

MICROMANUALES

**Proyectos de
música con
microordenadores**

R. A. Penfold



ANAYA
MULTIMEDIA

Proyectos de música con microordenadores

Proyectos de música con microordenadores

R. A. Penfold



MICROMANUALES

Título de la obra original :
COMPUTER MUSIC PROJECTS

Traducción de: Antonio Resines
Diseño de colección : Narcís Fernández

Reservados todos los derechos. Ni la totalidad ni parte de este libro puede reproducirse o transmitirse por ningún procedimiento electrónico o mecánico, incluyendo fotocopia, grabación magnética o cualquier almacenamiento de información y sistema de recuperación, sin permiso escrito de Ediciones Anaya Multimedia, Sociedad Anónima.

Copyright © 1986 by Bernard Babani (Publishing), LTD.

© 1986 EDICIONES ANAYA MULTIMEDIA, S. A.
Villafranca, 22. 28028 MADRID
Depósito legal: M. 23.609-1986
ISBN: 84-7614-101-7
Printed in Spain
Imprime: Anzos, S. A. - Fuenlabrada (Madrid)

Indice

Prefacio	7
1. Interfaces analógicos	9
Señales de control.	
Puertos de ordenador.	
Interfaz CV.	
Aumento de voltaje.	
Señal de puerta.	
Niveles de puerta.	
Puertos alternativos.	
Interfaz directo con el <i>bus</i> .	
Ajuste.	
<i>Software</i> .	
Secuenciación polifónica.	
Lector de teclado.	
Ajuste y utilización.	
Interfaces.	
2. Percusiones sintetizadas	69
Tambor de afinación fija.	
Configurador de envolvente.	
Tambor disco.	
Sonidos basados en el ruido.	
Sonidos metálicos.	
<i>Software</i> .	

3. Síntesis digital	95
Sintetizador digital.	
Grabación digital.	
Compander (Compresor-expansor).	
Preamplificador para micrófono.	
Eco.	
4. Interfaces MIDI.....	127
Control por ordenador.	
Datos en serie.	
Bytes de control.	
Modalidades.	
Omni (1).	
Poli (3).	
Mono (4).	
Interfaz MIDI.	
UART.	
<i>Software.</i>	

Prefacio

Los ordenadores tienen un gran potencial para revolucionar muchos aspectos de nuestras vidas, pero fuera de los ámbitos empresariales y de los juegos, su impacto, hasta el momento, ha sido un tanto limitado en la mayor parte de los campos. Una clara excepción a esto es el caso de la música electrónica. En los últimos años, muchos instrumentos y efectos electrónicos han cambiado hasta volverse casi irreconocibles con la introducción de circuitos digitales más y más complejos. Esto ofreció al usuario cada vez más prestaciones, pero afortunadamente no a un coste cada vez mayor, y los instrumentos de hoy presentan una magnífica relación calidad-precio. Por supuesto, no es esencial permanecer dentro de los límites de las “cajas negras”. Los instrumentos electrónicos modernos abren grandes horizontes a la experimentación. Un ordenador doméstico adecuado junto con algo de *hardware* hecho en casa y un instrumento electrónico de algún tipo, pueden representar un sistema inmensamente versátil, haciendo que el instrumento que se encuentra en el núcleo del sistema ofrezca un valor aún mayor a cambio de su precio.

El propósito de este libro es mostrar algunos de los modos en los que se puede utilizar un ordenador doméstico, con buenos resultados, para la producción

de música electrónica. Los temas cubiertos incluyen la secuenciación y su control por medio de interfaces analógicos y MIDI, los ordenadores como líneas de *delay* digital, y generadores de sonido controlados por ordenador. Hemos procurado mantener el contenido a un nivel accesible y tan desprovisto de jerga como ha sido posible, pero, necesariamente, hemos asumido que el lector tiene al menos cierta experiencia con el *software* y las técnicas de ordenador, y en la construcción de proyectos electrónicos sencillos. Este libro no está dirigido a los principiantes, pero, por otra parte, no es necesaria una gran comprensión del *hardware* y la programación de los ordenadores para construir y utilizar los circuitos aquí descritos.

R. A. PENFOLD

1

Interfaces analógicos

Incluso en los años predigitales los sintetizadores podían conectarse entre sí, de modo que un instrumento controlara a otro. Esto se lograba utilizando la puerta *in/out* y las entradas y salidas de CV. Aunque su objeto no era suministrar un método de control por ordenador, estas entradas permiten de hecho esta función. Existen ahora muy pocos instrumentos nuevos que tengan entradas convencionales de puerta y CV (voltaje de control), pero hay multitud de estos instrumentos disponibles (y a buen precio de segunda mano), si es que aún no se dispone de uno. Tanto un sintetizador polifónico como uno monofónico, en conjunción con un ordenador doméstico y un interfaz apropiado, puede producir excelentes resultados. Las memorias de la mayor parte de los ordenadores domésticos pueden acumular largas secuencias de notas. Alternativamente, si tenemos presente que la mayor parte de los sintetizadores pueden generar sonidos de percusión bastante aceptables, utilizando un programa de lazo cerrado breve, el sistema puede actuar como un generador de ritmo.

Señales de control

La señal de salida de la puerta tiene, normalmente, la forma de una señal lógica estándar de muy bajo voltaje (típicamente, 0,8 voltios o menos) en condiciones de espera, pero puede alcanzar +4 ó +5 voltios al tocar una tecla. Algunos sintetizadores tienen un generador de envolvente del tipo ataque/atenuación, en lugar de la variedad, más completa, ADSR (ataque, *delay*, *sustain* y *release*) y, en este caso, normalmente las entradas y salidas de puerta han sido reemplazadas por salidas y entradas de *trigger* o disparador. La diferencia entre los dos tipos de señal consiste en que la señal de puerta se vuelve positiva durante todo el tiempo que se mantenga apretada una tecla, mientras que la señal de *trigger* sólo se hace positiva unos breves milisegundos cada vez que se pulsa una tecla. La duración de esta pulsación carece de efecto alguno sobre la duración de los impulsos del *trigger*. A menudo, las salidas de éste pueden funcionar razonablemente, bien conectándolas a entradas de puerta, o viceversa, aunque no se obtendrán necesariamente resultados útiles en todos los casos. No obstante, en lo que al control por ordenador se refiere, no hay dificultad alguna, y un ordenador puede generar un impulso de control adecuado para su uso con cualquiera de los dos tipos de entrada.

No todos los sintetizadores utilizan niveles lógicos estándar de 5 voltios, aunque a veces responden adecuadamente a las señales de este voltaje. Con todo, en algunos casos, las señales de 5 voltios procedentes del ordenador, deben convertirse en niveles lógicos de 0 y +15 voltios (o los que fuere) para obtener un disparo fiable. Volveremos sobre este tema más adelante.

El voltaje de control adopta dos formas estándar. En algunos sintetizadores (fundamentalmente, los más

antiguos), los VCO son de tipo lineal. En otras palabras, existe una relación lineal entre el voltaje de control y la frecuencia de salida. Por ejemplo, si 1 voltio produjera una salida de 100 Hz, 2 voltios producirían 200, 3 producirían 300, y así sucesivamente. La mayor parte de los sintetizadores de fabricación reciente (probablemente todos) utilizan VCO (osciladores controlados por voltaje) logarítmicos, y el sistema estándar tiene establecida una ley de 1 voltio por octava. Por ejemplo, si 1 voltio produjera una salida de una frecuencia de 100 Hz, 2 voltios producirían 200 Hz, 3 voltios 400, 4 voltios 800, y así sucesivamente.

Este segundo método resulta muy conveniente desde el punto de vista del control por ordenador. El teclado genera una serie de voltajes de control, utilizando una serie de resistencias en un circuito atenuador, y todas las resistencias tienen el mismo valor. La disposición básica utilizada es la que se muestra en la figura 1, aunque en ella sólo se utilizan unos pocos interruptores y resistencias, mientras que, en la práctica, normalmente habría un mínimo de alrededor de treinta de cada uno de ellos. La fuente de voltaje variable se ajusta de modo que el teclado dé el voltaje correcto para cada nota, es decir, 1 voltio por cada doce notas. Esto viene a representar 83,33 milivoltios por nota.

Un convertidor de digital en analógico produce un voltaje de salida que aumenta de modo lineal y, por consiguiente, produce una salida de voltajes esencialmente idénticos a los del teclado. De hecho, un convertidor digital-analógico produce típicamente una salida que aumenta en pequeños pasos, con unas resoluciones que, comúnmente, son de unos 10 milivoltios. No obstante, puede utilizarse un amplificador de corriente continua de ganancia variable para aumentar los voltajes de salida, igualándolos a los suministrados por el teclado. Como cuestión de inte-

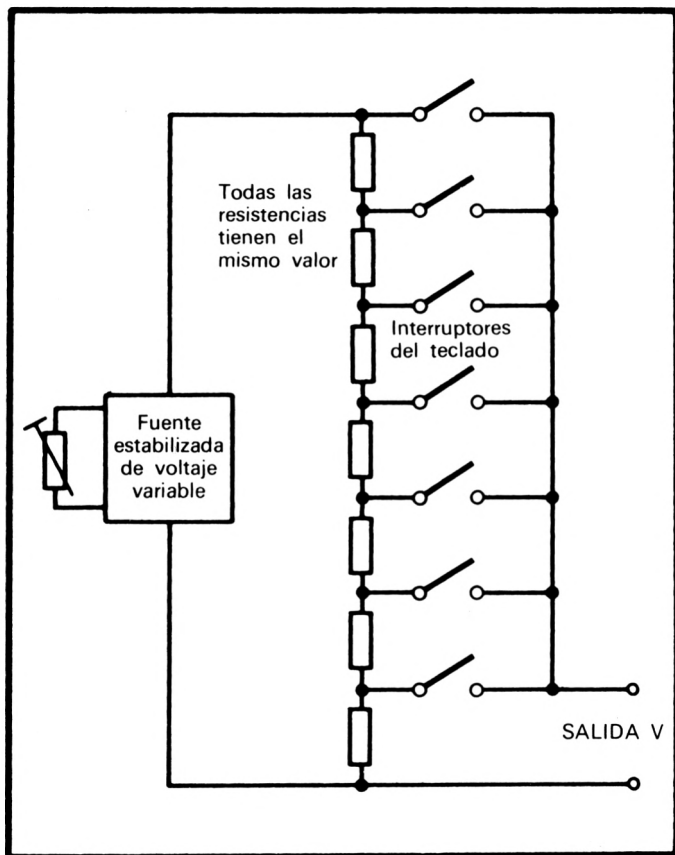


Figura 1.—Método básico para obtener CV a partir de un teclado, algunos sintetizadores analógicos utilizan un teclado “barrido digitalmente”. Con este tipo de instrumento, el circuito del teclado no dispone del circuito divisor de potencial estándar, sino que es barrido por un circuito digital que detecta qué tecla (en su caso) ha sido pulsada, y después alimenta el número apropiado a un convertidor digital-analógico. Entonces el convertidor genera un voltaje apropiado y

lo envía a los VCO. Esto puede parecer una forma innecesariamente compleja de hacer las cosas, pero los sintetizadores de este tipo tienen a menudo alguna forma de control digital incorporada, aunque sólo sea de un tipo muy básico, y no utilizan este sistema por capricho. A veces es posible controlar un sintetizador directamente a través de algún puerto apropiado para ordenador sin necesidad de interfaz alguno, pero esto probablemente implique hurgar en el interior del ordenador y, a menos que se esté muy seguro de lo que se está haciendo, lo más fiable es controlar la unidad a través de la entrada de CV y un interfaz apropiado.

Los sintetizadores que tienen un CV de características lineales, no pueden controlarse con facilidad desde un ordenador utilizando el tipo de convertidores analógico-digitales accesibles a los aficionados. Estos instrumentos no tienen incrementos iguales de voltaje de una nota a la siguiente, sino que va en aumento al crecer la frecuencia de salida. En teoría, debería ser posible obtener resultados satisfactorios utilizando un convertidor ordinario y un transformador lineal-logarítmico de voltaje, pero yo personalmente jamás he conseguido obtener buenos resultados utilizando este método y un par de colegas que experimentaron con él, no tuvieron más éxito. Es un tema que pasaremos por alto en el libro.

Puertos de ordenador

El tipo exacto de interfaz requerido depende del tipo de puerto de ordenador al que debe ir conectado aquél. Con ordenadores como el VIC-20, las cosas resultan sencillas, ya que ambos tienen un puerto apropiado para el usuario, con líneas de entrada y salida. Lo contrario se aplica a los aparatos que tienen un puerto de expansión, en donde los mecanismos

deben conectarse en un interfaz directo sobre los *buses* del ordenador. En el caso de algunos ordenadores (por ejemplo el Sinclair ZX81 y el ZX Spectrum), existen puertos paralelos añadibles que suministran una serie de entradas y salidas digitales (por lo menos en las fechas en las que se escribió esto). Si se dispone de un ordenador con un único puerto de expansión, pero existe en el mercado una placa de interfaz en paralelo, recomendaría el uso de ésta, ya que, casi con seguridad, hará que la conexión de la unidad con circuitos de música resulte mucho más directa. Si sólo se tiene una ligera experiencia en electrónica e interconexión de ordenadores, **sugeriría encarecidamente** no conectar circuitos por interfaz directo a los *buses* del ordenador. Más prudente es utilizar un ordenador con una puerta de uso incorporada, o una placa de interfaz en paralelo, en aquellos que no la tengan.

No es posible ofrecer aquí una exposición en profundidad sobre los interfaces de todos los ordenadores domésticos producidos hasta el momento, pero sí ofreceremos unos cuantos elementos básicos acerca de las conexiones por interfaz de algunos de los ordenadores más populares (y para nuestros fines, los más apropiados). Empezamos por el VIC-20, que tiene las conexiones de puerto de la figura 2 (a). La línea de +5 voltios es importante, y es la que se puede encontrar en la mayor parte de los ordenadores. Es importante porque permite que muchos tipos de interfaces y periféricos sean alimentados desde el ordenador sin necesidad de otras fuentes de alimentación. Las otras líneas de interés van de la PB0 a PB7. Estas son líneas de entrada/salida y actúan como pestillos cuando se emplean como salida.

Al utilizar estas líneas, la primera tarea es establecerlas como entradas o salidas, según se requieran. No hay por qué ponerlas todas en la misma modalidad, y cada una de ellas es programable

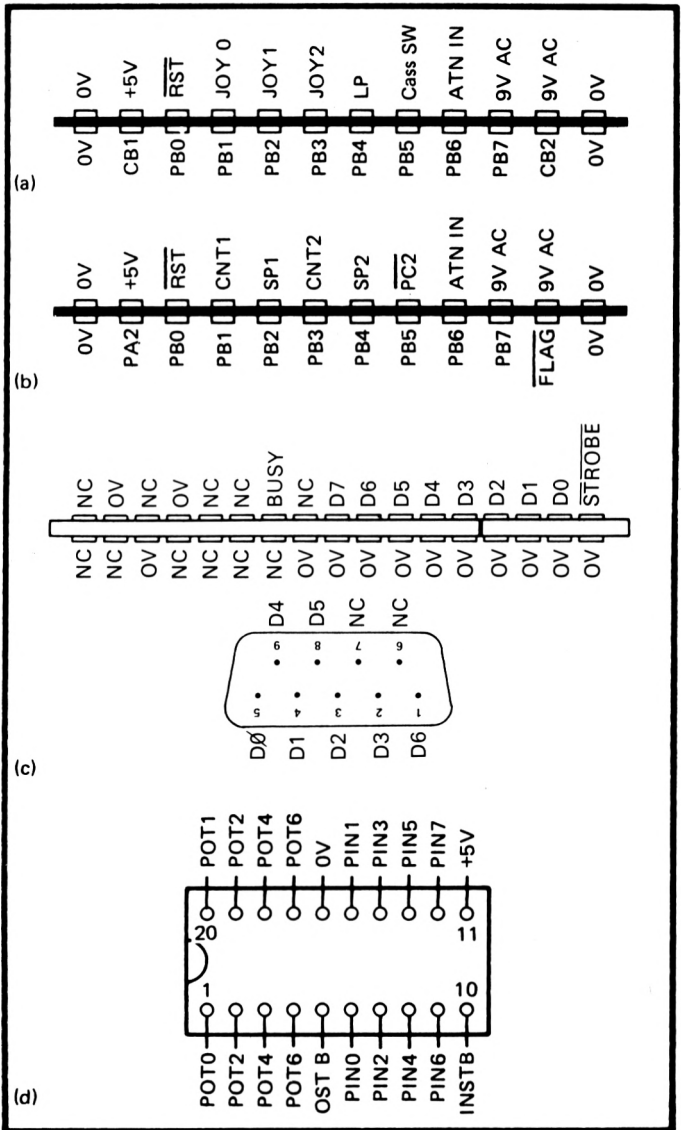


Figura 2.—Detalle de los puertos de usuario de varios ordenadores

individualmente como entrada o salida. El registro de dirección de datos se utiliza para establecer la modalidad operativa deseada para cada una de las líneas; en el VIC-20 se encuentra en las coordenadas 36138. Cada bit del registro de dirección de datos controla la línea correspondiente del puerto para el usuario. Fijar en uno 1 bit designa como salida a su línea; si se fija en 0, el resultado es una entrada. Para el que no esté familiarizado con el sistema de numeración binario, esto puede resultar un tanto confuso, pero la tabla ofrecida a continuación debería servir para simplificar las cosas.

Línea	PB0	PB1	PB2	PB3	PB4	PB5	PB6	PB7
Núm.	1	2	4	8	16	32	64	128

Esta muestra el número que debe mandarse (en sistema numérico decimal ordinario) al registro de dirección de datos para fijar cada línea como salida. El número necesario para establecer una línea como entrada es siempre cero. Las líneas se controlan simultáneamente enviando un valor único al registro de dirección de datos. Este número es fácil de calcular y no se trata más que de averiguar, en un principio, cuáles son los ocho números necesarios para fijar cada línea es el estado deseado, y de sumarlos después. Racionalizando un poco, sólo hay que decidir qué líneas han de ser salidas, buscar su número en la tabla de arriba y sumar la totalidad. En muchos casos se fijarán todas las líneas, bien como entradas (un valor DDR de 0) o como salidas (valor DDR 255). El valor se escribe sobre el registro de dirección de datos utilizando la instrucción POKE (por ejemplo, la instrucción POKE 37138,255 fijaría todas las líneas como salidas). Incidentalmente, al principio, todas las líneas están en modalidad de entrada.

Un punto importante a tener en cuenta es que el registro de dirección de datos sólo se emplea para establecer que las líneas sean entradas o salidas, y no se utiliza al escribir datos para las líneas o al recibirlos por ellas. Esto se logra empleando el registro periférico, que en el Vic-20 se encuentra en las coordenadas 37136. El escribir un 1 en un bit del registro periférico pone la línea correspondiente en alta (alrededor de +5 voltios), y escribir 0 la pone en baja (alrededor de 0 voltios). Si no se está familiarizado con el sistema binario de numeración, puede usarse una vez más la tabla de antes para averiguar el valor requerido para fijar las líneas en los estados deseados cuando se emplean como salidas. Por ejemplo, para que de PB0 a PB3 estén en alta y las demás líneas en baja, será necesario un valor de 15 ($1+2+3+4+8=15$). Desde BASIC, "POKE 37136,15" establecería los estados de salida requeridos. Al leer las líneas de *input*, se usa la función PEEK (por ejemplo, PRINT PEEK(37136)).

La figura 2 (b) muestra los detalles de las conexiones del puerto del Commodore 64. Es muy similar al puerto del VIC-20 y, de hecho, si sólo se precisa una alimentación de 5 voltios y una entrada/salida de 8 bits, es idéntica. Incidentalmente, las conexiones a ambos ordenadores se efectúan por medio de conectores hembra de 2 por 12 vías y 0,156 pulgadas. Es improbable que se pueda encontrar un conector adecuadamente polarizado, por lo que es importante asegurarse de que el conector quede insertado en la dirección correcta. Nótese que, aunque los ordenadores VIC-20 y el Commodore 64 tienen lo que a todos los efectos son puertos compatibles con *hardware* para el usuario, son ligeramente diferentes desde el punto de vista del *software*. Las únicas diferencias significativas para nuestros actuales propósitos son el registro de dirección de datos y las

coordinadas de los registros periféricos. Estas son 56579 y 56577, respectivamente.

Los ordenadores AMSTRAD tienen un puerto de usuario un tanto característico, ya que funcionalmente no es más que una ampliación de la matriz lógica del teclado.

No es un puerto programable para funcionar como entrada o salida. Por construcción, sólo admite entradas de datos. Concretamente, y para nuestro propósito, se pueden utilizar siete de sus bits de entrada, como pequeño puerto de entrada de información.

Desde código máquina podremos leer dicha información acudiendo a la rutina del *firmware* situada en &BB24. Desde BASIC, podemos consultar esta entrada con los valores proporcionados por la función JOY.

Como registro de salida, lo más útil será utilizar el puerto de la impresora, de cual podemos aprovechar siete bits para la información de salida. Desde código máquina escribiremos en este puerto mediante la rutina del *firmware* situada en &BD31. Desde BASIC, el envío de datos al puerto de impresora es realmente sencillo (PRINT#8,X), asegurándonos antes haber puesto la línea BUSY a nivel bajo.

El puerto de usuario utiliza un conector de nueve patillas del tipo SUB-D; para el puerto de la impresora es necesario un conector de tarjeta de 2*17 patillas, con un ancho de 2,54 mm cada uno. En la figura 2 (c) vemos los detalles de estos conectores.

No todos los ordenadores tienen un puerto para el usuario que sea un equivalente directo de los que poseen Commodore y AMSTRAD. Los ordenadores MTX500 y MTX512 tienen un puerto para el usuario en forma de un soporte DIL para circuito integrado de 20 puntas sobre la placa de circuito impreso. Una ranura existente en la parte trasera de la carcasa permite el paso de un cable de cinta hacia el mundo exterior. No me ha sido posible encontrar un

vendedor de conectores DIL de 20 puntas, pero a menudo basta con uno de 14 ó 16 para realizar las conexiones a los terminales de interés. En caso contrario, dos conectores de 14 puntas, recortados cuidadosamente hasta reducirlos a 10 cada uno, dan buen resultado. Los detalles de conexión al puerto del usuario del MTX aparecen en la figura 2 (d).

La diferencia evidente desde el punto de vista eléctrico entre este puerto y los anteriormente descritos, es que el puerto del usuario del MTX tiene líneas de entrada y de salida independientes, y no un único juego de ocho líneas de doble uso. Como era de esperar, las salidas son de POT0 a POT7 y las entradas, de PIN0 a PIN7. Esta disposición es, potencialmente, más versátil que tener ocho líneas de doble uso, pero no ocurre necesariamente así, ya que los puertos para el usuario del MTX carecen de algunas de las líneas adicionales que aparecen en los otros aparatos discutidos anteriormente, y esto puede ser una desventaja en el caso de algunas aplicaciones.

Aparte de las diferencias en el soporte mecánico, existe también una acentuada diferencia desde el punto de vista del *software*. Los ordenadores MTX y AMSTRAD están basados en el Z80, mientras que los Commodore utilizan un microprocesador 6502 (o, para ser más precisos, el 6510, muy similar, en el caso del Commodore 64). En el caso del 6502, los sistemas de entradas/salidas se establecen en la memoria y se puede leer o escribir sobre ellos, usando PEEK y POKE, como si fueran coordenadas ordinarias de la memoria. El microprocesador Z80 tiene memoria y mapas de entrada/salida separados y, ya se use código de máquina o BASIC, son necesarias instrucciones aparte para controlar cada tipo de circuito. Con los ordenadores MTX la instrucción BASIC OUT permite transferir datos a circuitos externos, mientras que la función INP permite la lectura de las señales enviadas a las entradas. Ya lea o

escriba en el puerto del usuario, se encuentra en la coordenada 7 (es decir, OUT7,X para introducir el valor X sobre las líneas de salida, y PRINT INP(7) para leer las líneas de entrada). Nótese que las ocho líneas de salida son del tipo de triple estado y que el OSTB (estrobo de salida) debe conectarse al terminal de 0 voltios para llevar a las salidas a su estado activo.

Esta breve descripción de los puertos de cuatro ordenadores debería servir para hacernos una idea de cómo se utilizan los puertos para paralelos. Cualquiera que sea el ordenador que se pretenda utilizar como centro de control del sistema de música electrónica, es importante averiguar todo lo que sea posible acerca de él. No solamente acerca de cómo conectarlo a circuitos externos, sino también acerca de la cuestión del *software*, preferiblemente incluyendo programación en código máquina. Sería justo decir que, cuanto mayor sea la comprensión del ordenador que se utilice, mejor podrá desempeñar las funciones que de él esperamos. Dejaremos el tema de los puertos de ordenador por el momento, pero más adelante, en este mismo capítulo, abordaremos métodos de interconexión a ordenadores sin puertos o con un puerto añadible.

Interfaz CV

Una vez resuelto el problema de cómo introducir o extraer datos de un ordenador, llegamos al problema de convertir datos digitales de salida en señales analógicas que puedan ser utilizadas para controlar la entrada de CV de un sintetizador. Los convertidores digital-analógicos son dispositivos considerablemente complejos que requieren algunos componentes de precisión para poder funcionar bien. Afortunada-

mente, existen ya en el mercado varios circuitos convertidores integrados de buena calidad y, a pesar de su costo razonablemente bajo, dan excelentes resultados. La gama Ferranti de convertidores es el tipo más fácil de encontrar, y utilizaremos aquí uno de ellos. El convertidor en cuestión es el ZN428E, que se utiliza en el circuito de interfaz mostrado en la figura 3.

IC1 es el *chip* convertidor del ZN428E y es un tipo R-2R estándar. Consiste, básicamente, en una fuente de voltaje de referencia de precisión que acciona una red compleja de resistencias. El voltaje de salida se extrae de ésta por medio de ocho interruptores electrónicos. Esto, de hecho, es análogo a los circuitos de teclado que aparecían en la figura 1, pero mientras que el circuito del teclado tiene un interruptor por nota, en este caso, tan sólo ocho de ellos nos permiten obtener hasta 255 notas diferentes. Esto, por supuesto, se logra seleccionando la mayor parte de las notas cerrando más de un interruptor. Es por este motivo por el que es más necesaria una red mucho más compleja de resistencias (llamada R-2R), y por lo que un atenuador sencillo del tipo que aparece en la figura 1 no resultaría suficiente. La red se conoce como R-2R simplemente porque emplea sólo dos valores de resistencia, en los que el valor de los componentes de mayor resistencia es exactamente el doble que el de los de valor más bajo.

Arriba afirmábamos que existen hasta 255 notas disponibles, pero esto es una extensión excesiva y poco práctica, y las dos entradas más insignificantes del IC1 se conectan simplemente a tierra. Esto suministra una entrada de 6 bits y una extensión de 63 notas, poco más de cinco octavas, en otras palabras. De hecho, es una extensión superior a la que pueden ofrecer la mayor parte de los sintetizadores desde su teclado, pero la mayor parte de los instrumentos

suministran las cinco octavas completas cuando se utilizan con una fuente de CV externa.

El IC1 tiene terminales de tierra analógicos y digitales separados, y no tienen por qué estar exactamente al mismo voltaje. No obstante, en este caso, se conectan ambos directamente a tierra y operan con el mismo potencial. La referencia de voltaje incorporada del IC1 es del tipo de potencial de referencia de 2,25 voltios, y requiere los componentes separados C1 y R1 para funcionar en condiciones. El voltaje de salida a fondo de escala es igual al potencial de referencia de 2,25 voltios y, por consiguiente, aumenta según incrementos de unidades de 10 milivoltios (0,01 voltios). Dado que no estamos empleando las entradas menos significativas de este circuito, la resolución queda reducida a pasos nominales de 40 milivoltios. Esto es ligeramente inferior a la mitad de lo requerido, pero el amplificador con base en el IC2 suministra el aumento de voltaje requerido.

El amplificador es un circuito amplificador operacional sin inversión. Normalmente es necesaria una alimentación de -5 voltios para que un amplificador operacional pueda suministrar voltajes a la línea de alimentación de 0 voltios. Esto no es necesario aquí, ya que el CA3140E especificado para el IC2 tiene una etapa de salida que puede suministrar voltajes directamente a la línea de 0 voltios. En la práctica, se utilice o no una alimentación negativa, existen errores de resalte (offset) que producen cierta falta de precisión a voltajes bajos. VR1 es un control nulo de resaltes que puede emplearse para recortar estos errores. VR2 es un control de ganancia de voltaje, y está ajustado para suministrar una escala adecuada desde el circuito. IC2 suministra una impedancia de salida relativamente baja, y debería ser capaz de mover cualquier entrada de CV sin problemas de carga.

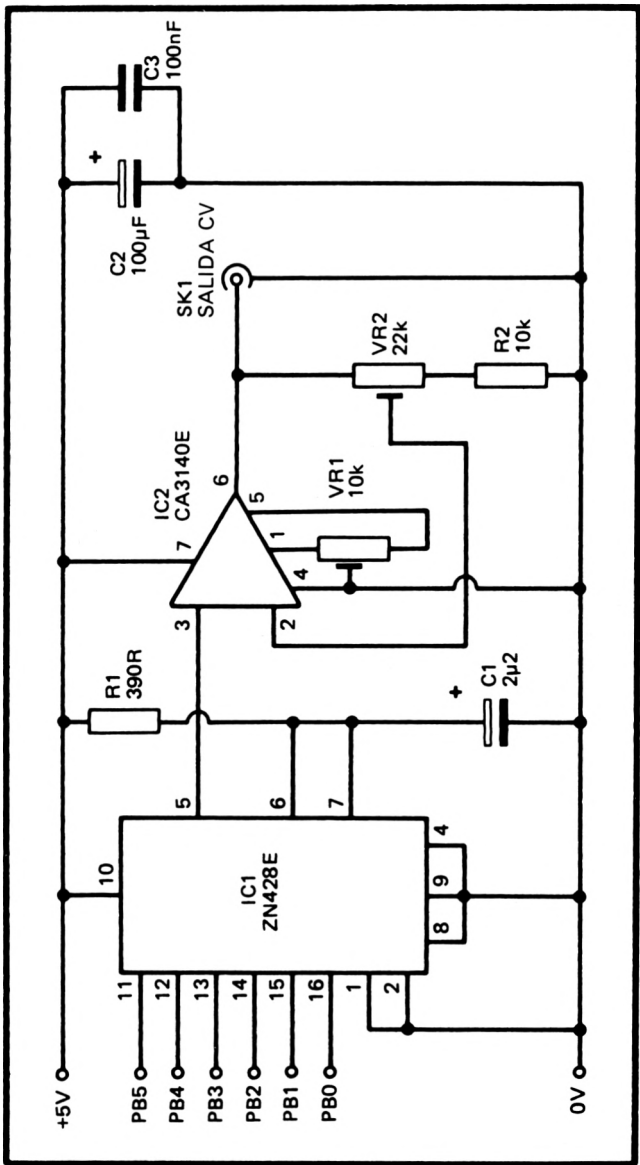


Figura 3.—Circuito simple para obtener voltaje de control a partir de un ordenador

En la figura 3 se ha asumido que la unidad será controlada desde el puerto de un VIC-20, un CBM 64 o un AMSTRAD, y las conexiones al interfaz se han señalado, por consiguiente, como "PB0", "PB1", etc. No obstante, la unidad puede moverse desde las seis salidas menos significativas de cualquier puerto normal para usuario o para paralelos (POT0 a POY5 en un ordenador MTX, por ejemplo). No es esencial utilizar los seis bits menos significativos, ni siquiera seis bits consecutivos, pero es recomendable hacerlo, ya que esto ayuda a que el *software* sea lo más directo posible.

Componentes del generador de CV (Versión completa. Figuras 3, 4 y 5)

Resistencias (de 1/4 W 5%)

R1	390R
R2,3	10k(2)
R4	100k
R5	1M
R6	22k
VR1	10k preajustada
VR2	22k preajustada
VR3	2M2 lin

Condensadores

C1	2 μ 2 63V electrolítico
C2	100 μ F 10V electrolítico
C3	100nF cerámico
C4	15nF poliéster
C5,6	470 μ F 16V electrolíticos (2)
C7,8	220 μ F 16V electrolíticos (2)
C9	330nF carbonato

Semiconductores

IC1	ZN428E
IC2	CA3140E
IC3	NE555
IC4	4001BE
D1,2,3,4	1N4002(4)

Miscelánea

SK1 Entrada de clavija estándar
Zócalo de IC DIL de 16 patillas
Zócalo de IC DIL de 14 patillas
Dos soportes de IC DIL de 8 patillas
Carcasa, placa de circuito, cable, etc.

Aumento del voltaje

Existe una ligera falla en el circuito básico de la figura 3, ya que no puede aportar la extensión completa de 63 notas. Algo más de cinco octavas a un voltio por octava requiere, obviamente, una salida de más de 5 voltios, que no puede obtenerse con una alimentación de 5 voltios. La etapa de salida del IC2 dará un voltaje máximo de alrededor de 3 voltios, con una alimentación positiva de 5 voltios. Tal vez las aproximadamente tres octavas que pueden obtenerse así sean suficiente para nuestras necesidades, pero puede que prefiramos utilizar un potencial de alimentación positiva más elevado para el IC2, obteniendo así la extensión completa de notas. Si el ordenador que se está empleando puede suministrar un voltaje de alimentación de alrededor de entre +9 y

Este utiliza el bien conocido circuito temporizador integrado 555 en modalidad oscilador, con una frecuencia operativa de unos pocos cientos de Hertzios. R3, R4 y C4 son los componentes temporizadores, y producen una salida que es virtualmente una onda cuadrada con una relación de 1 a 1. C5, D1, D2 y C7 forman un circuito rectificador y atenuador que genera un potencial de CC de alrededor de 3 ó 4 voltios. El voltaje general por este circuito es algo inferior al voltaje de alimentación de 5 voltios debido a las pérdidas que se producen en los rectificadores y en la etapa de salida del IC3. Este voltaje está conectado en serie con suministro de 5 voltios, lo que produce alrededor de 8 voltios en C7. C6, D3, D4 y C8 forman otro circuito similar al anterior, y la salida está conectada en serie al potencial que atraviesa C8. Esto da un total de unos 12 voltios en C8, y es este voltaje el que se emplea para alimentar el IC2. En situación de carga, el voltaje que atraviesa C8 disminuye sustancialmente, pero sigue dejando un potencial de alimentación de alrededor de 8 voltios para el IC2, que es suficiente para que la unidad disponga de algo más de cinco octavas de extensión.

Señal de puerta

En la mayor parte de los casos, la generación del voltaje de control es el aspecto más difícil de este tipo de interfaces, mientras que la señal de puerta es relativamente sencilla. Si el sintetizador que vamos a emplear utiliza niveles lógicos estándar de 5 voltios, no debería haber dificultad alguna en utilizar una salida digital del ordenador para controlar directamente la entrada de puerta o disparador del instrumento. Al haber utilizado sólo 6 bits para accionar el convertidor digital-analógico, quedan dos

líneas de un puerto de 8 bits libres, y se puede utilizar cualquiera de ellos para suministrar la señal de salida de puerta.

Si el sintetizador tiene una entrada de disparador, la duración de la pulsación de disparo probablemente no sea crítica, pero lo normal es una cifra mínima de unos 2 milisegundos. Al controlar la salida de puerta por medio del BASIC, debido a la velocidad relativamente lenta de este lenguaje de ordenador, resulta improbable que se pueda generar una pulsación de disparo excesivamente breve. Al utilizar código máquina las cosas cambian, y sería necesario un lazo de temporización para alargar las pulsaciones de salida hasta una duración apropiada.

Si el sintetizador tiene una entrada de puerta en condiciones, la duración de la pulsación de salida es más importante, ya que debe simular la acción de una tecla. En muchas envolventes ADSR una duración fija de pulsación de unos 10 milisegundos será suficiente, pero para poder sacar el máximo partido al sistema es necesario hacer que el tiempo de puerta sea programable. Otro enfoque considerablemente sencillo, desde el punto de vista del *software*, consiste en utilizar un generador de pulsaciones para obtener las de la puerta. La duración de la pulsación de salida del ordenador se vuelve entonces irrelevante, ya que el tiempo de pulsación de la puerta queda establecido por medio de un control del circuito generador de pulsaciones. Puede ajustarse este control para obtener la forma de la envolvente y el efecto que se desee, sin necesidad de complicar el *software* con un lazo temporizador extra. Esto no permite la alteración del tiempo de la puerta durante una secuencia de notas. En la práctica, no suele tener gran importancia.

La figura 5 muestra el circuito de un generador de pulsaciones de puerta. Está basado en una puerta de entrada NOR CMOS 4001BE quad2 (IC4). IC4a y IC4b van conectados en una configuración multivi-

bradora monoestable CMOS estándar. Este circuito no es reajutable. En otras palabras, si la pulsación de entrada es más larga que la duración requerida para el pulso de salida del circuito, funcionará adecuadamente y no habrá problemas con los disparos múltiples o la extensión de la pulsación de salida. La duración de la pulsación viene controlada por los valores de los componentes de temporización VR3, R6 y C9. VR3 permite que la longitud de la pulsación se ajuste dentro de una extensión aproximada de 5 a 500 milisegundos (0,5 segundos).

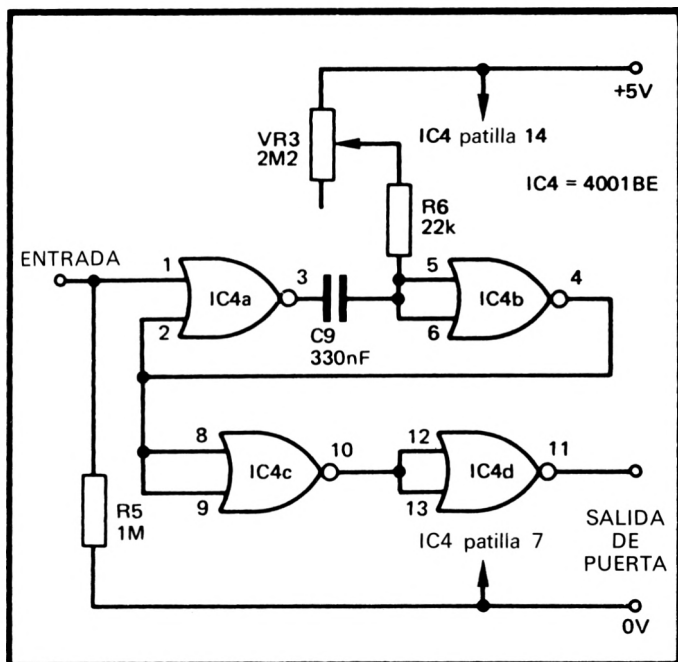


Figura 5.—Un generador de pulsos por puertas

El IC4c y el IC4d se utilizan como etapa amortiguadora sin inversión en la salida del circuito. R5 no

es más que una resistencia de protección que impide que las cargas estáticas dañen al IC4 si se deja la entrada en circuito abierto. En beneficio de aquellos que no estén familiarizados con los CMOS, tal vez debiéramos explicar que son vulnerables a los altos voltajes de energía estática que a menudo están presentes en los entornos modernos. Este tipo de dispositivos se entregan normalmente en algún tipo de paquete antiestático como la espuma conductora, y deben permanecer dentro del mismo hasta el momento de su instalación sobre la placa de componentes. Deben montarse siempre sobre soportes de circuitos integrados, y deben dejarse para el final, cuando todo el cableado y todos los demás componentes hayan sido ya instalados. En ese momento deben ponerse en su lugar con el mínimo de manipulación posible. Incidentalmente el CA3140E utilizado en el circuito interfaz CV tiene una etapa de entrada PMOS y, por consiguiente, requiere también estas precauciones en su manejo.

Niveles de puerta

Si el sintetizador de puerta que vamos a utilizar no requiere los niveles lógicos estándar de 5 voltios, puede resultar algo difícil conectarlo satisfactoriamente a un ordenador, aunque en teoría no existe motivo alguno que impida convertir niveles lógicos de 5 voltios en cualquier nivel de señal que se desee. Algunos sintetizadores que requieren niveles de señal de 0 a +15 voltios apenas funcionarán utilizando las señales estándar de 5 voltios, pero no es el caso de todos los sintetizadores. No es difícil conseguir un aumento del voltaje utilizando el circuito sencillo de un amplificador comparador operacional como el de la figura 6. No obstante, este circuito requiere una

alimentación de 15 voltios, y no es probable que podamos obtenerla del ordenador. Será necesario utilizar una alimentación por pilas de 15 voltios —por ejemplo, una batería de 6 voltios conectada en serie con una de 9— o habrá que construir un alimentador conectado a la red para el amplificador.

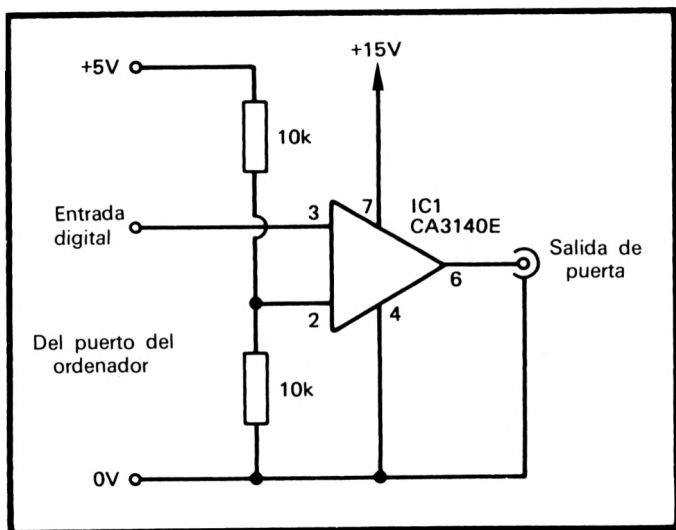


Figura 6.—*Conversión de nivel de 5 a 15 voltios*

No todos los sintetizadores requieren un voltaje de puerta. Algunos necesitan un cortocircuito a tierra para que se produzca el disparo. En estos casos se podría utilizar un interfaz de relé, pero el funcionamiento, en absoluto instantáneo, de los relés, puede producir resultados poco fiables. Es mejor utilizar algún tipo de interruptor electrónico y un transistor VMOS conectado del modo que muestra la figura 7. Da buenos resultados.

Los sintetizadores que requieren voltajes de puerta o disparo negativos son escasos, lo que es una suerte,

dado que éstos resultan difíciles de conectar a una salida lógica estándar. Dado que, como he dicho, son poco frecuentes y carezco de experiencia en su conexión, no serán tratados aquí.

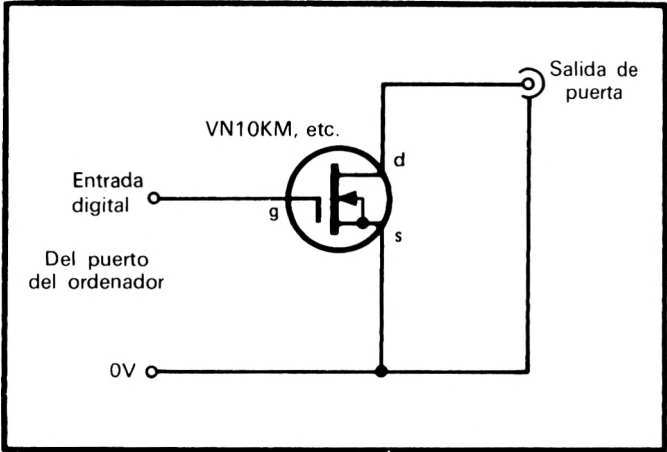


Figura 7.—Conversión de 5 voltios a puerta a tierra

Puertos alternativos

Hasta el momento no hemos considerado más que los interfaces realizados a través de un puerto para el usuario, o de algún tipo de puertos para paralelo añadibles. Existen alternativas dignas de breve mención aquí, aunque, en lo fundamental, probablemente lo mejor sea que sólo los más experimentados intenten realizar este tipo de interfaces.

Muchos ordenadores domésticos tienen una salida de ocho bits en forma de un puerto para impresora en

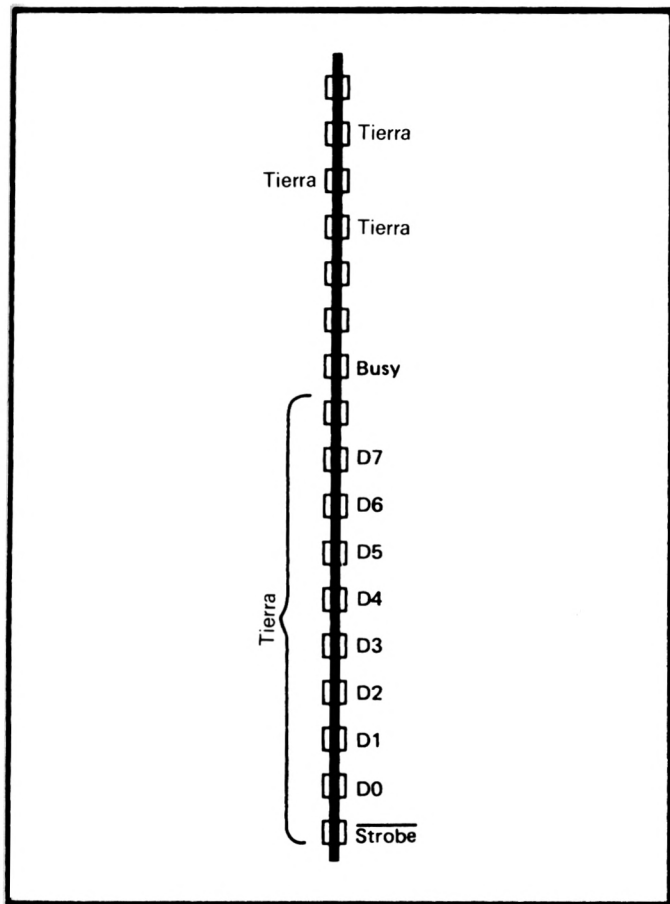


Figura 8.—Detalle de las conexiones para el Amstrad CPC 464-664-6128

paralelo, o de puertos para *joystick*. Si tomamos, en primer lugar, los puertos para impresora, el ordenador Amstrad CPC es un buen ejemplo. Este tiene las conexiones del puerto de la impresora detalladas en la figura 8. Para efectuar conexiones a este puerto es necesaria una patilla hembra de 2 por 17 vías y 0,1 pulgada de calibre (2,54 mm). Tal vez no sea posible obtener un conector de este tipo con el número correcto de vías, pero, con cuidado, es posible cortar un conector de mayor tamaño hasta el adecuado con una sierra de pelo.

Las líneas de interés en este caso son las D0 a D6, que pueden emplearse como siete salidas con pestillo (*latch*); de D0 a D5 serían útiles para accionar el convertidor digital-analógico, y D6 suministraría la señal de puerta o disparo. La mayor parte de los BASIC proveen instrucciones que permiten la escritura de datos al puerto de la impresora de byte en byte, pero, en la práctica, a menudo resulta más cómodo escribir los datos directamente sobre el *hardware* que suministra el puerto de la impresora. En este caso, se accede a las líneas de datos de la impresora por medio de un ingenio que se encuentra en las coordenadas &FE00. Por tanto, OUT(C),C con BC=&EF00,X escribirá el valor X sobre el interfaz. En la puerta de la impresora no hay línea de alimentación de +5 voltios, pero se puede obtener uno en el puerto de expansión.

Este sistema funcionará perfectamente bien con la mayor parte de los puertos para impresora, pero es justo decir aquí que existen una o dos excepciones. En algunos puertos de impresora en paralelo las líneas de salida se emplean también con otros fines y no se ponen en pestillo adecuadamente. Si se dispone de medios para monitorizar el estado lógico de la salida de línea de la impresora, es bastante fácil determinar si pertenece o no a esta categoría. Si la salida monitorizada es estática cuando no se están enviando

datos al puerto, las salidas deben ser del tipo con pestillo y no debería haber problema alguno para controlar el interfaz desde el puerto. Si existe una corriente continua de pulsaciones en la línea monitorizada, esto indica, sin lugar a dudas, que es inadecuada para la presente aplicación. El único puerto paralelo de este tipo con el que me he encontrado ha sido el del ordenador Oric 1, pero bien podría haber otros.

Los Atari 400, 600XL, 800 y 800XL tienen puertos para *joystick* que pueden emplearse como salidas digitales. Estos dos puertos suministran, cada uno, cuatro líneas (en los Atari 400 y 800, que tienen cuatro puertos para *joystick*, úsense los puertos 1 y 2). La figura 9 ofrece detalles sobre las conexiones de los dos puertos que, convenientemente, suministran también una salida de alimentación de 5 voltios.

Las ocho líneas de datos funcionan normalmente como entradas que leen cambios en cada *joystick*, pero las siguientes rutinas de *software* las ponen en modalidad de salida.

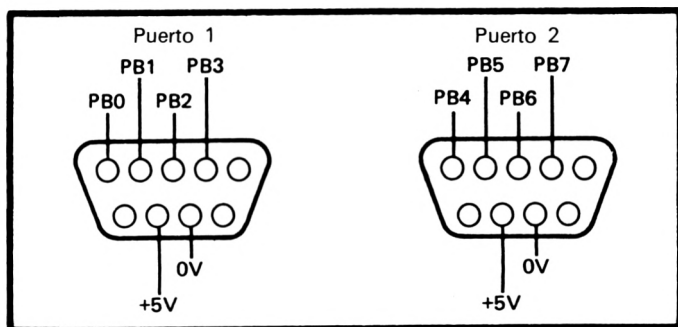


Figura 9.—Utilización de los puertos del Atari como salidas de 8 bits

POKE 54018,56
POKE 54016,255
POKE 54018,60

Seguidamente se escriben los datos para el puerto en las coordenadas 54016 utilizando la instrucción POKE (por ejemplo, POKE 54016,0 pondría de PB0 a PB7 en baja).

Interfaz directo con el *bus*

La mayor parte de los ordenadores tienen un puerto de salida de uso general que permite el acceso directo a los *buses* de coordenadas, datos y control del ordenador, además de otras líneas útiles, en la mayor parte de los casos. De hecho, en algunos ordenadores éste es el único puerto disponible para una expansión y el que hay que emplear en la presente aplicación. En la mayor parte de los casos no resulta difícil conectar dispositivos al puerto de expansión y, en algunos casos, resulta algo más complicado que utilizar un puerto para el usuario en condiciones. Con todo, tiene un par de inconvenientes en la medida en la que hace necesario un mayor conocimiento técnico para abordar con éxito este tipo de interfaces. Si se comete un error, existe el riesgo de producir daños en el *hardware* del ordenador, que tiene un precio muy elevado. Aunque que el riesgo no es especialmente grande, y que la mayor parte de los dispositivos lógicos no se dañan fácilmente por cosas como cortocircuitos accidentales en sus salidas, este tipo de interfaces no son recomendables para principiantes.

El Sinclair ZX Spectrum tiene un *bus* de expansión bastante típico Z80. En la figura 10 se ofrecen detalles sobre el puerto en cuestión. Las líneas de datos D0 a D7 llevan los datos para el *hardware*

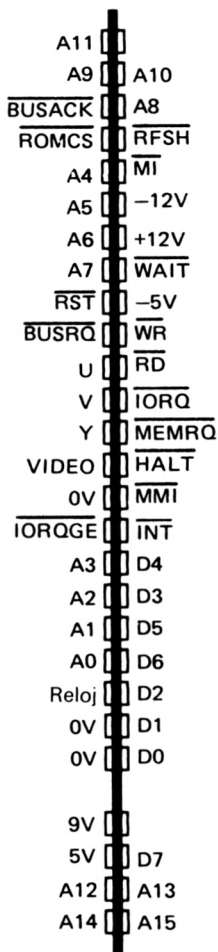


Figura 10.—Conexiones del bus de expansión del ZX Spectrum

interno o externo, y el problema consiste en extraer solamente los datos requeridos para la interfaz CV de entre la masa de señales existentes en estas líneas. Esto se hace verificando la presencia de un juego determinado de niveles lógicos en varias otras líneas, una combinación que sólo se produce cuando los datos para el interfaz CV están presentes en el *bus* de datos. En el sistema estándar Z80, el método empleado consiste en decodificar de A0 a A7, lo que da un total de 256 coordenadas entrada/salida de 0 a 255. Las ocho líneas de coordenadas más significativas (de A8 a A15) se utilizan sólo con dispositivos de memoria, y da 65536 coordenadas de memoria de 0 a 65535. Para que el microprocesador pueda seleccionar bien un circuito de memoria o un dispositivo de entrada/salida, tiene dos líneas de control: MEMRQ (solicitud de memoria) e IORQ (solicitud de entrada/salida). La línea apropiada está en baja, dependiendo del tipo de circuito al que intente acceder el microprocesador. Por tanto, el interfaz CV sólo debe activarse cuando IORQ se pone en baja y está presente la combinación apropiada de estados de la línea de coordenadas.

No todos los ordenadores basados en el Z80 adoptan este método estándar, y el Sinclair ZX Spectrum es un ejemplo de uno que no lo hace. El Spectrum utiliza las dieciséis líneas de coordenadas para direccionamiento entrada/salida, pero en una forma simplificada suministra un número relativamente pequeño de localizaciones utilizables de entrada/salida. La técnica básica consiste en disponer de dispositivos entrada/salida activados, poniendo en baja una de las líneas de coordenadas. La A5 está disponible para añadidos del usuario. Extraer datos es, por consiguiente, cuestión de obtenerlos del *bus* de datos cuando A5 e IORQ están en baja. Esto se logra enviando datos a las coordenadas entrada/salida 65503, y el BASIC de Sinclair incluye la instrucción

OUT. Gracias a la adopción de este método muy básico de codificación de las coordenadas, existen multitud de coordenadas que también podrían utilizarse, pero la única que ofrece total seguridad es la 65503, ya que deja todas las líneas de coordenadas altas, aparte de la A5, y evita operaciones indeseables sobre el *hardware* interno.

La conexión del generador de CV a un bus de datos se facilita por la inclusión de un pestillo de datos de ocho bits en la entrada del convertidor ZN428E. Si el ZN428E se alimenta de salidas con pestillos, el pestillo que lleva incorporado resulta evidentemente innecesario y se deja en la modalidad de “transparente”, haciendo que la entrada de control en la punta 4 esté permanentemente en baja (como en el circuito original de la figura 3). Si el dispositivo se alimenta por medio del *bus* de datos, la entrada de control deberá alimentarse desde el decodificador de coordenadas. Normalmente se mantiene en alta, y requiere una pulsación que lo pone en baja cuando se detectan los estados apropiados en los *buses* de coordenadas y control. La figura 11 muestra el diagrama del circuito de un decodificador de coordenadas sencillo para el ordenador Spectrum. Este incluye también una salida digital con pestillo para controlar la entrada de puerta del sintetizador.

De la decodificación de las coordenadas se encarga IC5, que es una puerta OR de dos entradas. Esta produce un estado de salida en alta si cualquiera de las entradas o ambas están en alta. Visto de otra manera, suministra una salida en baja sólo si ambas entradas están en baja. Esto nos da el resultado requerido al ser enviada una pulsación de baja a la entrada del control de IC1 cuando A5 e IORQ se ponen en baja simultáneamente. Hay otras tres puertas OR en el IC5, pero aquí nos las necesitamos y las pasaremos por alto.

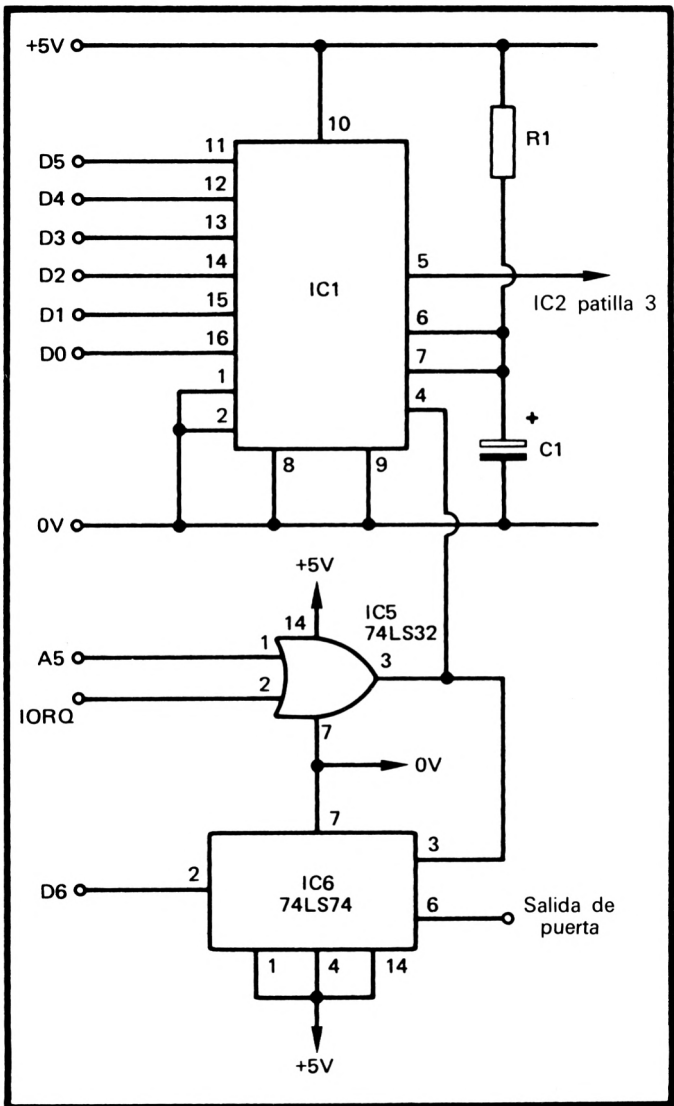


Figura 11.—Conexión del controlador con el conector de expansión del ZX Spectrum

IC6 es un *flip/flop* tipo 74LS74 dual, pero en este circuito sólo se emplea una sección del dispositivo, que actúa como pestillo de datos. Fija éstos en la línea D6 utilizando el mismo decodificador de coordenadas que el convertidor digital-analógico.

La unidad puede conectarse a un ordenador basado en el microprocesador 6502 o en un microprocesador compatible con el *bus*, como el 6809, pero el decodificador de coordenadas tendría que ser totalmente diferente al modelo sencillo de la figura 11. Normalmente, habrá que decodificar las 16 líneas de coordenadas (o al menos la mayoría) para poder poner el puerto añadible en la posición correcta o en un pequeño bloque de coordenadas. Muchos ordenadores basados en el 6502 ofrecen algo de ayuda en forma de una o unas salidas ya decodificadas. Por ejemplo, en el *bus* de 1 MHz del BBC modelo B, hay dos líneas decodificadas, NPGFC y NPGFD. La primera se pone en baja si se accede a cualquier coordenada en el rango &FD00 a &FDFF, mientras que la segunda se pone en baja con una pulsación si se accede a cualquier coordenada desde &FD00 a &FDFF. Una de estas líneas podría, por tanto, usarse para obtener la pulsación de control para IC1 e IC6.

Si no existe este tipo de prestación, no hay más alternativa que diseñar un decodificador para prácticamente la totalidad del *bus* de coordenadas. Algo similar ocurre en el caso de un ordenador que utilice el sistema estándar Z80 de interfaz, cuando hay que decodificar los últimos 8 bits menos significativos del *bus*, además del IORQ. Esta es una teoría que sólo podrá abordar alguien que tenga una razonable experiencia en el diseño de circuitos lógicos y, probablemente, no compensará el esfuerzo realizado. Si uno está interesado en el control por ordenador de instrumentos electrónicos, es claramente recomendable obtener un ordenador que coopere, en lugar de

perder el tiempo luchando con uno que esté mal equipado para la tarea.

Ajuste

Una vez alcanzada la etapa en la que la unidad se conecta al ordenador, es necesario llevar a cabo una serie de pruebas y montajes antes de que pueda utilizarse. Si la unidad se está conectando directamente a los *buses* del ordenador, es importante conectarla antes de encenderla. Aparte del riesgo de perjudicar el interfaz o el ordenador, es prácticamente seguro que se producirá una avería en el ordenador si se conecta un circuito a los *buses* con este encendido. Incluso en el caso de un periférico que se conecta al puerto para el usuario del ordenador, probablemente sea recomendable conectarlo antes de encender el mismo. Hay algo que desde luego no puede ponerse en cuestión; si tras su encendido, el ordenador no presenta su pantalla de arranque normal y no se comporta de un modo absolutamente habitual, debe apagarse inmediatamente para comprobar todo el cableado.

Suponiendo que no haya problemas, se conecta la salida de C1 del interfaz a la entrada de CV del sintetizador, y la salida de puerta a la puerta o la entrada del disparador del sintetizador. Probablemente será necesario el uso de patillas de *jack*, pero algunos sintetizadores pueden necesitar otro tipo de patilla. El conectar la unidad al sintetizador no pone necesariamente al sintetizador en la modalidad en la que puede ser controlado desde el exterior, y probablemente será necesario ponerlo en esta modalidad manualmente. Si existe alguna duda acerca de esto, se debe consultar el manual del sintetizador, que ofrece detalles acerca de

cómo disponer el instrumento para su control desde el exterior.

Asumiremos aquí que la señal puerta/disparador se deriva del bit 6 del *bus* de datos. Si se escribe un valor de 89 sobre el interfaz, éste pondrá la línea de salida de puerta en alta y enviará un valor de 25 al convertidor digital y analógico. Esto se corresponde con una nota dos octavas justas por encima de la nota más baja del teclado, y habrá que ajustar el VR2 para obtener una nota que sea la misma que se obtiene al accionar la tecla correspondiente. Si el sintetizador tiene una entrada de puerta, el instrumento debería suministrar una señal de salida continua siempre que la entrada de puerta se encuentre en alta, en el supuesto de que el control de *sustain* esté a un nivel razonablemente elevado. Con una entrada de disparador, las cosas son algo más difíciles. El configurador de envolvente debe ajustarse para que el tiempo de atenuación sea el máximo, y debe escribirse en el controlador un valor de 89 una y otra vez para mantener una salida continua y poder ajustar el VR2. Puede utilizarse un programa de bucle sencillo para enviar a la salida este valor repetidamente.

Al haber un valor de 65 en el interfaz, la puerta se mantiene aún “abierta” y el valor enviado al convertidor digital-analógico es 1. La nota obtenida debería ser igual a la que se produce al pulsar la nota más baja del teclado, pero en la práctica habrá que ajustar el VR1 hasta obtener exactamente la nota precisa. Incidentalmente, es el 1 y no el 0 el que da la nota más baja, y el 0 no es un valor convertidor válido en esta aplicación.

Todo este procedimiento deberá repetirse unas cuantas veces hasta que no se consiga mejorar ya el resultado, con el sintetizador recorriendo satisfactoriamente toda la extensión de dos octavas. Como explicábamos antes, incluso aunque el teclado del sintetizador sólo tenga una extensión de unas dos

octavas, si el IC2 tiene un voltaje de alimentación adecuado, es prácticamente seguro que se obtendrán resultados satisfactorios en una extensión de más de cinco octavas al utilizar el interfaz.

Software

El *software* puede ser cualquier cosa, desde una rutina muy simple, consistente en unas pocas líneas de BASIC, a un programa sofisticado tipo Compositor, complementado con gráficos para escritura musical, etc. Para la mayor parte de los fines, será suficiente una rutina simple, aunque es preferible realizar un programa más sofisticado si se pretende programar largas secuencias de notas. El número máximo de notas depende de factores como la capacidad de RAM del ordenador para almacenar datos y programas, y el tamaño del programa de secuenciación. Un ordenador doméstico de nuestros días viene a tener 30K o más de RAM disponible, e incluso, aunque hicieran falta varios K para el programa de secuenciador, siguen quedando suficientes RAM como para acomodar al menos unos cuantos miles de notas. Incluso un ZX81 con 1K de RAM puede suministrar secuencias de unas cuantas docenas de notas, y para muchos fines resulta suficiente.

La secuencia básica de acontecimientos que debe suministrar el *software*, tras las operaciones de montaje que pudieran ser necesarias en el puerto de salida, consiste en poner en alta la salida de puerta y enviar el primer valor de nota. Si la señal de puerta sale del bit 6 del puerto de salida, el número enviado al puerto de salida es 64, además del valor de la nota requerida. Ahora la salida de puerta deberá quedar de nuevo en baja. Si se utiliza la unidad con un sintetizador con una entrada de disparador, o si se emplea el circuito

generador de pulsaciones anteriormente descrito, la salida de puerta puede ponerse en baja inmediatamente. Esto se logra enviando el valor de la nota al interfaz. Si el ordenador está enviando una pulsación de puerta sin ayuda del generador de pulsaciones, existen dos opciones. Se puede utilizar una duración programable, pero fija, de la pulsación de la puerta, con lo que el tiempo de puerta requerido se establece al comienzo del programa y se emplea para todas las notas, o se puede hacer que el tiempo sea programable para cada una de ellas. Este segundo método ofrece, obviamente, un máximo de flexibilidad, pero puede resultar un poco tedioso al programar secuencias largas.

Finalmente, hay que establecer un pequeño retraso para enviar la siguiente nota con el intervalo de tiempo preciso. Una duración fija de las notas es un tanto limitada en sus posibilidades, por lo que este retraso debería ser programable individualmente para cada una de las notas. Esta secuencia de acontecimientos se repite después para cada nota. Si se necesita establecer un bucle infinito de la secuencia, sólo es necesaria una única línea de BASIC para que el programa vuelva de nuevo al principio.

Evidentemente, no es práctico ofrecer aquí un *software* complicado para una amplia gama de ordenadores, ni siquiera para unos pocos de ellos. No obstante, el sencillo programa que mostramos utiliza palabras comunes del BASIC, y responderá con la mayor parte de los ordenadores, además de demostrar el modo básico en el que debe funcionar el programa. Cualquier persona razonablemente familiarizada con el BASIC encontrará pocas dificultades para añadir refinamientos a esta rutina básica, diseñada para el VIC-20.

```
10 READ NV, D
20 IF NV = 0 THEN STOP
```

```

30 POKE 37136, 9NV+64)
40 POKE 37136, NV
50 FOR L = 1 TO D: NEXT
60 GOTO 10
70 DATA NV ,D,NV ,D,NV ,D, ... 0,0

```

La línea 10 READ lee el primer valor de nota y duración del enunciado DATA de la línea 70. La secuencia requerida de valores de nota y duración se almacenan en parejas en el enunciado DATA, con el valor de la nota siempre por delante del correspondiente valor de su duración. Si la secuencia de notas es larga, es recomendable distribuir los valores en una serie de enunciados de DATA, ya que esto permite que el material sea más fácil de manejar. En la mayor parte de las versiones del BASIC existe un límite a la longitud de la línea, de manera que las secuencias largas tendrían que distribuirse en varios enunciados de todos modos. Detrás de la pareja final de valores se introducen dos valores falsos de 0 y 0 en el enunciado. Junto con la línea 20, estos dos valores detienen la secuencia en lugar de permitir que el programa se venga abajo con un mensaje de error en los datos.

La línea 30 envía el primer valor de nota al convertidor y pone en alta la salida de puerta. Esta línea de programa es correcta para el ordenador VIC-20, pero, en el caso de otros ordenadores, las coordenadas deberán cambiarse al valor adecuado y, en caso necesario, habrá que cambiar la instrucción POKE por OUT (o lo que corresponda). La línea 40 pone en baja la salida de puerto y asumimos aquí que, o el sintetizador tiene una entrada de disparo, o que se usa el circuito generador de pulsaciones para obtener una pulsación de puerta de la duración requerida. Alternativamente, podría incluirse en el programa un bucle de retardo para establecer el tiempo de puerta requerido.

La duración de la nota viene controlada por el bucle de retardo FOR...NEXT de la línea 50. No es posible decir exactamente el valor que debe usarse aquí para obtener una duración determinada, ya que esto varía de un modelo de ordenador a otro. Con un BASIC lento es necesaria una duración de alrededor de 150 por segundo, mientras que un BASIC rápido puede necesitar un valor de 1.000 por segundo de duración. Esto es, evidentemente, algo que puede descubrirse tras un breve período de experimentación con el sistema. Si el ordenador tiene algún tipo de temporizador incorporado podría resultar más conveniente (y preciso) si se utiliza para controlar la duración de las notas. La línea 60 enlaza de nuevo con la línea 10, donde se lee el siguiente par de valores de nota y duración, y el proceso se repite.

Secuenciación polifónica

Es perfectamente posible controlar por ordenador bien un secuenciador polifónico con entradas de CV y puerta separadas para cada canal, o varios sintetizadores monofónicos, o una combinación de ambos. Un procedimiento sería hacer varios interfaces del tipo descrito anteriormente, pero existen un par de inconvenientes graves en este enfoque. Uno de ellos es simplemente el número considerable de salidas digitales que serían necesarias, al precisarse un puerto de 8 bits (si bien sólo se utilizan en realidad siete líneas) por cada interfaz que se emplee. Aun así, esto podría ser práctico en algunos casos; por ejemplo, utilizando un puerto para el usuario con un interfaz, un puerto de impresora con otro y, un tercero, conectado a un puerto de expansión. No obstante, aunque fuera posible accionar tres o cuatro interfaces desde el ordenador, sigue existiendo el segundo

inconveniente, que no es otro que el coste de hacer las cosas así. Los *chips* convertidores digital-analógicos no se encuentran entre los circuitos integrados más caros, pero están muy lejos de ser los más baratos.

La solución más práctica consiste en utilizar un convertidor digital-analógico junto con un multiplexor y una serie de circuitos de muestreo y conservación (una muestra conservada por canal). Esto sigue requiriendo un número considerable de salidas digitales, pero muchas menos que las que serían necesarias si hubiera que utilizar un convertidor para cada canal. Es sustancialmente más barato que emplear un convertidor por canal. Tiene, eso sí, la desventaja de resultar a menudo menos cómodo a la hora de usarlo, aunque no siempre sea éste el caso. El diagrama de la figura 12 expone, en líneas generales, el modo en que opera este sistema.

La salida del convertidor digital en analógico se envía a un multiplexor que, en este caso, es de cuatro canales, pero que puede tener cualquier número que se desee. El multiplexor no es, en realidad, más que una forma de interruptor electrónico que conecta la señal de entrada a uno de sus terminales de salida. Cuál de las salidas será la que reciba la señal de entrada depende de los niveles lógicos introducidos en las entradas de coordenadas del multiplexor. La idea consiste en seleccionar el canal 0 y escribir en él el voltaje requerido; después seleccionar el canal 1 y escribir el voltaje requerido sobre ese canal, y así sucesivamente, cambiando el voltaje en cada canal con arreglo a nuestras necesidades. El fallo obvio de este sistema es que no hay nada que mantenga la salida de cada canal al voltaje adecuado mientras el convertidor acciona uno de los otros canales. Esto se resuelve utilizando un circuito de muestreo y conservación para cada canal. Cada uno de estos circuitos

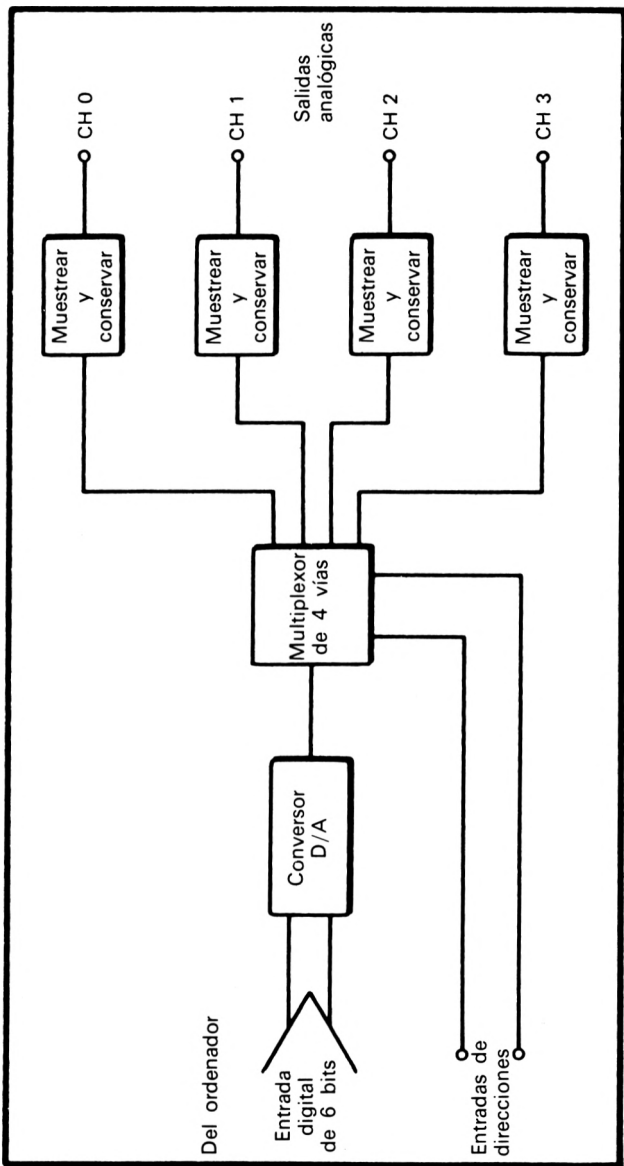


Figura 12.—Diagrama de bloques de un convertor D/A de 4 canales

es un circuito de memoria analógico que se limita a mantener el voltaje de salida en el último potencial de entrada que haya recibido. En algunos sistemas, estos circuitos pueden mantener con precisión el voltaje de cada una de las salidas durante un período de varios segundos o más, y no hay problema en utilizar un sistema así en una aplicación como ésta, en la que cada voltaje de salida será actualizado, como mínimo, cada dos o tres segundos. Hay otros sistemas menos convenientes, que requieren que cada canal se actualice muchas veces por segundo para dar buen resultado.

El diagrama del circuito de la figura 13 se corresponde con un práctico convertidor digital-analógico multicanal, que utiliza la técnica descrita más arriba, y está perfectamente capacitado para ser empleado como interfaz multicanal para secuenciador. Al igual que el circuito original, está basado en un *chip* convertidor Ferranti ZN428E.

El *chip* convertidor no aparece en la figura, pero esta parte del circuito es idéntica a la del diseño monofónico original. De hecho, la etapa de salida es también la misma, y el IC2, más sus circuitos asociados, son exactamente los mismos que los circuitos adyacentes en la figura 3. Sólo la parte central del circuito es diferente, y es allí donde se añaden el multiplexor y los circuitos de muestreo y conservación.

Como multiplexor analógico se utiliza un CMOS 4051BE. Este tiene una entrada de inhibición y una entrada de **bias** a las que se puede alimentar un **bias** negativo si el dispositivo debe utilizar voltajes de señal negativos respecto a la toma de tierra. En esta aplicación no es necesaria ninguna de estas dos prestaciones, y las patillas pertinentes (6 y 7) se conectan a tierra. Las tres entradas de coordenadas del IC1 se alimentan de tres salidas con pestillo del ordenador. Esto da un rango de coordenadas de 0 a 7,

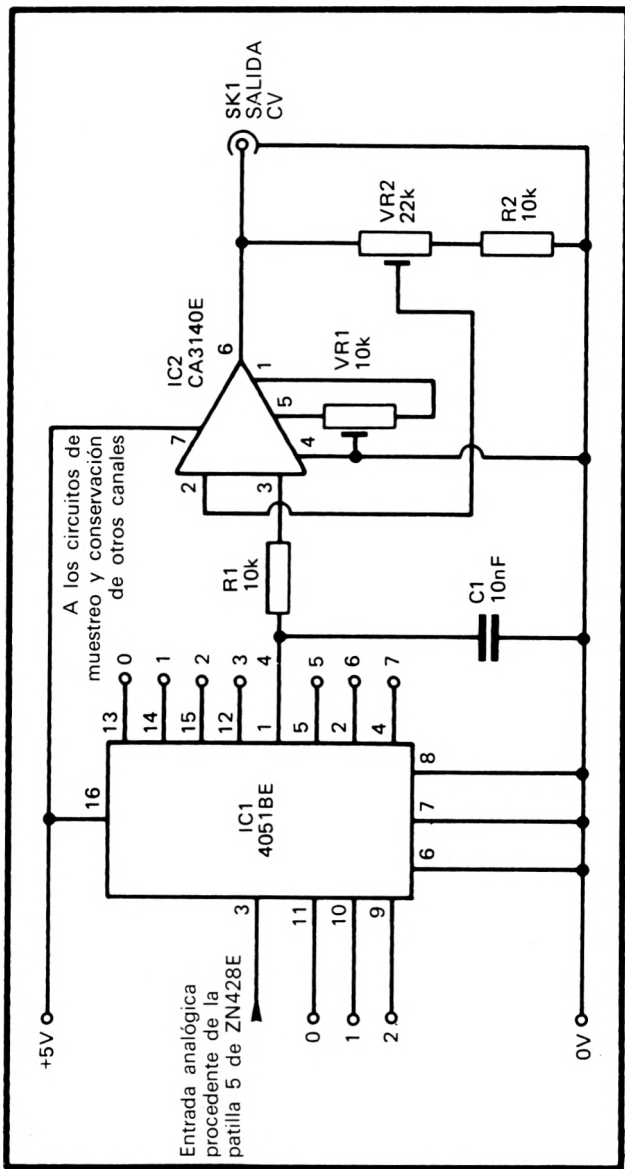


Figura 13.—Esquema de un generador multicanal de CV

y permite controlar hasta ocho canales. En la práctica, lo más probable es que no se necesiten más que cuatro canales y, si éste es el caso, la entrada 2 (patilla 9 del IC1) puede conectarse a tierra. Se utilizan entonces las salidas 0 a 3, mientras de la 4 a la 7 quedan inoperantes.

El circuito de muestreo y conservación de cada canal consiste en realidad en un único condensador (C1). Este se carga rápidamente hasta el potencial de salida suministrado por IC1 cuando se emplea el canal apropiado, y mantiene la carga cuando el IC1 pasa a un canal diferente. De hecho, la carga de C1 irá desapareciendo gradualmente, ya que tanto IC1 como IC2 suministran un camino de descarga e, inevitablemente, habrá un cierto factor de pérdida en el propio C1. No obstante, todos estos caminos de descarga ofrecen una resistencia muy elevada y no debería percibirse una reducción significativa en el potencial de carga de C1 hasta transcurridos varios segundos. R1 es una resistencia de entrada para proteger al IC2.

En la figura 13 sólo se ofrece un circuito de muestreo y conservación, y un amplificador de salida, pero, por supuesto, habrá que duplicar estos circuitos para cada canal que se utilice.

El circuito requiere más salidas que las que puede suministrar el puerto para el usuario de la mayor parte de los ordenadores. Muchos de ellos pueden, a pesar de todo, accionar este interfaz sin demasiada dificultad. Con el ordenador AMSTRAD, por ejemplo, una forma de enfocar la cuestión consiste en accionar el convertidor digital-analógico desde las seis salidas menos significativas del puerto de la impresora y accionar las entradas de coordenadas del IC1 desde tres líneas del puerto de expansión. Esto deja libres dos salidas de datos del puerto de la impresora y cinco líneas del puerto para el usuario, que pueden usarse como salidas de puerta. De hecho, podría

conseguirse un funcionamiento de ocho canales poniendo en servicio la línea de establecimiento de comunicación I/S como octava línea de salida de puerta. Pero sospecho que habrá pocas personas que necesiten toda la capacidad de ocho canales de la unidad.

Básicamente, se requiere del *software* que dé salida a nuevos valores de nota con los intervalos de tiempo apropiados, seleccionando, en primer lugar, el canal requerido utilizando las líneas que accionan las entradas de coordenadas y enviando después el valor al convertidor. Para una máxima comodidad, lo mejor es hacer que las entradas de coordenadas sean accionadas por las tres líneas menos significativas de un puerto de salida, ya que así el número necesario para seleccionar cada canal es el mismo que el número del canal. Por supuesto, la señal de puerta debe ser enviada por el ordenador y el *software* es mucho más manejable si se utiliza para suministrar la señal de puerta del circuito generador de pulsaciones anteriormente descrito. Con él, se trata ya sólo de dar salida a un nuevo valor de nota o a un juego de valores de nota, y después poner en alta las líneas apropiadas, para inmediatamente ponerlas de nuevo en baja. Si se utiliza una duración mayor para la pulsación de la puerta, y cada canal tiene un ritmo diferente, la temporización del programa podría desbordar a todo el que no fuera un programador de gran experiencia. Con una duración breve en la pulsación de la puerta y el circuito de generación de pulsaciones, las cosas son mucho más sencillas.

Un punto importante, que hay que tener en cuenta, es que el potencial de carga del condensador de muestreo y conservación tarda un breve tiempo en ajustarse a los cambios de voltaje. Estamos hablando de un tiempo de ajuste típico de alrededor de $100 \mu\text{S}$ y, al utilizar BASIC, la relativa lentitud de este lenguaje permite asegurar que los condensadores

siempre dispongan del tiempo suficiente para ajustarse adecuadamente a nuevos niveles de voltaje. Al utilizar código máquina sería necesario, casi con seguridad, introducir un corto bucle de retardo para garantizar que la salida del convertidor sea transmitida a cada circuito de muestreo y conservación durante el tiempo suficiente. Alternativamente, podría reducirse el C1 de cada canal a un valor de alrededor de 220 pF, pero los circuitos de muestreo y conservación sólo podrían mantener con precisión los circuitos de salida durante una fracción de segundo, con lo que cada canal tendría que ser actualizado varias veces por segundo para lograr resultados satisfactorios. Con todo, este segundo método podría resultar el más fácil de los dos, y en realidad de lo que se trata es de que cada uno escoja el sistema con el que más fácil le resulte trabajar. Incidentalmente, el código máquina tiene una ventaja en esta aplicación, en especial si se emplean varios canales. La velocidad del código máquina permite una sincronización precisa de los canales, mientras que con BASIC, en especial en las variedades más lentas, la sincronización podría ser notablemente “holgada”.

Componentes de un generador multicanal de CV (Figura 13)

Resistencias (1/4 W 5%)

R1,2	10K (2)
VR1	10K (preajustada)
VR2	22K (preajustada)

Condensador

C1	10nF carbonato
----	----------------

Semiconductores

IC1 4051BE
IC2 CA3140E

Miscelánea

SK1 Hembra de *jack* estándar
Zócalo DIL de IC de 16 patillas
Zócalo DIL de IC de 8 patillas
Carcasa, placa de circuito, cables, etc.

Nótese que, además, es necesario un circuito convertidor D/A CN428E (o similar) y que C1, IC2, R1, R2, VR1 y VR2 se duplican en cada canal (puede utilizarse un máximo de ocho).

Lector de teclado

Un lector de teclado podría considerarse como lo opuesto a un generador de voltaje de control. No es más que un circuito que lee la salida de voltaje de control de un sintetizador y la transforma en un valor digital que puede almacenarse en un ordenador. En otras palabras, es una forma de convertidor analógico-digital, en lugar de uno digital-analógico como el utilizado en el generador de CV.

Existen dos formas principales de utilizar un dispositivo de este tipo. Una de ellas consiste en permitir la entrada de valores de notas del teclado del sintetizador al utilizar un programa de secuenciador, lo que es mucho más fácil para la mayor parte de los usuarios que escribir las notas, y es el uso principal a que se destina este tipo de circuito. El segundo uso posible, que a muchos probablemente les parezca el

más interesante y útil, es utilizar el interfaz para registrar una secuencia de notas, incluyendo la duración de las mismas, de modo que la secuencia pueda después escucharse en tiempo real utilizando un interfaz del tipo anteriormente descrito. Aunque este tipo de sistemas reciben normalmente el nombre de registradores en tiempo real, en su mayor parte disponen de la prestación de poder variar la velocidad de reproducción dentro de límites bastante amplios. Esto es algo que es fácil de lograr en un sistema basado en un ordenador doméstico, y requiere el *software* necesario para alterar el valor de duración de las notas en lugar de *hardware* adicional.

El proceso de conversión analógico-digital no es tan simple como el de la conversión digital-analógica, y existen varios tipos de convertidores A/D que varían considerablemente en su velocidad operativa, su precisión y su precio. Para nuestros propósitos, no necesitamos un tipo especialmente rápido o preciso. Probablemente el mejor sea un convertidor del tipo de aproximaciones sucesivas, dado que éstos son fáciles de conseguir a precios accesibles. En general, ofrecen un nivel de rendimiento que es sustancialmente más que adecuado para esta aplicación.

Este tipo de convertidor consiste básicamente en un registro de 8 bits que acciona un convertidor digital-analógico, además de un comparador de alta velocidad y algo de lógica de control. Inicialmente, todas las salidas del registro se ponen a cero, aparte del bit más significativo. El voltaje de salida del convertidor digital-analógico se compara entonces con el voltaje de entrada. Si el primero es mayor, el bit más significativo del registro se queda en 1; en caso contrario, se fija en 0. En ese caso, el bit 6 del registro se fija en 1, la salida del convertidor digital-analógico se compara con el voltaje de entrada y el bit 6 permanece en 1 o vuelve a 0, según sean los resultados de la comparación. Este proceso se repite

con los demás bits del registro, bit a bit, hasta llegar al menos significativo. Al ir teniendo lugar este proceso, el número del registro se va convirtiendo gradualmente en una representación digital válida del voltaje de entrada. Es, pues, el número de 8 bits del registro quien suministra la salida digital convertida.

Por supuesto, el modo en que el número digital se relaciona con el voltaje absoluto varía de un diseño de convertidor a otro. El voltaje total de entrada del circuito es igual al voltaje total de salida del convertidor D/A utilizado en el diseño. En el caso de la unidad utilizada aquí, el valor total de la escala es de 2,55 voltios. Este circuito (figura 14) está basado en el circuito integrado Ferranti ZN449 y el convertidor D/A que aparece en este dispositivo es esencialmente el mismo que constituye la base del ZN428E, anteriormente descrito. Es fácil alterar el valor de toda la escala añadiendo un amplificador en la entrada analógica para aumentar la sensibilidad o, como en este caso, utilizando un atenuador para obtener una reducción de la sensibilidad.

Existen varias formas de conectar el dispositivo a un ordenador, pero, por el momento, asumiremos que la unidad está interconectada al puerto para el usuario de un ordenador como el AMSTRAD o el Commodore 64, lo que es, sin duda, el modo más sencillo de hacer las cosas. Las líneas PB2 a PB 5 se usan como *input* para leer las seis salidas más significativas del *chip* convertidor, IC1. Como en el circuito generador de CV, sólo se emplean seis bits, lo que da una extensión de más de cinco octavas (lo que es más de lo que ofrecen todos los sintetizadores que conozco). Las dos salidas menos significativas del IC1 se pasan por alto. Incidentalmente, el ZN449 especificado como IC1 pertenece a una serie de tres dispositivos, siendo los otros el ZN448 y el ZN447. El ZN449 es, con mucho, el más barato de los tres y es también el menos preciso. No obstante, dado que

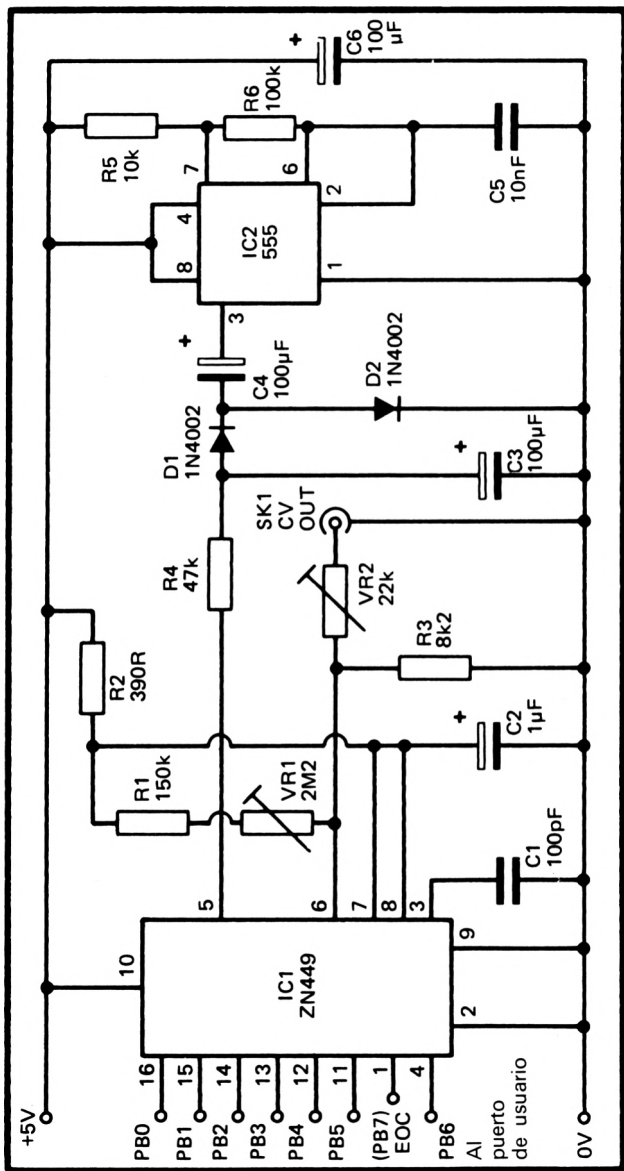


Figura 14.—Circuito lector de un teclado de CV

sólo necesitamos una resolución de seis bits para esta aplicación, su precisión es más que suficiente y no tiene sentido emplear una de las versiones más caras.

Con un convertidor digital-analógico se pueden introducir datos en el circuito en cualquier momento y, casi al instante, éstos serán convertidos en el valor analógico requerido. Existe un ligero retraso (llamado “tiempo de asentamiento”) hasta que la salida adopta el valor correcto, pero éste es muy corto, típicamente de 1µs o menos. En el caso del convertidor analógico-digital las cosas son menos complicadas, ya que el dispositivo no puede ser leído por las buenas en cualquier momento, obteniendo una lectura precisa de la entrada obtenida. El *chip* convertidor requiere antes una señal del ordenador para empezar la conversión, y después debe haber un retraso adecuado hasta que la conversión haya finalizado, y surja una representación adecuada del voltaje de entrada del registro de aproximaciones sucesivas.

En este circuito, PB6 actúa en modalidad de salida, y se emplea para suministrar un breve impulso negativo para iniciar cada conversión. El tiempo invertido en cada una de ellas es igual a nueve ciclos del reloj, y el ZN449 tiene un oscilador-reloj interno. C1 es el condensador de temporización para el circuito del reloj y, con el valor especificado de 100pF se obtiene una frecuencia para el reloj justamente inferior a 1 MHz (que es prácticamente la frecuencia de reloj más elevada que se puede emplear en un ZN449). Esto representa alrededor de 10 Ms por conversión. Al leer la unidad con BASIC, cada conversión habrá sido realizada holgadamente antes de que pueda ser leído el convertidor, dado que, incluso con una versión rápida de BASIC, cada instrucción tarda en completarse alrededor de 100 Ms o más. Al utilizar código de máquina será posible, casi con seguridad, tomar lecturas en algo menos de tiempo del que tarda el convertidor en realizar la

conversión. La solución más sencilla al problema consiste en emplear un breve retardo de *software* para evitar una lectura prematura del convertidor. Un método alternativo consiste en monitorizar el estatus de salida de “fin de la conversión” del IC1 (patilla 1) usando la línea PB7. Esta salida se pone en baja mientras se realiza una conversión y, por tanto, se puede utilizar el *software* para obtener un retardo y evitar que el convertidor sea leído antes de PB7 esté de nuevo en alta.

VR2 y R3 actúan como atenuador de entrada, que reduce la sensibilidad del circuito hasta obtener un voltaje de entrada de escala completa justamente por encima de los 5 voltios. VR2 se ajusta para obtener exactamente la sensibilidad precisa. IC1 no tiene un circuito incorporado de ajuste a cero, mientras que VR1 y R1 se emplean para garantizar la obtención de resultados precisos con potenciales de entrada bajos.

Es necesario una fuente de voltaje de referencia para la sección convertidora digital-analógica del IC1, y existe una fuente interna. Esta requiere los componentes sueltos R2 y C2. Es necesario un voltaje negativo para la etapa comparadora del IC1, que podría ser suministrado por un raíl negativo de alimentación del ordenador, en el que hay una fuente del voltaje adecuado. Incluso aunque exista un suministro apropiado, a menudo resulta más conveniente utilizar un sencillo circuito generador de voltaje negativo para derivar el voltaje negativo de **bias** del suministro de +5 voltios. Este es el método adoptado aquí y el circuito generador de **bias** consiste en el dispositivo 555 de temporización IC2 empleado en modalidad estable, que alimenta un sencillo circuito rectificador y atenuador. Este da un **bias** negativo de alrededor de -3 voltios que se envía al IC1 vía la resistencia de “cola” R4.

Con el ordenador AMSTRAD, por ejemplo, la forma de enfocar la cuestión consiste en utilizar las

seis entradas menos significativas (D0 a D5) del puerto de usuario, como entrada de los datos proporcionados por el conversor A/D. La señal que sale del ordenador para indicar el comienzo de la conversión puede ser la misma línea STROBE del puerto de impresora, ya que ofrece un nivel bajo únicamente en el momento en el que se escriben los datos sobre este puerto.

La señal de fin de conversión (EOC) puede leerse en el bit D6 del puerto de usuario. Es muy importante tener en cuenta que las entradas del puerto de usuario son activas a nivel bajo; por tanto, será necesario negar todas las entradas con puertas NOT.

Componentes del Lector de Teclado (figura 14)

Resistencias (1/4W 5%)

R1	150K
R2	390R
R3	8K2
R4	47K
R5	10K
R6	100K
VR1	2m2 preajustada
VR2	22K preajustada

Condensadores

C1	100pF placa cerámica
C2	1 μ F63V electrolítico
C3,4,6	100 μ F10V electrolíticos (3)
C5	10nF poliéster

SK1 Hembra *jack* estándar
Zócalo DIL de CI de 18 patillas
Zócalo DIL de CI de 8 patillas
Cable de ordenador
Carcasa, placa de circuito, cable, etc.

Ajuste y utilización

Para probar la unidad y dejarla lista para su uso es necesario un sencillo programa de prueba. El que sigue es para el ordenador AMSTRAD, pero no debería existir dificultad alguna para escribir una versión del mismo para cualquier otro ordenador.

```
10 PRINT #8,0  
20 PRINT JOY(0)—64  
30 GOTO 10
```

La línea 10 produce un pulso a nivel bajo en la línea STROBE del puerto de impresora para ordenar el comienzo de la conversión. La línea 20 lee a continuación la información del conversor, y le resta 64, de forma que el valor obtenido como muestra esté en el rango 0 a 63. El motivo de esta sustracción es que la señal de fin de conversión ocupa el séptimo bit del byte del puerto de usuario (D6). Al terminar la conversión, este bit quedará en activo (nivel bajo), añadiendo al byte de entrada el valor 64, que no es un dato que nos interese. La línea 30 enlaza el programa de vuelta a la línea 10, donde se inicia una nueva conversión. La acción del programa consiste en tomar una serie continua de lecturas que van apareciendo en el lado izquierdo de la pantalla.

Con el SK1 conectado a la salida de CV del sintetizador, y las dos resistencias preajustadas dispuestas a alrededor de la mitad de su valor máximo, se pulsa la tecla más aguda del teclado del sintetizador. Se ajusta entonces VR2, de modo que el valor devuelto a la pantalla sea igual al número de teclas del teclado. Seguidamente se aprieta la tecla más baja. Esto debería dar una lectura de 1, pero en caso necesario puede ajustarse VR1 hasta obtener una lectura estable de 1. Se repite todo este procedimiento dos o tres veces hasta que la unidad recorre con precisión toda la extensión del teclado, dando lecturas estables en todas las teclas del sintetizador.

No debería haber dificultades en utilizar la unidad para suministrar valores de nota para un programa de secuenciador estilo Compositor. Un punto importante a recordar es que hay que asegurarse que esté presente un nuevo valor de nota cuando el valor sea leído en el puerto para el usuario. Una forma de realizarlo es hacer que el puerto sea leído al pulsar una tecla determinada del ordenador; después sólo se trata de asegurarse de que se ha tocado la tecla apropiada en el sintetizador antes de accionar esa tecla del ordenador. Podría obtenerse un funcionamiento automático monitorizando la salida de puerta del sintetizador utilizando una entrada digital del ordenador. La línea D6 podría utilizarse para este fin, y entonces no habría más que leer el puerto del usuario para determinar el nuevo valor de la nota cada vez que se detectara una transición de alto a bajo en D6.

Si la unidad se emplea en una aplicación de secuenciador de tiempo real, es esencial monitorizar la salida de puerta del sintetizador, ya que hay que hacer que una línea de salida del ordenador simule la salida de la puerta durante la reproducción. No se trata simplemente de enviar una serie de valores de notas a la entrada de CV con los intervalos de tiempo apropiados. Existen dos formas básicas de enfrentarse

al *software* de secuenciación en tiempo real. Una de ellas consiste en monitorizar el puerto de entrada a intervalos regulares y almacenar los resultados en un bloque de memoria. Esto puede resultar muy sencillo desde el punto de vista del *software*, especialmente si la entrada de puerta y el convertidor A/D alimentan los mismos bits que se emplean para accionar la salida de puerta y el convertidor D/A. Durante la reproducción se trata, pues, de sacar los valores almacenados a través del puerto de salida con los intervalos de tiempo correctos. No obstante, incluso aunque la línea de entrada de la puerta alimente un bit diferente al utilizado para la salida de puerta, lo único necesario es una sencilla manipulación de valores para obtener un funcionamiento correcto del sistema.

La desventaja principal de éste es la necesidad de una gran cantidad de memoria para almacenar incluso secuencias breves de notas. El problema proviene de la necesidad de muestrear el puerto de entrada a intervalos bastante frecuentes, digamos unas cien veces por segundo, o, más aún, para obtener resultados realmente buenos. Si no se hace así, existe el riesgo de que las notas breves sean directamente pasadas por alto y, en el mejor de los casos, el *tempo* de la secuencia en la reproducción puede no ser muy preciso. Con un ritmo de muestreo de cien muestras por segundo, alrededor de 1K de RAM da tan sólo para unos diez segundos de grabación. Este sistema sigue siendo bastante práctico si se dispone de un ordenador con una cantidad de RAM considerable a disposición del usuario. Con alrededor de 30K de RAM, se puede lograr un tiempo de grabación máximo de alrededor de 5 minutos. Desde luego, es el mejor sistema si se dispone de un ordenador con abundante RAM libre y no se es particularmente experto en la escritura de *software*. Con un BASIC razonablemente rápido debe obtenerse un ritmo de

muestreo adecuado y no debería haber necesidad de recurrir al código máquina.

El problema del sistema delineado más arriba radica en que emplea varios bytes de RAM incluso para notas cortas, y, en las notas largas, puede llegar a utilizar más de un centenar de bytes. El segundo sistema es mucho más eficiente que todo esto y con la mayor parte de los ordenadores puede producir secuencias de muchos miles de notas de longitud 0, en términos de tiempo, de una duración de alrededor de 30 minutos o más. Incluso aunque no se desee almacenar largas secuencias (y probablemente pocos usuarios deseen hacerlo), la capacidad de almacenar varias secuencias relativamente cortas en la memoria de modo simultáneo puede, a menudo, resultar provechosa.

La idea básica del segundo sistema consiste en monitorizar el puerto de entrada y almacenar nuevos valores sólo cuando se detecta algún cambio. Además de los nuevos valores, hay que almacenar también duraciones. Estas están a menudo en forma de tiempo transcurrido desde el último cambio, en lugar de como tiempo total transcurrido desde el comienzo de la secuencia, ya que así se obtienen números más bajos y se emplea menos RAM. De hecho, este sistema necesita tan sólo alrededor de 4 bytes de RAM por nota. Puede resultar razonablemente fácil implementar, siempre y cuando el ordenador tenga una función temporizadora incorporada con una resolución razonable (alrededor de 10 ms). Sin un temporizador incorporado podría ser bastante difícil de implementar, especialmente en BASIC, y el primer sistema sería entonces la proposición más sensata a menos que sea uno razonablemente experto en la escritura de *software*.

Interfaces

Al decidir cuál es el mejor método para conectar la unidad al ordenador, debe tenerse presente que el lector de CV y el generador de CV no necesitan estar simultáneamente conectados al ordenador. Es conveniente operar sobre ellos desde dos puertos separados, ya que esto evitará muchas conexiones y desconexiones. Se evitará también el riesgo de conectar el lector de teclado al ordenador mientras éste tiene sus líneas dispuestas como salidas para accionar el generador de CV (lo que tendría como resultado la conexión de dos juegos de salidas). Esto podría no dañar el *hardware*, pero más vale no averiguarlo.

Si se desea conectar directamente el lector de teclado a los *buses* del ordenador, no tiene por qué resultar excesivamente difícil. Puede hacerse del mismo modo en que se conecta el generador de CV. La unidad sólo debe enviar datos al *bus* de datos durante los breves períodos en los que el microprocesador intenta leer la unidad, y las salidas deben pasar a un estado de alta impedancia en los demás momentos. No es necesario añadir un amortiguador de tres vías entre el IC1 y el *bus* de datos, ya que el IC tiene tres patillas de estados de salida. Estas están en estado inactivo cuando la patilla 1 está en alta y las salidas se activan poniendo en baja la patilla 1. Por consiguiente, las seis salidas utilizadas del IC1 pueden conectarse directamente a las patillas D0 a D5 del *bus* de datos, y el decodificador de coordenadas deberá enviar una pulsación negativa a la patilla 1 de IC1.

Tomemos un ejemplo práctico. Si la unidad se conecta al *bus* de expansión del ordenador AMSTRAD, la salida IORQ de éste puede emplearse para producir la pulsación de capacitación a la patilla 2 del IC1. Entonces el convertidor podría ser leído en

cualquier coordenada desde &FF00 hasta &FFFF. Sería necesario algún medio para suministrar la “pulsación de comienzo de la conversión” antes de cada lectura. Con este fin, se podría conectar la salida IORQ a la patilla 4 del IC1. Así, leer cualquier valor sobre unas coordenadas del rango &FF00 a &FFFF iniciaría una conversión.

La utilización de un circuito decodificador de coordenadas que no decodifique la línea de lectura/escritura de un *bus* 6502 o la línea de RD (lectura) de un *bus* Z80 supone un cierto riesgo. El problema es que una operación accidental de escritura al convertidor podría tener como resultado que el procesador intentara escribir datos sobre las salidas del convertidor, lo que es, evidentemente, indeseable. Esto puede evitarse utilizando un decodificador de coordenadas que procese la línea de lectura/escritura, que pasa a baja durante las operaciones de escritura y a alta en las operaciones de lectura. El Z80 tiene una disposición distinta, con líneas de lectura y escritura separadas, una de las cuales pasa a baja dependiendo del tipo de operación en curso. La decodificación de la línea de lectura/escritura permite que un dispositivo de entrada y otro de salida compartan las mismas coordenadas, pero no es esencial utilizarla, siempre y cuando se tenga la precaución de no escribir nunca accidentalmente datos sobre un dispositivo de entrada.

Si se emplea la conexión directa a los *buses* del ordenador, sigue siendo necesario tener una entrada digital para leer la línea de salida de puerta del sintetizador, que podría ser leída por cualquier entrada digital libre del ordenador. No obstante, todo suele resultar más cómodo si esta línea de entrada se encuentra en las mismas coordenadas que el convertidor. Se puede aportar fácilmente una entrada digital utilizando un amortiguador de tres estados, como muestra la figura 15.

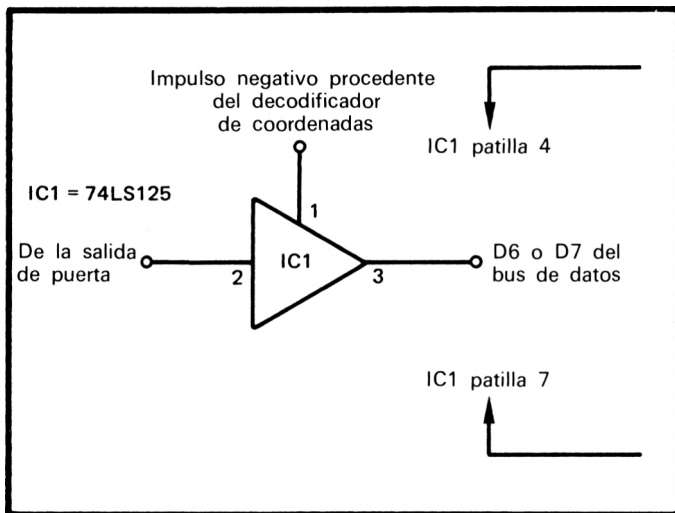


Figura 15.—Conexión de la salida de puerta con el bus de datos

2

Percusiones sintetizadas

Muchos sintetizadores de batería actuales son totalmente digitales. Esto es, no se limitan a emplear un circuito digital para generar el esquema rítmico deseado, sino que emplean también circuitos digitales para generar las señales de salida. La forma más común de generación digital de señales consiste en emplear ROM que contienen grabaciones digitales de instrumentos verdaderos y, como cabría esperar, esto produce normalmente muy buenos resultados. Este tipo de tarea puede abordarse empleando un ordenador doméstico apropiado como base del sistema, pero esto es algo que no discutiremos hasta el siguiente capítulo.

El costo de los sistemas totalmente digitales está descendiendo claramente con el paso del tiempo, pero la combinación del control digital y la generación analógica de sonido sigue representando lo que probablemente sea la solución más beneficiosa, en lo que a la relación calidad-precio se refiere, para la síntesis electrónica de percusión. En este capítulo describiremos algunos generadores analógicos sencillos de percusión, que resultan apropiados para su control digital. Todos los circuitos son baratos, pero ofrecen un abanico útil de sonidos. La utilización de todos estos circuitos, además de un ordenador

doméstico para controlar el sistema, compone un conjunto versátil pero de bajo costo, en especial si se tiene ya acceso a un ordenador doméstico apropiado y no hay que incurrir en el gasto de comprar un *ex profeso*.

Tambor de afinación fija

Los sintetizadores de percusión pueden ser muy complejos, pero se puede obtener un efecto sorprendentemente bueno empleando un circuito ultrasencillo, que solamente consiste en un oscilador de audiofrecuencia con el nivel de realimentación ajustado de modo que resulte justamente insuficiente para mantener la oscilación. Si se aplica una pulsación breve a la entrada del amplificador que hay en el corazón del oscilador, éste excita al oscilador, pero la oscilación no puede mantenerse y la señal de salida se desvanece gradualmente. Esto da una señal que consiste en un tono de un nivel inicial de señal elevado que, al principio muy rápidamente, y después de modo más gradual, va bajando hasta cero. En otras palabras, produce una señal muy similar a la obtenida en una caja de ritmos convencional, que utiliza un oscilador de onda sinusoidal o triangular que alimenta un VCA (amplificador controlado por voltaje) y un circuito generador de envolvente.

La salida de un circuito oscilador “amortiguado” sencillo es más realista de lo que se podría pensar, y es probablemente superior a muchos circuitos basados en un oscilador con circuitos de configuración de envolvente. Con todo, debe tenerse presente que la excitación de un oscilador es la analogía electrónica exacta a la acción de un tambor mecánico, en el que el impacto de la baqueta o maza hace que el parche produzca oscilaciones mecánicas que se desvanecen

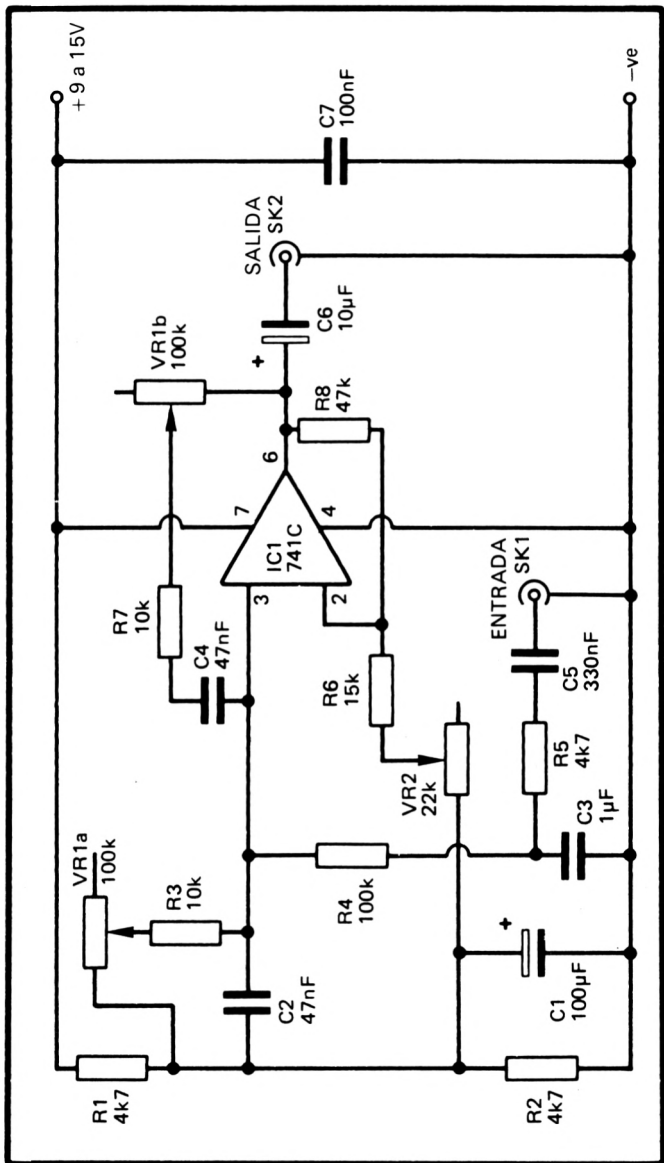


Figura 16.—Circuito de un simple pero efectivo sintetizador de tambor de nota única

rápido. Por tanto, no parece descabellado esperar buenos resultados de los sintetizadores de este tipo.

La figura 16 muestra el diagrama de circuito de un tambor de afinación fija que emplea el sistema delineado anteriormente. "Afinación fija" no es un término particularmente afortunado en este caso, ya que la afinación del tambor puede ajustarse manualmente en una banda de frecuencias muy ancha. Es un instrumento de afinación fija, en el sentido de que el tono permanece constante durante cada impulso tonal, y la frecuencia no está modulada electrónicamente al estilo de las baterías disco.

Todos aquellos que estén familiarizados con los equipos de comprobación de audio probablemente reconozcan el circuito como un oscilador Wien. Uno de los brazos de una red Wien está formado por C4, R7 y VR1b, mientras que el otro brazo está formado por C2, R3 y VR1a. A una determinada frecuencia la red suministra un desplazamiento de fase igual a cero, y en esa misma frecuencia presenta su nivel mínimo de pérdida (alrededor de 10 dB). Para producir oscilaciones en esa frecuencia, IC1 deberá tener una ganancia en lazo cerrado de aproximadamente 10 dB (tres veces), pero en esta aplicación el control de ganancia VR2 se ajusta para una ganancia de voltaje ligeramente inadecuada. VR2 opera, a todos los efectos, como control de amortiguación, ya que un nivel de ganancia bajo produce una señal de salida baja y un sonido de tambor bien amortiguado, mientras que si se lleva VR2 justamente al borde de la oscilación, se produce una amortiguación larga y un sonido de tambor muy resonante. VR1 permite variar la frecuencia operativa en una banda de frecuencias de unos 30 a 300 Hz, pero si se hace que R3 y R7 tengan un valor ligeramente inferior, puede aumentarse el límite superior de frecuencias si así se desea.

El ordenador suministra una breve pulsación de unos 5 ms de duración que es procesada, en primer lugar, por el condensador de bloqueo de CC, C5 y un filtro sencillo pasivo de paso bajo formado por R5 y C3. Sin este filtraje, la pulsación llegaría a la salida como un *clic* indeseable, pero el filtraje modifica el sonido convirtiéndolo en un *thud* más profundo, como el sonido inicial de un tambor ordinario. R4 acopla el impulso procesado a la entrada sin inversión del IC1, donde excita el circuito, produciendo una breve explosión de oscilaciones.

Este circuito puede parecer demasiado sencillo para ser verdad, pero todo tiene su precio. El problema radica en que puede resultar difícil montar la unidad con precisión para obtener exactamente el efecto deseado. Es probable que ajustar el control de frecuencias haga necesario ajustar de nuevo el control de realimentación. Este debe ajustarse con gran cuidado para lograr el efecto preciso; puede merecer la pena añadir un potenciómetro lineal en serie con VR2, para que actúe como control “fino” de la realimentación. Una vez que la unidad queda correctamente ajustada, debe producir un efecto que probablemente no pueda ser mejorado por ningún otro circuito analógico sencillo.

Por supuesto, si son necesarios varios tambores con afinaciones diferentes, tan sólo hay que construir varios de estos circuitos y accionarlos por separado desde salidas distintas del ordenador. Si éste tiene una salida de alimentación de entre 9 y 15 voltios, y un contenido en ruidos y vibraciones razonablemente bajo, ésta podría ser la fuente de alimentación de la unidad. Alternativamente, puede emplearse una pila de 9 voltios para alimentarla.

Componentes del sintetizador de tambor de afinación fija (figura 16)

Resistencias (1/4W 5%)

R1,2,5	4K7(3)
R3,7	10K(2)
R4	100K
R6	15K
R8	47K
VR1	100K doble lin
VR2	22K lin

Condensadores

C1	100 μ F16 electrolítico
C2,4	47nF carbonato (dos unidades)
C3	1 μ F carbonato
C5	330nF carbonato
C6	10 μ F25V electrolítico
C7	100nF cerámico

Semiconductores

IC1	741C
-----	------

Miscelánea

SK1,2 Hembra *jack* estándar
Zócalo DIL de CI de 8 patillas
Carcasa, placa de circuito, cable, etc.

Configurador de envolvente

No existe equivalente al método de generación del Sintetizador de Tambor de Afinación Fija cuando se trata de otros tipos de sonidos percusivos, como los platos o el gong. Todos ellos requieren un configurador de envolvente que procese la salida del circuito generador de sonidos para obtener buenos resultados. El circuito es del tipo sencillo de ataque y amortiguación en vez del tipo, más sofisticado, de ADSR (Attack-Decay-Sustain-Release), pero para sonidos de percusión un configurador de envolvente sencillo como éste es perfectamente adecuado. Más adelante, en este mismo capítulo, discutimos fuentes adecuadas de señal para su uso con el generador de envolvente.

El circuito se compone de dos secciones: una de ellas sirve para generar un voltaje de control a partir del impulso de entrada procedente del ordenador y la otra, para obtener un nivel de ganancia que varíe en simpatía con este voltaje. Al igual que el sintetizador de tambor de afinación fija, este circuito requiere impulsos de disparo de una duración de unos pocos milisegundos.

El generador de voltaje de control se basa en el IC1. El IC1a opera como un amplificador de baja ganancia de CC, que genera un impulso de salida de alrededor de 9 voltios a partir del impulso de entrada de 5 voltios. También suministra amortiguación. El CA3420E especificado como IC1 es un dispositivo que tiene una etapa de salida que puede producir voltajes de salida hasta el rail de alimentación de 0 voltios. Esta es una característica fundamental para el funcionamiento correcto de este circuito, ya que impide que la mayor parte de los demás amplificadores operativos duales puedan funcionar apropiadamente en él. El LM358 es prácticamente el

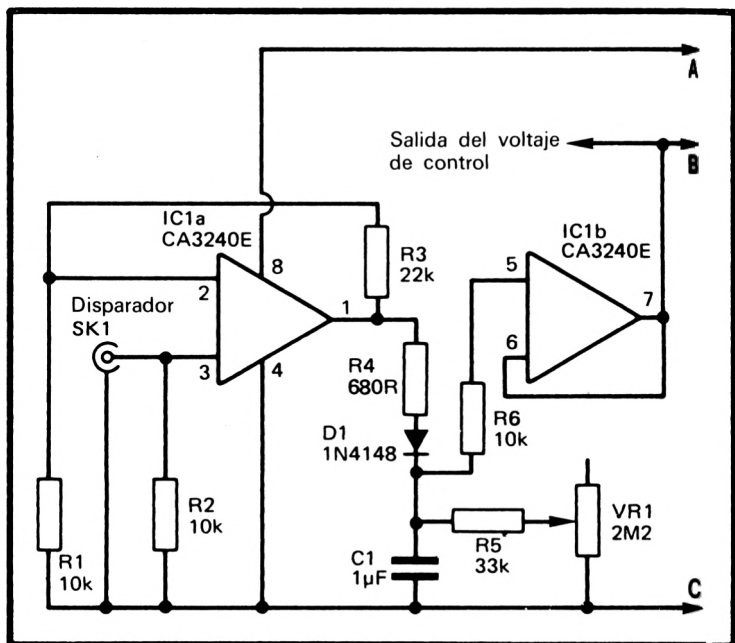
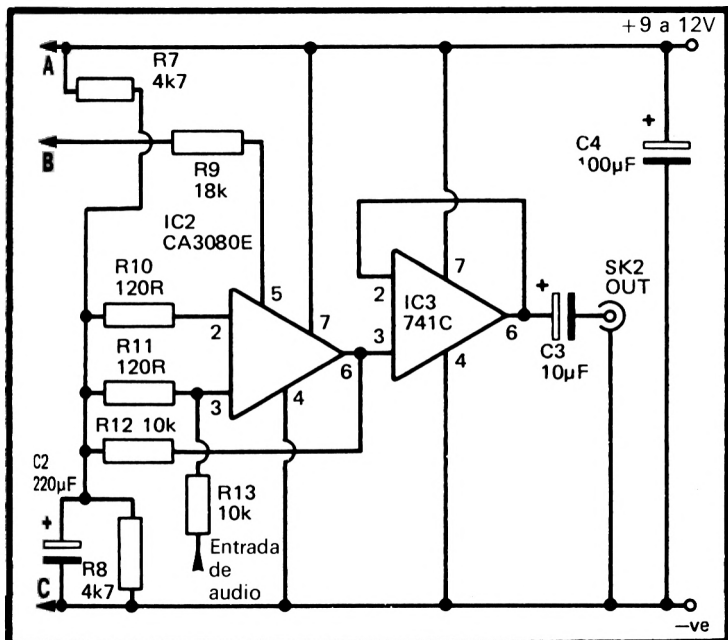


Figura 17.—Circuito de un conformador

único dispositivo alternativo capaz de actuar como sustituto adecuado en este circuito. Idealmente, la señal de entrada debería proceder de un dispositivo lógico MOS o CMOS, ya que éstos ofrecen un voltaje de salida a prácticamente la totalidad de los potenciales de los raíles. En este circuito, un potencial “bajo” de entrada de más de unos pocos centenares de milivoltios podría impedir que el circuito cortara la señal adecuadamente al final de la envolvente. El circuito pareció funcionar satisfactoriamente al ser empleado con varios dispositivos 74LS** en serie, pero en caso necesario, se podría emplear un amortiguador CMOS en la entrada para garantizar un funcionamiento apropiado. La salida del IC1 se



de envoltorio para sonidos de percusión

emplea para cargar C1, y la baja impedancia de salida de IC1, junto con la baja resistencia de R4 y D1, tiene como resultado que C1 quede casi totalmente cargado durante el breve período en el que está presente el impulso de salida. Cuando la salida de IC1a retorna a estado de baja, D1 impide que C1 se descargue de vuelta en la etapa de salida de IC1a, y el único camino significativo de descarga es a través de R5 y VR1. Esto ofrece un tiempo de descarga mucho más largo, de alrededor de 100 ms con VR1 al mínimo de su valor, que aumenta a varios segundos cuando VR1 está al máximo de su resistencia. C1 no descarga de modo lineal, sino de forma exponencial, lo que quiere decir que el voltaje disminuye muy

rápídamente al principio, pero después va amortiguándose cada vez más lentamente. Esto no es un inconveniente y, de hecho, resulta ventajoso, dado que produce sonidos de percusión muy reales. IC1b no es más que una etapa de amortiguación de la unidad de ganancia situada en la salida del generador de voltaje de control.

El amplificador controlado por voltaje es un tipo de amplificador operacional estándar de transconductancia. En este circuito, IC2 es el amplificador de transconductancia e IC3 actúa como una etapa de amortiguación de la salida, que da a la unidad una impedancia de salida baja. La ganancia del VCA depende de la corriente de **bias** alimentada a la punta 5 de IC2. Hablando con precisión, el circuito es un amplificador controlado por intensidad más que por voltaje. No obstante, se añade R9 en serie con la entrada de **bias** del amplificador de IC2 para que la corriente de entrada sea, a grandes rasgos, proporcional al voltaje aplicado, lo que convierte a todos los efectos el circuito para la operación controlada por voltaje.

Componentes del configurador de envolvente (figura 17)

Resistencias (1/4W 5%)

R1,2,6,12,13	10K(5)
R3	22K
R4	680K
R5	33K
R7,8	4K7(2)
R9	18K
R10,11	120R(2)
VR1	2M2 lin

Condensadores

C1	1 μ F carbonato
C2	220 μ F 16V electrolítico
C3	10 μ F 25V electrolítico
C4	100 μ F 16V electrolítico

Semiconductores

IC1	CA3240E
IC2	CA3080E
IC3	741C
D1	1N4148

Miscelánea

SK1,2	Hebras de <i>jack</i> estándar (2)
Tres zócalos DIL de IC de 8 patillas	
Carcasa, placa de circuito, cables, etc.	

Tambor disco

El circuito de la figura 16 podría emplearse como fuente de señal para el configurador de envolvente. En lugar de emplear el control de resonancia justamente por debajo del umbral de la oscilación, se adelantaría justo por encima de este punto, hasta que produjera un oscilación suave con una señal sinusoidal de buena calidad. No ofrece un efecto mejor que el método original de síntesis de un tambor de afinación fija y hace necesario el uso de más componentes, pero el tiempo de amortiguación necesario es más fácil de establecer.

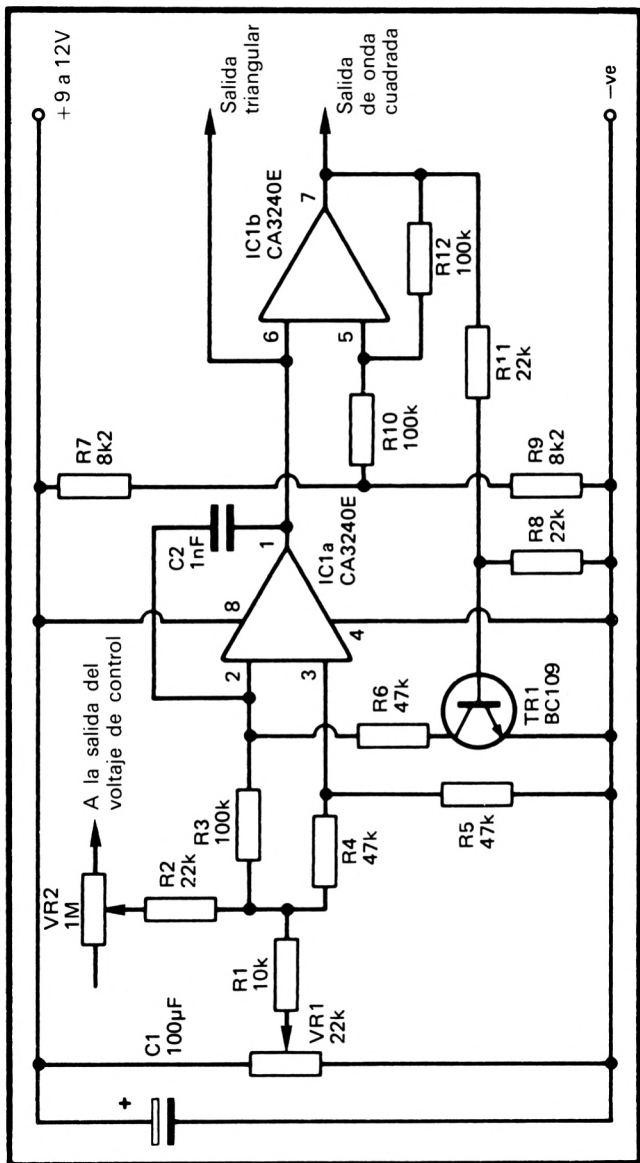


Figura 18.—Un VCO para generar percusión estilo disco

Si se necesita un sonido “disco” hace falta una fuente de sonido más compleja. Con este tipo de instrumento la afinación del tambor no es fija, sino que varía al ir amortiguándose el sonido. Esto requiere el uso de un oscilador controlado por voltaje (VCO), alimentado por la salida de voltaje de control del configurador de envolvente. La figura 18 suministra el diagrama de circuito de un VCO.

Los VCO empleados en los circuitos para música están casi invariablemente basados en amplificadores operativos de transconductancia, pero este circuito es una excepción, dado que está basado en un amplificador operacional dual. El circuito es, virtualmente, un oscilador de onda triangular/cuadrada del tipo que emplea un Integrador Miller y un Disparador Schmitt. Aquí, IC1a actúa como integrador y IC1B aporta la función de disparador. El voltaje de control se obtiene por adición de TR1 y sus circuitos asociados. Esta configuración logra una buena linealidad en una banda ancha de barrido.

El voltaje de control podría obtenerse directamente de la salida de voltaje de control del configurador de envolvente, pero esto podría restringir a la unidad a una banda de gran anchura y ofrecería escaso control sobre el efecto obtenido. En lugar de ello, R1 y R2 actúan como mezclador pasivo que combina el voltaje de control procedente de el configurador de envolvente con un **bias** ajustable procedente de VR1. VR2 permite que varíe la cantidad de voltaje de barrido añadido al **bias** fijo con una banda muy ancha. Por consiguiente, VR1 actúa como un control de afinación, mientras que VR2 es el control de la anchura de la banda de barrido. Junto con el control del tiempo de amortiguación del generador de envolvente, todos ellos ofrecen un control considerable sobre el sonido producido y permiten la obtención de toda una gama de efectos útiles.

Componentes del VCO (figura 18)

Resistencias (1/4W 5%)

R1	10K
R2,8,11	22K (3)
R3,10,12	100K (3)
R4,5,6	47K (3)
R7,9	8K2 (2)
VR1	22K lin
VR2	1M lin

Condensadores

C1	100 μ F 16V electrolítico
C2	1nF carbonato

Semiconductores

IC1	CA3240E
TR1	BC109

Miscelánea

Zócalo DIL de IC de 8 patillas
Placa de circuito, cable, etc.

Sonidos basados en el ruido

Muchos sonido percusivos, como es el caso de los platos y las palmas, están basados en el ruido. Un sonido de plato realmente bueno requiere algunos circuitos avanzados, pero pueden obtenerse buenos

resultados empleando una fuente de ruido blanco, en especial si existen opciones de filtraje.

El ruido blanco es el familiar “soplo” que normalmente intentamos minimizar en los circuitos electrónicos. Hay muchos modos de generar este tipo de ruido y prácticamente cualquier componente electrónico genera, al menos, una pequeña cantidad del mismo. Para producir un circuito generador de ruido razonablemente sencillo, necesitamos un componente que genere un nivel de ruido bastante alto. Probablemente la mejor elección sea un transistor de silicio ordinario. En este circuito (figura 19), TR1 y la resistencia de carga R1 actúan como fuente de ruido. R1 introduce un **bias** inverso en la unión base-emisor de TR1, lo que hace que éste entre en avalancha como un diodo zener; también, como un diodo zener, genera crestas de ruido. De hecho, podría emplearse un diodo zener de bajo voltaje en lugar de TR1, pero casi con seguridad daría un voltaje de señal de salida menor.

Prácticamente cualquier transistor NPN de silicio funcionará, en mayor o menor medida, en la posición TR1, e incluso uno de tipo PNP puede funcionar si se transponen las conexiones a la base y el emisor. Si se utiliza con una alimentación de 12 voltios, la elección del transistor no resulta crítica, aunque algunos dispositivos producirán mayor salida que otros. Si la unidad se alimenta por medio de una pila de 9 voltios, las cosas resultan algo más delicadas, ya que los voltajes de ruptura de algunos dispositivos son demasiado elevados como para permitir un funcionamiento por debajo de los 9 voltios. Una pila de 9 voltios suministra entre 7,5 y 8 voltios bajo carga y cuando está a punto de agotarse. El BC184 funcionó bien empleando una pila de 9 voltios, pero probablemente la solución más práctica sea instalar un soporte para transistor en el panel de componentes en lugar de TR1, y después probar con una variedad de dispo-

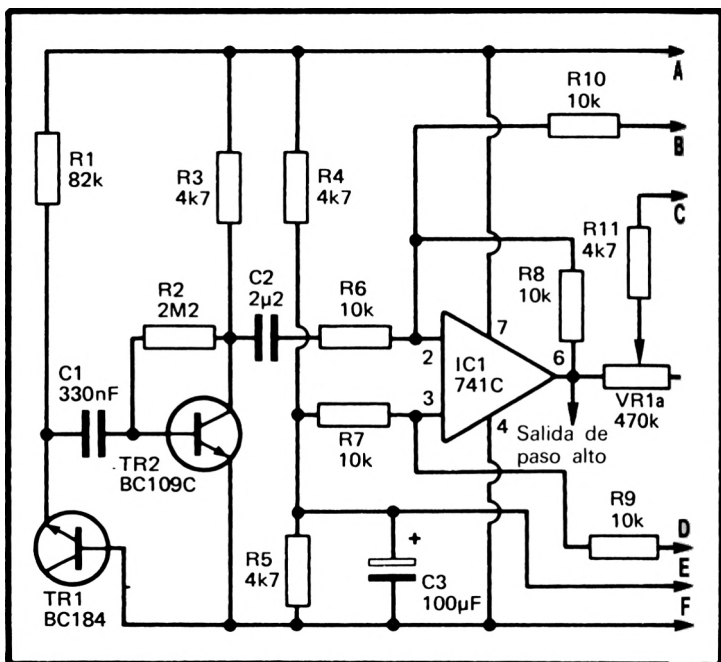
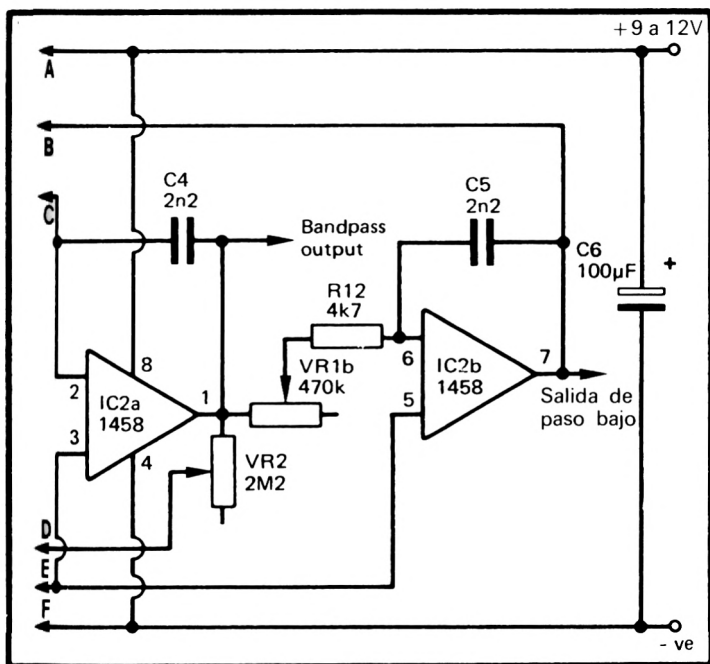


Figura 19.—Generador de ruido

sitivos de la caja de recambios hasta encontrar aquel que dé mejores resultados.

La producción de ruido en TR1, aunque relativamente elevada, probablemente sea bastante baja en términos absolutos (típicamente de alrededor de uno o dos milivoltios RMS). TR2 se emplea, pues, como amplificador emisor común de alta ganancia que aumenta la señal hasta niveles más útiles.

El resto del circuito, que es un filtro de estado variable de alta resonancia, suministra un filtraje versátil. Es un circuito convencional de tres amplificadores operacionales con salidas de paso alto, paso de banda y paso bajo. En esta aplicación, probablemente sean las salidas de paso alto y de banda las



para sonidos tipo címbalo

que produzcan los efectos más útiles, pero pueden preferirse los efectos producidos por la salida de paso bajo. Si se desea, se puede introducir un conmutador de tres vías para lograr una selección por conmutador de los tres tipos de filtraje. La frecuencia de corte puede variarse por medio de VR1 desde unos pocos cientos de Hz hasta los límites superiores de la banda de frecuencias de audio. VR2 es el control de “Q” o resonancia, el cual permite que la banda pasante se haga muy estrecha, con ganancia elevada en la zona central de respuesta. Si se emplea un filtraje de paso alto, el control de resonancia puede producir un pico pronunciado en la respuesta cerca de la frecuencia de corte, originando lo que es, en realidad, un cruce

entre los filtrajes de paso alto y de banda. También se obtiene un pico cerca de la frecuencia de corte cuando se acciona un filtro en la modalidad de paso bajo.

Utilizando el filtraje de paso alto, una frecuencia de corte y una resonancia bastante altas, y un tiempo de amortiguación de unos dos segundos, se obtiene un efecto tipo platos. Una mayor resonancia y un tiempo de amortiguación breve producen un efecto tipo palmada. No obstante, existe una amplia gama de efectos y vale la pena dedicar algún tiempo a la experimentación con los controles para encontrar los efectos más a nuestro gusto. No se deben pasar por alto los interesantes efectos que pueden obtenerse utilizando también una frecuencia de corte baja.

Componentes del generador de ruido (figura 19)

Resistencias (1/4W 5%)

R1	82K
R2	2M2
R3,4,5,11,12	4K7 (5)
R6,7,8,9,10	10K (5)
VR1	470 doble lin
VR2	2M2 lin

Condensadores

C1	330nF carbonato
C2	2 μ 2 carbonato o poliéster
C3,6	100 μ F16V electrolíticos (2)
C4,5	2n2 carbonato (2)

Semiconductores

Tr1	BC184 (véase el texto)
Tr2	BC109C
IC1	741C
IC2	1458C

Miscelánea

Placa de circuito, cables, etc.

Sonidos metálicos

Los sonidos “metálicos”, como las campanas o los gongs, son, en su mayor parte, altamente complejos. Mientras que la mayor parte de los sonidos musicales se producen por medio de un elemento vibrador unidimensional, como una cuerda o un tubo, los instrumentos como un gong o una campana son bi o tridimensionales. En lo que concierne a los sonidos producidos, esto tiene por resultado la generación de componentes de frecuencia que no están armónicamente relacionados, mientras que la señal de salida de, por ejemplo, un instrumento de cuerda no contiene prácticamente ninguna frecuencia no relacionada armónicamente.

Es posible producir electrónicamente una señal que contenga fuertes componentes no armónicos utilizando dos osciladores y un tipo de mezclador llamado modulador de anillo. Un modulador de anillo suprime ambas señales de entrada en la salida, pero genera fuertes señales sumadas y restadas. Como ejemplo sencillo, asumamos que los dos osciladores operan a 1 y 1,5 kHz. Las frecuencias suma y diferencia son: 3,5 kHz ($1 \text{ kHz} + 2,5 \text{ kHz} = 3,5 \text{ kHz}$) y 1,5 kHz

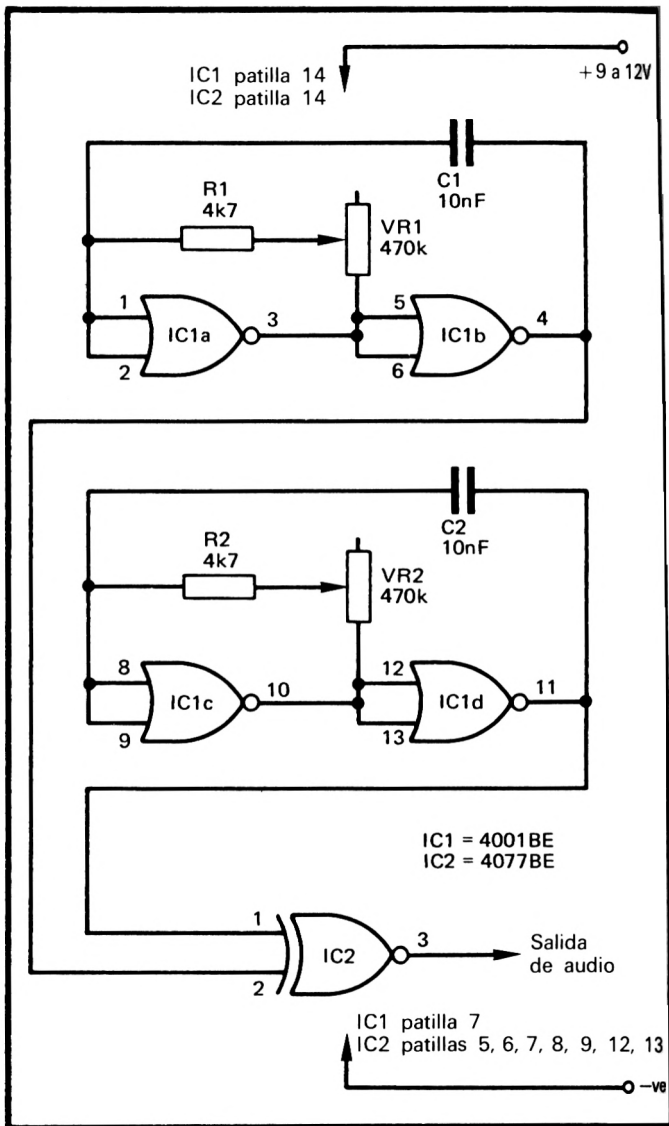


Figura 20.—Modulador de anillo para sonidos metálicos

(2,5 kHz — 1 kHz = 1,5 kHz), respectivamente. Esto supone que las señales de entrada son ambas ondas sinusoidales puras y, si tienen otra forma de onda, se genera una serie compleja de frecuencias de entrada. En la práctica las señales de entrada de onda cuadrada parecen dar buenos resultados.

El doble oscilador con el circuito de modulación, en anillo de la figura 20, es extremadamente sencillo, pero junto con el configurador de envolvente puede suministrar buenos sonidos metálicos. Los dos osciladores son CMOS estables estándar, y sus frecuencias de salida pueden ajustarse con límites muy amplios por medio de VR1 y VR2. Los moduladores de anillo llegan a ser circuitos bastante complejos, pero las cosas pueden simplificarse mucho en este caso, dado que nos enfrentamos a señales de onda cuadrada. De hecho, el modulador de anillo consiste tan sólo en una puerta lógica, el IC2. También existen otras tres puertas NOR exclusivas de dos entradas en IC2, pero no se utilizan en este circuito. Las entradas no utilizadas se conectan a tierra para protegerlas de voltajes estáticos imprevisibles.

Una puerta NOR exclusiva es un tipo de circuito lógico infrecuente y poco usado y, probablemente, muchos lectores lo desconozcan. Una puerta NOR ordinaria tiene una salida que pasa a baja si cualquiera de las entradas pasa a alta, o si lo hacen ambas entradas. Una puerta NOR exclusiva sólo difiere de esto en que la salida pasa a alta, y no a baja, si ambas entradas pasan a alta. Para nuestros fines del momento, esto da excelentes resultados, con la aparición de fuertes señales de adición y sustracción en la salida.

Se puede obtener una gama útil de efectos probando VR1, VR2 y el control de amortiguación con diferentes valores. Para sonidos metálicos estándar (aparte de los de tipo campana y gong), los dos osciladores deberían estar separados con arreglo a

algún intervalo musical o muy ligeramente desafinados respecto a ese intervalo para producir una nota de batido grave en la señal de salida. Pueden obtenerse sonidos discordantes desafinando de modo notable ambos osciladores.

Componentes del modulador de anillo (figura 20)

Resistencias (1/4W 5%)

R1,2 4K7 (2)

Condensadores

C1,2 10nF poliéster (2)

Semiconductores

IC1 4001BE
IC2 4077BE

Miscelánea

Placa del circuito, cables, etc.

Software

Se pueden utilizar tres tipos generales de *software* con los sintetizadores. El primero es un programa que permite que los tambores sean disparados directamente accionando teclas del ordenador, con lo que

se puede tocar y perfeccionar esquemas rítmicos. El segundo tipo es una especie de grabador en tiempo real, en el que se pueden grabar las secuencias y después reproducirlas en un lazo continuo. El tercero es un programa estilo Compositor, que permite introducir esquemas rítmicos en forma de datos numéricos. Este método es el más útil para personas inexpertas en el terreno de la percusión que desean obtener una pista de apoyo sobre la que tocar con un sintetizador u otro instrumento.

Es evidente que el *software* tendrá que variarse con arreglo al tipo de ordenador que se emplee, pero si se trabaja con BASIC es muy fácil generar los impulsos de disparo. En primer lugar, se activan como salidas las líneas del puerto para el usuario (en caso necesario) y después, simplemente, se pone en alta cada salida, pasándola inmediatamente a baja de nuevo para generar impulsos de disparo. Este sencillo programa de demostración es para el Commodore 64, y permite que los tambores alimentados por las líneas PBO a PB3 del puerto para el usuario sean disparados tocando las teclas "1" a "4", respectivamente.

```
5 POKE 56579,255
10 GET A$:IF A$ = "" THEN 10
20 IFA$ ="1" THEN POKE 56577,1:POKE 56577,0
30 IFA$ ="2" THEN POKE 56577,2:POKE 56577,0
40 IFA$ ="3" THEN POKE 56577,4:POKE 56577,0
50 IFA$ ="4" THEN POKE 56577,8:POKE 56577,0
60 GOTO 10
```

Las líneas del puerto para el usuario se fijan todas como salidas en la línea 5 y el teclado se lee en la línea 10. Las líneas 20 a 50 generan un impulso en la salida apropiada correspondiente a la tecla accionada; después, la línea 60 enlaza el programa con la línea 10, donde se lee de nuevo el teclado del ordenador.

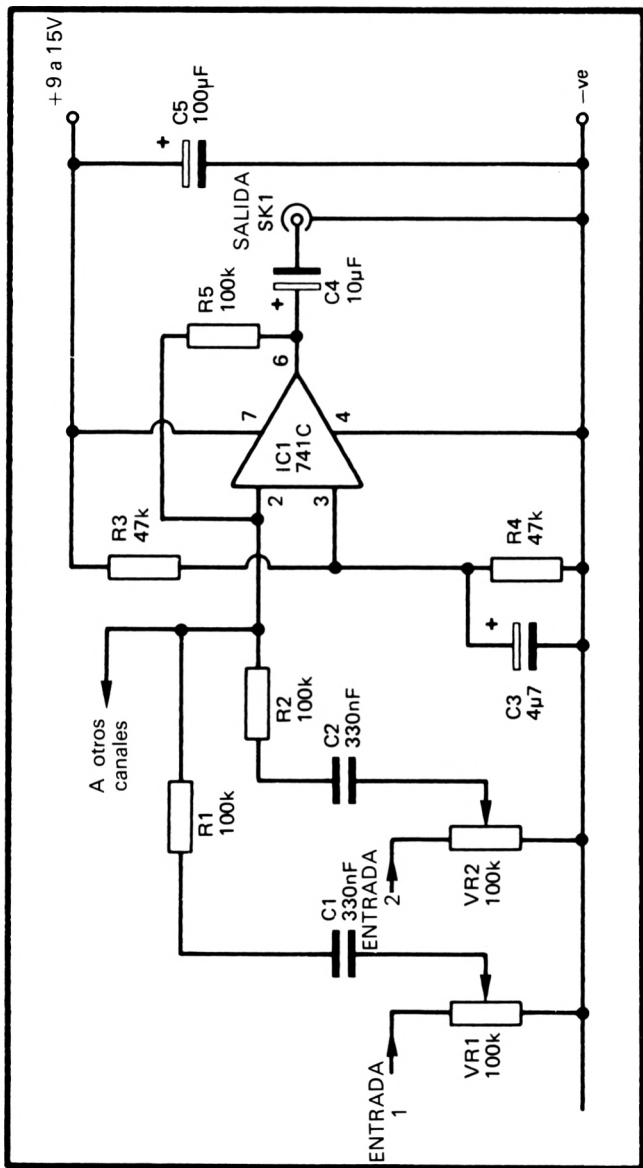


Figura 21.—Circuito mezclador simple

Cada sintetizador de percusión tiene un *jack* de salida y es necesario, por tanto, un mezclador exterior para combinar las señales. Alternativamente, puede emplearse el circuito sencillo de modalidad sumadora de la figura 21 para mezclar las señales y suministrar una salida única. Aunque este circuito tiene sólo dos entradas, como puede verse en la figura 21, se pueden tener todas las que se deseen añadiendo simplemente un potenciómetro, un condensador de bloque de corriente continua y una resistencia de entrada para cada canal adicional.

Componentes de un mezclador sencillo (figura 21)

Resistencias (1/4W 5%)

R1,2,5	100K (3)
R3,4	47K (2)
VR1,2	100K log (2)

Condensadores

C1,2	330nF (2)
C3	4 μ 7 63V electrolítico
C4	10 μ F 25V electrolítico
C5	100 μ F 16V electrolítico

Semiconductor

IC1	741C
-----	------

Miscelánea

SK1	Hembra de <i>jack</i> estándar
	Placa del circuito, cables, etc.

3

Síntesis digital

Muchos instrumentos musicales electrónicos de nuestros días emplean algún tipo de síntesis digital para generar la señal de salida. Existen dos formas básicas de enfocar la síntesis digital y, desde el punto de vista del *hardware*, el método más sencillo consiste en emplear un *software* que genere las formas de onda requeridas. En la práctica, suele ser más versátil utilizar un sistema de grabación digital para almacenar sonidos “reales” en la memoria en forma digital y después reproducirlos cómo y cuándo se quiera. Con todo, este sistema requiere más *hardware*, dado que tanto los circuitos de grabación como los de reproducción son necesarios. Este segundo sistema tiene otro uso en la música electrónica, una utilidad que es, por lo menos, tan importante como la síntesis de sonidos. Puede ser empleado como una línea de retardo (*delay*) digital, y con un ordenador que tenga alrededor de 30K de memoria es posible obtener retardos de hasta un segundo, lo que permite obtener un excelente efecto de eco.

Un punto que tenemos que señalar aquí es que, aunque la síntesis digital no resulta difícil desde el punto de vista del *hardware*, en especial si se dispone de un ordenador como el AMSTRAD, que está

ampliamente equipado con puertos de entrada y salida, el *software* requiere bastante experiencia, y el *hardware* no sirve en realidad para nada sin un *software* adecuado. El problema fundamental es que el BASIC no resulta apropiado para la síntesis digital de señales que no sean extremadamente sencillas, dado que es excesivamente lento. No es posible, por ejemplo, producir una señal sinusoidal de 1 kHz si el ordenador sólo puede enviar datos a un ritmo de unos pocos cientos de valores por segundo. Es esencial la utilización del código máquina o lenguaje ensamblador si se pretende obtener resultados aceptables.

Sintetizador digital

Empezaremos por un circuito que permite la síntesis digital de formas de onda a partir del *software*, en lugar de emplear un proceso de grabación. Este es un punto de partida evidente, ya que es sencillo, muestra los principios básicos implicados en la operación y puede ser empleado como circuito de reproducción de un sistema de grabación, como describiremos más adelante. El modo en que se sintetizan digitalmente las formas de onda viene explicado en el diagrama de forma de onda de la figura 22.

Con un sistema digital no es posible suministrar señales que varíen continuamente de voltaje como lo hacen las señales analógicas. El ordenador puede enviar una serie de valores y un convertidor digital-analógico los altera, convirtiéndolos en una serie de voltajes, pero la señal de salida salta casi instantáneamente de un voltaje a otro en simpatía con los cambiantes valores recibidos por el ordenador. Esto da una forma de onda escalonada en la salida, del tipo que puede verse en la figura 22, que es el equivalente digital de una forma analógica de onda en sierra. En

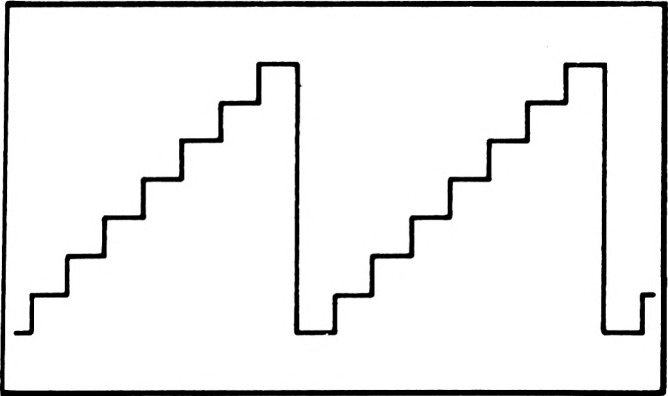


Figura 22.—Onda de forma de diente de sierra generada digitalmente

este ejemplo no hay más que siete escalones en la forma de onda, lo que la convierte en una representación digital un tanto pobre de una onda en sierra. Con un sistema de 8 bits podría haber hasta 256 niveles de voltaje diferentes (incluyendo el 0), lo que daría mucho mejor resultado. De hecho, una onda en sierra con 256 pasos, vista en un osciloscopio, resultaría casi con seguridad indistinguible de su equivalente analógico. Desde luego, los escalones no eran perceptibles en mi osciloscopio cuando lo comprobé. El oído humano discierne más que el osciloscopio medio y los componentes de distorsión de alta frecuencia producidos por los escalones de la onda podrían ser audibles, pero no tendrían ninguna consecuencia en la mayor parte de las aplicaciones prácticas. De hecho, el nivel de distorsión podría perfectamente ser inferior al de muchos circuitos analógicos generadores de onda en forma de sierra.

Desafortunadamente no siempre es posible emplear la resolución completa de 8 bits del sistema, y esto se debe al ritmo máximo, relativamente bajo, con el que la mayor parte de los ordenadores domésticos pueden

enviar datos a la salida. Como valor típico se pueden escribir alrededor de cien mil valores por segundo en un convertidor digital-analógico y, aunque esto pueda parecer más que suficiente, debe tenerse presente que las formas de onda complejas requieren entre 200 y 500 escalones por ciclo, si se pretende ampliar toda la resolución del sistema. A 100.000 pasos por segundo esto da una frecuencia de salida máxima en la banda de 200 a 500 Hz.

Es posible obtener frecuencias de salida más elevadas, pero sólo empleando una menor resolución y, por consiguiente, a costa de una mayor distorsión. Esto no representa el inconveniente que podría parecer a primera vista, ya que si bien aumenta el nivel de distorsión, muchos de los resultados de ésta tienen frecuencias que están por encima del límite superior del espectro de audio. Es normal incluir un filtro de paso bajo en la salida del convertidor digital-analógico para atenuar la distorsión fuera de banda que podría, en caso contrario, producir problemas en los equipos tales como los altavoces y amplificadores, que se alimentan con la señal de salida. El filtro de salida elimina también los ruidos que genera el convertidor digital-analógico al cambiar de un valor a otro. Incidentalmente, el filtro de paso bajo tiende a eliminar los escalones de la forma de onda de la señal de salida y, aunque la señal pueda componerse de unos pocos escalones, registrada con un osciloscopio puede tener una calidad respetable.

La figura 23 muestra el diagrama de circuito de un convertidor digital-analógico junto con un filtro de paso bajo para la síntesis digital de formas de onda. El convertidor digital-analógico se basa en el dispositivo ZN428E, que mencionamos anteriormente, por lo que no describiremos este componente ni los procedimientos para conectarlo a un ordenador. En este circuito, al contrario que en la anterior aplicación, se utilizan las ocho entradas.

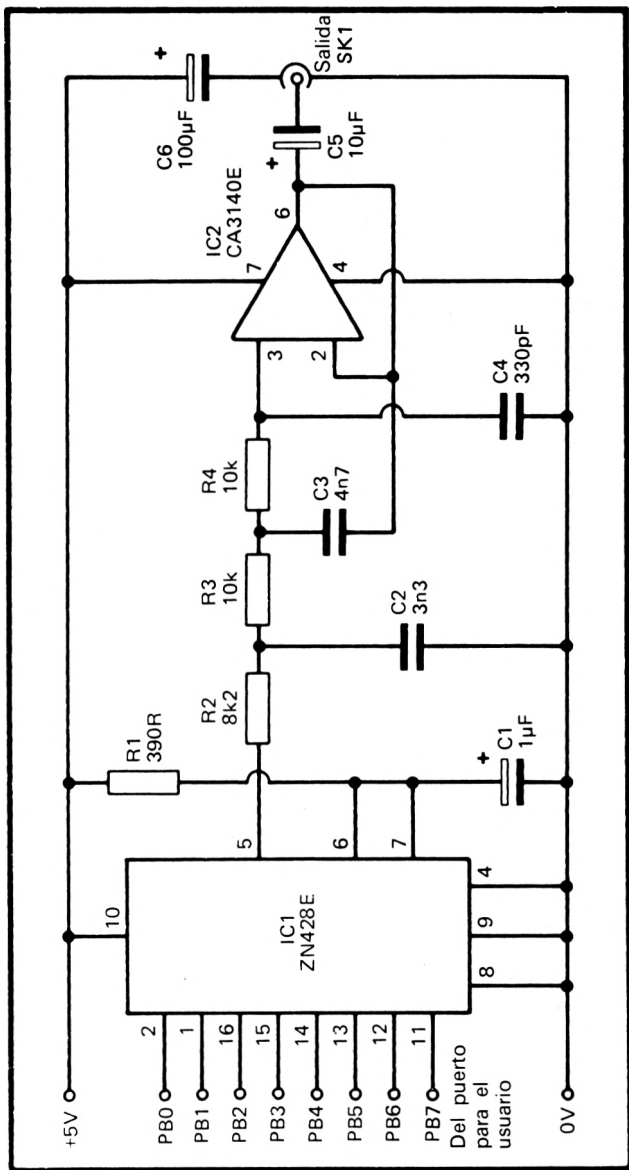


Figura 23.—Convertor D/A para síntesis

El filtro de salida es un circuito activo de tercer orden (18 dB por octava) que tiene como amplificador-amortiguador a IC2. La frecuencia de corte se encuentra aproximadamente a 10 kHz, lo que da algo menos de la banda de frecuencias completa, pero es lo suficientemente elevado como para obtener buenos resultados. Esta frecuencia de corte y el ritmo de *roll-off* garantiza que sólo alcancen la salida de la unidad cantidades mínimas de señal de alta frecuencia.

Hay que dominar dos aspectos del *software*. En primer lugar, hay que dejar a un lado una sección de RAM que se utilizará para almacenar una serie de números que den la forma de onda deseada, o se pueden suministrar varios bloques de formas de ondas de la RAM con los números correspondientes a varias formas de ondas. Para las formas de onda sencillas no resulta demasiado difícil establecer una secuencia adecuada de números, y un programa para introducirlos con POKE en la sección asignada de RAM. Como ejemplos, una forma de onda en sierra del tipo que aparece en la figura 22 sólo haría necesario que estuvieran en RAM los números 0 a 255 en secuencia. La forma de onda en diente de sierra invertido (con una rampa descendente) requeriría la presencia de los números 255 a 0. Una onda triangular necesitaría la presencia de los números 0 a 255 y de 255 a 0. En la práctica, es en general más fácil almacenar todas las formas de onda en bloques de 256 bytes (páginas) de RAM y, por consiguiente, sería lo mejor sacrificar un cierto grado de resolución y utilizar una secuencia de números pares del 50 al 255, descendiendo después otra vez a 0. Las señales en onda cuadrada y los impulsos son relativamente fáciles, y simplemente requieren una serie de ceros para establecer el período de salida “bajo” y una serie de 255 para establecer la duración de la salida “alta”. La mayor parte de los ordenadores tienen una función

SIN que puede emplearse para generar la serie de números necesaria para producir ondas sinusoidales de una calidad razonable. Se pueden utilizar números al azar para generar una señal de ruido.

Con uno o varios bloques de RAM cargados con números adecuados, el siguiente paso es que una rutina en el código máquina escriba estos números a su vez en el convertidor digital-analógico y hacer un bucle infinito si es necesaria una forma de onda repetitiva. Esta es la única parte del programa que tiene que escribirse en código máquina, pero es esencial que esté escrito en este código (o en lenguaje ensamblador) para obtener una velocidad operativa adecuada. Siempre y cuando los bloques de RAM que contienen los juegos de números sean de una longitud de 256 bytes y no crucen los límites de la página, no debería haber dificultad alguna para desarrollar una rutina apropiada; incluso para un principiante en el código máquina esto no debería plantear grandes problemas. Una cosa a tener presente es que las interrupciones deberán desactivarse cuando se emprenda cualquier forma de síntesis, grabación o reproducción digital. Si no se hace así, es probable que las rutinas como el barrido de teclado y la función del temporizador incorporado produzcan modulación y distorsión en la señal de salida.

Como comentábamos más arriba, la rutina en código máquina producirá tan sólo una frecuencia de salida. No obstante, la frecuencia de salida puede reducirse no añadiendo instrucciones de operación o añadiendo bucles de retardo para controlar la velocidad con la que ascienden o descienden las formas de onda. Se pueden obtener frecuencias más elevadas utilizando un byte sí y otro no del bloque de RAM, o tal vez incluso sólo uno de cada diez o veinte, si se busca una frecuencia de salida realmente elevada. Este produce una reducción en la resolución de la señal de salida, pero, como explicábamos antes,

ésta no debería ser perceptible para nadie al escuchar la señal de salida. Este método para aumentar la frecuencia de salida ofrece un control bastante grosero, pero si se usa en conjunción con la ausencia de instrucciones de operación o con lazos de temporización para ralentizar el ritmo de salida, se puede obtener una amplia gama de frecuencias. Cuando se utilizan tan sólo unos pocos números para sintetizar formas de ondas, hay que intentar emplear siempre números que abarquen un abanico muy amplio, preferiblemente aproximándose al rango máximo utilizable de 0 a 255. Esto da una potente señal de salida de un voltaje cercano al nivel total de salida de 2,55 voltios de pico a pico.

Componentes del convertidor D/A (figura 23)

Resistencias (1/4W 5%)

R1	390R
R2	8k2
R3,4	10K (2)

Condensadores

C1	1 μ F 63V electrolítico radial
C2	3n3 carbonato o poliéster miniatura
C3	4n7 carbonato o poliéster miniatura
C4	330pF placa cerámica
C5	10 μ F 25V electrolítico
C6	100 μ F 10V electrolítico

Semiconductores

IC1	ZN428E
IC2	CA3140E

SK1 Hembra de *jack* estándar
Soporte DIL de IC de 16 patillas
Soporte DIL de IC de 8 patillas
Cable de ordenador
Placa de circuito, carcasa, cables, etc.

Grabación digital

En teoría, es posible sintetizar formas de onda muy complejas y obtener prácticamente cualquier frecuencia con una precisión razonable. Es incluso posible sintetizar una forma de onda que produzca dos o tres frecuencias de salida, dando una forma de salida polifónica. Que valga o no la pena hacer esto es otra cuestión, y podría hacer falta una gran cantidad de esfuerzo y *software* complejo para lograr unos resultados que pueden obtenerse más fácilmente de otros modos. En realidad, esto es una cuestión de preferencias personales, al igual que lo es el decidir si ésta es o no una línea de trabajo interesante. En última instancia, es muy improbable que pueda producirse el *software* necesario para utilizar a fondo las posibilidades del sistema. Obtener el máximo resultado de un sistema de este tipo requiere alguna forma de sistema de grabación digital que permita almacenar los sonidos y posteriormente introducirlos en RAM. Luego, éstos pueden ser reproducidos con la forma de la señal original, o la señal puede modificarse (por ejemplo, se puede alterar su afinación).

Hay más de un modo de utilizar un sistema de este tipo y vale la pena dedicar algún tiempo a examinar algunos de ellos. El procedimiento más sencillo consiste en hacer una grabación digital de un instrumento de percusión y después limitarse a

reproducir la grabación cada vez que ese sonido es necesario. Aunque la mayor parte de los ordenadores domésticos no dan un tiempo de grabación superior a uno o dos segundos, esto resulta obviamente suficiente para algunos instrumentos de percusión. Un enfoque diferente consiste en realizar una grabación muy breve (de unos 100 ms) y hacer un lazo continuo durante la reproducción para obtener una señal constante. Se utiliza después un configurador de envolvente convencional para procesar el sonido, igual que se hace en un sintetizador analógico. En otras palabras, puede adoptarse un cruce entre la síntesis analógica convencional y la síntesis digital. Al utilizar esta técnica, es importante hacer un lazo en la grabación en una sección de RAM que dé valores de comienzo y final razonablemente bien emparejados. En caso contrario, existe el peligro de que se produzca un ruido audible cada vez que la grabación enlaza de nuevo con el principio.

Otro modo de emplear el sistema consiste en efectuar una grabación en una sección de RAM, pero con el programa enlazando continuamente con el comienzo del bloque de RAM y volviendo a grabar sobre él. Si se escribe el valor de cada byte de RAM en el circuito de reproducción inmediatamente antes de utilizarlo para la regrabación, se obtiene una acción equivalente a una línea de retardo. Utilizando *chips* analógicos de línea de retardo de bajo precio, es imposible obtener retardos superiores a 100 ms con una banda pasante razonablemente ancha, pero con una de tipo digital basada en un ordenador doméstico adecuado, es posible obtener un retardo de un segundo o más con una banda pasante considerable de alrededor de 10 kHz. Un sistema digital de 8 bits no da una calidad equiparable con la de la alta fidelidad, pero sí ofrece niveles de distorsión que soportan perfectamente la comparación con la mayor parte de los circuitos analógicos de línea de retardo. Una línea

de retardo digital puede emplearse para obtener cualquiera de los efectos en los que se utilizan líneas analógicas de retardo, incluyendo el *chorus* y el *flanger*. Probablemente, la utilización más popular sea la obtención de un largo efecto de eco, dado que esto es algo que está más allá de las posibilidades de la mayor parte de las líneas de retardo analógicas.

Lo único necesario para obtener un sistema de grabación digital, basado en un ordenador doméstico, es un circuito de reproducción como el descrito anteriormente en este capítulo, y un convertidor analógico-digital, para que actúe como circuito de grabación (o “digitizador de audio” como a veces se llama a estos circuitos). La señal de audio debe muestrearse a un ritmo que sea al menos el doble de la frecuencia de entrada máxima. Si se utiliza un filtro de paso bajo en la entrada del digitizador para restringir a 10 kHz la banda pasante de la señal de entrada, esto permite emplear una frecuencia de muestreo mínima de 20 kHz, pero el ritmo de muestreo debería ser, preferiblemente, de 30 kHz o más. Esto permite obtener un tiempo de retardo razonablemente largo con la mayor parte de los ordenadores domésticos, requiriendo entre 20 y 30K de RAM por cada segundo de retardo. Un ordenador doméstico típico con alrededor de 30K de RAM disponibles podría, por tanto, suministrar un retardo máximo de entre 1 y 1,5 segundos.

Con un ritmo de muestreo de 20.000 a 30.000 muestras por segundo, el convertidor analógico-digital ZN449 (como el utilizado en el circuito lector de teclados anteriormente descrito) es más que adecuado en este aspecto, con su capacidad máxima de muestreo superior a las 100.000 muestras por segundo. No es ideal debido a que no ofrece una linealidad óptima de 8 bits, pero esto puede resolverse empleando una de las versiones de más calidad. El ZN447 es el que ofrece mayor rendimiento y es el

mejor para esta aplicación. No obstante, el ZN449 dará un rendimiento razonable y resulta satisfactorio si no se quiere incurrir en el coste adicional del ZN447. También se puede llegar a un compromiso entre el coste y el rendimiento usando el ZN448 (que es el dispositivo empleado en el prototipo de digitizador). La figura 24 muestra el diagrama de circuito del digitizador de audio.

El IC1 se utiliza de modo muy similar al del *chip* convertidor del circuito lector de teclado, pero hay unas pocas diferencias importantes que merece la pena subrayar aquí. La más evidente es que en esta aplicación se emplean las ocho salidas del IC1 y, por consiguiente, las ocho líneas de datos del puerto para el usuario quedan ocupadas. Hay que encontrar algún otro medio para accionar la entrada "iniciar conversión" del IC1. Con los ordenadores VIC-20 y BBC modelo B. Probablemente la mejor opción sea utilizar la línea de inicio de diálogo (*handshake*) CB2, del puerto para el usuario. Esta puede disponerse para que opere en una modalidad en la que produzca un breve impulso negativo cada vez que se envíen datos al puerto del usuario. Así pues, puede producirse un pulso de iniciación de la conversión introduciendo cualquier valor en el puerto para el usuario (que se limita a ignorar el dato, dado que todas sus líneas están dispuestas como entradas). CB2 se fija en la modalidad apropiada escribiendo un valor de 160 en el registro de control de periféricos en &FE6C en el BBC modelo B, y de 37148 en el VIC-20.

El ordenador Commodore 64 no tiene la línea CB2 en el puerto para el usuario, pero tiene en su lugar la PC2. Esta produce un breve impulso negativo tras cualquier operación de lectura o escritura sobre el puerto para el usuario. Esto resulta muy conveniente en esta aplicación, ya que se puede utilizar PC2 para enviar automáticamente un impulso de iniciación de la conversión tras cada lectura del convertidor. Es

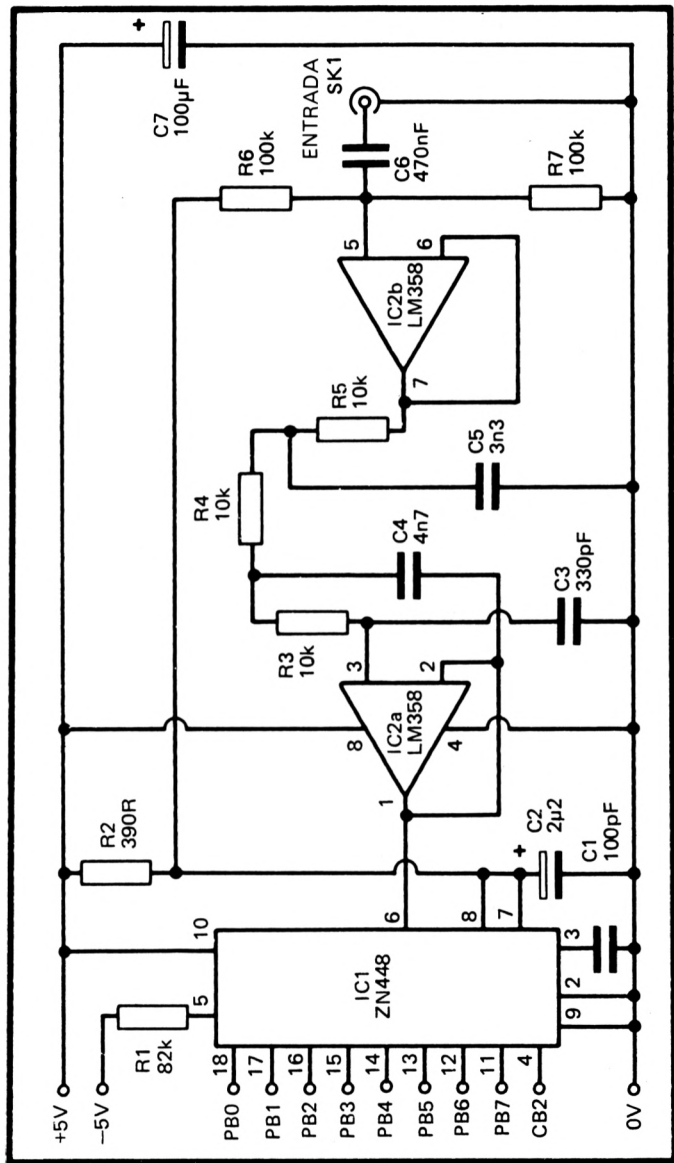


Figura 24.—Circuito digitalizador de audio

evidente que no todos los ordenadores tienen un equivalente del PC2 o el CB2, pero cualquier salida digital puede emplearse para accionar la entrada de iniciación de la conversión, empleando el *software* para generar el impulso negativo en caso necesario.

En el AMSTRAD, la forma más adecuada de conseguir un pulso negativo es utilizar la línea STROBE, la cual produce el pulso de nivel bajo durante un breve instante cada vez que se envía algún dato al puerto de impresora.

El método de interfaz directa al *bus* de datos de un ordenador, anteriormente descrito, resulta también adecuado en este caso. El impulso de iniciación de la conversión puede obtenerse a través de un circuito decodificador de coordenadas. Al utilizar el sistema como línea de retardo es evidentemente necesario que los circuitos de grabación y reproducción estén conectados simultáneamente al ordenador y, por tanto, no es posible accionar ambos circuitos desde el puerto del usuario. Allá donde sea aplicable, la solución más sencilla consiste en accionar el circuito de reproducción desde un puerto de impresora en paralelo y conectar el circuito de grabación al puerto para el usuario. La mayor parte de los ordenadores no tienen los dos tipos de puerto. En este caso, es necesario conectar al menos uno de los circuitos a los *buses* del ordenador.

Si queda libre alguna entrada digital, puede emplearse para monitorizar la salida de estatus de “fin de la conversión” del IC1. Con todo, no es éste el método que yo recomendaría y probablemente resulte mucho más fácil emplear una serie de instrucciones de no operación para suministrar un retardo que impida una lectura prematura del convertidor.

La resistencia “de cola” R1 se alimenta desde una salida de -5 voltios del ordenador. Podría emplearse aquí el método de derivar una alimentación negativa del rail de $+5$ voltios empleado en el circuito lector

del teclado, lo cual se tendría que hacer así si el ordenador no dispone de un rail de alimentación negativa adecuado. Si el ordenador dispone de una salida de -5 voltios, es preferible emplear este suministro, ya que así se evita la posibilidad de que se produzca un acoplamiento fortuito del oscilador del generador de alimentación negativa con el camino de la señal en el circuito de grabación.

El filtro de entrada del IC1 es esencialmente el mismo de la salida del circuito de reproducción. Va precedido por una etapa amortiguadora de unidad de ganancia (IC2b) que asegura que el filtro sea alimentado por medio de una impedancia de fuente razonablemente baja. La entrada del circuito tiene un **bias** procedente de R6 y R7 utilizando la fuente de voltaje de referencia de IC2. Esto da una lectura de quiescencia de alrededor de 128. La señal de entrada de audio varía por encima y por debajo del nivel de quiescencia. En esta aplicación, un **bias** de entrada de 0 voltios no sería el apropiado, ya que tendría como resultado la rectificación en media onda de la señal a través del sistema.

Es necesario un nivel nominal de señal de entrada de 2,55 voltios de pico a pico para accionar a fondo la entrada del circuito y, en interés del buen rendimiento, el nivel máximo de entrada debería aproximarse siempre a esta cifra. En caso contrario, los niveles de ruido y distorsión serán más elevados de lo realmente necesario.

El *software* tendrá que ser diseñado, obviamente, para que se ajuste a la propia aplicación al microprocesador empleado en el ordenador y al rango de coordenadas de la RAM utilizable. En una aplicación de grabación sencilla es necesario algún método de retención para evitar que se realice la grabación antes de tiempo. Podría emplearse algún tipo de disparo automático, pero no hay por qué recurrir a nada más complicado que accionar una sola tecla del ordenador

para iniciar la grabación. Hará falta un contador para detener la grabación cuando la sección de RAM asignada haya quedado llena, que puede ser bien un registro de 16 bits en el microprocesador o un par de bytes de RAM, dependiendo del tipo de microprocesador que se esté empleando. El programa de reproducción no es más, en realidad, que una ligera variación del de grabación, en el que se toman los valores de RAM en lugar de escribirlos sobre ella y en el que la generación de pulsaciones de conversión resulta innecesaria. Con todo, hay algo de lo que se debe estar pendiente: vigilar que el programa se reproduzca a la misma velocidad a la que se grabó, ya que en caso contrario se producirá una variación en la afinación de la señal de salida. Por supuesto, es perfectamente correcto diseñar las cosas deliberadamente de modo que se obtenga una variación de la afinación, si es eso lo que se desea. Si el programa funciona a la máxima velocidad de grabación posible, tal vez resulte que el ritmo de muestreo es bastante elevado, lo que da un tiempo de grabación bastante bajo para la cantidad de RAM utilizada. Se pueden emplear algunas instrucciones de no operación, o un corto bucle de retardo, para ralentizar las cosas un poco en caso necesario. Pero se debe tener cuidado de no permitir que el ritmo de rastreo se vuelva demasiado lento, ya que esto daría una señal de salida de muy baja calidad.

En una aplicación de línea de retardo, hay que diseñar el *software* para que envíe a la salida el valor almacenado en cada byte de RAM inmediatamente antes de volver a escribir sobre él, y el programa debe reiniciarse continuamente. Idealmente, el tiempo que tarda el programa en enlazar con el principio debería ser igual al tiempo que tarda el programa en desplazarse normalmente de una coordenada a otra. No obstante, no parecen producirse graves consecuencias si aparece una ligera discrepancia. La longitud del

retardo se varía alterando el tamaño del bloque de RAM utilizado. Alternativamente, el tamaño del bloque de RAM puede ser constante y tan grande como sea posible, pero los valores escritos sobre el circuito de reproducción pueden tomarse 1K, 2K o el valor que sea, por detrás del punto en el que se escriben los nuevos valores. Probablemente este segundo método sea la solución más elegante, pero también es la más difícil de implementar, y existe el ligero peligro de que funcione con demasiada lentitud.

Al utilizar algunos ordenadores basados en el Z80, surge un ligero problema debido a la forma en que está dispuesto su mapa de entradas/salidas. En lugar de utilizar sólo las ocho líneas de coordenadas menos significativas para las coordenadas de entrada/salida, se emplean las dieciséis. Los ordenadores Sinclair ZX Spectrum y Amstrad CPC entran dentro de esta categoría. Esto hace que las instrucciones que utilizan el registro B como contador sean inutilizables, ya que el registro B es necesario para suministrar los 8 bits más significativos de las coordenadas. A continuación ofrecemos una rutina básica de línea de retardo que funciona con los ordenadores basados en el Z80, cualquiera que sea el método utilizado en el mapa de entrada/salida:

```
DI
LD DE,1
LD BC,coordenadas del puerto
LD HL,coordenadas del primer byte de RAM
LD IY,$FFFF- tamaño del bloque de RAM
LDA,(HL)
OUT (C),A
INI
ADD IY,DE
JP C,iniciar en coordenada más 4 (decimal)
NOP
```

NOP
NOP
NOP

JP NC,coordenadas de iniciación más 14 (decimal)

La primera instrucción desactiva las interrupciones y, como señalábamos anteriormente, en el caso de no hacerse esto, se producirá una señal de audio moderada en la salida. Seguidamente se carga el par de registro DE con un valor de 1 y el BC con las coordenadas del puerto de entrada/salida. El par de registro HL se emplea para suministrar las coordenadas de la RAM, que es alimentada, en principio, con las coordenadas de iniciación (las coordenadas más bajas del bloque de RAM). El par de registro IY se utiliza como un contador de 16 bits que determina el tamaño del bloque de RAM empleado para grabar. El valor almacenado ahí es igual a FFFF en hex (65535 en decimal) menos el número de bytes utilizados en el bloque de RAM. Por supuesto, hay que tener cuidado en seleccionar un área de memoria ocupada por RAM disponible, recordando siempre que es necesaria una pequeña cantidad de memoria para almacenar el programa.

La siguiente instrucción carga el acumulador con los datos de las coordenadas contenidas en el par HL (las primeras coordenadas del bloque de RAM) y la siguiente instrucción envía el valor introducido en el acumulador al puerto de salida. Aunque el registro B no se mencione específicamente en esta instrucción, esto obedece tan sólo a que no sería utilizado para suministrar parte de las coordenadas de salida con el sistema estándar de entradas y salidas del Z80. Aun así, se encarga de suministrar el número de 8 bits de A8 a la posición A15 del *bus* de coordenadas. La siguiente instrucción toma el valor del puerto de entrada cuyas coordenadas están en el par BC y lo introduce en las coordenadas de memoria contenidas

en el par HL. El par HL sufre entonces un incremento automático de 1, pero al registro B se le sustrae 1. Seguidamente, el valor de 1 del par DE se suma al par IY. Aparte de incrementar el valor de IY en 1, esto permite comprobar los registros IY para determinar si han recorrido ya su ciclo atravesando \$FFFF, volviendo a 0 (lo que fijaría el indicador de arrastre). Si esto ya ha ocurrido, el programa salta de vuelta al punto en el que se introduce en el par de registro BC la dirección del puerto de entrada/salida, y, desde ahí, continúa como anteriormente. Nótese que deben volverse a introducir las coordenadas del puerto, ya que la instrucción INI alteró el contenido del registro B (aunque esto no es necesario en un ordenador que utilice coordenadas de entrada/salida de tan sólo 8 bits). Si el indicador de arrastre no está fijado, se llevan a cabo algunas instrucciones de no operación, lo que no hace más que igualar los dos lazos y reduce el ritmo de muestreo a un nivel más adecuado. Seguidamente, el programa enlaza de nuevo con la instrucción LD A,(HL). Dado que HL sufrió un incremento automático debido a la instrucción INI, el programa se desplaza al siguiente byte de RAM cuyo contenido se envía al puerto de salida, que después es sustituido por el valor del puerto de entrada.

No se utilizan instrucciones RET al final del programa, dado que el proceso de enlazado impide que se llegue a esta instrucción. Probablemente la única forma de salir del programa sea apretar el botón de *reset*. Hemos asumido aquí que el puerto de entrada tiene las mismas coordenadas que el puerto de salida, pero si no fuera éste el caso, sólo hay que añadir una instrucción LD BC,**** entre las instrucciones OUT e INI para que se produzca el cambio a las coordenadas correctas de entrada.

Este programa sirve para ilustrar las funciones básicas de las que debe disponer el *software*, pero,

obviamente, sólo es aplicable a los ordenadores basados en Z80 y en una aplicación de línea de retardo. Con todo, si se comprende claramente el modo en que debe funcionar el sistema, y si uno es capaz de escribir rutinas sencillas en código máquina o lenguaje ensamblador para un ordenador, no debería haber dificultad alguna en idear rutinas simples apropiadas para la aplicación a emplear.

Componentes del digitizador de audio (figura 24)

Resistencias (1/4W 5%)

R1	82K
R2	390R
R3,4,5	10K (3)
R6,7	100K (2)

Condensadores

C1	100pF placa cerámica
C2	2 μ 2 63V electrolítico
C3	330 pF placa cerámica
C4	4n7 carbonato o poliéster en miniatura
C5	3n3 carbonato o poliéster en miniatura
C6	470nF carbonato o poliéster en miniatura
C7	100 μ F 10V electrolítico

Semiconductores

IC1	ZN448
IC2	LM358

SK1 Hembra de *jack* estándar
Soporte DIL de IC de 18 patillas
Soporte DIL de IC de 8 patillas
Cable de ordenador, placa del circuito, carcasa,
etcétera.

Compander (Compresor-expansor)

Aunque el sistema básico de grabación y reproducción aquí descrito da una calidad de audio razonable, un aspecto en el que está lejos de resultar ideal es en términos del ruido producido en su funcionamiento. La importancia del nivel de ruido depende de la aplicación, pero dado que tiene una relación señal/ruido de alrededor de 46 dB, resultaría inadecuado en este aspecto para muchas aplicaciones. Esta situación puede mejorarse grandemente utilizando un sistema *compander* de reducción de ruidos, con el que se puede lograr una notable relación señal/ruido de alrededor de 80 dB.

Un *compander* es, en realidad, dos circuitos, un compresor que se emplea durante la grabación y un expansor que se emplea durante la reproducción. El compresor es un circuito que tiene un nivel de ganancia que varía en simpatía con la amplitud media de la señal de entrada. El circuito está dispuesto de modo que la ganancia en voltaje se reduce al aumentar la amplitud media. Esto eleva las señales de bajo nivel, con lo que quedan muy por encima del nivel de ruido, pero no se produce aumento alguno en la ganancia en las señales de alto nivel, con lo que no se recortan ni se distorsionan.

Por sí misma, la distorsión permite que la señal deseada destaque por encima del nivel de ruido de fondo, pero tiene el efecto indeseable de que

distorsiona la dinámica de la señal, dando diferentes grados de intensidad sin que la señal se extinga nunca. El expansor contrarresta esto, suministrando un nivel de ganancia de voltaje que disminuye en simpatía con las reducciones en la amplitud media de la señal. Las señales de alto nivel, como ocurre con el caso del circuito compresor, pasan a través de éste sin cambiar de amplitud, pero las señales de bajo nivel quedan atenuadas reinstaurándose así la dinámica de la señal original. Como mínimo, si el compresor y el expansor tienen características complementarias, el efecto global es no producir cambio alguno en los niveles dinámicos.

Esto puede parecer irrelevante, pero lo que hay que tener en cuenta es que las señales de bajo nivel se atenúan durante la reproducción para hacerlas descender a su nivel original. Esta atenuación se aplica también a cualquier ruido del sistema y el nivel de "soplo" de fondo queda reducido. Con un nivel de señal alto, el sistema tiene poco o ningún efecto, pero esto no tiene importancia, dado que la señal será lo suficientemente fuerte como para enmascarar el ruido de fondo y la baja relación señal/ruido no resultará audible para los oyentes de la señal de audio.

Los sistemas compresores-expansores pueden ser bastante complejos y pueden resultar muy difíciles de ajustar adecuadamente una vez montados. Afortunadamente, existen ya este tipo de sistemas en circuito integrado, lo que permite construir un sistema muy eficaz, a un precio razonablemente bajo, utilizando muy pocos componentes y sin necesidad de ninguna clase de ajuste. El dispositivo utilizado en los circuitos compresor y expansor de las figuras 25 y 26 es un NE571. También el NE570 funciona en el circuito y sólo difiere del otro en que ofrece una calidad de audio ligeramente superior. El NE571 probablemente sea el más adecuado de los dos dispositivos, ya que es más barato que el NE570 y

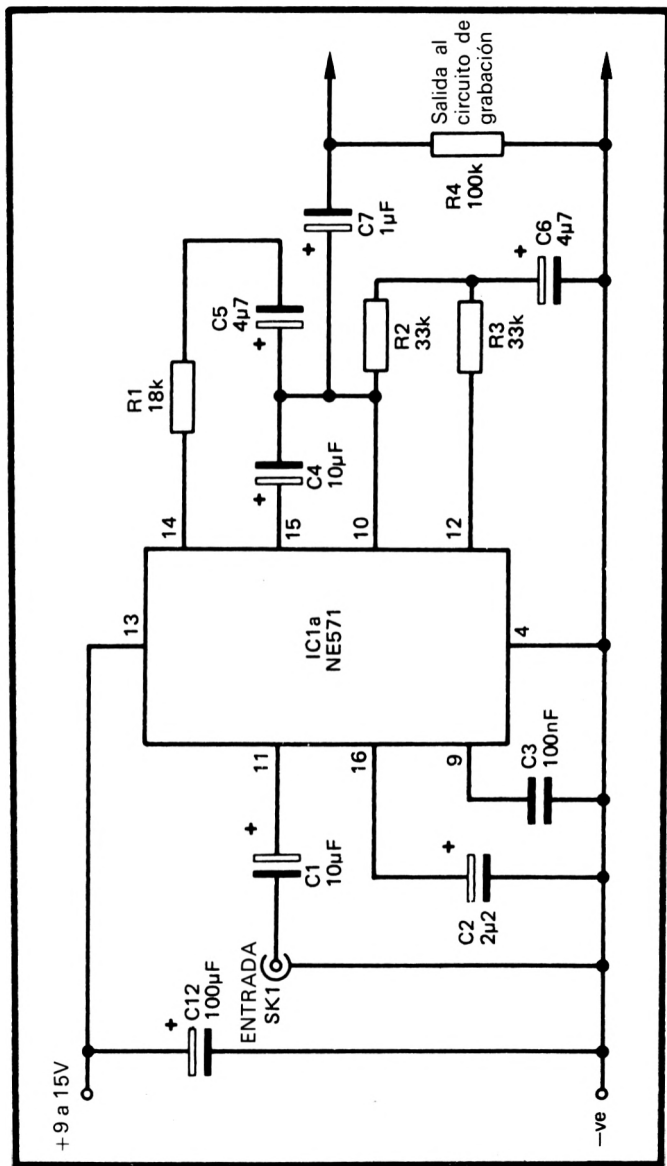


Figura 25.—Sección compresora de un circuito de reducción de ruido

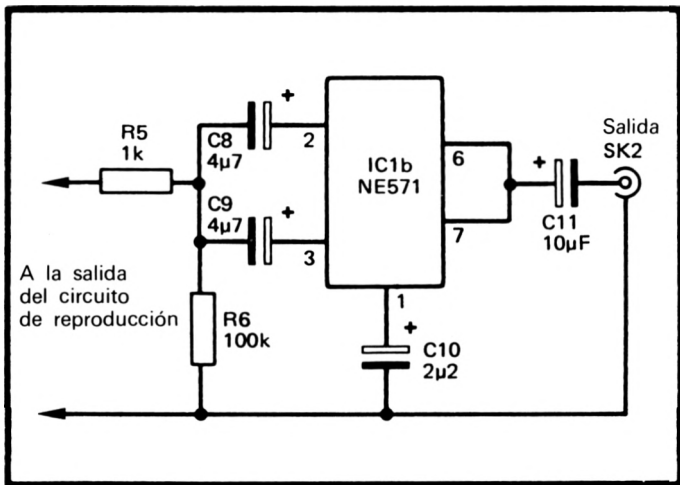


Figura 26.—Esquema de la sección expansora de un compresor

tiene un nivel de rendimiento perfectamente adecuado para esta aplicación.

Sólo hace falta un NE570 o NE571 para un sistema compresor-expansor, dado que cada dispositivo contiene dos juegos idénticos de circuitos, uno de ellos para su uso con el compresor y el otro para el expansor. Los bloques básicos de circuitos se conectan de modo diferente para darles una u otra función. Cada sección del dispositivo está formada por un rectificador de precisión, un amplificador controlado por voltaje, una fuente de voltaje de referencia (utilizada para el **bias**) y un amplificador operativo.

Si consideramos, en primer lugar, el funcionamiento del compresor, el amplificador controlado por voltaje se conecta en el circuito de realimentación negativa del amplificador operativo y éste se introduce en la línea de la señal. El rectificador de precisión procesa la señal de salida del circuito y, tras suavizar el voltaje en CC resultante, éste se emplea

para accionar la entrada de control del amplificador controlado por voltaje. Con un nivel bajo de señal, la señal de salida y el voltaje de control son bajos, lo que produce una escasa ganancia en el amplificador controlado por voltaje. No obstante, esto lleva poca realimentación al amplificador operativo que, por consiguiente, tiene una gran ganancia de voltaje. A niveles de señal más altos, la señal de salida y el voltaje de control se vuelven mucho mayores, produciendo una mayor ganancia en el amplificador controlado por voltaje. Esto produce una gran cantidad de realimentación hacia el amplificador operativo que, por consiguiente, tiene un nivel de ganancia de voltaje reducido, con lo que se obtiene la deseada compresión.

El expansor es algo más simple y el amplificador controlado por voltaje se introduce en la línea de la señal. De hecho, también el amplificador operativo se inserta en la línea de la señal, pero se limita a actuar como un amplificador amortiguador en la salida del amplificador controlado por voltaje. La señal de entrada va al rectificador de precisión e, igual que antes, su señal queda suavizada y se emplea para suministrar el voltaje de control del amplificador controlado por voltaje. Al ser accionado el rectificador de precisión por la señal de entrada, el voltaje de control que va al amplificador controlado por voltaje es proporcional a la amplitud media de la señal de entrada. Por tanto, el voltaje de control aumenta con la señal de entrada, aportando el incremento necesario en la ganancia y en la expansión de volumen.

El NE570 y el NE571 requieren un voltaje de alimentación mínimo de 6 vatios y, por consiguiente, no funcionarán adecuadamente con la alimentación a 5 vatios obtenida de un ordenador. Cualquier alimentación razonablemente bien filtrada de entre +9 y +15 vatios dará buenos resultados. Una batería de

9 vatios será suficiente como fuente de alimentación si el ordenador carece de una salida de alimentación apropiada.

Los circuitos expansor y compresor tienen una característica de 2:1 que, en teoría, significa que duplican la relación señal/ruido de alrededor de 62 a 92 dB. En la práctica, hay que tomar en cuenta factores como el ruido de los circuitos compresores y expansores. A pesar de todo, esto sigue permitiendo obtener una relación señal/ruido de alrededor de 80 dB, cifra que, de hecho, es superior a la de muchos sistemas de alta fidelidad.

Componentes del sistema de reducción de ruidos (figuras 25 y 26)

Resistencias (1/4W 5%)

R1	18K
R2,3	33K (2)
R4,6	100K (2)
R5	1K

Condensadores

C1,4,11	10 μ F 25V electrolíticos (3)
C2,10	2 μ 2 63V electrolíticos (2)
C3	100 nF poliester
C5,6,8,9	4 μ 7 63V electrolítico
C12	100 μ F 16V electrolítico

Semiconductor

IC1	NE571 o NE570
-----	---------------

Soporte DIL de IC de 16 patillas, clavijas de entrada y salida, placa de circuito, cable, etc.

Preamplificador para micrófono

Si se desea emplear el sistema para registrar sonidos “en vivo”, es necesario añadir un preamplificador de micrófono delante del circuito grabador, ya que los micrófonos tienen un nivel de señal demasiado bajo como para poder accionar adecuadamente el circuito de grabación. El diagrama de circuito de la figura 27 es el de un preamplificador sencillo de micrófono para su uso con un micrófono dinámico de alta impedancia o uno de características similares (como un micrófono de electroto con un transformador de salida incorporado).

La señal de salida de un micrófono de alta impedancia es, como mucho, de unos pocos milivoltios RMS y podría estar, incluso, por debajo de un milivoltio RMS. En términos de voltaje de pico a pico, esto se corresponde con un nivel de señal muy bajo, de unos 2,5 milivoltios o, lo que es lo mismo, la milésima parte del nivel necesario para accionar el circuito, de 2,55 voltios de pico en pico. Por consiguiente, el preamplificador deberá dar una ganancia en voltaje de alrededor de 60 dB.

Este diseño se basa en un amplificador operativo dual. Una sección del dispositivo (1C1a) actúa como un amplificador sin inversión con una ganancia de voltaje de un factor de alrededor de 48 (34 dB). R3 fija la impedancia de entrada del circuito nominalmente en 22K, lo que es apropiado para la mayor parte de los micrófonos de alta impedancia. R1, R2 y C2 forman una derivación central de las líneas de

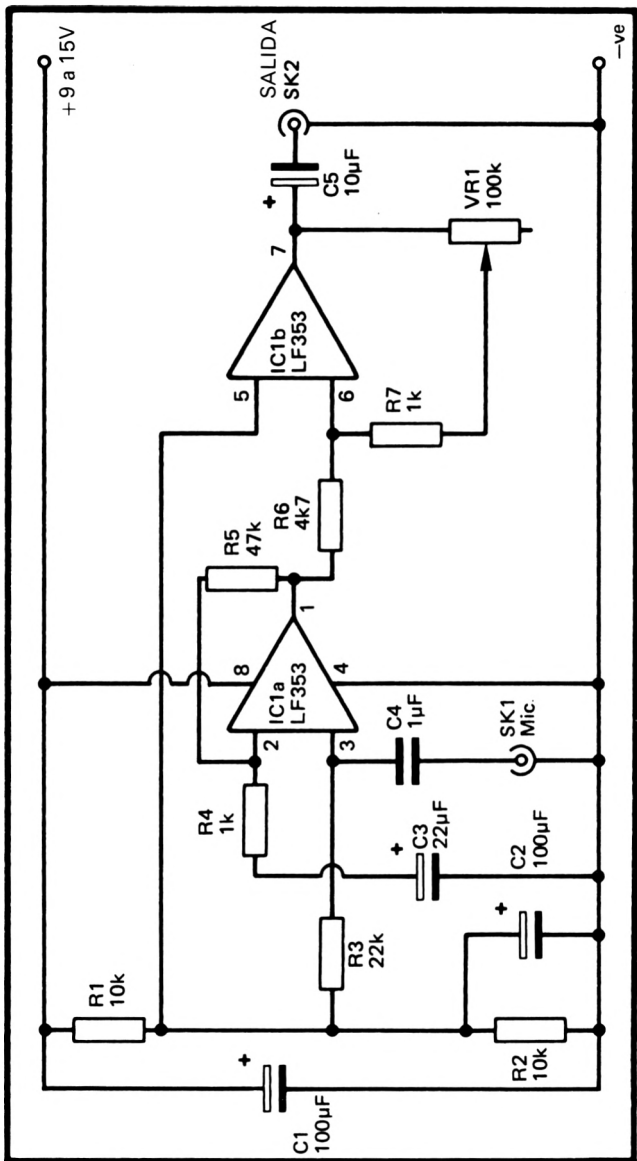


Figura 27.—Circuito preamplificador para microfono

alimentación, que se utiliza como fuente de **bias** para las dos secciones de IC1. Incidentalmente, IC1 es un tipo *bifet* que da un nivel de ruido muy bajo. También ofrece un rendimiento excelente en lo que a distorsión se refiere. Nótese que al ser IC1 un dispositivo Jfet y no un MOSfet, no requiere precauciones antiestáticas para su manejo.

La segunda etapa del IC1 opera en su modalidad de inversión, y se acciona directamente desde la salida del IC1a. R6, R7 y el control de ganancia VR1 constituyen el circuito de realimentación negativa. La ganancia en voltaje de esta etapa se puede variar por medio de VR1 de un factor de alrededor de veinte veces (26dB), con VR1 en su resistencia máxima, hasta una pérdida de alrededor de 14 dB, con VR1 en su resistencia mínima. Esto hace que el circuito produzca la ganancia global máxima de voltaje de aproximadamente 60 dB.

El circuito es bastante sensible a la recogida accidental de ruidos eléctricos. Los sistemas digitales, incluidos los ordenadores, producen prolíficas cantidades de este tipo de ruido. Es recomendable construir el preamplificador como una unidad independiente dentro de una carcasa de metal conectada al rail de alimentación negativo. Probablemente, la mejor fuente de alimentación sea una pequeña pila de 9 vatios, que es una alimentación totalmente libre de ruidos. Por supuesto, el cable del micrófono debe ser un cable blindado con la pantalla conectada al rail de alimentación negativo del preamplificador. Aunque el circuito tiene un nivel alto de ganancia de voltaje junto con una banda pasante razonablemente ancha, dado que la entrada y la salida están fuera de fase, no hay demasiado riesgo de que la realimentación accidental produzca inestabilidad, por lo que la disposición de los componentes no es particularmente crítica.

Componentes del preamplificador de micrófono (figura 27)

Resistencias (1/4V 5%)

R1,2	10K (2)
R3	22K
R4,7	1K (2)
R5	47K
R6	4K7
VR1	100K lin

Condensadores

C1,2	100 μ F16V electrolíticos (2)
C3	22 μ F 16V electrolítico
C4	1 μ F carbonato o poliéster en miniatura
C5	10 μ F 25V electrolítico

Semiconductor

IC1	LF353 o TL072
-----	---------------

Miscelánea

- SK1,SK2 Hembras *jack* de 3,5 mm
- Soporte DIL de IC de 8 patillas
- Placa del circuito, carcasa metálica, etc.

Eco

Para obtener un efecto repetitivo de eco es necesario tener un mezclador en la entrada del circuito digital de retardo. Una de las entradas se alimenta con la señal de entrada y, en la otra, se introduce la salida de

la línea de retardo. Variando la cantidad de señal retardada que se mezcla con la señal de entrada, puede alterarse el tiempo de atenuación del efecto de eco. Hay que tener precaución de no emplear un exceso de realimentación, ya que esto hará que la señal de eco aumente de nivel en lugar de desvanecerse gradualmente, dando lugar a lo que acabaría siendo una señal de ruido.

El circuito mezclador de la figura 21, descrito anteriormente, es perfectamente adecuado para esta aplicación. La señal de entrada sería enviada directamente a C1 y se omitiría VR1. La señal retardada se introduciría en la entrada 2 del circuito.

4

Interfaces MIDI

En el momento en el que escribo esto, el interfaz MIDI empieza a convertirse en una característica estándar en los instrumentos electrónicos, pero en realidad no es nuevo, ya que su existencia se remonta a dos o tres años atrás, al igual que los instrumentos que utilizan el sistema MIDI. La idea del interfaz MIDI es simple, tal vez obvia, y sólo tiene como fin permitir que se pueda utilizar un instrumento para controlar a otro, de modo que ambos puedan combinarse de modo eficaz para funcionar como un único instrumento. De hecho, es perfectamente factible enlazar varios instrumentos por medio de MIDI, haciéndolos funcionar como un instrumento grande y sofisticado.

El enlace entre instrumentos electrónicos fue también posible en el pasado, pero de un modo un tanto rudimentario. Además, si uno poseía instrumentos de más de un fabricante, resultaba difícil conseguir que uno de ellos controlara a los otros. En algunos casos lo único necesario era un cable especial, o un interfaz sencillo, pero en otros resultaba imposible crear un enlace eficaz entre los instrumentos.

MIDI, que significa **M**usical **I**nstrument **D**igital **I**nterface (Interfaz digital para instrumentos musicales), tiene como misión permitir que un instrumento

equipado con MIDI (u otro mecanismo) pueda ser conectado a cualquier otro instrumento que disponga de él, al margen de qué instrumento sea y quiénes sean sus fabricantes. Lo que es más, no es simplemente un sistema básico que se limita a transmitir información sobre puertas o valores de notas. Está diseñado para poder comunicar otras informaciones, tales como la fuerza con la que se aprieta una tecla en un teclado sensible al tacto. Por otra parte, un instrumento que sólo requiera algo relativamente básico, como impulsos de puerta, puede seguir siendo utilizado con un instrumento más sofisticado, y será diseñado de tal forma que ignore toda información que no sea relevante para su funcionamiento. Por tanto, es perfectamente posible emplear, por ejemplo, un sintetizador de 16 canales sensible al tacto con una caja de ritmos electrónica, siempre y cuando, claro está, que ambos tengan una interfaz MIDI.

Control por ordenador

Aunque hasta el momento sólo hemos considerado la posibilidad de conectar un instrumento MIDI a otro, lo que hace que el interfaz MIDI resulte especialmente útil es la posibilidad del control por ordenador. El conectar un ordenador doméstico a un instrumento MIDI no es muy difícil en la mayor parte de los casos y puede aumentar grandemente la versatilidad del instrumento. En especial, existe la posibilidad de utilizar, por ejemplo, un sintetizador polifónico y una caja de ritmos con un programa tipo Composición. Muchos ordenadores domésticos tienen generadores de sonido incorporados que pueden generar una música bastante notable cuando se emplean con un buen programa de Composición, pero no puede compararse con los resultados que se pueden

obtener con los sintetizadores, pianos eléctricos y demás instrumentos equipados con MIDI que hay hoy en día en el mercado. Con un ordenador doméstico, un *software* adecuado y uno o dos instrumentos equipados con MIDI, lo que se tiene es virtualmente una orquesta personal.

Datos en serie

Antes de entrar a considerar un interfaz práctico para conectar un instrumento MIDI a un ordenador, tal vez valga la pena explicar algunas ideas básicas acerca del funcionamiento del MIDI. Una diferencia importante entre el sistema MIDI y cualquier otro de los interfaces descritos en los anteriores capítulos de este libro, es que es totalmente digital. Si es necesaria alguna conversión analógico-digital (y muchos instrumentos electrónicos son hoy totalmente digitales), ésta tiene lugar en el interior del instrumento MIDI y no es algo por lo que deba preocuparse el usuario.

Otra diferencia entre el interfaz MIDI y otros examinados hasta ahora en este libro, es que es de tipo serial. En el interfaz del generador de CV los 6 bits de datos se escriben simultáneamente sobre el convertidor digital-analógico, utilizando seis líneas diferentes; es decir, el procedimiento habitualmente denominado “en paralelo”. En un sistema seriado los datos se transmiten por una sola línea (más una línea de tierra). Pero esto, obviamente impide la transmisión simultánea de todos los bits. En su lugar, deben ser transmitidos de uno en uno y, en el caso de la mayor parte de los sistemas seriados, incluyendo el MIDI, el bit menos significativo es el que primero se envía, seguido después por toda la secuencia hasta el bit más significativo. Esto tiene la ventaja de que sólo hacen falta dos cables para conectar una pieza de

equipo a otra, en lugar de entre siete y nueve cables (dependiendo del número de bits empleados en el sistema). Lo que ya es menos obvio es que, en los sistemas de datos en paralelo, normalmente es necesario tener cables apantallados, y que éstos sean cortos (de alrededor de dos metros o menos) para evitar la corrupción de datos debida al acoplamiento accidental de un cable con otro. Con los sistemas de datos seriados se pueden emplear normalmente cables bastante largos sin ningún problema.

En la forma fundamental, anteriormente delineada, un sistema serial no puede funcionar apropiadamente, dado que el equipo receptor no tendrá forma de saber en qué momento comprobar la línea de transmisión para determinar el estado de cada bit. Un modo de resolver el problema es utilizar un sistema sincrónico en el que se utiliza una tercera línea que transmite una señal de reloj, o algún otro tipo de señal de sincronización, para indicar el comienzo de cada corriente de datos. Con todo, la mayor parte de los sistemas seriados son asíncronos, y el sistema MIDI pertenece a esta categoría. Con este sistema no existen líneas de conexión adicionales, sino que se transmiten bits adicionales junto con los bits de datos. El más importante es el bit de “iniciación”, que indica al equipo receptor que debe muestrear la línea de datos a intervalos regulares a partir de ese momento, hasta que se haya verificado el estado de final de cada byte, y estos bits reciben el nombre de bits de “parada”. Se utilizan tan sólo como parte de un procedimiento sencillo para comprobar si hay errores. En ocasiones, se añaden bits extra llamados bits de “paridad”, y también éstos se emplean como parte de un proceso simple de comprobación de errores. No obstante, éstos no se aplican a los interfaces MIDI. La figura 28 ilustra el modo en que se transmiten los datos seriados, lo que podría servir para clarificar un tanto el sistema.

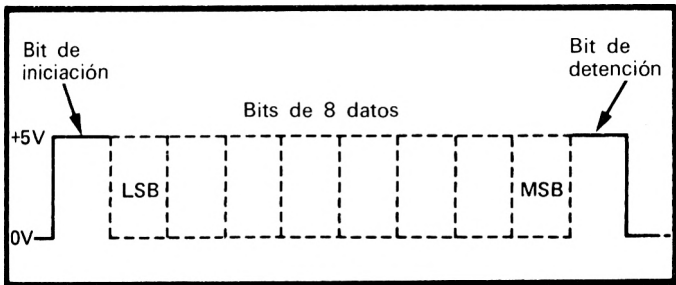


Figura 28.—Estructura del conjunto de datos en serie utilizados por el sistema MIDI

Existen una serie de formatos diferentes de palabra de uso común en los sistemas seriales, pero para propósitos de MIDI se emplea un bit de arranque, 8 bits de datos, 1 bit de parada y ninguna paridad. Los sistemas de datos seriales emplean toda una variedad de tasas en baudios. Existen varias estandarizadas que van de 50 a 19.200 baudios. La tasa en baudios no es más que el número de bits transmitidos por segundo si existe una corriente continua de datos. El sistema MIDI actúa originalmente a 19.200 baudios, pero recibió críticas por ser demasiado lento (a pesar del hecho de que es la tasa más rápida de uso común, aparte de la del sistema MIDI final). De hecho, una tasa en baudios de 19.200 no es tan rápida como pudiera parecer, si se tiene presente que hay que transmitir 10 bits para transferir un byte de información. Esto da un máximo absoluto de 1.920 bytes por segundo o, lo que es lo mismo, de poco más de 0,5 milisegundos por byte. Incluso teniendo presente que normalmente hay que transmitir tres bytes para producir cada respuesta del instrumento, y que pueden controlarse varios canales, seguirían siendo necesarios tan sólo unos pocos milisegundos para enviar bloques de datos. Resulta debatible si esto podría producir problemas audibles, pero, en el

sistema final, la tasa en baudios se aumentó hasta 31.250.

El sistema MIDI es esencialmente el mismo de los puertos seriales RS232C y RS423 instalados en muchos ordenadores domésticos, pero en pequeños detalles existen problemas que hacen que resulte muy improbable que se pueda emplear una salida seriada de un ordenador para accionar una entrada de MIDI. El primer problema es que los interfaces MIDI emplean niveles TTL estándar de 5 vatios y normalmente tienen entradas opto-aislantes para evitar problemas con los lazos a tierra. Los puertos RS232C utilizan niveles de señal que son nominalmente de + y -12V (+ y -5V en el caso del RS423). Esto podría resolverse con etapas de procesamiento de la señal adecuadas, pero el problema de la tasa de baudios es algo más difícil. 31.250 baudios no es una tasa estándar, y no es tampoco una tasa que se pueda alcanzar con ningún ordenador doméstico que yo conozca. Tal vez fuera posible modificar el interfaz serial para operar con una tasa en baudios correcta, pero, casi con seguridad, esto haría que resultara inutilizable para otros propósitos y seguramente no sería una buena idea.

Como veremos más adelante, los circuitos integrados, ideados fundamentalmente para su uso en los interfaces RS232C y otros similares, pueden llegar a trabajar perfectamente en una aplicación MIDI. Utilizando uno de estos dispositivos no resulta demasiado difícil añadir una salida MIDI a la mayor parte de los ordenadores. Incidentalmente, la tasa en baudios de 31.250 puede parecer una elección un tanto extraña, pero en realidad resulta conveniente, ya que se puede obtener una frecuencia de reloj apropiada para el dispositivo de interfaz dividiendo una señal de 1 mHz por 32; una señal de 2 mHz por 64, o una señal de 4 mHz por 128.

Bytes de control

El *hardware* MIDI necesario para enlazar dos piezas de equipo sólo resuelve la mitad del problema y, para que este enlace pueda tener algún valor práctico, debe existir algún mecanismo estándar para interpretar los datos enviados de una pieza de equipo a otra. Se emplea un sistema de códigos, de manera que todo aquel que desee experimentar con el control por ordenador de equipos MIDI debe tener al menos un grado de comprensión aceptable de cómo operan estos códigos.

Hacen falta 3 bytes para enviar cada nota al sintetizador, y éstas suministran las siguientes funciones:

Byte 1: Información de estatus que indica que los siguientes dos bytes se refieren al disparo de una nota y el número del canal de MIDI.

Byte 2: Valor de la nota.

Byte 3: Valor de la velocidad.

Hay 16 canales MIDI disponibles, y éstos se seleccionan empleando los números 0 a 15. Esto puede producir cierta confusión, ya que a menudo se encuentran referencias de que los canales MIDI van del 1 al 16. La diferencia es que el rango 1 a 16 se emplea solamente con fines de identificación, mientras que los números del 0 al 15 son los que se usan, de hecho, para seleccionar estos canales. Así pues, si el sintetizador con el que estamos trabajando funciona en el canal 12, el valor del canal empleado para enviarle datos sería el 11. Algunos instrumentos sólo actúan en un canal particular, pero en el caso de la mayor parte de los instrumentos modernos se puede disponer que el instrumento actúe en cualquier canal que se desee.

Por supuesto, la idea de disponer de varios canales es permitir alimentar varios instrumentos desde una salida de MIDI y que cada instrumento sólo responda a los datos concretos que le están destinados (que contendrán el número correcto de identificación del canal). Es posible hacer que dos instrumentos empleen el mismo canal, pero sólo si se debe alimentar a ambos con los mismos datos. En el caso de los sintetizadores polifónicos u otros instrumentos multicanal, es prácticamente seguro que habrá un canal de MIDI independiente para cada voz, de modo que cada una de las voces puede ser controlada independientemente. Incidentalmente, la mayor parte de los instrumentos MIDI tienen tres patillas hembras para MIDI: Estas son las de entrada, salida y una tercera que es el MIDI THRU. Esta última simplemente permite que los datos enviados a la patilla de entrada puedan mandarse a otro instrumento. De este modo, pueden controlarse varios instrumentos desde una única salida MIDI, si así se desea.

El valor de la velocidad sólo se aplica a los instrumentos sensibles al tacto, pero este byte también será transmitido por los instrumentos que carecen de esta prestación. Normalmente transmitirán el valor máximo de 127. Es esencial que se transmita este byte para mantener la compatibilidad entre diversos tipos de instrumentos MIDI. Si algunos instrumentos transmitieran más bytes por nota que otros, esto haría, evidentemente, que no pudieran funcionar juntos correctamente. Por supuesto, un instrumento no sensible al tacto se limitará a prescindir del valor del byte de la velocidad y no podrá emplearlo en modo alguno. Si examinamos ahora cada byte con algo más de detalle, descubrimos que el byte de encabezamiento está dividido en dos *nibbles* de 4 bits. El *nibble* más significativo está fijado en 1001, lo que indica que se va a disparar una nota. El menos significativo contiene el número del canal. Así pues,

10010011 especificaría que se va a disparar una nota (1001) en el canal 4 (0011). Pocos ordenadores permiten la entrada de datos en forma binaria y, en todo caso, ésta es una forma un tanto tediosa de hacer las cosas. Racionalizando las cosas, el primer byte es igual a 143 más el número de identificación del canal de MIDI (o 144 más el valor verdadero del canal de MIDI).

El valor de la nota pertenece al rango de 1 a 127, y las notas están espaciadas un semitono con el DO medio a un valor de 60. Esto da una extensión de diez octavas. Incluso aunque el sintetizador tenga tan sólo una extensión de teclado de cuatro o cinco octavas, probablemente haya hasta diez octavas completas disponibles a través del interfaz MIDI, o que, al menos, se puede lograr una extensión de este orden; no obstante, esto es algo que varía de un instrumento a otro. Un poco de experimentación servirá para determinar el rango de valores utilizable en cada instrumento.

Se emplea también un rango de valores de 1 a 127 para el byte de la velocidad. Los números bajos representan poca velocidad o una tecla pulsada con suavidad; los números elevados representan una velocidad elevada o una tecla pulsada con fuerza.

Hasta el momento hemos considerado este proceso como el disparo de una nota, pero lo que estamos haciendo realmente es activar la nota con la puerta. En otras palabras, estamos simulando la pulsación de la tecla, pero también debemos liberar la nota. Esto se hace de un modo muy similar a cuando se usa la puerta, y una vez más requiere tres bytes. El segundo y tercer byte son, de hecho, los mismos que se emplearon para activar la nota, pero el primer byte es ligeramente diferente. El código binario del *nibble* más significativo es 1000, pero el byte menos significativo es, de nuevo, el valor del canal de MIDI. El número decimal a emplear aquí es, pues,

127 más el número de identificación del canal de MIDI, o 128 más el valor verdadero del canal de MIDI.

Generalmente, es posible controlar otros parámetros por medio de un interfaz MIDI empleando el código de encabezamiento apropiado y los datos correctos, pero las prestaciones varían considerablemente de un instrumento a otro. El manual del instrumento debería ofrecer detalles completos acerca de lo que puede controlarse a través del interfaz MIDI, junto con los detalles apropiados del código de control. No es esencial utilizar otros códigos que los delineados más arriba, dado que permiten controlar los parámetros importantes y permiten lograr una secuenciación multicanal sensible al tacto. Cada voz puede ajustarse para que produzca el efecto deseado antes de iniciar la secuencia, y no es necesario abordar este tipo de tarea a través del interfaz MIDI, a menos que hagan falta cambios en el transcurso de una secuencia.

Modalidades

Existen tres tipos de modalidades de operación del MIDI que recibieron originalmente los nombres “Omni”, “Poly” y “Mono”, pero estos nombres han sido sustituidos hoy por los números 1, 3 y 4, respectivamente. Los instrumentos más sofisticados tienen las tres modalidades, pero los tipos más sencillos tienen tan sólo una o dos. La idea básica consiste en escoger una modalidad que, cuando sea necesario, rebaje al instrumento sofisticado que controla el sistema al mismo nivel que el menos sofisticado que está siendo controlado.

Omni (1)

Todos los instrumentos MIDI tienen esta modalidad como valor de omisión cuando se encienden. En esta modalidad el equipo receptor responderá a la información nota *on/off* al margen del canal por el que se transmitan. El modo concreto en el que las voces individuales de un instrumento polifónico responden a las notas recibidas depende del método de asignación interna utilizado en el instrumento en cuestión. El punto fundamental de esta modalidad es que permite controlar toda una serie de instrumentos desde uno solo (o controlador), y todos los instrumentos tocarán al unísono.

Poly (3)

En esta modalidad pueden asignarse los instrumentos a un determinado canal y, por tanto, los datos pueden dirigirse a un único instrumento, aunque se estén alimentando varios desde una única fuente. Al igual que con la modalidad Omni, el modo en que responde un instrumento a cada nota depende de su método de asignación interna.

Mono (4)

Si un instrumento tiene varios canales, cada uno con una voz diferente, es posible, empleando esta modalidad, asignar cada voz a un canal de MIDI distinto, pero deben ser canales consecutivos. Si se está empleando un ordenador para obtener una secuenciación de múltiples partes, ésta es la mejor

modalidad a escoger, ya que ofrece un control individual sobre cada voz. Para la secuenciación monofónica bastará con la modalidad de omisión (Omni). Existe, incidentalmente, una modalidad 2, pero (al menos mientras escribo esto) es exclusiva de los sintetizadores Yamaha. Se limita a hacer que todos los canales de todos los sintetizadores toquen las mismas notas.

Interfaz MIDI

Virtualmente, cualquiera de los interfaces seriados utilizados normalmente para los puertos seriales RS232C, como el 6850 o Z80 DART, pueden suministrar un interfaz MIDI. La alta tasa en baudios no es en realidad un problema, dado que la mayor parte de los sistemas seriales de interfaz pueden operar a unas tasas en baudios máximas de entre 250 y 500 kilobaudios. La mayor parte de los dispositivos seriales de interfaz están diseñados para conectarse directamente a los *buses* de una familia concreta de microprocesadores, y no sería fácil hacerlos actuar desde un puerto para el usuario o cualquier otro puerto de salida en paralelo. Existe un tipo de dispositivo serial de interfaz que es la excepción a la regla, el tipo UART (Receptor/Transmisor Universal Asíncrono). Utilizando uno de ellos es posible producir un interfaz MIDI que debería funcionar con cualquier ordenador capaz de suministrar nueve líneas digitales de salida. Las figuras 29 y 30 corresponden al circuito de un interfaz MIDI “universal” de este tipo, y la unidad está basada en el UART industrial estándar 6402.

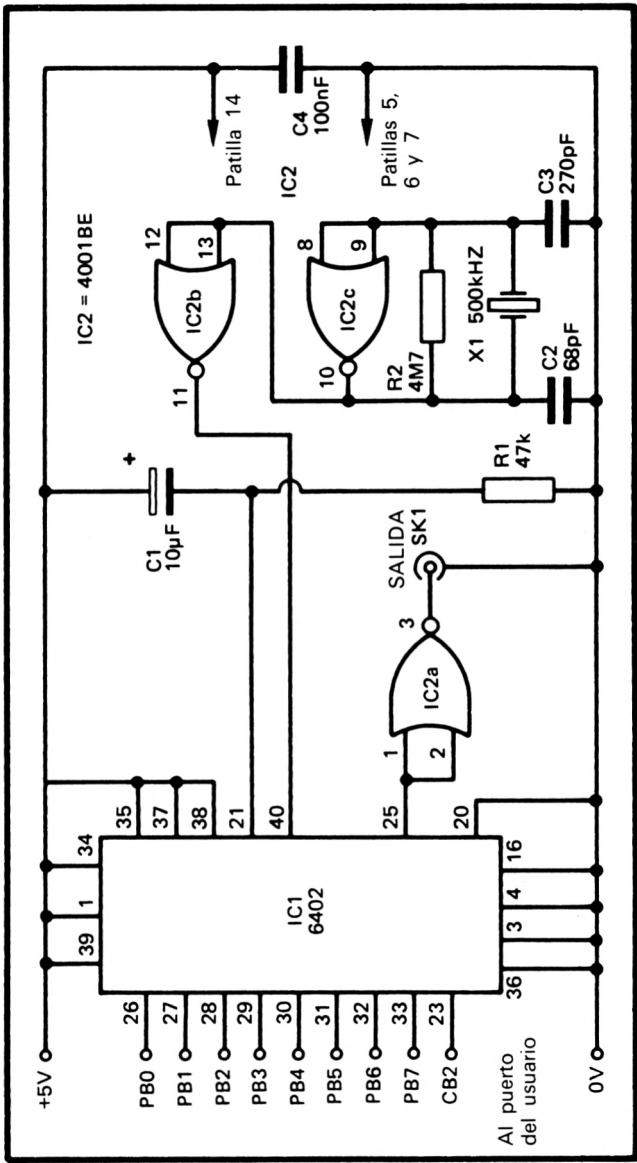


Figura 29.—Interfaz MIDI

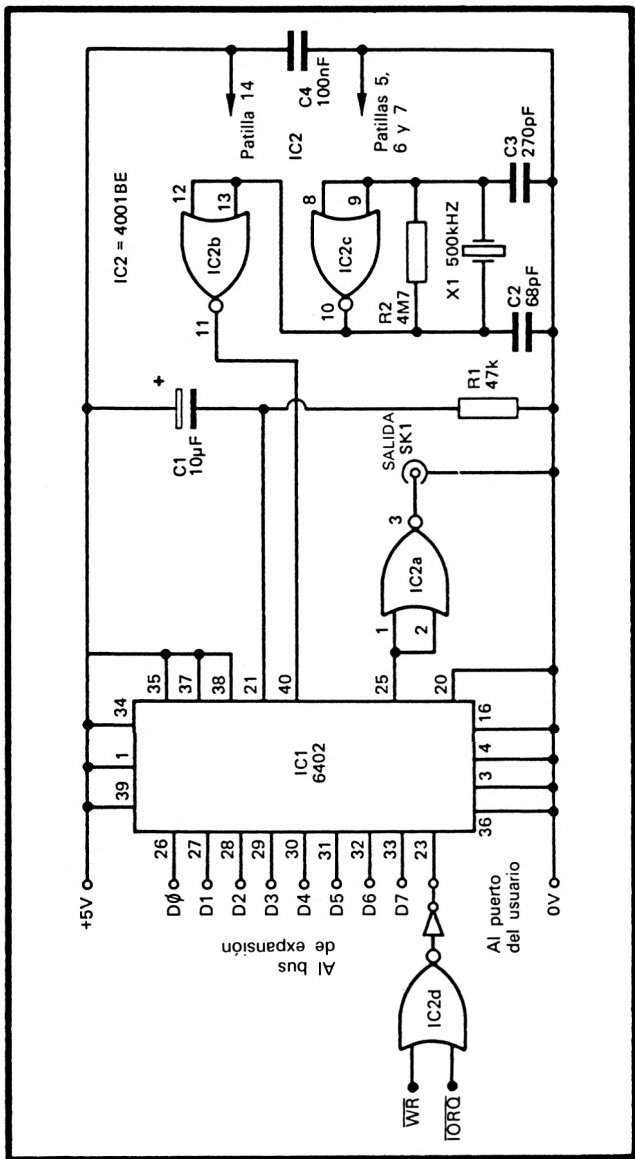


Figura 30.—Interfaz MIDI para AMSTRAD

UART

Las figuras 29 y 30 incluyen los detalles de conexión para su uso con los puertos para usuario del del VIC-20. PB0 a PB7 se emplean para accionar las seis entradas de datos del IC1 del UART y CB2 acciona la entrada de TBRL (carga en el registro del amortiguador del transmisor). Este último debe ponerse en baja con un impulso para poder cargar los 8 bits de datos en el registro del amortiguador del transmisor, de donde son enviados al registro del transmisor y transmitidos desde la salida serial del IC1. Si el interfaz se emplea con el Commodore 64, sería posible obtener resultados similares empleando PC2 en lugar de CB2. En el caso de otros ordenadores podría ser necesario utilizar una salida digital ordinaria haciendo que el *software* generara el impulso negativo sobre TBRL. Otra posibilidad digna de investigación sería accionar el circuito desde una salida de impresora en paralelo, utilizando la línea Strobe para accionar TBRL. No obstante, esto sólo funcionaría con un puerto de impresora capaz de suministrar códigos completos de 8 bits (algunos sólo pueden suministrar 7 bits de datos, y el bit más significativo está siempre en baja).

Para solucionar este problema veamos una posible vía de tratamiento. El ordenador AMSTRAD sólo dispone de 7 bits significativos en el puerto de impresora, lo cual nos obliga a utilizar el *bus* de expansión para poder ofrecer los 8 bits que necesita el interfaz MIDI. Para obtener el pulso negativo que accione a TBRL, acudimos a las líneas INR e IORQ del *bus* de expansión, conectadas de forma que sólo ofrezcan el nivel bajo cuando escribamos con la instrucción OUT del Z-80 o del BASIC, dentro del rango de direcciones &F800 a &F8FF.

C1 y R1 suministran un largo impulso de *reset* al IC1 en el encendido. IC2a es una puerta NOR de dos entradas, pero se utiliza aquí como un invertidor en la salida serial de IC1. Esto es esencial, ya que la salida serial directa de IC1 tiene la polaridad equivocada y no tendría efecto alguno si se conectara a una entrada MIDI. IC1 tiene una acción incorporada de dividir por dieciséis en la entrada de reloj y, por consiguiente, requiere una frecuencia de reloj que sea dieciséis veces la tasa en baudios o, en este caso, 500 kHz. La señal de reloj es aportada por dos puertas de IC2 que se emplean en un sencillo circuito de oscilador de cristal. Por cierto, que un resonador cerámico de 500 kHz es adecuado como sustituto barato de un X1 de cristal. La cuarta puerta del IC2 no se emplea en este circuito.

El IC1 es capaz de manipular una amplia gama de formatos de palabra diferentes, y cinco de sus entradas están ligadas a uno u otro de los raailes de suministro para programar el formato requerido. El método de conexión de las figuras 29 y 30 da la modalidad correcta con un bit de arranque, 8 bits de datos, 1 bit de detención y ningún bit de paridad.

Tal vez deba explicarse que el 6402 es capaz tanto de efectuar transmisiones seriales como de recibirlas, pero en este circuito sólo se utiliza la sección transmisora. Es improbable que una aplicación práctica requiera que el ordenador lea datos de un aparato MIDI y, por consiguiente, no tendría mucho sentido intentar implementar la sección receptora.

El tipo estándar de patilla empleado para los interfaces MIDI es el DIN de cinco puntas (180 grados). La figura 31 muestra el modo correcto de conectar la salida del interfaz a la entrada MIDI del sintetizador u otro instrumento.

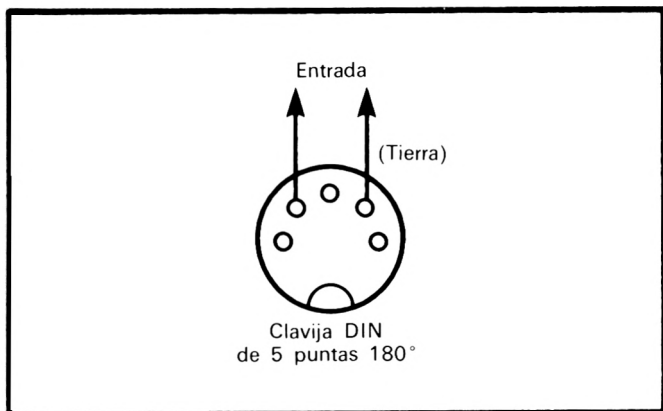


Figura 31.—Señales del conector de entrada MIDI

Componentes de un interfaz MIDI

(figuras 29 y 30)

Resistencias (1/4W 5%)

R1 47K
R2 4M7

Condensadores

C1 100 μ F 25V electrolítico
C2 68pF cerámico
C3 270pF cerámico
C4 100nF cerámico

Semiconductores

IC1 6402
IC2 40001BE

Miscelánea

SK1 Hembra DIN 5 vías
X1 Resonador de cristal o cerámico de 500 kHz
Soporte DIL de IC de 40 patillas
Soporte DIL de IC de 14 patillas
Cable DIN de 5 vías
Placa del circuito, carcasa, etc.

Software

Este programa sencillo para ordenador muestra la forma básica en que se envían las notas al instrumento.

```
10 READ A,B
20 IFA=0 THEN END
30 OUT 7F840,A
40 FOR D=1 TO B: NEXT
50 OUT 7F840,0
60 GOTO 10
70 DATA 60, 500, 62, 500, 64, 500, 65, 500, 67,
    500, 69, 500, 71, 500, 72, 1000, 0, 0
```

Los valores de nota (A) y duración (B) son leídos (READ) en la línea 10 del enunciado DATA de la línea 70. Los valores de muestreo dan una escala creciente de DO mayor partiendo del DO medio, pero obviamente estos valores podrían ser sustituidos por cualesquiera otros. Los dos valores finales han de ser 0, y esto pone fin a la secuencia en la línea 20 en lugar de permitir que el programa se venga abajo. Las notas se pasan por puerta en la línea 30. Aquí estamos empleando el canal 40, pero cualquier otro canal sería apropiado y el número de canal resulta, por tanto, irrelevante. El bucle FOR..NEXT de la línea 40

establece la duración de cada nota antes de ser sacada por puerta en la línea 50. El programa enlaza con el principio por medio de la línea 60 hasta que se detiene la secuencia.

Este simple programa debería servir para dar una idea general del modo en que se puede efectuar una secuenciación por medio de un interfaz MIDI. Para la secuenciación multicanal, sería preferible emplear código máquina para obtener una mejor sincronización entre los canales. Cuando se utiliza BASIC no hay peligro de que un byte sea enviado al interfaz antes de que el anterior haya sido procesado, pero la mayor velocidad del código máquina sí plantea esta posibilidad. Una forma de resolver el problema sería monitorizar la salida de TBRE (vaciar el registro del amortiguador del transmisor) en la patilla 22 del IC1. Esta pasa a alta cuando el amortiguador está vacío y el dispositivo está listo para el siguiente byte de datos. Junto con un *software* adecuado, esto podría suministrar una acción de detención apropiada. Con todo, en la práctica, probablemente sería más fácil utilizar un bucle temporizador para espaciar los bytes de datos alrededor de 350 μ s.

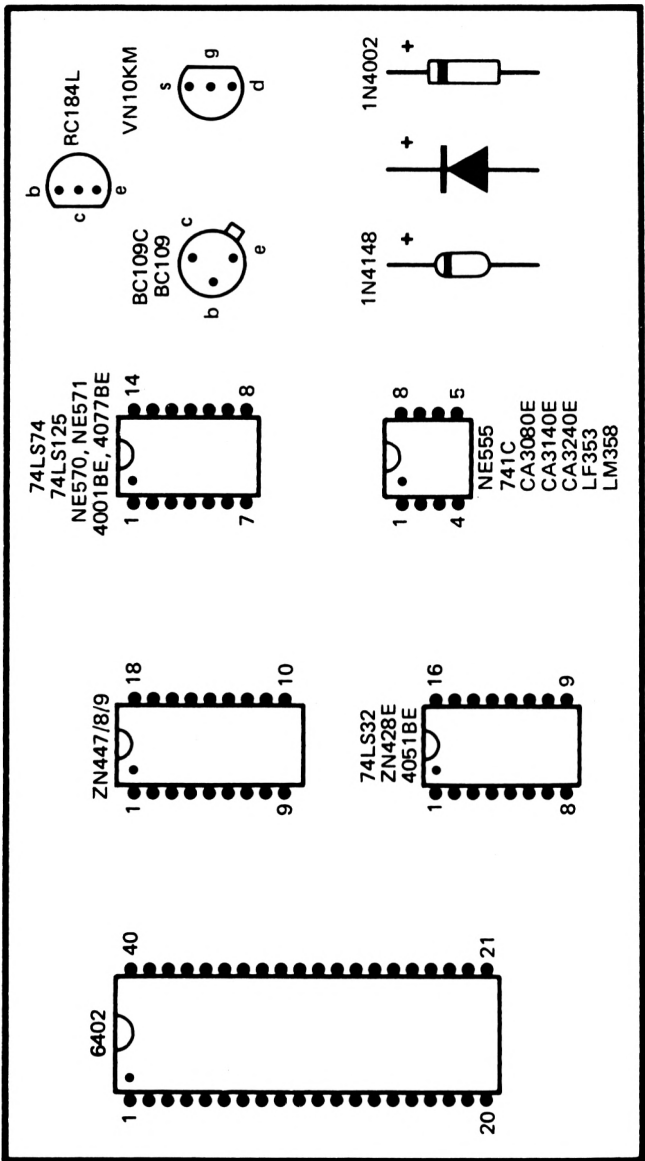


Figura 32.—Vista superior de los IC y de las bases de los transistores

ANAYA MULTIMEDIA

Colección «MICROINFORMATICA»

- Angell, I. O. y Jones, B. J.:** DISEÑO DE GRAFICOS Y VIDEO-JUEGOS (incluye cassette).
- Barnett, G.:** EL LIBRO GIGANTE DE LOS JUEGOS PARA COM-MODORE 64/128.
- Beechhold, Henry F.:** EL LIBRO DEL HARDWARE. No destape su ordenador personal... sin leer antes este libro.
- Birmingham Educational Computing Centre:** INTRODUC-CION A LA TECNOLOGIA DE LA INFORMACION. PREINFOR-MATICA.
- Bishop, Peter:** PROGRAMACION AVANZADA EN BASIC.
- Brown, Peter:** PASCAL A PARTIR DEL BASIC.
- Cavalcoli, Aldo:** EL ORDENADOR PERSONAL: COMO ELEGIR-LO Y UTILIZARLO.
- Coccione, L. y Winter, G.:** LOS ORDENADORES NO MUER-DEN.
- Dachslager, H., Hayashi, M. y Zucker, R.:** PROGRAMACION EN BASIC: UN METODO PRACTICO.
- Dewhirst, J. y Tennison, R.:** TU PRIMER LIBRO DEL ZX SPECTRUM.
- D'Opazo, J. y Grupo GOLEM:** PROGRAMACION EN LOGO.
- Durst, J.:** «SPRITES» Y GRAFICOS EN LENGUAJE MAQUINA (ZX SPECTRUM).
- Galende Domínguez, F.; Sánchez López, A.; Alfaraz López, M. y Sánchez García, J. A.:** COMETAS EN TU MI-CRO: EL HALLEY. Cálculos de órbitas y parámetros de cometas en BASIC.
- Gavin, Maurice:** ASTRONOMIA: EL UNIVERSO EN TU ORDE-NADOR.
- Gibbons, John P.:** PROGRAMACION AVANZADA DEL COM-MODORE 64. Ampliación del BASIC y rutinas gráficas.
- Greenwood, Gareth:** CODIGOS Y CLAVES SECRETAS. Cripto-grafía en BASIC.
- Hammond, R.:** EL ORDENADOR Y TUS HIJOS.
- Hartnell, Tim:** EL LIBRO GIGANTE DE LOS JUEGOS PARA ORDENADOR.
- Hartnell, Tim:** INTELIGENCIA ARTIFICIAL: CONCEPTOS Y PROGRAMAS.
- Hartnell, Tim:** EL LIBRO GIGANTE DE LOS JUEGOS PARA ZX SPECTRUM.
- Hartnell, Tim, y otros:** EL LIBRO GIGANTE DE LOS JUEGOS PARA DRAGON.
- Hartnell, Tim:** EL SUPERLIBRO DE LOS JUEGOS PARA OR-DENADOR.
- Heller, R. S. y Martin, C. D.:** BITS Y BYTES: INICIACION A LA INFORMATICA.

- Hollerbach, Lew:** MICROINFORMATICA: CONCEPTOS BASICOS.
- Hurley, R.:** JUEGOS GRAFICOS DE AVENTURA PARA ZX SPECTRUM.
- Johnson, David:** DESCUBRE LAS MATEMATICAS CON TU MICRO.
- Johnston, J.:** MICROS: TAMAÑOS, FORMAS Y SABORES.
- Johnston, J.:** MICROS: BIPS, PITIDOS Y LUCES.
- Johnston, J.:** MICROS: MENUS, BUCLES Y RATONES.
- Kosniowski, Czes:** MATEMATICAS DIVERTIDAS EN BASIC.
- Kramer, S.:** PROGRAMACION AVANZADA DEL ZX SPECTRUM.
- Lacey, Andrew:** LIBRO GIGANTE DE LOS JUEGOS PARA MSX.
- Núñez, Agustín:** PROGRAMACION DEL INTERFACE 1 Y MICRODRIVE.
- Otero, M. A.; Pueyo, M. A. y Cajaraville, J. A.:** PRIMEROS PASOS EN LOGO. El mundo de la tortuga Fan. Libro del profesor. Libro del alumno.
- O'Shea, T. y Self, J.:** ENSEÑANZA Y APRENDIZAJE CON ORDENADORES. Inteligencia artificial en educación.
- Pentiraro, Egidio:** EL ORDENADOR EN EL AULA.
- Pritchard, Joe:** DESCUBRE TU MSX. Programación y aplicaciones.
- Pritchard, Joe:** LENGUAJE MAQUINA MSX. Introducción y conceptos avanzados.
- Pritchard, J.:** RUTINAS EN LENGUAJE MAQUINA PARA AMSTRAD. CPC 464-664-6128.
- Rosso, Vincenzo de:** COMO SE PROGRAMAN LOS ORDENADORES.
- Sato, T.; Mapstone, P. y Muriel, I.:** MSX: GUIA DEL PROGRAMADOR Y MANUAL DE REFERENCIA.
- Servello, Fausto:** ¿QUE ES LA TELEMATICA?.
- Snover, S. L. y Spikell, M. A.:** JUEGOS MATEMATICOS DE INGENIO EN BASIC.
- Thomasson, Don:** PROGRAMACION AVANZADA DEL AMSTRAD. Entradas y salidas de la ROM.
- Webb, David:** LENGUAJE MAQUINA AVANZADO PARA ZX SPECTRUM.
- Zaks, Rodney:** EL LIBRO DEL BASIC.

MICROMANUALES

Los MICROMANUALES son libros de referencia condensados que le ofrecen información práctica sobre todas las áreas de la microinformática: programación, uso de programas, proyectos, etc.

Otros títulos de la colección:

- dBASE III
- Introducción a los sistemas operativos
- Introducción a las comunicaciones
- Programación del 8088/8086
- SYMPHONY
- Técnicas y proyectos de interfaces
- Proyectos *hardware* para Amstrad y MSX





AMSTRAD

CPC



MÉMOIRE ÉCRITE
MEMORY ENGRAVED
MEMORIA ESCRITA



<https://acpc.me/>

[FRA] Ce document a été préservé numériquement à des fins éducatives et d'études, et non commerciales.

[ENG] This document has been digitally preserved for educational and study purposes, not for commercial purposes.

[ESP] Este documento se ha conservado digitalmente con fines educativos y de estudio, no con fines comerciales.