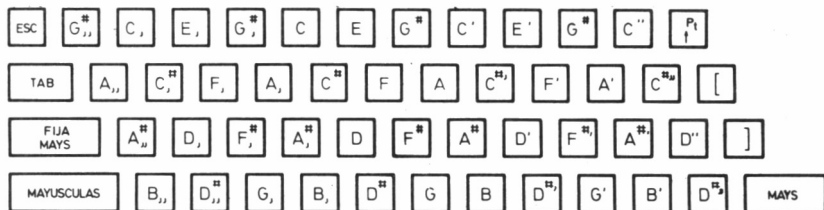
código del canal B y sumarle el código de sincronización con A, más el código de sincronización con C, es decir, $2+8+32=42$.

SU TURNO

Si realmente hemos pensado en componer alguna melodía para nuestro ordenador, precisamente el peor camino que podemos elegir para ello es usar nuestro micro. Si entre los trastos de nuestra habitación podemos rescatar un sintetizador de doble teclado, puede ser la mejor manera de empezar a realizar una composición, pero como la mayoría de nosotros no encontraremos más teclado que el de nuestro ordenador, utilizaremos un programa que nos permitirá confeccionar alguna que otra composición musical.

Para ello, vamos a hacer uso de dos programas (19.4.16); el primero convierte las teclas de nuestro Amstrad en un teclado musical. Las posiciones de las notas no son precisamente las de un piano, pues podremos suponer que si nuestro ordenador tuviera instaladas las teclas en fila horizontal, necesitaríamos una carcasa de metro y medio de ancho. En el dibujo podremos ver cómo se han distribuido las notas por nuestro Amstrad (19.4.15).

19.4.15.



PROGRAMA 19.4.16

```

10 TIEMPO=30
20 FOR I=22 TO 71
30 IF INKEY(I)=0 THEN X=I-21:ELSE 50
40 ON X GOSUB 80,100,110,120,140,160,180,200,220,240,260,280,300,320,3
40,360,380,400,420,440,460,480,500,520,540,560,570,590,610,630,650,670
,690,710,730,750,770,790,810,830,850,870,890,910,930,940,960,970,990,1
000
50 NEXT
60 IF INKEY(79)=0 THEN FOR S=1 TO 4:PRINT CHR$(8);CHR$(16);:NEXT
70 CALL &BB03:GOTO 20
80 SOUND 2,100,TIEMPO,7:PRINT"100 ";
90 RETURN
100 RETURN
110 RETURN
120 SOUND 2,119,TIEMPO,7:PRINT"119 ";
130 RETURN
140 SOUND 2,113,TIEMPO,7:PRINT"113 ";
150 RETURN
160 SOUND 2,142,TIEMPO,7:PRINT"142 ";
170 RETURN
180 SOUND 2,106,TIEMPO,7:PRINT"106 ";
190 RETURN
200 SOUND 2,134,TIEMPO,7:PRINT"134 ";
210 RETURN
220 SOUND 2,127,TIEMPO,7:PRINT"127 ";
230 RETURN
240 SOUND 2,159,TIEMPO,7:PRINT"159 ";
250 RETURN
260 SOUND 2,150,TIEMPO,7:PRINT"150 ";
270 RETURN
280 SOUND 2,190,TIEMPO,7:PRINT"190 ";
280 SOUND 2,190,TIEMPO,7:PRINT"190 ";
290 RETURN
300 SOUND 2,179,TIEMPO,7:PRINT"179 ";
310 RETURN
320 SOUND 2,225,TIEMPO,7:PRINT"225 ";
330 RETURN
340 SOUND 2,169,TIEMPO,7:PRINT"169 ";
350 RETURN
360 SOUND 2,213,TIEMPO,7:PRINT"213 ";
370 RETURN
380 SOUND 2,253,TIEMPO,7:PRINT"253 ";
390 RETURN
400 SOUND 2,201,TIEMPO,7:PRINT"201 ";
410 RETURN
420 SOUND 2,239,TIEMPO,7:PRINT"239 ";
430 RETURN
440 SOUND 2,301,TIEMPO,7:PRINT"301 ";
450 RETURN
460 SOUND 2,284,TIEMPO,7:PRINT"284 ";
470 RETURN
480 SOUND 2,358,TIEMPO,7:PRINT"358 ";
490 RETURN
500 SOUND 2,338,TIEMPO,7:PRINT"338 ";
510 RETURN
520 SOUND 2,268,TIEMPO,7:PRINT"268 ";
530 RETURN
540 SOUND 2,319,TIEMPO,7:PRINT"319 ";
550 RETURN
560 RETURN
570 SOUND 2,379,TIEMPO,7:PRINT"379 ";
580 RETURN
590 SOUND 2,478,TIEMPO,7:PRINT"478 ";
600 RETURN

```

```

610 SOUND 2,568, TIEMPO,7:PRINT"568 ";
620 RETURN
630 SOUND 2,451, TIEMPO,7:PRINT"451 ";
640 RETURN
650 SOUND 2,426, TIEMPO,7:PRINT"426 ";
660 RETURN
670 SOUND 2,536, TIEMPO,7:PRINT"536 ";
680 RETURN
690 SOUND 2,402, TIEMPO,7:PRINT"402 ";
700 RETURN
710 SOUND 2,506, TIEMPO,7:PRINT"506 ";
720 RETURN
730 SOUND 2,602, TIEMPO,7:PRINT"602 ";
740 RETURN
750 SOUND 2,758, TIEMPO,7:PRINT"758 ";
760 RETURN
770 SOUND 2,716, TIEMPO,7:PRINT"716 ";
780 RETURN
790 SOUND 2,902, TIEMPO,7:PRINT"902 ";
800 RETURN
810 SOUND 2,851, TIEMPO,7:PRINT"851 ";
820 RETURN
830 SOUND 2,676, TIEMPO,7:PRINT"676 ";
840 RETURN
850 SOUND 2,638, TIEMPO,7:PRINT"638 ";
860 RETURN
870 SOUND 2,804, TIEMPO,7:PRINT"804 ";
880 RETURN
890 SOUND 2,1204, TIEMPO,7:PRINT"1204 ";
900 RETURN
910 SOUND 2,956, TIEMPO,7:PRINT"956 ";
920 RETURN
930 END
940 SOUND 2,1135, TIEMPO,7:PRINT"1135 ";
950 RETURN
960 RETURN
970 SOUND 2,1073, TIEMPO,7:PRINT"1073 ";
980 RETURN
990 RETURN
1000 SOUND 2,1012, TIEMPO,7:PRINT"1012 ";
1010 RETURN
1020 RETURN

```

■ 19.4.16.

Se pueden emplear hasta 44 teclas, que reproducirán 44 notas con una duración prefijada, si bien al menos durará el sonido tanto tiempo como tengamos la tecla pulsada; quizá un poco más. Podemos controlar dicha duración alterando en valor TIME de la línea 10.

Cuando utilizamos el programa, sólo las teclas señaladas en la figura tienen efecto musical. La tecla DEL borrará los números de nota que aparecen sobre la pantalla, por lo cual, podemos corregir una nota que no nos haya agradado, al escuchar su sonido.

ESC produce el final del programa. Si tecleamos un RETURN en vez de END en la línea 930, el programa sólo podrá ser interrumpido por medio de un CONTROL MAYS ESC. Para no llegar a este extremo, es aconsejable que grabemos el programa antes de ejecutarlo. Cuando ya hayamos tocado la melodía deseada, y corregido los errores, habremos de parar el programa, y por medio del desdoblamiento de cursores, copiar en una línea DATA los datos aparecidos en la pantalla. Después borraremos las líneas de programa y grabaremos nuestras líneas DATA musicales para utilizarlas más tarde en una composición.

El siguiente programa (19.4.17) es un buscador de acordes. Con él podemos introducir un número de tono, y la nota correspondiente será interpretada. Al introducir otro número produciremos armonía y las dos notas serán ejecutadas al mismo tiempo para escuchar el resultado. Si es de nuestro gusto, podemos buscar otra nota para encontrar una nueva armonía, más perfecta. Si no nos gusta el resultado obtenido, bastará con pulsar la barra espaciadora para que podamos elegir cualquier otro número.

```
10 INPUT"NUMERO DE NOTA";N
20 SOUND 1,N,30000,6
30 INPUT"ARMONIA 1 ";N1
40 SOUND 130,0,0
50 SOUND 2,N1,30000
60 A$=INKEY$:IF A$="" THEN 70
70 IF A$=CHR$(32) THEN 40
80 INPUT"ARMONIA 2 ";N2
90 SOUND 132,0,0
100 SOUND 4,N2,30000
110 A$=INKEY$:IF A$="" THEN 120
120 IF A$=CHR$(32) THEN 90
130 INICIO=TIME
140 WHILE TIME < INICIO +2000:WEND
150 CLEAR
160 END
```

■ 19.4.17.

ENVOLVENTES DE SONIDO



Si nos tapamos los ojos con las manos o con un pañuelo, fácilmente podremos adivinar, mediante el sentido del oído, cuando una misma melodía es interpretada con un instrumento u otro. ¿Qué es lo que diferencia una nota tocada en un trombón de la misma en una guitarra eléctrica?

Estas dos notas poseen la misma amplitud y tono, pero las escuchamos de manera muy distinta cuando llegan hasta nuestros pabellones auditivos. Debemos tener en cuenta dos factores: la forma de la onda y algo llamado envolvente.

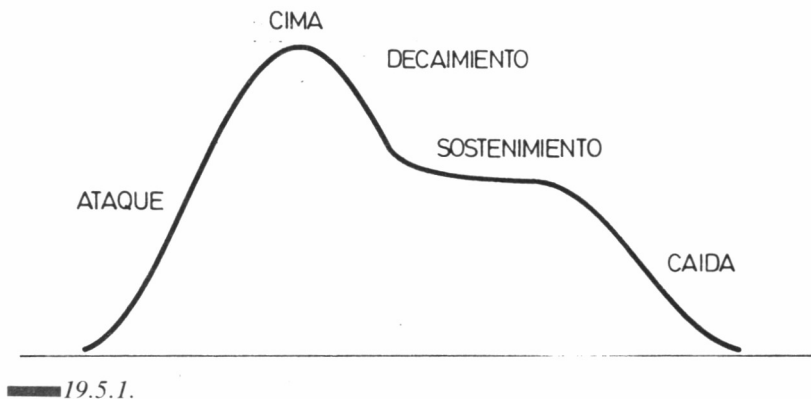
La forma de la onda de un instrumento clásico musical es infinitamente más compleja que la que puede producir, por ejemplo, el chip de sonido de nuestro ordenador. No es algo fácil de estudiar.

Las envolventes son muy diferentes. Una nota de un instrumento está constituida por infinidad de ondas. Cuando producimos una nota por medio de un instrumento, las primeras ondas son más pequeñas que las posteriores, hasta llegar a la onda más amplia, momento a partir del cual la amplitud irá disminuyendo hasta cero.

Es decir, la nota necesita una fase de «arranque» hasta llegar a la «cima» de máxima amplitud, para posteriormente iniciar una fase de descenso o disminución de la amplitud. Sin embargo, lo que ocurre después de llegar a la cima de amplitud, depende en gran manera del

instrumento del cual haya nacido la nota. Por ejemplo, si este es de percusión o punteado (un tambor y una guitarra, respectivamente, por ejemplo) ocurrirá el fenómeno que acabamos de describir.

En la figura (19.5.1) podemos observar las partes de una envolvente, es decir, el arranque, cima, leve descenso, una parte estable en la cual la amplitud se mantiene constante, y la fase de disminución total o caída.



Este tipo de envolventes se llama «de amplitud», pues reflejan este aspecto de las ondas. Sin embargo, un instrumento puede ser empleado de diversas maneras, con las cuales se producen las envolventes «de tono».

Si observamos a un violinista tocando, interesándonos en especial en el movimiento de sus manos, observaremos que los dedos de la mano que presionan las cuerdas se agitan de un lado a otro del instrumento. Este típico movimiento de las yemas de los dedos, sobre las cuerdas que están siendo presionadas, permite al intérprete manipular el tono de la nota, aumentándolo y decreciéndolo a placer.

Esta acción lleva el nombre técnico de «vibración», cuya finalidad es conseguir que las notas suenen más intensamente, aunque, por otra

parte, demasiada vibración puede producir que la melodía suene indecisa.

Dicha vibración no es fácil de obtener en otros instrumentos, y mucho menos en nuestro ordenador, pero como sabemos que el Amstrad es capaz de alcanzar cotas calificadas de increíbles, no dejaremos de intentarlo. El ordenador necesita trabajar con una envolvente de amplitud y otra de tono. Pronto veremos como se definen.

La instrucción SOUND debe poseer los datos de canal, tono, duración y volumen. El dígito de duración ha de ser cero, y el del volumen debe ser, normalmente también cero. Si intentamos reproducir el sonido de un instrumento, el dígito de volumen ha de ser siempre cero, pues la melodía empezará desde cero y crecerá en amplitud, y recordemos que a mayor amplitud corresponde mayor volumen, y viceversa.

También podemos asignar otros valores al volumen de inicio de la melodía, pero éstos se suelen emplear para sonidos especiales. El volumen, que oscila de 0 a 7 normalmente, oscila de 0 a 15 cuando ha sido definida una envolvente.

Después debemos definir la envolvente, pero la instrucción SOUND sólo especificará qué envolvente ha de ser empleada en cada momento, pudiendo elegir entre 15 tipos distintos. Si no están definidas, la instrucción SOUND no puede llamarla.

Naturalmente, con frecuencia sólo queremos definir una envolvente, con lo cual, indicar en la instrucción SOUND el número 1, bastará para conseguirlo. En la instrucción ENV, debe ir especificado al principio el número de envolvente. Esta orden BASIC es quizá una de las más difíciles de manejar, pero a corto plazo puede resultar bastante beneficiosa.

DEFINIENDO ENVOLVENTES

Es muy complejo reproducir la envolvente de una nota producida por un instrumento musical. La verdadera envolvente de amplitud posee una figura complicadísima, y las partes que antes hemos diferenciado (ascenso, cima...) no son más que una aproximación. Todo esto quiere decir que la curva de una envolvente será tan complicada que es muy difícil de crear.

La envolvente más aproximada la podemos conseguir por medio de líneas rectas, formando figuras en escaleras, o curvas. Aun así, podemos obtener una figura bastante parecida a la real. Pero, empecemos

mos a crear envolventes. Existen tres maneras de realizarlas, como veremos.

Los tres caminos o métodos para generar una envolvente poseen otros tantos nombres técnicos: hardware, software absoluto y software relativo. Si echamos una ojeada al manual, podremos comprobar que al método que más espacio se le dedica es al de software relativo, quizá el más complicado del trío. Sin embargo, de vez en cuando nos puede interesar aplicar los otros dos sistemas; así pues, empecemos por ellos.

Las envolventes de hardware utilizan formas que están predefinidas en la parte de la ROM que se ocupa del chip de sonido. Podemos convertir cinco de estas envolventes en simplemente una, o tomar una y mezclarla con otra de software, técnica que veremos más adelante. Todo se hará según sea el propósito, el objetivo, el efecto que persigamos.

La forma de la instrucción ENV para crear o tomar envolventes de hardware es la siguiente:

ENV dígito 1, =dígito 2, dígito 3

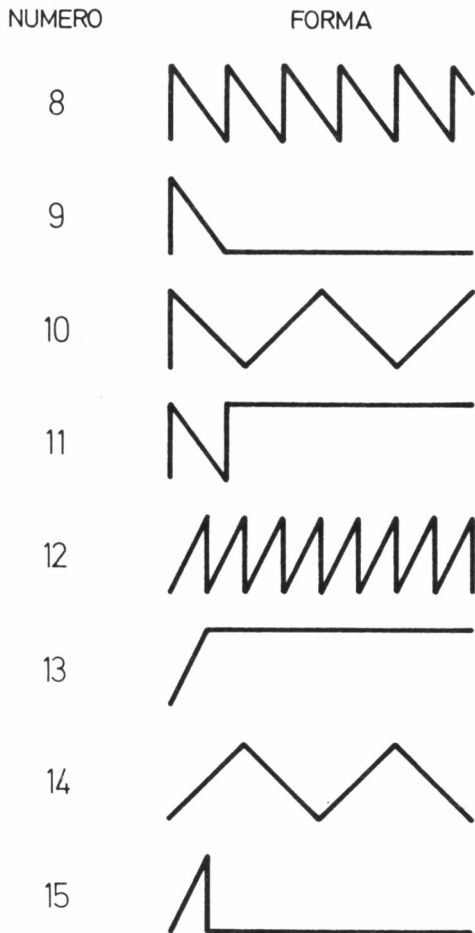
En la cual, el primer dígito indica el número asignado a la envolvente (de 1 a 15). El segundo nos obliga a elegir la envolvente que deseamos emplear (de 8 a 15), y el último asigna el período de la envolvente (de 0 a 65535). Pero vamos a internarnos algo más profundamente en el campo de las envolventes hardware.

USO Y FORMAS DE LAS ENVOLVENTES DE HARDWARE

Este tipo de envolventes nos abren un amplio campo a la experimentación, para poder considerar todas las posibilidades que se nos brindan con su uso. Existen ocho tipos, como ya dijimos (de 8 a 15), de envolventes de hardware. Podemos observar en la figura (19.5.2) la forma de todas ellas, pero a menudo tendremos cierta dificultad para asociar estas formas, con el sonido que estamos escuchando. Esto es debido a que el efecto conseguido no depende tan sólo de la forma de la envolvente, sino de su período de tiempo.

Si el período es demasiado corto, alguna de las envolventes actuará muy rápidamente. Por ejemplo, los 9 y 15 son casi inaudibles, así como es casi imposible escuchar el comienzo de las envolventes 11 y

13. El caso contrario, es decir, un período exageradamente grande, producirá que el tiempo para cambiar el volumen sea tan largo que sólo escuchemos una parte de la envolvente.



Un período que ronde el valor 1000 puede ser lo más adecuado en este caso. El siguiente programa (19.5.3), permite escuchar los ocho tipos de envolventes con varios períodos, para que nosotros mismos podamos juzgar cuál es el ideal.

```
10 FOR I=8 TO 15:RESTORE
20 PRINT"ONDA NUMERO ";I
30 FOR A=1 TO 4
40 READ AS:PRINT" PERIODO ";AS
50 ENV 1,=I,AS
60 SOUND 2,239,0,0,1
70 FOR L=1 TO 6000:NEXT
80 NEXT:NEXT
90 DATA 10,100,1000,10000
```

■ 19.5.3.

A priori, los sonidos más interesantes y espectaculares los producen las envolventes 8, 12 y 14 con un período de 10. Esto, a pesar de todo, depende únicamente del sonido que deseemos crear. ¿Podría ser este el sonido buscado para imitar un fagot o un órgano? Estaremos de acuerdo que si lo que buscamos es imitar el sonido de un barco saliendo de puerto nos viene como anillo al dedo.

Algunos sonidos son resultado de la mezcla del tono especificado en la instrucción de sonido y la repetición de la envolvente. Se pueden crear sonidos muy interesantes de este modo. Veamos un ejemplo bastante sencillo (19.5.4).

```
10 ENV 1,=8,10
20 ENV 2,=8,100
30 SOUND 2,20,0,0,1
40 SOUND 2,50,0,0,2
```

■ 19.5.4.

El uso de un período muy corto en envolventes de alto tono, produce la superposición de dos notas. Con un largo período, el sonido se convierte en un suave susurro. Pocas aplicaciones podemos encontrarle para la música, pero para efectos especiales puede sernos de gran utilidad.

Antes de continuar, es mejor aclarar a que es a lo que estamos llamando continuamente número de período. Quizá nos engañe esta denominación, pues puede hacernos entender que representa la duración, el tiempo de la envolvente, lo cual no es cierto.

Lo que se establece, es el tiempo de cambio de una parte a otra de la envolvente, algo llamado el «tiempo de paso». Tomemos como ejemplo una envolvente de número sugerente, como la 13. Está formada por una línea diagonal ascendente y otra recta y constante. Esta sección inclinada se halla representada por dieciséis pasos, y el tiempo que tardará en pasar por cada uno de ellos es precisamente el número de período.

El ordenador, de todas formas, dispone automáticamente dos segundos de pausa detrás de cada envolvente de hardware. Tal vez esto puede empezar a hacernos pensar que es imposible realizar música con este tipo de envolventes.

Para controlar más o menos el tiempo de las envolventes de hardware, es aconsejable conocer cómo se produce una pausa. Para ello se necesita añadir los números 1, 0 y N, a la orden ENV, donde precisamente N es el número de pausa, que oscila entre 0 y 255.

Con un número de pausa de 255, obtendríamos 2,55 segundos de pausa o retraso, pues los números del 1 al 255 dan pausas en unidades de 100. Cero, caso aparte, produce la máxima pausa, de 2,56 segundos. Las pausas más normales suelen ser de 10 a 50 musicalmente hablando. Veamos, como ejemplo la siguiente melodía (19.5.5):

```
10 ENV 2,=9,1000,1,0,40
20 ENV 4,=9,8000,1,0,40
30 FOR I=1 TO 30
40 READ A,B
50 SOUND 2,A,0,0,B
60 NEXT
70 DATA 253,2,213,2,213,2,213,2
80 DATA 253,2,213,2,213,2,213,2
90 DATA 190,2,239,2,284,2,239,2
100 DATA 239,2,253,2,253,4
```

110 DATA 253,2,213,2,213,2,213,2
 120 DATA 253,2,213,2,213,2,213,2
 130 DATA 190,2,239,2,284,2,338,2
 140 DATA 190,2,213,2,213,4

19.5.5.

Esta parece haber sido programada de la manera más simple, es decir, con una serie de líneas de notas de las que deberemos calcular su duración. Pero no es tan fácil como eso, pues no podemos utilizar un número de duración si estamos empleando una envolvente, sino que debemos programar una envolvente para cada duración de nota. Los números 2 y 4 del programa son los números de las envolventes.

A cada envolvente se le ha otorgado un período apropiado a la duración de su nota, pero hemos de tener cuidado al calcularlos. Podríamos esperar que un valor de 2000 produjese una duración en tiempo del doble que 1000, pero no es así, pues no se trata de duraciones de nota, sino de valores de tiempo descendente.

Una vez hayamos escuchado el programa ejemplo, que emplea una envolvente número 9, probemos una modificación. Esta vez, la envolvente estará constituida por una envolvente 13, una rampa ascendente, seguida de una 9 y una rampa descendente. La combinación de las dos constituye una envolvente de forma triangular.

Intentemos ahora introducir en el programa anterior

10 ENV 2,=13,500,=9,500,1,0,40
 20 ENV 4,=13,4000,=9,4000,1,0,40

y comparémosla con ésta

10 ENV 2=13,500,1,0,7,=9,500,1,0,40

En esta última podremos escuchar con meridiana claridad el punto más alto de la envolvente.

Sin embargo, más allá de las envolventes de hardware, podemos construir aquella envolvente que más nos plazca o satisfaga a nuestros oídos. Para ello, tenemos a nuestra disposición las envolventes absolutas y relativas. La diferencia entre las dos es que las absolutas permiten cambios de volumen más bruscos que las relativas. Estos dos métodos de envolvente son muy parecidos, así pues los estudiaremos juntos.

En primer lugar, podemos anunciar que el número de secciones es como mucho cinco, como en las envolventes de hardware. En cada una de estas secciones, podemos tener una línea recta o una serie de escalones que conforman el ángulo de una línea inclinada. Esto quiere decir, que es posible representar o simular mediante líneas, una envolvente clásica con ascenso, leve descenso, mantenimiento y descenso, y todavía tendríamos una sección de reserva.

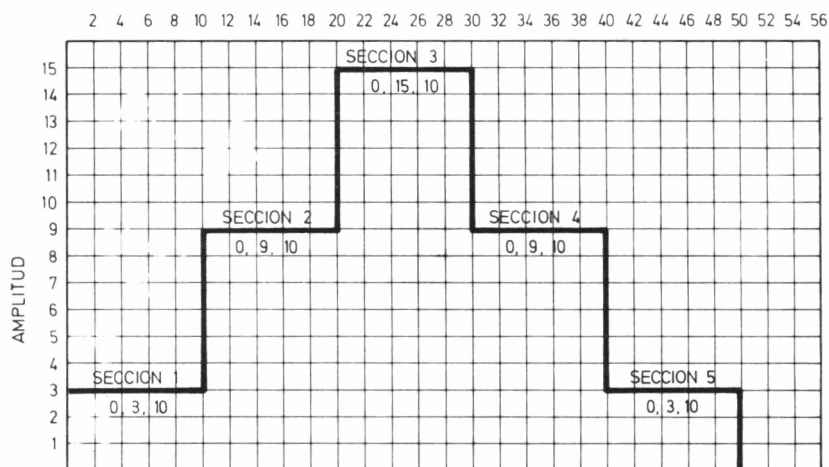
En las envolventes absolutas tan sólo nos está permitido usar un volumen por cada sección de envolvente. Debemos especificar tres números muy determinados en cada sección: cuenta de pasos, altura del paso y tiempo del paso. El primero es un número que oscila entre 0 y 127. Si utilizamos cero, la envolvente es absoluta. La altura de paso es el número de la amplitud, y por lo tanto del volumen. Este número puede ser positivo o negativo, y va de 127 a -128, pero en la práctica sólo se emplean del 0 al 15. Finalmente, el tiempo de paso es exactamente lo que indica el término, y oscila entre 0 y 255, siendo cero una pausa de 2.56 segundos.

Cuando usamos envolventes absolutas, las envolventes de software por excelencia, tenemos que diseñar cada una de ellas sobre un papel preparado para ello. Tomemos un papel cuadriculado y tomémosle las medidas que aparecen en la figura. Podemos observar una envolvente, en la cual las líneas verticales representan niveles de volumen (de 0 a 15) y las horizontales ofrecen los tiempos y el número de pasos, que como podemos observar, no es mayor que 5 en cada sección. En cada paso, poseemos un número de volumen fijo, que es el segundo de los tres que aparecen debajo de cada línea. El último número nos informa del tiempo en que será ejecutada cada sección (19.5.6).

En el siguiente programa, aplicamos la envolvente a una melodía que ya habíamos escuchado antes. Podemos comprobar que todas las notas suenan unas detrás de otras, sin ningún tipo de pausa; esto es porque no hemos programado un silencio, mediante una bajada de volumen, al final de la envolvente. Las absolutas pueden llegar a resultarnos útiles cuando no necesitemos cambiar demasiado el volumen. En el siguiente programa (19.5.7), el volumen comienza alto, y se baja una unidad en cada paso, menos en el último que es cero. Esto se traduce en un silencio, que además es mayor, pues hemos aumentado su tiempo.

Un uso específico para las envolventes absolutas es la producción de eco. Si creamos una sección con un volumen alto, después un silencio, y tras él un volumen más bajo, podemos crear un sencillo efecto bastante real. El siguiente programa (19.5.8) sirve para corroborar

esta teoría; en él, tomamos la melodía que ya hemos empleado en programas precedentes y cada nota es sonorizada en forma de eco. Cada envolvente posee esta vez cuatro secciones.



19.5.6.

```

10 ENV 2,0,3,10,0,9,10,0,15,10,0,9,10,0,3,10
20 ENV 4,0,3,20,0,9,20,0,15,20,0,9,20,0,3,20
30 FOR I=1 TO 30
40 READ A,B
50 SOUND 2,A,0,0,B
60 NEXT
70 DATA 253,2,213,2,213,2,213,2
80 DATA 253,2,213,2,213,2,213,2
90 DATA 190,2,239,2,284,2,239,2
100 DATA 239,2,253,2,253,4
110 DATA 253,2,213,2,213,2,213,2
120 DATA 253,2,213,2,213,2,213,2
130 DATA 190,2,239,2,284,2,338,2
140 DATA 190,2,213,2,213,4

```

19.5.7.


```

10 ENV 2,0,15,10,0,0,10,0,7,10,0,0,20
20 ENV 4,0,15,20,0,0,20,0,7,16,0,0,16
30 FOR I=1 TO 30
40 READ A,B
50 SOUND 2,A,0,0,B
60 NEXT
70 DATA 253,2,213,2,213,2,213,2
80 DATA 253,2,213,2,213,2,213,2
90 DATA 190,2,239,2,284,2,239,2
100 DATA 239,2,253,2,253,4
110 DATA 253,2,213,2,213,2,213,2
120 DATA 253,2,213,2,213,2,213,2
130 DATA 190,2,239,2,284,2,338,2
140 DATA 190,2,213,2,213,4

```

■ /9.5.8.

ENVOLVENTES RELATIVAS

Cuando dibujamos sobre un papel la forma de una envolvente relativa, llegamos rápidamente a la conclusión de que es considerablemente más complicada que una envolvente de hardware o absoluta. Sin embargo, ahora que ya sabemos cómo se crea una envolvente absoluta, no nos será tan difícil aprender cómo se crea una relativa.

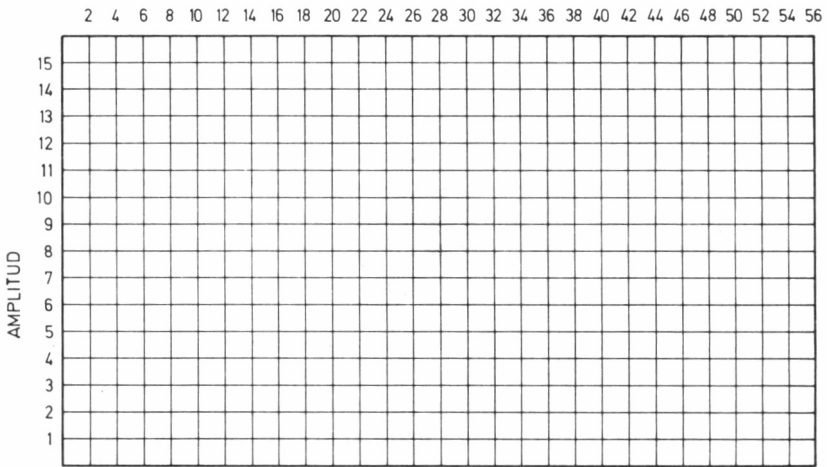
En primer lugar, debemos especificar el número de secciones de la envolvente; un máximo de cinco, como de costumbre. Tras el comando BASIC ENV, aparece un valor del 1 al 15 que es el número de referencia de la envolvente. Después tenemos tres números para cada sección de la envolvente. El primero de éstos es el contador de pasos. Para un envolvente absoluta, este número es cero, pero para una relativa, este número es el volumen relativo al paso anterior, es decir, el incremento. Recordemos que el volumen inicial será decidido por la instrucción SOUND.

El segundo valor de cada sección es la altura de paso, como en las envolventes absolutas, con números permitidos desde -128 a 127, pero sólo utilizables de 0 a 15. Y el último valor es el tiempo de cada paso de la envolvente.

Ahora debemos estudiar la relación que mantienen cada uno de estos valores con los demás. El contador de pasos, por ejemplo, multiplicado por el tiempo de paso, da el tiempo total de cada sección de la envolvente. Este no deberá ser excesivamente corto, pero tampoco

demasiado largo. Otra relación es el producto del tiempo de paso por la altura de paso, que da el volumen. Si tenemos diez pasos, cada uno de una altura de uno, tendremos un cambio de volumen en esta sección de 10 unidades de volumen. Si el volumen comienza en cero, en la instrucción SOUND el volumen se elevará hasta 10. Si el volumen empieza en 5, ascenderá hasta el máximo, que es 15.

Como sólo poseemos cinco secciones de envolvente, lo más útil es utilizar cuatro para realizar una elevación, leve bajada, sostenimiento y caída, dejando una sección de reserva para realizar algún silencio si fuera necesario. Para empezar, tomemos un papel de dibujar envolventes, como el de la figura (19.5.9), en la cual se muestra cómo ha de estar marcado.

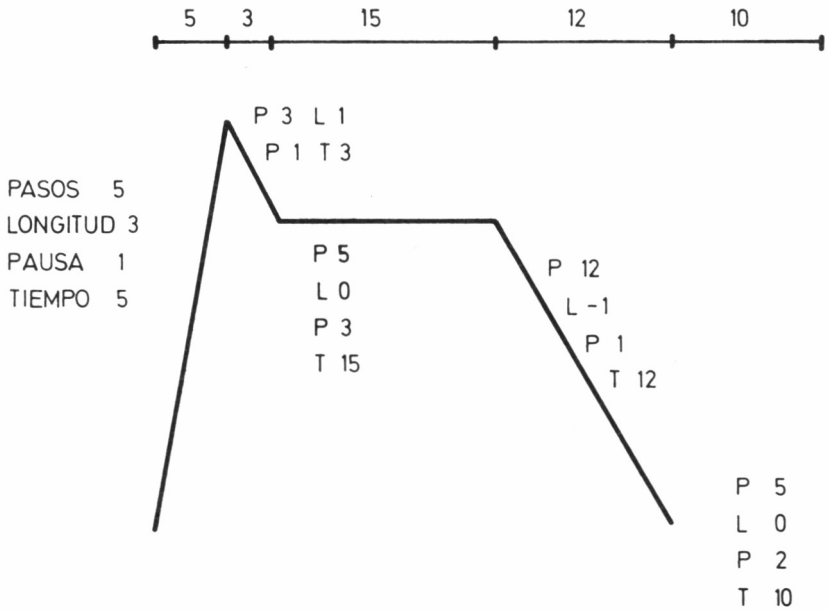


19.5.9.

Utilizamos el papel gráfico simplemente por comodidad, pero sería una buena idea utilizar papel de calco y lápiz. Este sistema es más eficaz, porque al principio cometeremos gran cantidad de errores y dibujaremos muchas secciones, aunque después caeremos en la cuenta que no las podemos programar, o no nos ofrecen los resultados apetecidos.

Con papel de calcar y lapicero, podemos ensayar cuanto queramos, borrar las secciones que no deseemos, y posteriormente pasar a tinta las definitivas o las que ofrezcan mejores resultados. Debemos tener mucho cuidado en la confección de una envolvente. La forma de una sección nunca inspira como sonará posteriormente.

No intentemos dibujar los pasos porque es tiempo perdido; simplemente bastará con que tracemos subidas y bajadas con líneas inclinadas, y mostremos el tiempo total de cada sección, que es lo importante. Si sumamos los tiempos de las secciones, hallaremos el tiempo total de la nota. En el ejemplo (19.5.10), el ascenso consta de cinco pasos de tres unidades cada uno, lo cual nos da un resultado de quince unidades de volumen.



19.5.10.

Se trata de un ascenso muy lento, mucho más que el que podríamos conseguir con un instrumento de cuerda o de percusión. El tiempo total utilizado en este ascenso es de cinco unidades. El leve descenso consta de tres pasos, la unidad de volumen es uno, la misma que la de tiempo. Esto hace que el volumen baje de 15 a 12, y disponga para la operación de tres unidades de tiempo.

El sostenimiento tiene cinco pasos de volumen cero con una unidad de tiempo de tres. El tiempo total será igual a 15 unidades, sin cambio en absoluto de volumen. La caída absoluta final es muy lenta. Para ella, utilizamos doce pasos de medida uno, para conseguir que el volumen baje de nuevo de doce a uno. La unidad de tiempo es también de uno, así que el tiempo empleado es de doce unidades.

Al final, el volumen se mantiene estable, en cero y en los últimos cinco pasos de tiempo 2, lo cual se traduce en un silencio de 10 unidades de tiempo. El tiempo absoluto o total para toda la nota o para toda la envolvente es de 42 unidades, que son 0.42 segundos.

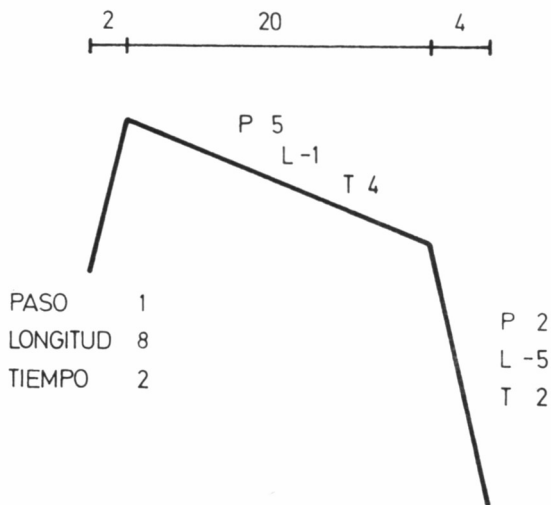
En el siguiente programa (19.5.11), las envolventes son diseñadas con tiempos más largos para la envolvente más pequeña, lo cual produce el sonido típico de una envolvente que posee un ascenso y una caída relativamente lentos. Cada nota suena como si temiera aparecer.

```
10 ENV 2,5,3,1,3,-1,1,5,0,3,12,-1,1,5,0,2
20 ENV 4,5,3,2,3,-1,2,5,0,6,12,-1,2,5,0,2
30 FOR I=1 TO 30
40 READ A,B
50 SOUND 2,A,0,0,B
60 NEXT
70 DATA 253,2,213,2,213,2,213,2
80 DATA 253,2,213,2,213,2,213,2
90 DATA 190,2,239,2,284,2,239,2
100 DATA 239,2,253,2,253,4
110 DATA 253,2,213,2,213,2,213,2
120 DATA 253,2,213,2,213,2,213,2
130 DATA 190,2,239,2,284,2,338,2
140 DATA 190,2,213,2,213,4
```

19.5.11.

Sin embargo, podemos producir ascensos más fuertes y rápidos de dos maneras. La primera es reducir los pasos, aumentar la amplitud, y disminuir el tiempo de paso. La otra forma es emplear alguna amplitud especificada en la instrucción SOUND. Si el sonido empieza con un volumen de cinco, por ejemplo, es más fácil realizar un ascenso más fuerte, más agudo.

También podemos colocar el volumen en 15 y tan sólo realizar las fases de la segunda a la tercera, lo cual proporcionará un ascenso más agudo que el que podemos conseguir por el segundo sistema. En la siguiente figura (19.5.12), observamos una nota muy corta con ataques o ascensos con más pasos.



19.5.12.

Esta ha sido realizada empleando un volumen inicial de siete y elevándolo hasta quince, con pasos de dos unidades. La envolvente decae entonces sin la sección de sostenimiento. Esto, como podemos escuchar en el programa que listamos a continuación (19.5.13) produ-

ce un sonido de cuerda punteada. El ritmo de la música será más rápido a causa de la longitud de la caída de la nota.

Cuando estemos diseñando envolventes hemos de tener mucho cuidado. La alteración de una sola de sus secciones puede modificar la extensión total y el tiempo. Debemos practicar y experimentar, en otras palabras, jugar con las envolventes para obtener un perfecto control sobre ellas.

```
10 ENV 2,1,8,2,5,-1,4,2,-5,2
20 ENV 4,1,8,2,5,-1,8,2,-5,4
30 FOR I=1 TO 30
40 READ A,B
50 SOUND 2,A,0,7,B
60 NEXT
70 DATA 253,2,213,2,213,2,213,2
80 DATA 253,2,213,2,213,2,213,2
90 DATA 190,2,239,2,284,2,239,2
100 DATA 239,2,253,2,253,4
110 DATA 253,2,213,2,213,2,213,2
120 DATA 253,2,213,2,213,2,213,2
130 DATA 190,2,239,2,284,2,338,2
140 DATA 190,2,213,2,213,4
```

■ 19.5.13.

EL ORDENADOR CREA NUESTRAS ENVOLVENTES

Existen muchos programas de creación de envolventes, pero no son realmente útiles, si no tenemos mucha experiencia en el tema. Gracias a este tipo de programas, podemos descubrir gran cantidad de efectos de sonido, aunque si intentamos encontrar alguno específico, es posible que este sistema desemboque en una lamentable pérdida de tiempo.

El siguiente programa (19.5.14) nos permitirá encontrar envolventes rápida y fácilmente. Hemos disminuido las opciones, a pesar de que ofrezcan buenos resultados, para que simplemente con cambiar algunos pocos valores podamos alterar el sonido de la nota y escu-

charlo. Podremos ver nuestra lista de opciones, realizando los cambios deseados y recordando los anteriores.

Es un método más rápido de crear una envolvente que tener que dibujarla sobre un papel cuadrulado y numerado, y podemos ahorrar mucho tiempo cuando busquemos un sonido que no sepamos exactamente cómo es. Cuando hayamos realizado ya nuestra creación, la envolvente será impresa por impresora, si es que disponemos de este periférico.

PROGRAMA 19.5.14

```

10 CLS:A$="ERROR-INTENTALO OTRA VEZ"
20 PRINT TAB(16)"ENVOLVENTE"
30 LOCATE 1,2
40 WHILE B$<>"Q"
50 PRINT"ATAQUE-- RAPIDO,NORMAL,LENTO (R/N/L) ":C$=D$
60 GOSUB 470:IF E$="P" THEN PRINT"--";C$:GOTO 110:ELSE D$=E$
70 IF D$="R" THEN B1=1:B2=15:B3=1
80 IF D$="N" THEN B1=3:B2=5:B3=1
90 IF D$="L" THEN B1=5:B2=3:B3=2
100 IF INSTR("RNL",D$)=0 THEN PRINT A$:GOTO 50
110 PRINT"LEVE CAIDA-- NINGUNA,RAPIDA,MEDIA O TOTAL (N/R/M/T) ":C$=F$
120 GOSUB 470:IF E$="P" THEN PRINT"--";C$:GOTO 180:ELSE F$=E$
130 IF F$="N" THEN B4=1:B5=0:B6=1
140 IF F$="R" THEN B4=1:B5=-3:B6=1
150 IF F$="M" THEN B4=3:B5=-1:B6=2
160 IF F$="T" THEN B4=15:B5=-1:B6=2:ENV 1,B1,B2,B3,B4,B5,B6,10,0,10:GOTO 400
170 IF INSTR("NRMT",F$)=0 THEN PRINT A$:GOTO 110
180 PRINT "SOSTENIMIENTO-- NINGUNO, FEQUEVO, MEDIO O LARGO (N/P/M/L) ":C$=G$
190 GOSUB 470:IF E$="P" THEN PRINT"--";C$:GOTO 250:ELSE G$=E$
200 IF G$="N" THEN B7=1:B8=0:B9=1
210 IF G$="P" THEN B7=2:B8=0:B9=2
220 IF G$="M" THEN B7=3:B8=0:B9=4
230 IF G$="L" THEN B7=5:B8=0:B9=5
240 IF INSTR("NFML",G$)=0 THEN PRINT A$:GOTO 180
250 PRINT"CAIDA-- RAPIDA, NORMAL O LENTA (R/N/L) ":C$=H$
260 A$=B1*B2+B4*B5
270 GOSUB 470:IF E$="P" THEN PRINT"--";C$:GOTO 320:ELSE H$=E$
280 IF H$="R" THEN B10=1:B11=-A$:B12=1
290 IF H$="N" THEN B10=2:B11=-(A$/2):B12=2
300 IF H$="L" THEN B10=3:B11=-A$/3:B12=3
310 IF INSTR("RNL",H$)=0 THEN PRINT A$:GOTO 180
320 PRINT"TIEMPO DE SILENCIO-- LARGO, NORMAL O CORTO (L/N/C) ":C$=I$
330 GOSUB 470:IF E$="P" THEN PRINT"--";C$:GOTO 320:ELSE I$=E$
340 IF I$="L" THEN B13=10:B14=0:B15=10
350 IF I$="N" THEN B13=5:B14=0:B15=5
360 IF I$="C" THEN B13=3:B14=0:B15=5
370 IF INSTR("LNC",I$)=0 THEN PRINT A$:GOTO 180
380 ENV 1,B1,B2,B3,B4,B5,B6,B7,B8,B9,B10,B11,B12,B13,B14,B15
390 SOUND 120,0,0
400 SOUND 2,239,-20,0,1

```

```

410 PRINT"PULSA Q PARA ACABAR, OTRA PARA EMPEZAR"
420 B$=INKEY$:IF B$="" THEN 420
430 WEND
440 PRINT"LOS DIGITOS DE LA ENVOLVENTE SON:"
450 PRINT"ENV 1,"";B1;"";B2;"";B3;"";B4;"";B5;"";B6;"";B7;"";B8;"";
";B9;"";B10;"";B11;"";B12;"";B13;"";B14;"";B15
460 END
470 E$=INKEY$:IF E$=""THEN 470 ELSE RETURN

```

19.5.14.

ENVOLVENTES DE TONO

Al igual que las envolventes de amplitud, que deciden el cambio del volumen durante el transcurso de una nota, también podemos alterar el tono de una nota mientras está sonando. Es una manera de obtener vibraciones. El grado de vibración tiene que ser cuidadosamente escogido y debe ser más alto para notas más altas que para las bajas. La cantidad de vibración debe ser regulada.

Normalmente, grandes cambios de tono suelen sonar ridículos. El uso de las envolventes de tono precisa que se utilice un dato más en la instrucción SOUND, además de emplear la instrucción ENT para definir las.

ENT se parece mucho a ENV en la manera de construir envolventes. Como es habitual, detrás de la instrucción encontramos, en primer lugar, el número de envolvente, que puede ir de 1 a 15. Una pequeña diferencia estriba en que podemos elegir valores negativos, con lo que conseguiremos que la variación de tono se repita hasta el final de la nota.

Siguiendo al número de envolvente, es posible definir cinco secciones de cambio de tono, siendo cada sección un conjunto absoluto o relativo. La diferencia entre ambas, radica en que el conjunto absoluto posee un número de tono inalterable durante un tiempo determinado, y en el relativo podemos especificar un grado de cambio.

El período de tono será un número de 0 a 4095, pero lo más práctico es usar uno de 20 a 2000. El tiempo de pausa dista de 0 a 255. El cero produce una pausa de 2.56 segundos. Si la primera sección de una envolvente de tono es una sección absoluta, el número de período

do de tono será ignorado. Por ejemplo, con ENT 1,=239,50 sonará la nota C' durante medio segundo.

Las envolventes de tono relativas emplean tres números en cada sección. El primer número corresponde con el de pasos, el segundo con la altura de paso, y el tercero con el tiempo de pausa. La altura de paso no es la amplitud de un determinado paso, sino el tono. Si estamos escuchando la nota 300, un paso de +1 significa un cambio a 280, que es un tono más bajo. Estas alturas de paso pueden ser negativas (tono más alto) o positivas (tono más bajo).

Los números de pausa son usualmente intervalos en unidades de 1/100 segundos. En el siguiente programa (19.5.15), podemos escuchar una nota que comienza con una pequeña vibración. El tono se altera en cuatro pasos, cada uno una unidad positiva, después de ocho negativas, y cuatro positivas otra vez, acabando así con la misma nota. Veamos ahora que en la instrucción SOUND, todos los números están especificados, pero existe un cero como envolvente de amplitud. Esto es debido a que debe haber siempre antes de un número de envolvente de tono, uno de envolvente de amplitud. Si como ocurre en el ejemplo, no usamos envolvente de amplitud, debemos colocar un cero en esa posición.

```
10 ENT 1,4,1,1,8,-1,1,4,1,1  
20 SOUND 2,239,100,15,0,1
```

■ 19.5.15.

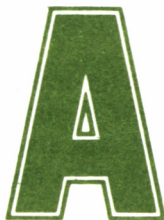
Para terminar, un ejemplo de cómo las notas pueden ser tocadas con una envolvente de tono y amplitud al mismo tiempo; escuchemos

el siguiente programa (19.5.16). Utiliza una envolvente de amplitud que produce un rápido ascenso, leve descenso, no muy grande sostenimiento, y una rápida caída, con un silencio al final. La envolvente de tono produce una subida en el tono de tres pasos, lo baja seis, y lo vuelve a subir otros tres. Utilizaremos estos dos tipos de envolventes para algo completamente distinto: el ruido.

```
10 ENV 1,1,15,1,2,-3,1,5,0,5,1,-9,1,5,0,4
20 ENT -1,3,1,1,6,-1,1,3,1,1
30 FOR I=1 TO 8
40 READ A
50 SOUND 2,A,0,0,1,1
60 NEXT
70 DATA 478,426,379,358,319,284,253,239
```

■ 19.5.16.

EL RUIDO



ello a lo cual nosotros llamamos ruido es en nuestro ordenador un sonido aleatorio. Para las notas musicales, hemos definido un tono y una amplitud, para el ruido nada de esto hace falta. Podemos generar eléctricamente algo llamado «ruido blanco», que es una mezcla de sonido que contiene todas las frecuencias y una gran cantidad de amplitudes.

Cuando vemos en la pantalla de un osciloscopio la forma del ruido blanco, más bien parece una banda de espesa niebla, sin formas cuadradas ni puntas. Este ruido es parecido al que nosotros obtenemos cuando tenemos parado nuestro dial a mitad de camino entre dos emisoras de radio. ¿Qué es lo que escuchamos? Simplemente ruido.

Los ruidos naturales no tienen nada que ver con este, así que por mucho que golpeemos dos cacerolas jamás podremos imitar el que produce nuestro ordenador. Este no es un caso de ruido blanco, sino de que el sonido que producimos posee una envolvente que crece, se mantiene y muere. Algo parecido ocurre cuando golpeamos un instrumento de percusión. El sonido es similar al de una nota musical, en este caso, pero una vez más las frecuencias que produce están relacionadas con un valor único, que depende de la longitud, el volumen y la superficie del instrumento. Todos los sonidos naturales, como los relámpagos o la lluvia al caer, son de este tipo.

A partir de ahora toda descripción empieza a ser difícil y lo que hemos de hacer es agudizar nuestros oídos al máximo. Si antes era difícil predecir lo que un programa de sonido va a interpretar, en el caso del ruido es casi imposible, además de que pequeños cambios pueden determinar muy diferentes efectos para nuestro oído. Las posibilidades son tan grandes que nadie puede decir haber creado, escuchado y catalogado todos los efectos de sonido posibles. Todos los programas que veremos pueden ser utilizados en nuestros propios juegos o utilidades.

Nuestro Amstrad nos permite elegir entre quince «tonalidades» de ruido. Esta selección se lleva a cabo designando un número de ruido, del 1 al 15, tras el número del tono de la envolvente. Introduciendo 0, como es normal, no se producirá ningún sonido; todo lo contrario del ruido.

El ruido puede ser interpretado por separado o en conjunción con otros sonidos. Si ponemos un cero como número de tono del canal de sonido, sólo se producirá ruido. Nótese que sólo disponemos de un generador de ruido, y que no podemos utilizar o escuchar diferentes ruidos por cada uno de los canales simultáneamente.

Para empezar a habituar nuestro oído al ruido que habrá de soportar a lo largo de este capítulo, nada mejor que escuchar el ruido de tonos de 1 a 31. El manual quizá nos diga que sólo podemos usar valores de 0 a 15, pero en otros lugares podremos encontrar los números disponibles llegan al 31.

El sonido en el siguiente programa (19.6.1) puede parecer el vapor que se libera de una olla a presión cuando quitamos la pesa. El tono del ruido comienza alto y acaba bajo. Cambiando ligeramente el programa, podemos lograr efectos muy interesantes, por ejemplo, cambiando el tiempo del programa, haciéndolo más rápido, podremos escuchar un sonido parecido al que oiríamos en una playa con olas embravecidas.

El programa que tenemos a continuación (19.6.2) es un tanto curioso, pues reproduce el sonido que emitiría una locomotora de vapor de hace ya bastantes décadas. Quizá no sea la mejor simulación de locomotoras que hemos escuchado jamás, pero usando una envolvente podríamos acercarnos aún más a una locomotora convincente.

Las posibilidades del ruido se pueden ampliar infinitamente utilizando envolventes de ruido. No necesitamos elaborar minuciosamente una envolvente de software relativa, sino tan sólo emplear las que ya tenemos definidas, es decir, las de hardware. La envolvente de hardware número 9 es particularmente útil para una gran variedad de sonidos, tales como disparos.

```

10 FOR I=1 TO 31
20 SOUND 2,0,50,15,0,0,I
30 FOR A=1 TO 1000:NEXT
40 NEXT

```

■ 19.6.1.

```

10 FOR I=0 TO 79
20 IF I MOD 4=0 THEN A2=15 ELSE A2=5
30 SOUND 2,0,10,A2,0,0,5
40 FOR A=1 TO 150:NEXT
50 NEXT

```

■ 19.6.2.

Sería bastante bueno conseguir que el ordenador interpretase dos sonidos débilmente diferentes. En el siguiente ejemplo (19.6.3), el tono es 10 en el primero, y 12 en el segundo. Esto produce un cierto efecto de estéreo. Notemos que el programa, a medida que se va ejecutando, está cortado por varias pausas totalmente imprescindibles para escuchar el efecto estéreo. Probemos el sonido que resulta de su supresión.

Repetir una envolvente de hardware puede ser muy útil para conseguir efectos de sonido, mucho más que para la música. Una importante aplicación de este tipo de combinación es la formación de sonidos rítmicos, como por ejemplo el producido por el motor de pistones de un antiguo aeroplano.

Con un tiempo muy pequeño en la envolvente de hardware, que ha de ser de forma 9, podremos crear un sonido bastante convincente. Pero ahora experimentemos cambiando la envolvente por una núme-

```

10 ENV 1,=9,1000
20 FOR I=1 TO 12
30 SOUND 4,0,0,0,1,0,19
40 FOR A=1 TO 1000:NEXT
50 SOUND 132,0,0
60 SOUND 1,0,0,0,1,0,12
70 FOR A=1 TO 1000:NEXT
80 SOUND 129,0,0
90 NEXT

```

■ 19.6.3.

ro 14. Así, podremos escuchar el sonido característico de las hélices de un helicóptero. Para ello, haremos uso de una ENV 1,0=12,200, también muy útil para conseguir este tipo de sonidos.

La combinación de envolventes hardware puede resultar explosiva, particularmente si mezclamos las número 13 y 9. Asociadas con pequeños tiempos, vienen a producir sonidos parecidos a disparos, de los que ya empezamos a tener un buen acopio; pero con largos tiempos, podremos escuchar el sonido que produciría un martillo golpeando una superficie no resonante (19.6.4).

```

10 ENV 1,=13,20000,=9,20000
20 FOR I=1 TO 12
30 SOUND 2,0,0,0,1,0,10
40 NEXT

```

■ 19.6.4.

Obviamente, podemos hacer uso de las envolventes relativas de software para controlar el sonido, y una vez que empezamos a trabajar en este campo, las posibilidades son ilimitadas. Veamos un ejemplo: sonido de un deportista haciendo footing. Podemos alterar el tiempo de alguna de las partes de la envolvente, y de la frecuencia

predominante del ruido, para producir un gran número de efectos (19.6.5).

Otra dimensión en la cual nos podemos introducir dentro de este interminable mundo del ruido, es la aportada por las envolventes de tono para el sonido. Su efecto es alterar el tono a medida que se está produciendo el sonido. Si colocamos una longitud de paso positiva, el sonido comenzará con tono alto y finalizará con tono bajo. Esto produce efectos para naves espaciales, pistolas y un sinfín de posibilidades que nos pueden ayudar en la creación de fantásticos juegos.

```
10 FOR I=1 TO 20
20 ENV 1,5,5,2,4,-2,1,5,0,3
30 SOUND 2,0,0,0,1,0,31
40 NEXT
```

■ 19.6.5.

El sonido del siguiente programa (19.6.6) se ha creado por combinación de una envolvente de hardware 13 y un sonido como un silbido. La envolvente de hardware crea al principio un sonido muy agudo que el comando ENT se encarga de transformar. Los dos sonidos al tiempo pueden crear unos efectos extraordinarios. Realicemos algunos cambios y escuchemos la diferencia.

```
10 ENV 1,=13,6000
20 ENT 1,200,1,1
30 SOUND 2,0,0,1,1,10
```

■ 19.6.6.

También podemos crear dos sonidos diferentes con un solo comando. Aunque esta afirmación no se hace del todo cierta en el si-

guiente programa... ¡al menos da esa impresión! Este efecto podría ser muy útil para nuestros propios juegos de guerras interestelares (19.6.7).

```
10 ENV 1,=11,5000
20 ENT 1,200,1,1
30 SOUND 2,0,0,0,1,1,10
```

■ 19.6.7.

Ahora probaremos en el siguiente programa (19.6.8) la combinación de una envolvente de hardware repetitiva con una de tono. Sin embargo, el comando ENT no sólo puede utilizarse para notas descendentes, sino también para las ascendentes. Si es esto lo que deseamos, debemos colocar un valor de comienzo en la instrucción de sonido SOUND.

```
10 ENV 1,=8,300
20 ENT 1,200,1,2
30 SOUND 2,0,0,0,1,1,31
```

■ 19.6.8.

En el siguiente programa (19.6.9) podemos observar el resultado comentado en el párrafo anterior. Se emplea una altura de paso negativa en el comando ENT, con un número de período de tono de comienzo de 500 en la línea SOUND.

A modo de nuevo experimento, podemos probar sonidos rítmicos acompañados por música. En el próximo programa (19.6.10) obten-

dremos una imitación de tambores. Para ello usa dos envolventes, una con una rampa ascendente y una rápida caída, y la otra con un rápido ataque y una lenta caída.

```
10 ENV 1,=14,5000  
20 ENT 1,200,-4,2  
30 SOUND 2,500,0,0,1,1,1
```

■■■■/19.6.9.

```
10 ENV 1,15,1,1,1,-15,1,5,0,4  
20 ENV 2,1,15,1,15,-1,1,5,0,5  
30 SOUND 2,0,0,0,1,0,20  
40 SOUND 2,0,0,0,2,0,25  
50 SOUND 2,0,0,0,2,0,30
```

■■■■/19.6.10.

Dichas envolventes pueden sincronizarse en otro canal con una melodía, porque de otra manera el ritmo y la música irían absolutamente descoordinados. El siguiente programa (19.6.11) puede ser un buen ejemplo de este efecto.

Algunos tipos de sonidos de percusión poseen cambios del tono del ruido durante el tiempo del sonido. Estos son realmente difíciles de sintetizar. El siguiente programa (19.6.12) emplea una envolvente con un ataque o ascenso inicial medio y un bucle que va cambiando los números de ruido de las instrucciones SOUND. Al ataque le sigue una leve caída y dos golpes de ruido.

```

10 ENV 1,15,1,1,1,-15,1,5,0,4
20 ENV 2,1,15,1,15,-1,1,5,0,5
30 FOR I=1 TO 8
40 READ B1,B2,B3
50 SOUND 17,B1,40,7
60 SOUND 10,0,0,0,1,0,20
70 SOUND 17,B2,30,6
80 SOUND 10,0,0,0,2,0,25
90 SOUND 17,B3,30,6
100 SOUND 10,0,0,0,2,0,20
110 NEXT
120 DATA 239,213,190,213,239,284
130 DATA 284,253,239,284,319,358
140 DATA 319,284,253,239,213,190
150 DATA 213,190,239,213,213,213

```

■ 19.6.11.

```

10 FOR I=1 TO 3
20 ENV 1,3,5,1
30 FOR A=23 TO 15 STEP -2
40 SOUND 2,0,0,0,1,0,A
50 NEXT
60 FOR S=1 TO 400:NEXT
70 FOR B=1 TO 2
80 SOUND 2,0,10,7,0,0,5
90 FOR S=1 TO 400:NEXT
100 NEXT
110 FOR S=1 TO 400:NEXT
120 NEXT

```

■ 19.6.12.

RUIDO CON LAS ENVOLVENTES DE TONO

Podemos crear una extensa selección de efectos de sonido utilizando una envolvente de tono junto a notas musicales. Normalmente, para efectos musicales, las envolventes de tono se aplican a composi-

ciones muy pequeñas, pero cuando las empleamos para realizar grandes cambios de tono, se pueden obtener efectos realmente interesantes.

Veamos el siguiente programa (19.6.13) que produce un sonido de aviso. Se consigue con una envolvente de tono que tiene 200 pasos de cambio de tono, -5 ascendente y 2 descendente. Cuando se aplica a un sonido que empieza en la nota 500, el resultado es un cambio rápido a un tono alto.

```
10 ENT 1,200,-5,2
20 FOR I=1 TO 20
30 SOUND 2,500,100,7,0,1
40 NEXT
```

■ 19.6.13.

Aquí debemos elegir cuidadosamente la duración del sonido, ya que de otra manera podríamos encontrarnos con efectos bastante lejanos a los que perseguimos. En el ejemplo, podemos observar que la instrucción de sonido ha sido colocada en medio de un bucle que se repite veinte veces. El sonido parece una alarma de evacuación.

Con tonos más altos, podríamos generar efectos aún más interesantes. En el siguiente ejemplo (19.6.14), podremos escuchar un agudísimo sonido de alarma, que se obtiene con valores bajos para el período de tono. La primera nota es el número de tono 20, una nota muy alta, y la orden ENT especifica 10 pasos de unidad de cambio. Se trata pues de un rápido cambio de tono. Si añadimos a la instrucción SOUND un número de sonido, como 20, podremos escuchar un cierto ruido de hélices de helicópteros. Este es otro ejemplo de cómo un pequeño cambio en el programa puede producir un efecto completamente distinto.

Un dato importante a tener en cuenta, es que no debemos colocar las envolventes en medio de un bucle, si deseamos repetir una envolvente varias veces. Con un valor negativo en la instrucción ENT, la envolvente será repetida tanto como dure la nota especificada en la línea SOUND. Esta facilidad puede sernos de gran utilidad si deseamos que el tono de ciertas notas suba y baje, como en el siguiente

programa. Este emplea una envolvente de tono que especifica un cambio de 50 pasos hacia abajo, 100 hacia arriba, y 50 otra vez hacia abajo (19.6.15).

```
10 ENT 1,10,-1,1
20 FOR I=1 TO 20
30 SOUND 2,20,10,7,0,1
40 NEXT
```

■ 19.6.14.

```
10 ENT -1,50,2,2,100,-2,2,50,2,2
20 SOUND 2,200,5000,7,0,1
```

■ 19.6.15.

Cuando esto se lleva a cabo con un número negativo de envolvente, escuchamos un sonido parecido al de una sirena. Podemos realizar algunos trucos con este tipo de envolventes, realizando secciones no simétricas, es decir, en las que el tono del final de la envolvente no es el mismo que el del principio.

En el siguiente ejemplo (19.6.16), el tono se eleva durante cincuenta pasos, pero sólo descende durante noventa y sube otros cincuenta. Esto produce que exista un tono al final diez pasos más alto que el del principio; un efecto realmente interesante. Recordemos también que podemos crear envolventes de tono con porciones de tono ininterrumpido. Esto se consigue con una sección que tiene por número de cambio de tono el cero, como podemos ver en el siguiente ejemplo. En el sonido producido, la nota más alta se mantiene más tiempo que la nota baja.

```
10 ENT -1,50,-2,1,90,2,1,50,-2,1
20 SOUND 2,200,1000,7,0,1
```

■ 19.6.16.

Lo siguiente es combinar envolventes de amplitud y tono con notas musicales. Con la cantidad de posibilidades que nos ofrecen las envolventes de volumen o amplitud, si les sumamos las de las envolventes de tono, podemos obtener un pozo inagotable de experimentos en el campo de los efectos musicales.

Observemos el siguiente ejemplo (19.6.17). Este utiliza una envolvente de hardware número 9 de larga duración, y una envolvente de tono que desciende y se eleva de nuevo. También podemos apreciar el efecto de la combinación de una envolvente repetitiva de hardware y una de tono, incluso podemos realizar algunos cambios para obtener efectos aún más interesantes; por ejemplo, dando el valor 30 al período de la envolvente de tono.

```
10 ENV 1,=8,200
20 ENT -1,20,1,1,40,-1,1,20,1,1
30 SOUND 2,200,0,0,1,1
```

■ 19.6.17.

En el siguiente programa (19.6.18) escucharemos el sonido producido por un insecto. Asimismo, podemos cambiar número de la nota a 800 en la instrucción de sonido.

```
10 ENV 1,=10,1000
20 ENT -1,30,-1,1,5,6,1
30 SOUND 2,200,0,0,1,1
```

■ 19.6.18.

En el ejemplo que aparece a continuación (19.6.19), escucharemos una sirena de policía. Para ello se ha empleado una envolvente de hardware número 13, la cual posee una rampa alta donde encuentra su máximo volumen, todo ello combinado con una envolvente de tono que lo sube o baja, pero decae finalmente. También podemos tomar una envolvente repetitiva con un corto período de tiempo, y con mayor cambio de tono, como podemos observar en el segundo programa (19.6.20).

```
10 ENV 1,=13,12000
20 ENT -1,28,-1,1,5,6,1
30 SOUND 2,80,0,0,1,1
```

■ 19.6.19.

```
10 ENV 1,=14,100
20 ENT -1,26,-1,1,5,6,1
30 SOUND 2,60,1000,0,1,1
```

■ 19.6.20.

Ahora intentemos algo completamente diferente. Sonidos aún más versátiles pueden obtenerse si usamos envolventes de software relativas. Con una que posea un rápido ataque o ascenso y una caída muy leve y lenta, uniéndola además a una de tono, podemos obtener un interesante efecto de «boing boing» (19.6.21).

```
10 ENV 1,=14,100
20 ENT -1,26,-1,1,5,6,1
30 SOUND 2,60,1000,0,1,1
```

■ 19.6.21.

Otro buen resultado obtendremos con notas altas, y rápidos ataques y caídas, además de celéricos cambios de tono. Una vez más, debemos tener cuidado con el tiempo. Como siempre, podemos jugar con el número de tono en la instrucción SOUND. Finalmente, podemos intentar escuchar algunas campanas. Una nota de este tipo necesita un agudo ataque y un largo y leve descenso. Escuchemos el siguiente ejemplo (19.6.22).

```
10 ENV 1,1,15,1,5,0,5,15,-1,4
20 ENT 1,4,0,4,10,-1,1,10,0,2
30 SOUND 2,800,0,0,1,1
```

■ 19.6.22.

MÁS DIFÍCIL TODAVÍA

Qué tal si para terminar combinamos una envolvente de amplitud, de tono, ruido y tono en un sólo sonido. Así, acabaremos con estos cuatro programas. En el primero (19.6.23), podemos escuchar una combinación de silbidos y una especie de traqueteo. El segundo programa (19.6.24) puede resultar mucho más interesante para los creadores incansables de juegos de extraterrestres.

```
10 ENV 1,3,5,1,10,0,5,5,-3,1
20 ENT 1,200,-1,1
30 SOUND 2,30,0,0,1,1,1
```

■ 19.6.23.

```
10 ENV 1,15,1,1,10,0,10,1,-15,2
20 ENT -1,1,5,1,2,-5,1,1,5,1
30 SOUND 2,30,0,0,1,1,1
```

■ 19.6.24.

Si estamos realizando un programa tipo Hitchcock, podemos ejecutar un buen sonido que exterioriza la gradual aproximación de algo realmente siniestro (19.6.25). Y para muchos de nuestros mejores programas de juegos de acción trepidante, utilicemos unos impresionantes tiros con eco, que podremos escuchar con el cuarto programa (19.6.26).

```
10 ENV 1,15,1,100
20 ENT -1,2,2,2,4,-2,2,2,2,2
30 SOUND 2,50,0,0,1,1,1
```

■ 19.6.25.

```
10 ENV 1,1,15,1,15,-1,10
20 ENT -1,2,10,2,4,-10,2,2,10,2
30 SOUND 2,200,0,0,1,1,31
```

■ 19.6.26.

APENDICE I

DICCIONARIO MUSICAL



escribir música no es fácil. Existen gran cantidad de términos cuya función es describir cómo es una melodía. Si queremos transformar una tonadilla del pentagrama a nuestro ordenador, encontraremos muchos de estos términos que nos ayudarán a elegir sin errores, el tiempo, el tono y el volumen.

A

- ACELERANDO: De lento a rápido.
- ACCIDENTE: Signos que no forman parte del pentagrama, pero que transforman el tono de las notas.
- ACOMPAÑAMIENTO: Notas que son interpretadas para que armonicen con la melodía principal.
- ACORDE: Conjunto de tres o más sonidos simultáneos, combinados armónicamente.
- AD LIB: Quiere decir que los tiempos fijados pueden dejar de ser respetados.
- ADAGIO: A poca velocidad.
- AIRE: Tonada, melodía.
- ALLEGRETTO: Algo más lenta que el Allegro.

- ALLEGRO: Deprisa pero ordenado.
- ANACRUSIS: Nota o notas que en la parte débil de un compás preceden al comienzo de un ritmo.
- ANDANTE: Marcha ni lenta ni rápida.
- ANDAMENTO: Movimiento y desarrollo de una composición musical.
- ANIMATO: Animado y alegre.
- APOYATURA: Nota pequeña y de adorno, cuyo valor se toma de la nota siguiente.
- ARMADURA: Piezas con que se arman algunos instrumentos.
- ARMONICO: Cada uno de los sonidos, más agudos que el fundamental, que acompañan naturalmente a éste y le dan un timbre especial.
- ARPEGGIO: Sucesión de sonidos de un acorde.
- ATACAR: Producir un sonido de manera que sobresalga entre los demás.
- ATONAL: Música concebida sin sujeción a una tonalidad determinada.

B

- BAJO: Sonido grave.
- BARRA: Línea vertical que corta el pentagrama para indicar la división de compases.
- BEMOL: Nota cuya entonación es un semitono más baja que su sonido natural.
- BIS: Se emplea para dar a entender que una cosa se ha de repetir.
- BREVE: Paradójicamente, nota muy larga, equivalente en su valor al de dos compases mayores.

C

- CADENCIA: Manera de terminar una frase musical. Ritmo. Resolución de un acorde disonante sobre otro consonante.
- CALDERON: Signo que representa la suspensión del movimiento del compás. Improvisación que se ejecuta durante la suspensión de un compás.
- CANON: Composición de contrapunto en que sucesivamente van entrando las voces, repitiendo cada una el canto de la que le antecede.

- CANTABLE: Que se canta despacio.
- CIFRA: Notación musical por medio de números, es decir, nombrar las notas con números.
- CLAVE: Signo que se pone al principio del pentagrama para determinar el nombre de las notas.
- COMA: Cada una de las cinco partes en las que se divide un tono.
- COMPAS: Cada uno de los períodos de tiempo iguales en que se marca el ritmo de una composición musical.
- COMPASAR: Dividir en compases las composiciones.
- CON BRIO: Enérgicamente.
- CONTRAPASO: Segundo paso que canta una voz cuando otra canta el primero.
- CONTRAPUNTO: Concordancia armoniosa de voces contrapuestas.
- CROMATISMO: Sistema musical que procede de semitonos.

D

- DA CAPO: Repitiendo desde el principio.
- DIAPASON: Intervalo que consta de cinco tonos.
- DIESI: Sostenido.
- DIGITAR: Emplear los dedos en ciertos instrumentos para la nueva ejecución de la obra.
- DIMENSION: Medida de los compases.
- DISCORDE: No estar acordes las voces de los instrumentos.
- DISONANCIA: Acorde con consonante. Sonido desagradable.
- DOMINANTE: Quinta nota de la escala de cualquier tono.

E

- ESCALA: Sucesión de las siete notas musicales.

F

- FABORDON: Contrapunto.
- FERMATA: Calderón.
- FIGURA: Nota musical.
- FILARMONIA: Pasión por la música.
- FUGA: Composición que tiene por base su tema y su contrapunto.

G

- GAMA: Tabla con que se enseña la entonación de las notas.
- GRUPETO: Conjunto de tres o más notas de adorno, que se ejecutan a expensas del valor correspondiente a la nota siguiente.
- GUIA: Voz que va delante en la fuga y a la cual siguen las demás.

L

- LARGO: Con ritmo lento.
- LEIMA: Uno de los semitonos utilizados en música griega.
- LIGADURA: Artificio con que se liga la consonancia con la disonancia.
- LLAVE: Clave.

M

- MAESTOSO: De movimiento pausado y solemne.
- MANO: Escala.
- MEDIANTE: Tercer grado de la escala.
- MODULACION: Pasar de unos tonos a otros según las reglas de la armonía.
- MORDENTE: Adorno musical formado por una o más notas que, antepuestas a la principal, se ejecutan rápidamente.

N

- NATURA: Escala natural del modo mayor.
- NEUMA: Signo que se empleaba en la antigua notación musical.
- NOTA: Cualquiera de los signos con que se representan los sonidos.
- NOTACION: Escritura musical.

P

- PARTITURA: Texto completo de una composición musical.
- PAUSA: Breve interrupción. Silencio.
- PAUTADA: Pentagrama.
- PAUTAR: Señalar en el papel las rayas necesarias para escribir las notas musicales.
- PENTAGRAMA: Conjunto de cinco rectas horizontales, paralelas y equidistantes, sobre las cuales se escribe la música.

- PRESTO: Con movimiento rápido.
- POLIFONICO: Combinación armónica de varias melodías o motivos simultáneos.

S

- SECO: Sonido corto y sin resonancia.
- SEMITONO: Cada una de las dos partes en las cuales se divide el intervalo de un tono.
- SINCOPIA: Sonido que comienza en una parte débil del compás y se prolonga hasta una parte fuerte.
- SINFONICO: Conjunto de voces y sonidos que suenan a la vez.
- SOLFA: Música.
- SOSTENIDO: Aquella nota cuya entonación excede en un semitono mayor a la que corresponde a su sonido natural.
- SUSPENSION: Prolongación de una nota que forma parte de un acorde sobre el siguiente, produciendo acorde.

T

- TONALIDAD: Sistema de sonidos que sirve de fundamento a una composición musical.
- TÓNICA: Primera nota de una escala musical.
- TONO: Mayor o menor elevación del sonido correspondiente a la mayor o menor rapidez de las vibraciones que lo producen.
- TREMOLO: Sucesión rápida de muchas notas iguales, de la misma duración.
- TRINO: Sucesión rápida y alternada de dos notas de igual duración, entre las cuales media la distancia de un tono o un semitono.

U

- UNISONO: Que tiene el mismo tono que otro sonido.

V

- VALOR: Duración del sonido que corresponde a cada nota.

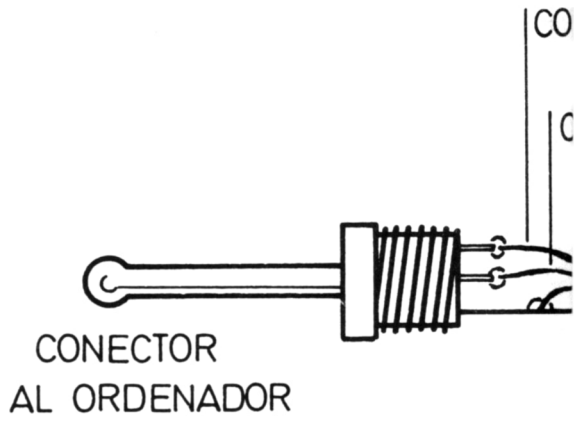
CONECTANDO NUESTRO AMSTRAD A UN SISTEMA HI-FI



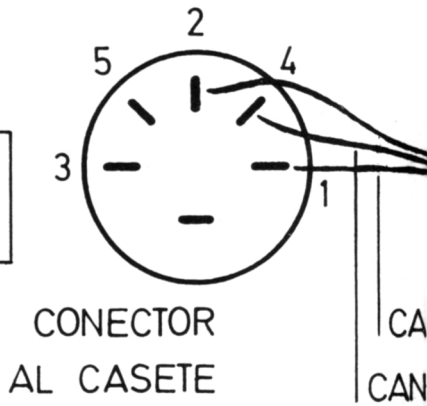
Es posible unir nuestro ordenador con un casete estéreo. Si este casete forma parte de una cadena de alta fidelidad, necesitaremos un conector adecuado. Conseguirlo no tendrá demasiado problema en una tienda especializada, o incluso podemos construirlo nosotros mismos, como podemos ver en el dibujo.

Tomamos un simple cable de estéreo, aislado, y con dos conectores. Uno ha de ser conectado al ordenador y el otro al casete estéreo. Encendamos el casete, coloquemos una cinta virgen, y pongamos el casete a grabar, pero no dejemos que la cinta avance, pulsando la tecla de pausa que suelen tener todos estos aparatos.

Ahora ejecutemos el programa de sonido en el ordenador, ajustemos el volumen del casete o del sistema Hi-Fi. Algunos casetes poseen un volumen automático, pero otros permiten ajustarlo manualmente. Cuando los controles de volumen están perfectamente ajustados, podemos ejecutar de nuevo el programa y grabarlo en una cinta, para después manipular el sonido grabado a nuestro antojo. Recordemos parar el casete cuando la música haya terminado, rebobinemos la cinta y tendremos un sonido «sensiblemente» mejor que el reproducido por el altavoz interno del ordenador.

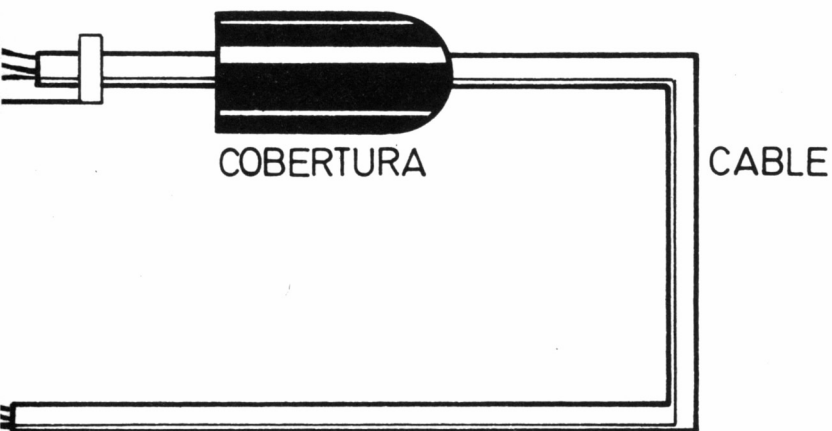


AMBOS CONECTORES
SON UNIVERSALES



CONEXION DEL CANAL DERECHO

CONEXION DEL CANAL IZQUIERDO



CANAL IZQUIERDO

CANAL DERECHO

E

l sonido ocupa un lugar preponderante dentro de la programación y los poseedores de un AMSTRAD CPC tenemos la suerte de disponer de un ordenador con auténtica potencia en este campo. Aprenderemos en este volumen de la GRAN BIBLIOTECA AMSTRAD qué es realmente el sonido, cómo se forma y de qué está compuesto, encontrando posteriormente fórmulas sencillas para componer música, acompañarla, producir armonía y efecto estereofónico. Todo ello, unido a la explicación concienzuda de todos los comandos BASIC referentes al sonido que utiliza nuestro AMSTRAD.

GRAN BIBLIOTECA
AMSTRAD

450 ptas.
(incluido IVA)

Precio en Canarias, Ceuta y Melilla: 435 ptas.