

## Periféricos de entrada

### 4.1 TECLADOS

Se denomina «teclado» al género de periféricos de entrada, constituidos por un conjunto de botones pulsadores, de tal modo que cada botón se corresponda con un determinado carácter, función, instrucción o idea. El tipo de teclas, así como su número y distribución, vendrán determinados por la aplicación concreta que se desee realizar, por lo que no existen modelos genéricos, sino desarrollos específicos.

El número de teclados conectados a un determinado sistema es sumamente variable, oscilando entre cero (tal como en sistemas muy simples de instrumentación o automatización) y varias decenas (tal como en sistemas multiterminal de recogida y consulta de bancos de datos); no obstante, en sistemas basados en microprocesadores, lo más común es disponer de una unidad, desde donde se suministran al sistema las informaciones básicas en cuanto a selección y control de programas, e introducción de variables.

Físicamente, el teclado acostumbra a ir asociado a otro periférico de salida, tal como una impresora, una pantalla o un visualizador, con lo que el operador obtiene una comunicación bidireccional con el sistema. Este conjunto de teclado y visualizador se trata en muchos aspectos de forma conjunta y habitualmente recibe el nombre de consola.

El teclado, junto con el sistema de vídeo, es el periférico más popular. Es la principal herramienta de entrada de datos al sistema o al menos de su control ya que las grandes cantidades de datos van a través de los sistemas de comunicación o de los dispositivos de almacenamiento intercambiables. Recientemente otros dispositivos como por ejemplo el ratón han ganado popularidad, pero generalmente se usan más para controlar al sistema que para introducir los datos a procesar por el sistema. Necesitamos distinguir entre el teclado estándar, el cual es familiar a todos los usuarios de ordenadores, y varios tipos de teclados especiales, los cuales son diseñados para muchas aplicaciones específicas.

Externamente, el teclado consiste en un conjunto de teclas diseñadas para ser pulsadas por el dedo, y algunas veces uno o más leds luminosos. Internamente el teclado es un circuito electrónico, que detecta cuando cada tecla es pulsada y/o liberada, y envía esta información al procesador principal. A menudo asociamos el teclado con el vídeo, debido a que cada vez que pulsamos una

tecla, aparecerá en la pantalla el carácter correspondiente. Sin embargo, en realidad, esto no ocurre así sino que la señal es transmitida desde el teclado al procesador, y éste entonces escribe el carácter en la pantalla. En el caso de terminales inteligentes, el procesador implicado es comúnmente un procesador dentro del dispositivo. En terminales 'tontos' (carecen de la posibilidad de procesamiento de datos), se depende del ordenador principal, que devolverá el carácter en la pantalla. A veces, este eco puede ser suprimido, como por ejemplo cuando se teclea una contraseña de acceso ('password').

La tecnología empleada en los teclados es relativamente simple. Estos son periféricos que no necesitan de gran sofisticación, porque son utilizados directamente por el hombre y con sus manos, lo que impone restricciones de tamaño, de mecánica, etc. Otra restricción importante es que los teclados comunes de las consolas de ordenador provienen, en su aspecto físico, de la evolución de las máquinas anteriores (máquinas de escribir, teletipos, etc.). Este lastre es evolutivo y muy común en todos los campos de la informática.

El elemento básico de los teclados es el pulsador, elemento electromecánico que conforma la tecla. La idea general consiste en un dispositivo que permanece normalmente en reposo, y que en este estado suministra una señal determinada. Al accionarlo, el movimiento mecánico producido es traducido a una variación de la respuesta eléctrica, que es detectada por alguna circuitería. Este hardware, manejando señales ya digitales, lo comunica al procesador central. A continuación estudiaremos los modelos de pulsadores más interesantes.

## 4.2 TIPOS DE PULSADORES

El componente básico de un teclado es el pulsador individual. Generalmente, cada tecla controla un simple interruptor que permanece abierto mientras el pulsador está en descanso y se cierra cuando el pulsador es accionado.

Existe un sinnúmero de variaciones, relacionándose más adelante los tipos más usuales. Una primera clasificación entre ellos, se puede hacer atendiendo a la forma en la que se produce el cambio de estado, si es de contacto físico o no lo es. El concepto de conmutación de estado sólido se aplica a veces al describir elementos sin contacto, aunque hablando con propiedad solamente se puede aplicar a una limitada variedad entre la que se cuentan los de efecto Hall o los de elementos fotosensibles.

Dentro de los pulsadores con contacto físico podemos incluir:

- ◆ El de contacto convencional
- ◆ De láminas flexibles
- ◆ De bóveda
- ◆ Reed
- ◆ Elastómeros

y entre los que no tienen contacto físico:

- ◆ Capacitivos
- ◆ Inductivos
- ◆ De efecto Hall

Los pulsadores de contacto son los más simples y baratos y por lo tanto se utilizan bastante. El movimiento que provoca el operario actúa directamente uniendo sus dos contactos que en reposo están separados. Esta unión modifica el nivel eléctrico de uno de los contactos en base al estado del segundo contacto, hecho que es detectado y comunicado al ordenador. El movimiento

mecánico actúa directamente sobre los contactos y permiten, sin grandes complicaciones, configuraciones de contactos múltiples. Los interruptores mecánicos son utilizados básicamente por su economía. El principal inconveniente de los pulsadores de contacto es que éste no es perfectamente instantáneo y suelen producirse rebotes (Fig. 4.1). Un circuito antirebote típico se muestra en la figura (4.2). Este circuito está constituido por dos puertas NAND de dos entradas conectadas formando un biestable tipo RS donde las dos entradas R y S se conectan a estado alto a través de sendas resistencias, lo que garantiza que el biestable mantiene su estado cuando el contacto pasa de una posición a otra. El contacto pone una de las entradas (R o S) en estado bajo y nunca se produce la situación de ambas entradas en estado bajo. Supongamos que inicialmente el pulsador está en la posición A, lo que provoca un estado bajo en una de las entradas de la puerta NAND superior, con lo que independientemente de como esté la otra entrada producirá a la salida ( $Q$ ) un estado alto. Por el contrario, la puerta inferior tendrá ambas entradas en estado alto y la salida ( $\bar{Q}$ ) estará en estado bajo. Cuando pulsamos la tecla, el contacto móvil abandona su posición A, con lo que las entradas R y S estarán simultáneamente en estado alto y la salida del biestable no cambia hasta que el contacto móvil llegue a la posición B que producirá un estado alto en ( $\bar{Q}$ ) y un estado bajo en ( $Q$ ). Si se produjese un rebote y el contacto móvil abandonase la posición B durante un instante, el estado del biestable quedaría inalterado.

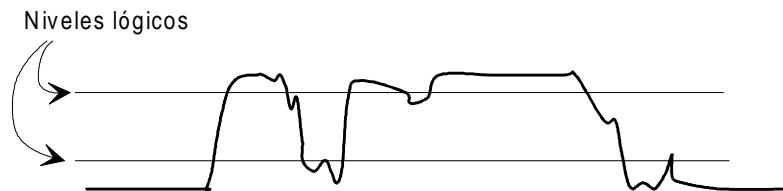


Fig. 4.1 Rebotes en un pulsador de contacto

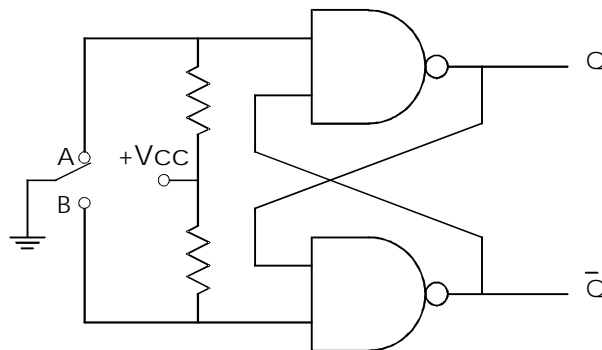


Fig. 4.2 Circuito antirebote típico

Si consideramos a los rebotes como una interferencia o ruido que contamina la señal, podemos eliminarlo con la ayuda de un filtro. En estos casos una sencilla red R-C formada por una resistencia en serie con un condensador en paralelo suele ser suficiente.

Otra forma, también muy corriente de evitar los rebotes, es emplear un monoestable. En este tipo de circuitos el pulsador es utilizado para disparar el monoestable que genera un pulso cuya anchura es independiente del tiempo que permanezca activado el contacto. De esta forma, cuando un pulsador golpea repetidas veces el contacto antes de asentarse definitivamente, la salida es fija, independiente e insensible a estas oscilaciones. Para garantizar esto, la anchura del pulso que proporciona el monoestable debe ser mayor al tiempo de oscilación del contacto, para evitar un redisparo accidental del monoestable.

Otro problema asociado a los pulsadores de contacto es que son sensibles a las condiciones ambientales, como oxidación de los contactos, polvo, humedad, etc, aunque veremos que hay algunos tipos de pulsadores que tienen el contacto en una cavidad sellada herméticamente, lo que obviamente incrementa su coste. Por último señalar que los pulsadores por contacto están sometidos también al desgaste debido al rozamiento y golpeteo de contactos lo que provoca una erosión en el mismo haciéndolo de esta forma muy dependiente del uso que se haga de él. A continuación vamos a ver algunos tipos elementales de pulsadores de contacto y posteriormente veremos los tipos más importantes de pulsadores sin contacto.

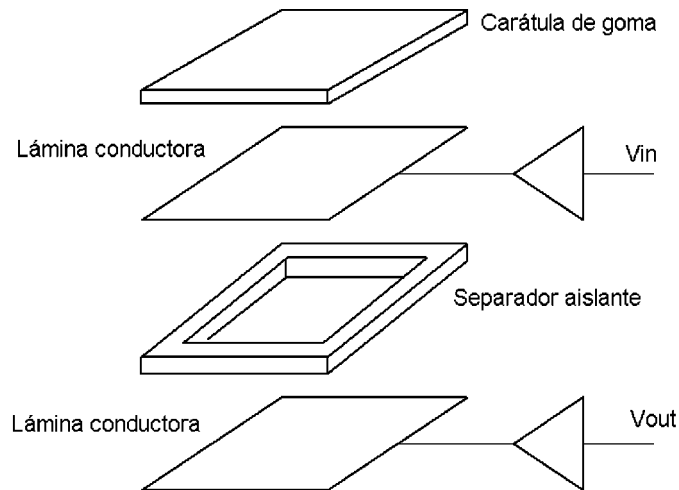


Fig. 4.3 Pulsador de láminas flexibles

#### 4.2.1 Pulsador de lámina flexible

Este tipo de interruptor (fig. 4.3), formado por una serie de láminas sobrepuestas, se basa en la deflexión de un diafragma flexible, metalizado por su cara inferior, que permite establecer contacto con un circuito impreso a través de aperturas practicadas en un separador aislante.

Una cubierta de silicona protege los contactos contra los contaminantes. Algunas versiones más económicas emplean láminas flexibles de silicona conductora, que sustituyen la cubierta protectora y el diafragma metalizado. En algunos casos se emplea una base serigrafiada de tinta conductora como sustrato. No obstante, estas soluciones, aunque resultan mucho más económicas tienen una vida útil mucho más corta debido a la menor resistencia al desgaste de la película conductora frente a la lámina metálica.

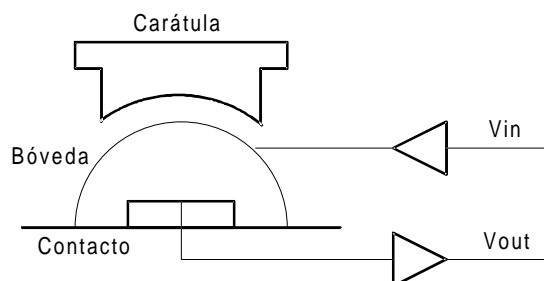


Fig. 4.4 Pulsador de bovedilla

#### 4.2.2 Pulsador de bovedilla

Otro ejemplo de pulsador de contacto es el de bóveda (Fig. 4.4). En él existe una lámina de conductor en forma de bóveda que al ser pulsada, se deforma hasta tocar el conductor que hay

debajo con la cúpula de la bóveda. Al liberarlo, la elasticidad le hace recuperar su forma original y el contacto desaparece. Emiten un clásico chasquido audible que advierte de su correcta operación. Su principal ventaja es que mantienen el contacto sellado y completamente aislado del medio ambiente pero presentan una vida reducida debido a que la metalización no puede ser muy gruesa para permitir la flexibilidad de la bovedilla y se desgasta rápidamente.

#### 4.2.3 Pulsador elastómero

Es muy similar al pulsador de bovedilla. Están constituidos por un circuito impreso, en el que están definidos tantos pares de contactos como debe contener el conjunto del teclado, y un elemento elastómero (silicona) que forma la parte móvil del contacto (fig. 4.5).

La superficie de contacto del circuito impreso está serigrafiada con grafito conductor como protector de la oxidación y mejorador de la conductividad eléctrica, mientras que la parte móvil está formada por una pieza inyectada en material elastómero con un pequeño inserto de silicona conductiva en su centro, de tal modo que al ser pulsada establece conexión entre los contactos definidos en la parte fija del circuito impreso. La sección de la parte móvil es extraordinariamente delicada en cuanto a diseño, puesto que de ella depende la vida útil (número de operaciones) del pulsador y su respuesta táctil.

Este tipo de pulsador ofrece un coste reducidísimo, siempre y cuando el volumen de producción permita anular sensiblemente los costes de amortización de los moldes de inyección, brindando una vida útil comprendida entre 5 a 50 millones de pulsaciones, suficientes para gran parte de las aplicaciones. Es el tipo de pulsador que se emplea habitualmente en los mandos a distancia de los equipos de audio-video domésticos.

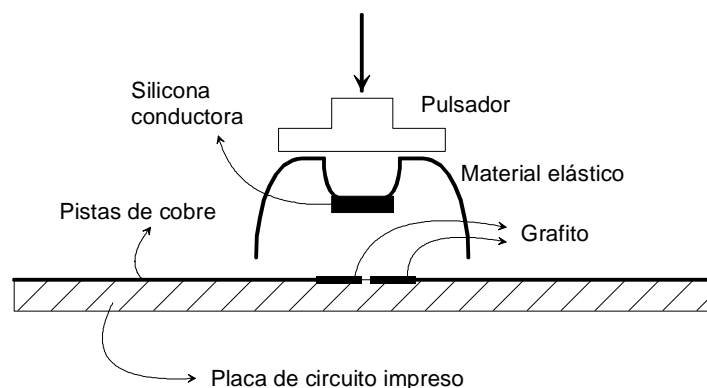


Fig. 4.5 Pulsador basado en elastómero

#### 4.2.4 Pulsadores Reed

El último tipo de pulsador de contacto que veremos es el pulsador REED (fig 4.6). Consiste en una cápsula hermética que alberga dos conductores separados, uno fijo y otro móvil en una atmósfera inerte. Cuando se pulsa, un pequeño imán se aproxima a la cápsula lo suficiente como para atraer al contacto móvil y conectarlo con el fijo. Tiene la ventaja de que el contacto está aislado, volviéndose inmune a la suciedad que es un elemento muy frecuente en todos los teclados especialmente en entornos industriales. Dada la acción indirecta sobre los contactos, no se transmiten sobrecargas mecánicas que provoquen fatiga y desgastes prematuros. Por lo anterior y por la hermeticidad del encapsulado que impide la contaminación de los contactos, este tipo de pulsadores ofrecen una vida útil unas 5 veces superior al clásico pulsador mecánico. El contacto Reed resulta bastante caro y es por lo tanto poco frecuente no siendo rentable fabricar teclados completos, mas bien se emplean para pequeños teclados o contactos aislados en ambientes industriales, donde la duración y fiabilidad son factores predominantes.

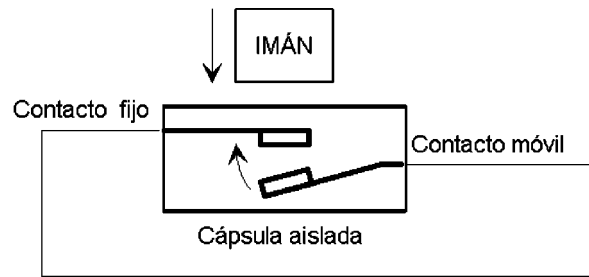


Fig. 4.6 Pulsador Reed

#### 4.2.5 Pulsadores capacitivos

Este tipo de pulsadores emplea un cambio en la capacidad de un condensador para entregar una salida (fig. 4.7). Emplean dos superficies vecinas sobre un mismo circuito impreso, estando una de ellas excitada por la señal alterna de un oscilador; si se aproxima paralelamente una placa conductora sobre ambas superficies, se provoca un acoplamiento entre ellas, con lo que aparece una fracción de la señal alterna de entrada, en la salida.

La señal de salida del pulsador debe ser convenientemente amplificada y convertida a niveles lógicos.

Existen múltiples variantes tales como los que utilizan contactos sensitivos, sin ningún elemento móvil. Otros diseños emplean bovedillas metálicas cóncavas como elemento de acoplamiento. Los pulsadores capacitivos ofrecen la elevada fiabilidad de los interruptores sin contactos móviles.

Dados los bajos niveles de señal entregada por estos pulsadores se presenta una acusada sensibilidad a interferencias y unos serios condicionamientos en la estructura metálica soporte, trazado de pistas en el circuito impreso y electrónica de amplificación, detección y conversión. Por todo ello sólo aparecen disponibles formando parte de teclados completos producidos por fabricantes especializados.

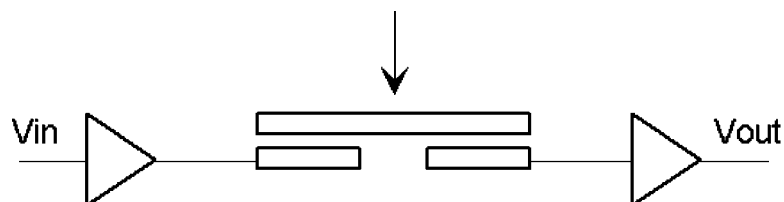


Fig. 4.7 Pulsador de tipo capacitivo

#### 4.2.6 Pulsador de efecto Hall

Los sensores de efecto Hall (fig. 4.8) están formados por un elemento semiconductor recorrido por una corriente continua, y un campo magnético perpendicular a ella que provoca una deformación de las líneas equipotenciales sobre la superficie del semiconductor, apareciendo una tensión de salida proporcional al producto de la corriente de polarización por la intensidad del campo magnético aplicado.

La conmutación se obtiene al aproximar un imán permanente al sensor, que desarrolla una tensión de salida que es amplificada y convertida en digital.

Generalmente el conjunto formado por el sensor, amplificador, disparador de Schmitt, monostable opcional y etapa de salida, forma un circuito integrado monolítico asociado a cada pulsador.

Se distinguen dos tipos fundamentales: estático y dinámico. Los pulsadores estáticos conducen a su salida, mientras exista campo magnético a su entrada. Los pulsadores dinámicos conducen durante cierto período (típicamente 20 ps) cuando el campo de entrada supera el nivel de conmutación, pero no lo hacen durante el resto del tiempo que dicho campo permanezca a nivel elevado, ni durante el alejamiento del imán; para ello incorporan un monostable que dispara en el flanco de subida de la señal magnética.

La salida puede estar formada por un transistor de colector abierto, simple o doble, o bien por una puerta lógica «Y» aceptando señales externas de validación y sincronismo.

Dada la ausencia de contactos, la baja impedancia de todas las señales de interconexión y la insensibilidad a polvo, suciedad y contaminantes, este tipo de pulsadores ofrece la mayor fiabilidad (esencialmente duración infinita), sólo limitada por el desgaste del elemento móvil y el resorte de retorno. Este último, en algunos casos, es sustituido por un sistema magnético de retorno, que proporciona simultáneamente una realimentación al tacto. Su principal inconveniente es su elevado coste.

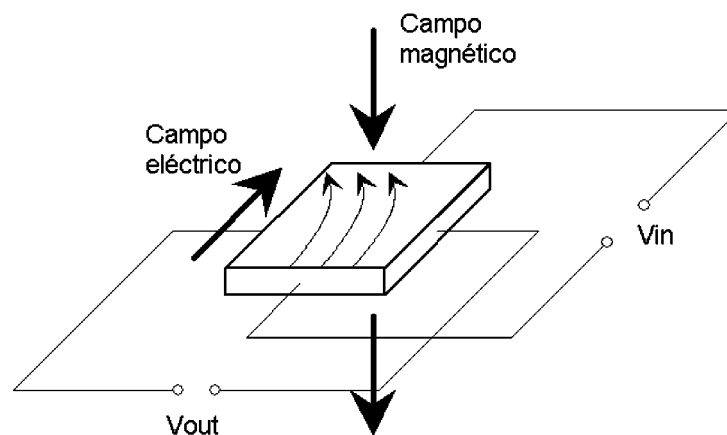


Fig. 4.8 Pulsador de efecto Hall

Basados en el efecto Hall, consisten en la aparición de una diferencia de potencial, cuando un semiconductor es atravesado por una corriente continua en presencia de un campo magnético, según la figura (4.8). De una forma simplificada, podemos describir el efecto Hall de la siguiente forma. Al aplicar una diferencia de potencial entre los extremos del semiconductor, se producirá en su seno una corriente eléctrica, formada por un flujo de electrones y/o huecos. En esta situación, las cargas eléctricas en movimiento se ven afectadas por el campo magnético, que les aplica una fuerza:

$$F = q(\bar{v} \times \bar{B})$$

En donde:  $q$  es la carga del electrón  
 $\times$  indica un producto vectorial  
 $\bar{v}$  es la velocidad del electrón  
 $\bar{B}$  es el vector campo magnético

Como el producto vectorial es perpendicular a ambos vectores, se obtiene así una fuerza que desvía a los electrones haciendo que describan una trayectoria curvilínea. La componente

transversal de este movimiento induce en los extremos perpendiculares a la corriente una diferencia de potencial. Esta tensión es amplificada y comparada con una referencia que dará como resultado un estado de 1 ó 0.

El mecanismo actuador del pulsador, lo que hace es acercar dos imanes permanentes al semiconductor hasta que se consigue la tensión de Hall suficiente para ser detectada.

#### 4.2.7 Pulsador inductivo

Este tipo de pulsadores se basa en una variación de la permeabilidad magnética en el medio de acoplamiento de dos circuitos inductivos (fig. 4.9).

Los dos circuitos inductivos están definidos a ambas caras de un mismo circuito impreso. Uno de dichos circuitos está recorrido por una corriente de alta frecuencia (impulsos), mientras que el otro podrá captar dicha señal si existe un buen acoplamiento, y esto se produce cuando la tecla correspondiente es pulsada.

El acoplamiento a través del aire y del propio sustrato del circuito impreso es muy débil, por lo que el nivel de señal obtenido a la salida de los secundarios asociados a las teclas no pulsadas es suficientemente bajo como para ser discriminado como «O» lógico. La pulsación de la tecla introduce un núcleo de alta permeabilidad magnética (ferrita) en el orificio del circuito impreso situado en el eje común a ambas espiras, con lo que el acoplamiento se incrementa de forma notable, induciendo una corriente secundaria (como consecuencia de cada impulso que recorre el circuito primario) de suficiente nivel como para ser discriminada como de nivel lógico alto «1».

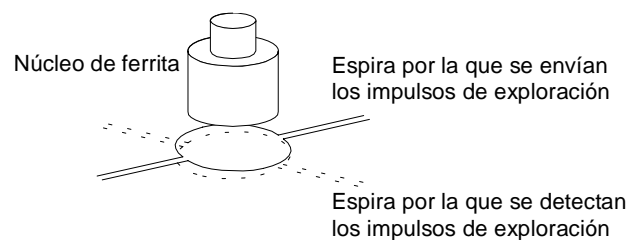


Fig. 4.9 Pulsador de tipo inductivo. Cada una de las espiras, está por una cara de un circuito impreso.

Su fiabilidad es tan elevada como la del resto de técnicas que no implican un contacto físico, es decir: capacitivos, núcleos magnéticos o de efecto Hall. Actualmente se están imponiendo de forma notable dada su combinación de alta fiabilidad y coste reducido, siendo este último consecuencia de no precisar un circuito integrado para cada tecla (como en el caso del efecto Hall), ni un cableado con alta incidencia del coste de mano de obra, ni presentar la sensibilidad al diseño asociada a los pulsadores capacitivos (téngase presente que el circuito correspondiente a cada secundario, usualmente un grupo de 10 a 16 teclas, es un circuito cerrado de muy baja impedancia, por lo que la sensibilidad a interferencias de origen externo está sensiblemente reducida).

### 4.3 CODIFICACIÓN

Como señales de salida de un teclado pueden utilizarse las conexiones correspondientes a todos y cada uno de los distintos conjuntos tecla-pulsador que lo constituyen. Esto puede ser válido para teclados simples formados por un reducido número de teclas, pero es claramente engorroso si el número total de teclas supera ciertos umbrales.

A título de ejemplo considérese que un teclado mínimo de 12 teclas; no es posible procesarlo mediante una sola palabra en sistemas basados en procesadores de 8 bits (aplíquese lo anterior a teclados alfanuméricos de más de 70 teclas).

Evidentemente se han buscado soluciones mucho más efectivas basadas en la codificación de los datos de salida.

Esta codificación consiste en numerar de forma binaria cada uno de los distintos códigos emitidos por el teclado, de tal modo que el número total de bits precisos para expresar cualquier código no supere los umbrales de maniobrabilidad.

La codificación más usual para teclados numéricos reducidos es la hexadecimal (o su subconjunto BCD si nos limitamos a dígitos decimales); en el caso de teclados alfanuméricos se amplía la codificación, siendo el código más usual el ASCII de 6 o 7 bits según sea reducido o completo, o bien el EBCDIC de 8 bits.

Por circuitos codificadores de un teclado, no solamente se entienden los circuitos precisos para reducir el número de conexiones, sino además el resto de electrónica asociada a teclas y pulsadores; esto incluye, naturalmente, los circuitos destinados a generar las variantes asociadas a cada modo y los destinados a prevenir las pulsaciones simultáneas, que se comentarán más adelante.

#### **4.3.1 Conexión a codificador**

Si el número de pulsadores es pequeño, se pueden emplear codificadores para la identificación de la tecla pulsada, como se esquematiza en la figura (4.10) y que consiste simplemente en llevar la señal digital que entregan los pulsadores a un codificador de prioridad BCD que suministra directamente el código de la tecla pulsada. A este circuito se le añaden algunos elementos para generar la validación de la pulsación (VAL). La red formada por la resistencia y el condensador tienen como objetivo retardar la salida de validación de forma que cuando esta se active, los posibles rebotes ya hayan pasado. Con esta sencilla solución, podemos eliminar los rebotes de todas las teclas.

En estado de reposo, cuando ninguna tecla está activa, las entradas del codificador están en estado alto pues están conectadas a la alimentación a través de las resistencias de la parte superior de la figura. Las salidas del codificador están también en estado alto, por lo que la salida de la puerta AND también lo estará. Pasado el tiempo necesario para que el condensador se cargue a través de la resistencia, la entrada correspondiente de la puerta NAND también estará en estado alto, mientras que la otra (B) estará en estado bajo por pasar a través de un inversor. La señal de validación estará en ese momento en alto y el monoestable en su estado estable que podemos suponer, sin pérdida de generalidad, que es el nivel bajo.

Cuando pulsamos una tecla, la correspondiente entrada del codificador se pone en estado bajo y en su salida aparecerá el código correspondiente bajando una o varias de sus salidas. Esto provocará que la puerta AND ponga su salida en estado bajo. Este estado bajo se propaga a la puerta NAND a través de dos canales independientes: por una parte un inversor y por otra a través de la red RC. El cambio en A, aparecerá a la salida del inversor tras un breve instante pasando a estado bajo. Sin embargo la otra entrada requiere que se descargue el condensador y esto llevará algún tiempo que dependerá del producto R·C, que se escoge de forma que sea mucho mayor que el tiempo de respuesta del inversor tal y como se muestra en las curvas de la figura (4.10). Esto hace que la otra entrada de la puerta NAND baje lentamente. Durante la descarga del condensador, ambas entradas de la puerta NAND estarán en estado alto y su salida será por tanto un estado bajo. Cuando el condensador se haya descargado casi por completo, la entrada correspondiente de la puerta NAND pasará a estado bajo con lo que su salida conmutará y retornará al estado alto. De esta forma hemos conseguido un pulso que se utiliza para disparar el monoestable.

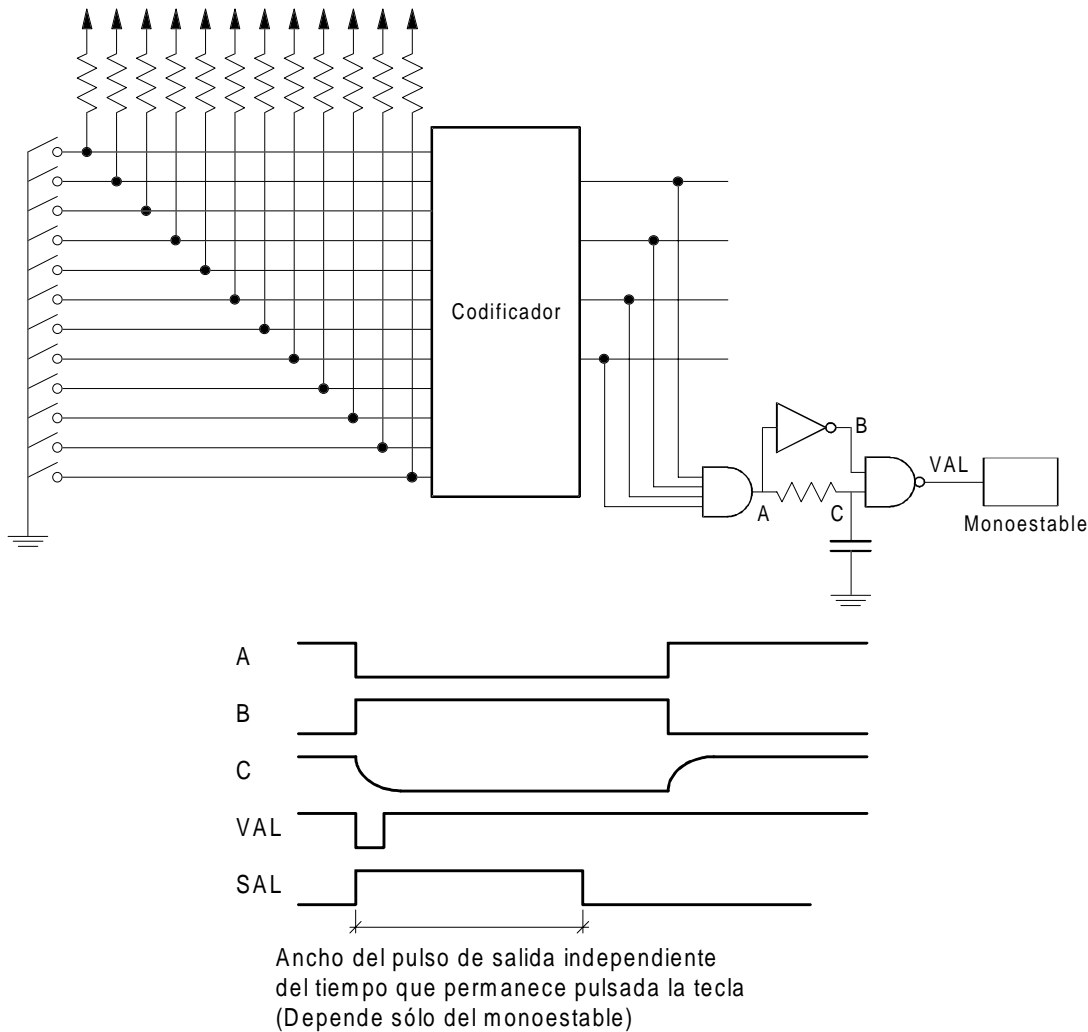


Fig. 4.10 Ejemplo de teclado numérico

### 4.3.2 Conexión matricial

Cuando el número de teclas sea más elevado (caso típico de teclados alfanuméricos) no es practicable seguir empleando los circuitos codificadores antes descritos, ya que se requeriría un codificador de 80, 100 o más entradas (tantas como teclas). En estos casos el más utilizado, incluso cuando el número de teclas no es elevado, es el teclado matricial (fig. 4.12). El 'truco' está en conectar los pulsadores de forma que el contacto se efectúe entre una fila y una columna de una matriz tal como se refleja en la figura (4.12).

El funcionamiento, es similar al del teclado con codificador que se ha descrito en el apartado anterior. Sin embargo, ahora deben conectarse a tierra dos líneas simultáneamente, para activar los dos decodificadores conectados ahora tanto a las filas como a las columnas. Esto se puede conseguir con un pulsador de doble contacto o con un transistor de doble colector, solución que resulta ideal para los pulsadores de efecto Hall, tal y como se muestra en la figura (4.11).

La técnica más usual consiste en conectar las teclas en forma matricial, de tal modo que el número total de teclas conectable es igual al número de intersecciones. Este tipo de conexión resulta ideal para teclados con pulsadores de tipo inductivo o de efecto Hall dinámico.

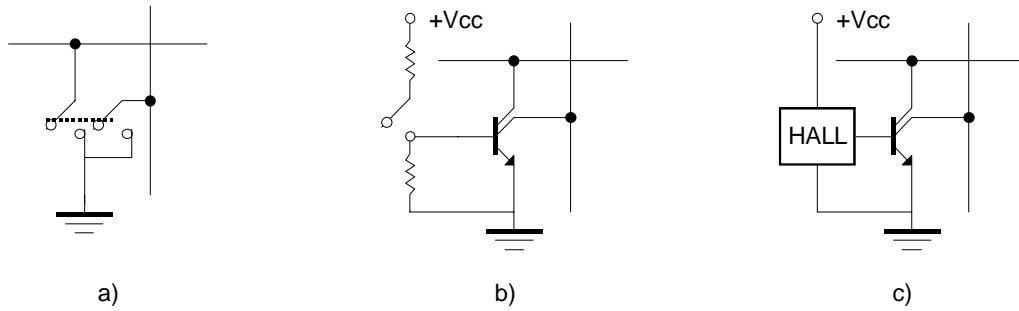


Fig. 4.11 Detalle de una celda de interconexión en un teclado matricial.  
 a) Con conector de doble contacto, b) con transistor de doble colector, c) con celda de efecto Hall

De esta forma se consigue que el número de terminaciones eléctricas a controlar (filas más columnas) sea sólo del orden del doble de la raíz cuadrada del número de teclas. En el ejemplo de la figura (4.12) se controlan 128 teclas con una matriz de 8 filas por 16 columnas (24 hilos) que suele ser suficiente para la mayoría de las aplicaciones.

Para suministrar al procesador central el código de la tecla pulsada se emplean codificadores de prioridad que generan un código intermedio de 7 bits. Este código se utiliza para direccionar una EPROM en la que están escritos los códigos de las letras. De esta forma, se habilita también la posibilidad de que las teclas cambien el código, como los paginadores, dentro de la EPROM. Añadiendo a este esquema un microcontrolador para comunicar con el procesador central, alguna circuitería adicional y una parte mecánica, obtenemos un teclado muy parecido a los que existen actualmente.

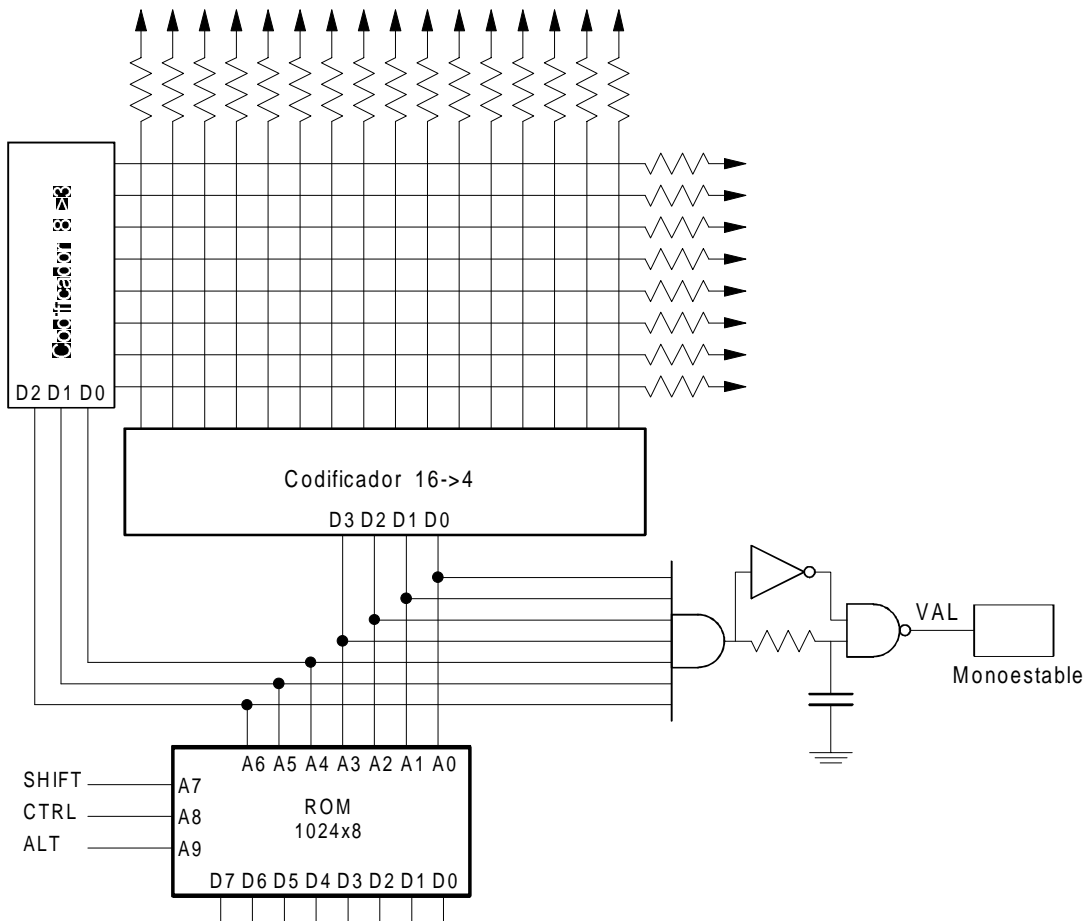


Fig. 4.12 Ejemplo de teclado matricial. Todas las resistencias se conectan a la alimentación (estado alto).

La detección de la tecla pulsada consiste en que al pulsarse una tecla se activan dos líneas a la vez, una correspondiente a una fila y la otra a una columna. Existe un codificador para las filas y otro para las columnas. Las salidas de los decodificadores se conectan a las direcciones de una ROM, junto con las teclas de cambio de modo (SHIFT, CTRL, ALT, etc.), en la que están almacenados los códigos según un determinado sistema (ASCII, EBCDIC, etc.). Problema: la dirección de memoria debe mantenerse cierto tiempo.

#### **4.3.3 Exploración secuencial**

En casos donde no resulta conveniente una conexión matricial como sucede con todos los teclados mecánicos así como los Reed se acude como norma general a realizar los circuitos codificadores empleando técnicas de exploración secuencial.

Un circuito clásico es el ilustrado en la figura (4.13) basado en un contador de 7 bits, un multiplexor y un decodificador de 4 a 16. Las teclas codificadas forman una matriz en la que cada tecla conecta una salida del decodificador con una entrada del multiplexor. El decodificador está seleccionado por los 4 bits menos significativos y el multiplexor por los 3 más significativos del contador. Cuando se pulsa una tecla, se cierra una conexión, de tal modo que cuando el contador alcanza el código apropiado, el multiplexor conmuta su salida y dispara un monostable redispensible, que detiene el conteo. El monostable se redispensa continuamente mientras la tecla está pulsada con lo que se produce automáticamente la autorepetición.

En el esquema de un teclado con exploración secuencial que se muestra en la figura (4.13), se puede ver que las entradas al multiplexor están en estado bajo debido a que están conectadas permanentemente a tierra a través de las resistencias. De esta forma, la salida del MUX permanecerá en estado bajo sea cual sea la combinación de señales de control. Por otra parte, las salidas del decodificador también estarán en estado bajo excepto la que corresponda al código de entrada que pasará momentáneamente a estado alto durante un ciclo de reloj.

Si no hay ninguna tecla pulsada, las entradas del MUX no se ven afectadas, pero supongamos que se pulsa una tecla. De esta forma conectamos una de las salidas del decodificador con una de las entradas del MUX. La salida del MUX cambiará únicamente cuando estén seleccionadas simultáneamente la salida del decodificador mediante los 4 bits C0-C3 del contador y la entrada del MUX correspondiente a la línea de la columna pulsada. Esta combinación de valores C0-C3, C4-C6 se produce una vez en cada recorrido completo del contador y en ese momento el valor de cuenta del contador nos da directamente el código de la tecla pulsada.

Este código por provenir de un contador binario de 7 bits estará comprendido entre 0000000 y 1111111 pero en la mayoría de las aplicaciones puede interesar otro tipo de código, como por ejemplo el código ASCII con lo que la salida del contador no se utiliza directamente sino que sirve para direccionar una ROM que realiza la conversión de código necesaria. Una PROM o circuitería equivalente realiza la codificación y adaptación de modos, aunque esta función puede ser realizada por el procesador mediante acceso a una tabla residente en memoria RAM y por lo tanto fácilmente modificable por el programa.

Esta memoria ROM que realiza la conversión de código, puede tener además otras entradas adicionales para cambiar el código de salida, tal y como se muestra en las figuras (4.12) y (4.13). De esta forma podemos entender esta memoria como paginada en la que se accede a las distintas páginas mediante esas entradas adicionales y donde cada página contiene un conjunto completo de códigos de tecla.

La conversión que realiza esta memoria ROM, puede hacerla la circuitería integrada en el propio teclado o también, como ya se ha adelantado, un programa y la memoria RAM del sistema, aunque en la mayoría de las aplicaciones relacionadas con ordenadores es una combinación de

ambas. El hecho de que la conversión de código la realice el procesador por software tiene la ventaja de que podemos cambiar la tabla de conversión también por software simplemente cargando un fichero del disco. Esta es la técnica empleada habitualmente para internacionalizar los códigos. De esta forma el teclado devuelve siempre los mismos códigos y el mismo teclado es válido para cualquier país. Para adaptar el teclado a uno u otro idioma basta con cambiar la serigrafía de las teclas y el fichero correspondiente.

Un esquema completo para codificar un teclado de hasta 128 teclas por el método de exploración secuencial resultaría bastante voluminoso si se realiza mediante integrados simples SSI y MSI, máxime si se incorporan circuitos complicados de selección de modos y protectores de sobrepulsaciones. Estos casos han sido resueltos mediante circuitos integrados MOS-LSI, que realizan todas estas funciones.

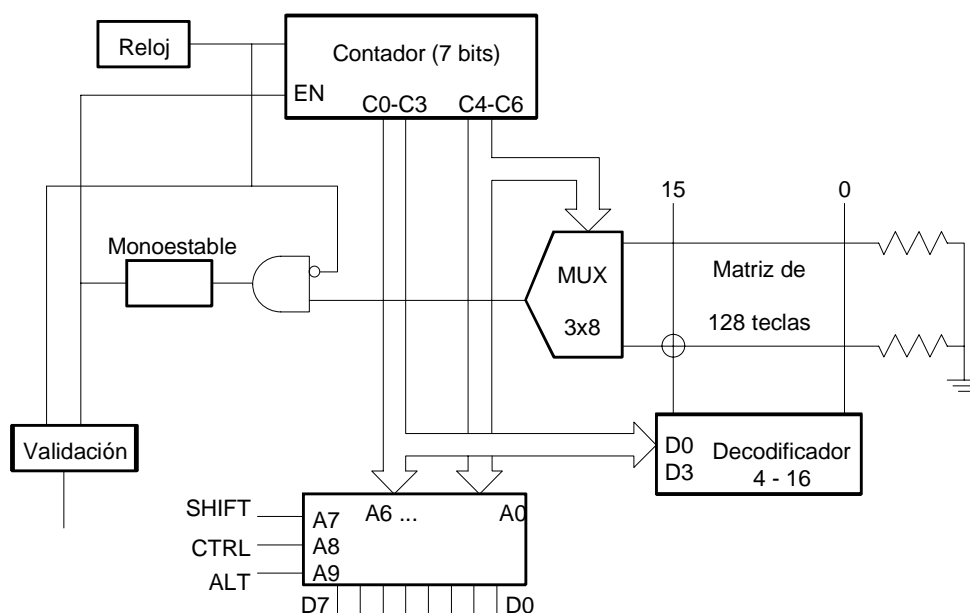


Fig. 4.13 Ejemplo de teclado matricial con exploración secuencial

#### 4.3.4 Codificación por microprocesador

Una técnica que se está expandiendo fuertemente consiste en la utilización de microcontroladores como elementos de lógica activa en la codificación de teclados. Esta técnica permite, con un mínimo de componentes, realizar funciones que hasta el presente raramente eran llevadas a cabo directamente por el teclado como periférico. Entre estas funciones cabe mencionar: exploración secuencial; protección contra pulsaciones simultáneas; transcodificación; modos múltiples; selecciones de modo complejo; salidas en paralelo o en serie; memoria FIFO en caso de pulsación más rápida que el posible acceso por parte de la CPU; autorrepetición en teclas seleccionadas (se denomina autorrepetición al hecho de que si se mantiene pulsada una tecla, pasado un tiempo prudencial, entre 0,5 a 1 segundo, se repiten las validaciones del código asociado a un ritmo aproximado de 10 Hz); autorización o inhibición total o parcial del teclado; generación de señal audible para realimentación acústica; paridad; detección de errores de operación; etc.

Los módulos de pulsador de estado sólido disponen de una entrada y una salida, preparadas para exploración. La señal es válida cuando la entrada de interrogación (exploración) está activada y la tecla pulsada. Dado que el módulo pulsador es un interruptor de estado sólido, con salida digital, podrá conectarse directamente con el microprocesador. No es preciso preveer rutinas para eliminación de rebotes, ni circuitos detectores especiales, tales como los que precisan los pulsadores capacitivos o de núcleos de ferrita; con todo ello se dispone de mayor espacio en la ROM para permitir incrementar sus prestaciones.

### 4.3.5 Doble codificación

El control del teclado puede efectuarse totalmente mediante un único microprocesador, tal como se ha visto en el apartado precedente, o bien distribuyendo el control entre el microprocesador localizado en el propio teclado y el situado en la unidad central. En este último caso pueden emplearse diversas soluciones, siendo la de doble codificación una de las más extendidas.

La técnica de doble codificación consiste en generar dos códigos por cada tecla, el primero en el momento de su pulsación («make») y el segundo en su liberación («release»). Ambos códigos presentan una notable similitud; de hecho sus siete bits de menor peso son habitualmente idénticos, diferenciándose en que el octavo bit es un «0» en la pulsación y un «1» en la liberación o viceversa. Los siete bits de peso inferior acostumbran a codificar de forma binaria la posición física de la tecla, siendo, por tanto, independientes del carácter o función asimilados a dicha tecla; como consecuencia, para obtener el código alfanumérico asociado a la tecla en cuestión, es preciso efectuar una transcodificación mediante el microprocesador de la unidad central. La tarea adicional que comporta esta rutina queda ampliamente compensada por la flexibilidad que se obtiene, puesto que cualquier tecla puede adoptar el significado que se desee, el cual puede ser modificado sin más que alterar la tabla de transcodificación activa. Asimismo, cualquier tecla puede adoptar la característica de tecla de modalidad (tal como: shift, control, alt., etc.) dado que el microprocesador de la unidad central es quien decide, en base a su microprogramación, la funcionalidad asignada a cada tecla. Asimismo, dado que la unidad central de control conoce en todo momento qué teclas están pulsadas (puesto que ya ha recibido el código de pulsación y aún no el de liberación) pueden estructurarse secuencias complejas que comporten combinaciones simultáneas de múltiples teclas, o bien funciones tales como la autorrepetición «typematic» consistente en repetir a una frecuencia de 10 a 15 c/s el código de la tecla cuya pulsación haya sido mantenida un tiempo superior a unos 500-750 ms.

## 4.4 PULSACIÓN SIMULTÁNEA DE VARIAS TECLAS

Existen diversos métodos para controlar la forma en que el teclado informará al procesador central de cómo y cuando se activan los pulsadores. Las técnicas que se utilizan intentan por un lado evitar conflictos ante el manejo erróneo del teclado, y por otro, agilizar el manejo y evitar esperas superfluas.

El problema se plantea cuando se pulsán varias teclas simultáneamente y, una vez hecho esto, cuando se van liberando sucesivamente.

En los teclados conectados a ordenadores (donde está corriendo un sistema operativo), lo más usual es que el manejador genere una interrupción por cada tecla que se pulse. Incluso, en algunos casos, se genera también una interrupción cuando la tecla se libera, para dotar al manejador de toda la información posible y dejar que actúe como quiera.

Este problema común a cualquier tipo de teclado aparece cuando se pulsán dos o más teclas simultáneamente. Si no se toma ninguna precaución, lo más común es que se provoque una suma inclusiva de bits dando lugar a la generación de un tercer código que no corresponde a ninguna de las dos teclas pulsadas, perdiendo asimismo la información correspondiente a estas últimas.

Aunque pueda parecer que este es un problema de operatoria ajena a los equipos, la tecnología ha desarrollado una serie de soluciones que permiten soslayar los defectos humanos de manipulación. Las soluciones más usuales son las siguientes:

- ◆ sobrepulsación de 2 teclas
- ◆ inhibición de N teclas
- ◆ sobrepulsación de N teclas.

#### **4.4.1 Sobrepulsación de dos teclas**

(En inglés: «2-Key rollover».) Cuando se pulsan varias teclas simultáneamente, sólo se transmite el código asociado a la primera, quedando la segunda y consecutivas bloqueadas hasta liberar la primera. Si una segunda tecla fue pulsada tras la primera y liberada antes que ésta, no queda registrada, perdiéndose su información. Este método garantiza que no aparezcan códigos erróneos, pero no impide pérdida de información.

#### **4.4.2 Inhibición de N teclas**

(En inglés: «N-Key lockout».) Cuando se pulsan varias teclas simultáneamente no se generan códigos a la salida. Cuando una sola tecla está pulsada, el teclado genera su código, pero cuando se pulsa una segunda tecla mientras la primera permanece activa, el teclado no generará ningún código mientras no se libera la primera. Una vez que se libera la primera, el código correspondiente a la segunda aparecerá a la salida. Por tanto si se pulsan N teclas simultáneamente, permanecerá inhibida la codificación hasta que todas las teclas regresen a la posición de reposo, excepto una.

Este procedimiento es muy similar al de sobrepulsación de dos teclas, diferenciándose por el hecho de que en aquél durante la pulsación múltiple se dispone del código de la primera tecla; mientras que en éste, durante la pulsación múltiple, la salida permanece inhibida.

Merece el mismo comentario respecto a posibles pérdidas de información.

#### **4.4.3 Sobrepulsación de N teclas**

(En inglés: «N-Key rollover».) Cuando se pulsa una tecla, se genera su código correspondiente. Si la primera tecla permanece deprimida mientras se pulsa una segunda, se generará la salida correspondiente a la segunda tecla.

Si se pulsa una tercera tecla mientras las dos primeras (o alguna de ellas) están todavía activadas, se genera el código correspondiente a esta tercera tecla.

En un caso extremo, todas las teclas del teclado excepto una pueden ser pulsadas; cuando se activa la última tecla, se generará su código asociado.

Este método se encuentra comúnmente en máquinas eléctricas de escribir, donde ha demostrado su virtud de poder incrementar notablemente la velocidad de tecleo sin generación de errores, ni pérdidas de información.

Generalmente se acepta que los procedimientos de «inhibición de N teclas» o «sobrepulsación de dos teclas» son suficientes cuando aparece una indicación visual, tal como iluminación de una pantalla de TRC, impresión sobre papel o similares. El procedimiento de «sobrepulsación de N teclas», deseable en todos los casos, es absolutamente necesario cuando no se dispone de información visual asociada al teclado.

## **4.5 RATONES Y TABLETAS GRÁFICAS**

Las nuevas tendencias de la programación actual nos hacen trabajar con iconos, ventanas, menús, etc. y se precisa un periférico apuntador: algo que sea capaz de mover gráficamente el puntero que aparece en pantalla. Naturalmente, existen diversas formas de mover ese puntero por la pantalla, que son los periféricos apuntadores: ratones, tabletas gráficas, lápices ópticos, trackballs, pantallas táctiles, etc. Hay claramente dos ganadores: los ratones y las tabletas gráficas de precisión o tabletas digitalizadoras.

Aunque fue Apple quien lo popularizó con su ordenador Macintosh, lo cierto es que la tecnología actual del ratón es obra de Xerox, quienes reinventaron el concepto y los primeros diseños creados por Douglas Engelbert, en el Stanford Research Institute en 1967.

En 1970, Xerox creó el primer ratón digital que incluía circuitos de conversión analógico digitales. El primer ratón para PC fue creado por Mouse Systems, compañía que dio origen a todo un estándar. Microsoft introdujo su primer ratón en 1983, un año antes que Apple en Macintosh.

#### 4.5.1 Funcionamiento básico del ratón

El funcionamiento del ratón es sencillo y ha evolucionado con el paso de los años. Los primeros ratones, electromecánicos dieron paso a los ratones optomecánicos, que son los que se conocen en la actualidad: una combinación mecánica (la bola y los rodillos) con detección digital (fotosensores). Los ratones ópticos son lo último que ha lanzado la tecnología del sector: se utiliza una alfombrilla (o tableta) especial, que contiene una rejilla de líneas dibujadas en ella. El ratón óptico no contiene partes mecánicas, y esa es una gran ventaja puesto que se traduce en menos fallos. A todo esto hay que añadir los botones del ratón: Su número puede variar entre uno (como el ratón Macintosh) y tres. El número normal en los PC's es dos para los compatibles Microsoft, y tres para los compatibles Mouse Systems.

El ratón de tipo optomecánico contiene una bola (fig. 4.14), generalmente de caucho, teflón o goma, que gira soportada sobre tres ejes: Uno horizontal, otro vertical y un tercero oblicuo, que con la ayuda de un muelle sirve para mantener la bola en contacto permanente con los otros dos ejes. Cada uno de los ejes está unido a un pequeño "plato" circular con pequeñas rendijas, que giran en torno a los rodillos al mover la bola (el ratón) sobre la mesa. En la figura (4.14) el tercer eje estaría oculto detrás de la esfera y empujaría la esfera contra los dos ejes que mueven las ruedas ranuradas que interrumpen periódicamente los haces de los fotosensores.

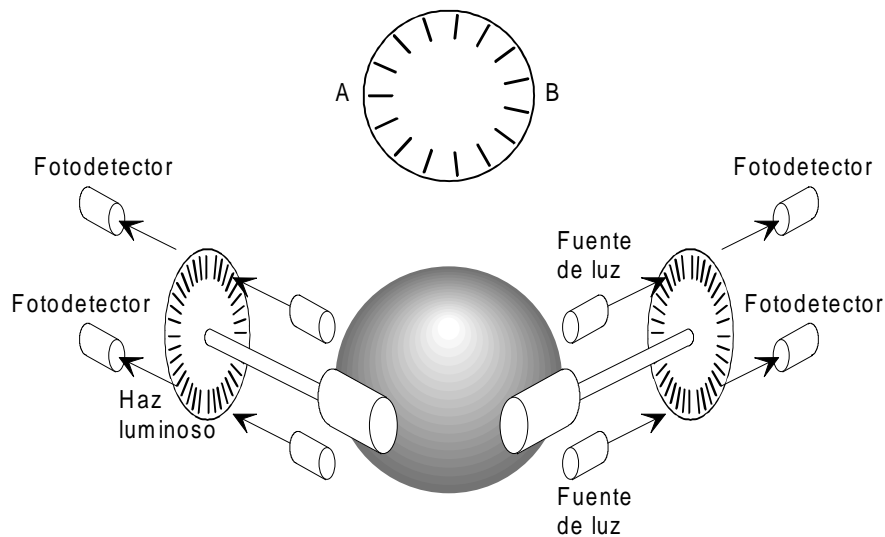


Fig. 4.14 Esquema de un ratón optomecánico

Dos pequeños haces de luz inciden sobre dos fotosensores. La diferencia entre luz-oscuridad en las rendijas debido al giro del plato permite detectar el movimiento, las rejillas están ligeramente desplazadas (en el gráfico puntos A y B); es decir no hay orificios diametralmente opuestos. Esto hace que uno de los dos fotosensores se dispare el primero indicando el sentido de giro, como se muestra en la figura (4.15). La información se convierte en bits en función del movimiento y se transmite al ordenador para desplazar el puntero por la pantalla.

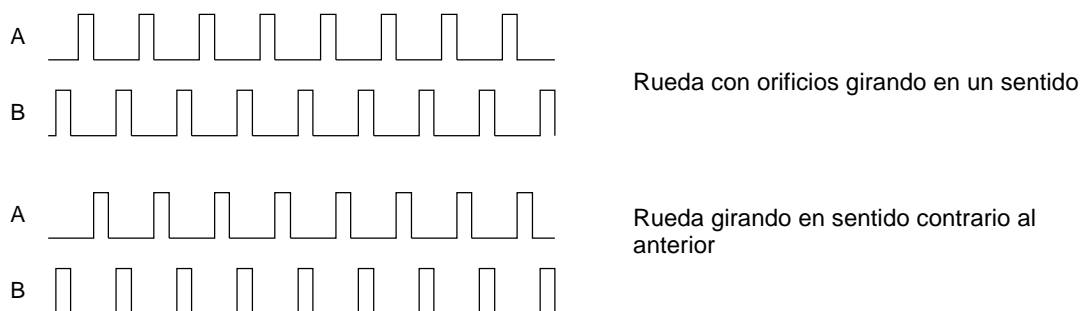


Fig. 4.15 Forma de los trenes de pulsos de los fotodetectores que permiten averiguar el sentido de giro.

La tecnología de los ratones ópticos es completamente diferente. La parte interior del ratón contiene dos agujeros y dos focos de luz (A y B) que emiten sendos haces, que reflejados en la alfombrilla especial, y pasando a través de unas lentes, son detectados por un par de fotosensores (C y D). Los fotosensores pueden medir los pasos de luz (oscuridad), reflejados entre las líneas o puntos de las rejillas, convirtiéndolos en información sobre el movimiento. Hay distintas tecnologías basadas en este sistema: Unas utilizan luz polarizada y otras luz roja e infrarroja sobre una alfombrilla con líneas negras y azules, para distinguir desplazamientos horizontales y verticales.

#### 4.5.2 Nuevos ratones

En la actualidad están surgiendo nuevos tipos de ratones que aún no tienen mucha implantación en el mercado. Un modelo de ratón curioso es el 'ratón a distancia', el cual es un ratón convencional que en vez de cable utiliza infrarrojos para transmitir las señales al ordenador. Este tipo, evita el molesto cable, pero su campo de acción sobre la mesa está limitado, debido a que necesita una línea recta libre de cualquier obstáculo entre él y el receptor que está conectado físicamente al ordenador ya que al permanecer estático no es preciso que sea un elemento independiente con lo que se elimina un dispositivo y la molesta conexión al sistema central. El principio de funcionamiento es el mismo que en los mandos a distancia de los equipos de audio y vídeo doméstico.

Más recientemente se han introducido los ratones inalámbricos que emplean una señal de radiofrecuencia, con lo que no se necesita una línea recta despejada entre el ratón y el receptor. Estos ratones pueden actuar desde cualquier posición de una habitación y esto supone un problema cuando varios ratones de este tipo, conectados a distintos equipos deben convivir en un entorno cercano. Para solventar este problema emplean múltiples canales de emisión de forma que cada par emisor-receptor disponga de un canal exclusivo. De esta forma no se producen interferencias.

Otro modelo de ratón es el 'ratón lápiz'. Es un mini-ratón con una minibola colocada en la punta de un lápiz grande. Se puede manejar como un lápiz sobre la mesa o los papeles, y la bola rueda de la misma forma.

Los ratones denominados "trackball" son una extraña variante de los ratones. Si cogemos un ratón, le damos la vuelta y movemos la bolita con el dedo, ya tenemos un trackball. Son tan similares que la mecánica interna es prácticamente la misma. El trackball puede instalarse sobre una consola de ordenador, de modo que no es necesario el cable que conecta el ratón a la máquina, por lo que se ha popularizado entre los ordenadores portátiles.

En este tipo de equipos se han introducido recientemente dos tipos de ratones de reducidas dimensiones y ausentes de partes mecánicas. El modelo popularizado por Toshiba e incluido en todos sus portátiles, incorpora entre las teclas un pequeño apéndice montado sobre unos pequeños elementos piezoeléctricos. Los elementos piezoeléctricos proporcionan una tensión eléctrica cuando son deformados en una determinada dirección. Si colocamos dos elementos con sus

direcciones más sensibles en sentido perpendicular, ya tenemos una forma de determinar la dirección y sentido del desplazamiento.

El otro modelo "touch-pad", consiste en un pequeño rectángulo sensible al tacto. Al mover el dedo sobre este rectángulo el dispositivo lo detecta y envía al sistema la información correspondiente. Al igual que con las pantallas táctiles donde podemos encontrar sus antecesores más directos, son muy diversas las tecnologías que se emplean para detectar la posición del dedo.

#### **4.5.3 Tabletas gráficas**

Desde el inicio de la informática actual, las tabletas gráficas, -también llamadas tabletas digitalizadoras o simplemente digitalizador- han tenido gran importancia en muchos campos de aplicación, pero especialmente en dibujo y diseño asistido por ordenador (CAD).

El trabajo realizado por estas tabletas, consiste en transformar en imágenes los dibujos que se realizan sobre ellas y presentarlas en la pantalla.

Las tecnologías de las tabletas gráficas, al igual que los ratones han evolucionado con el paso del tiempo. En las primeras tabletas sensoras, el usuario debía presionar el lápiz sobre la superficie, lo que producía el consiguiente desgaste de la tableta; además la precisión, o número de puntos detectables, no era muy grande. Actualmente, las tabletas pueden utilizarse con un lápiz normal (presión), con un ratón especial (campos magnéticos) o con un lápiz propio (señales transmitidas por cable). En muchos casos, la precisión obtenida es del orden de 1000 puntos por pulgada. Incluso se puede dibujar sin tocar la tableta con el lápiz: un papel colocado encima no parece importarles a las tabletas actuales, lo que puede ser de gran utilidad en muchos casos.

Las últimas tabletas incluyen un lápiz o ratón sin cable, que proporciona una apariencia más real a la hora de dibujar en ellas. En muchas ocasiones las tabletas pueden emular el funcionamiento de los ratones por software para garantizar su compatibilidad.

La ventaja de las tabletas es que son periféricos de movimiento absoluto, frente a los ratones que son de movimiento relativo. Un simple 'click' en cualquier lugar de la tableta y el puntero saltará hasta allí automáticamente, independientemente de su posición inicial.

Funcionan detectando la posición absoluta del cursor o lápiz sobre su superficie. Los modelos antiguos detectan la presión de un lápiz sobre una rejilla de diminutos contactos situada bajo la tarjeta. Esto permite calcular rápidamente las coordenadas x e y sobre las que se encuentra. En los modelos más actuales, el lápiz o cursor emite señales o crea un campo magnético sobre la tableta, con lo cual también se puede detectar su posición. Estos sistemas permiten escribir sin ni siquiera tocar la tableta. Este hecho ofrece la posibilidad de interponer un documento entre la tableta y el lápiz.

## **4.6 LECTORES DE CÓDIGO DE BARRAS**

El código de barras es una de las técnicas de recogida de datos usada con ordenadores de más rápido crecimiento aunque en sectores bastante específicos. Existen escáner ópticos que han llegado a ser muy familiares debido a su uso en supermercados y en otros pequeños comercios. Tienen también un gran campo de aplicación en la industria, transportes de mercancías, sistemas automatizados de almacenaje, etc. Se han diseñado de modo que deben recoger la información que se halla impresa como una secuencia de barras de ancho y espaciado variable, u ocasionalmente como una serie de círculos concéntricos. Un código de barras simple se muestra en la figura (4.16). El código de barras puede verse como una versión ampliada del código Morse con barras estrechas representando los puntos y barras anchas representando las rayas. No obstante, la analogía no es del todo precisa ya que en el caso de los códigos de barras, los huecos o espacios en

blanco también se utilizan para almacenar información, y además, tanto las barras como los espacios pueden ser de distintos anchos y no sólo anchos y estrechos. Las barras se leen haciendo un único registro de una línea (figura 4.16). El medio en el que se encuentra el código de barras es mucho más variable que en muchas otras aplicaciones de barrido y en particular, no es necesario que sea plano. Los códigos de barras pueden usarse en latas, botellas y bolsas de plástico, y por supuesto, también en papel. Sin embargo, la especificación de los propios códigos de barras se define estrictamente y deben ser impresos con cierta precisión, algunos tipos antiguos de impresoras por ejemplo, no tienen la suficiente definición. Existen varios conjuntos de códigos (incluyendo desde los que se usan en almacenes hasta los que se usan para libros u otras aplicaciones específicas), pero en general lo principal es codificar cada producto de manera única. La codificación se realiza usualmente utilizando los primeros caracteres de código para identificar un fabricante y éste asigna los restantes caracteres según considera adecuado. Esto implica que el código de barras en sí mismo no transporta información significativa (aunque existen excepciones) sino que se usa normalmente de forma indirecta para direccionar información almacenada en una tabla.

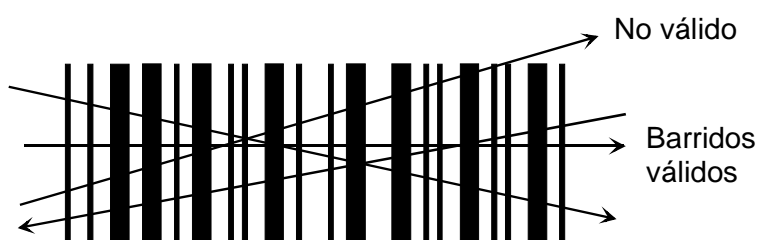


Fig. 4.16 Ejemplo de código de barras simples, sobre el que se muestran varias trayectorias de barrido válidas y una no válida debido a que no recorre todas las barras.

El código de barras es una tecnología de identificación automática que permite recoger datos en tiempo real de manera exacta y rápida pero por sí mismo no resuelve problemas. La combinación de códigos de barras con el hardware y las aplicaciones de software apropiados permitirá la mejora del funcionamiento, de la producción y en último término, de la rentabilidad.

Los símbolos de código de barras pueden imprimirse a bajo coste usando una amplia variedad de técnicas de impresión, y todo el símbolo puede ser escalado uniformemente aumentando o disminuyendo su tamaño según los requisitos de cada aplicación. El código de barras es una tecnología unidimensional (sólo el ancho de las barras y los espacios contienen información). La altura de estos elementos puede considerarse como una medida de la redundancia de datos de los símbolos de código de barras. No obstante, existen códigos de barras bidimensionales e incluso basados en círculos concéntricos aunque en este último caso la información sigue siendo bidimensional.

Los sistemas de código de barras ofrecen a menudo una seguridad muy alta de los datos, el error de sustitución puede ser a menudo mejor que 1 error en un millón de caracteres.

#### 4.6.1 Simbología de códigos de barras

Simbología es el término usado para describir reglas no ambiguas que especifican la forma en que se codifican los datos según el ancho de barras y espacios. Esta codificación se realiza de forma análoga al lenguaje. Cuando nos comunicamos por medio del habla o la escritura podemos usar cualquier lengua siempre que ambas partes estén de acuerdo sobre el idioma elegido y lo conozcan. Lo mismo sucede en el uso del código de barras y dependiendo de los datos que vayan a comunicarse se usarán diferentes simbologías. La comunicación obviamente no se producirá a menos que los equipos de lectura e impresión usen una simbología compatible. Esta consideración

es importante ya que existen docenas de simbologías de código de barras diferentes desde el comienzo de esta tecnología.

A continuación veremos los parámetros y características de la simbología de códigos.

- ◆ Conjunto de caracteres. Este término describe el rango de caracteres de datos que puede codificarse dentro de una simbología dada. Algunas simbologías pueden codificar sólo números denominándose por tanto simbologías numéricas. Otras simbologías pueden codificar información alfanumérica mientras que otras soportan códigos únicos de 128 enteros o el conjunto de caracteres ASCII.
- ◆ Tipo de simbología. La simbología del código de barras puede dividirse en dos categorías generales: discreta y continua. En el primer caso cada carácter se coloca sólo y está separado de caracteres vecinos mediante un hueco intercaracteres. El ancho del hueco no contiene información. En el proceso de decodificar cada carácter es tratado individualmente. En el segundo caso no existen huecos intercaracteres sino que cada carácter comienza con una barra y termina con un espacio. Debido que no existen huecos intercaracteres, un símbolo continuo requiere menos longitud para codificar una cantidad dada de datos. Contrarrestando estas ventajas en la densidad, está el hecho de que el rango de tecnologías de impresión demandadas disponibles es algo más restringido para códigos continuos que para simbología discreta.
- ◆ Anchura del elemento. En un símbolo de código de barras el dato es almacenado en los anchos de las barras y espacios. Existen dos tipos básicos de código de barras: los que emplean sólo dos anchos de elementos (estrecho y ancho) y los que usan anchos múltiples.
- ◆ Longitud variable o fija. Algunas simbologías por su estructura misma codifican sólo mensajes de longitud fija. Otras simbologías se usarán en entornos de longitud fija debido a consideraciones de seguridad de los datos mientras que otras podrán usarse para codificar datos de longitud variable.
- ◆ 'X' y 'Z'. 'X' es el término usado para describir el ancho nominal de los elementos estrechos de un código de barras (tanto barras como espacios). Cuando se examina un símbolo desconocido, es bastante común medir y calcular el ancho medio de los elementos estrechos, esto estrictamente hablando no es 'X' sino 'Z'.
- ◆ Densidad. Las simbologías de códigos de barras se diferencian en la cantidad de datos que pueden codificarse en una longitud dada. Para poder hacer comparaciones significativas debe tenerse en cuenta el valor de 'X' cuando se examinan densidades relativas. Hemos de señalar que la densidad está normalmente especificada sólo por los caracteres de datos. La longitud completa del símbolo incluye caracteres de comienzo/parada, zonas vacías y caracteres de chequeo.
- ◆ 'Self-checking'. Una simbología se denomina 'self-checking' si un único defecto de impresión no deja que un carácter sea transpuesto en otro carácter válido en la misma simbología.
- ◆ Código de comienzo, código de parada. Un código de comienzo es un patrón particular de barras y espacios que está situado al comienzo de un símbolo para indicar al escáner dónde comienza el código. El código de parada se encuentra al final del código para indicar el final de los caracteres de datos. Permiten además detectar fácilmente el sentido de lectura (directo o inverso)
- ◆ Carácter de chequeo. Un carácter de chequeo es un carácter (o caracteres) situado en una posición predeterminada en un código y cuyo valor está basado en algunas relaciones matemáticas de los otros caracteres del código. El escáner usa este carácter para validar que el dato correcto ha sido decodificado.
- ◆ Bidireccional. Una simbología es bidireccional si el símbolo puede barrerse indistintamente a izquierda o derecha sin que ello afecte a los datos decodificados. Casi todas las simbologías que se usan actualmente son bidireccionales. Para esto se requiere, o que haya códigos de inicio y final o que no haya caracteres que tengan una representación simétrica de barras y espacios.
- ◆ 'Self-clocking'. (autoreloj) Los escáners necesitan una información de referencia con objeto de tener una forma de medir la posición relativa de los bordes de todos los elementos. Algunas viejas simbologías incluían una pista de reloj separada. Las simbologías modernas están

diseñadas de tal forma que el escáner no necesita una pista de reloj separada para recobrar la información sobre el ancho (propiedad de 'self-clocking').

Algunas simbologías que se usan en la actualidad son la UPC ('Universal Product Code') que usa un código de 10 dígitos y se usa para identificar únicamente un producto y su fabricante. Otras simbologías estándar de la industria son 'Interleaved 2 of 5', 'Codabar', 'Code 128', 'Code 93', 'Code 49', 'Code 39', PostNet o 'Code 16K'. También hay códigos bidimensionales y circulares.

#### 4.6.2 Equipamiento de lectura

Un lector de código de barras es un dispositivo que se usa para extraer la información que está codificada ópticamente y la convierte en datos digitales compatibles con el ordenador. El lector de código de barras necesita realizar siete funciones básicas para decodificar la información de un símbolo de código de barras:

- 1.- Encontrar los elementos correctos.
- 2.- Determinar los anchos de cada una de las barras y espacios del símbolo.
- 3.- Cuantificar los anchos de los elementos en un número de niveles apropiado a la simbología que se esté usando (dos para el 'código 39', 'interleaved 2 of 5', cuatro para 'UPC' y 'código 93', cinco para 'código 128', etc.).
- 4.- Asegurar que los anchos de los elementos cuantificados son consistentes con los de las reglas de codificación de la simbología usada. Comparar el patrón de los anchos de los elementos cuantificados con una tabla de valores almacenados para esa simbología y determinar los datos codificados.
- 5.- Si es necesario, cambiar el orden de los datos. La dirección leída se determina examinando los caracteres de comienzo/parada.
- 6.- Confirmar que existen zonas vacías válidas y los dos extremos del símbolo.
- 7.- Confirmar que cualquier carácter de chequeo es consistente con el dato decodificado.

El segundo paso, es decir, la medida del ancho de los elementos se realiza por medio de un sistema de barrido electro-óptico en combinación con el software del microprocesador del lector de código de barras. Los siguientes cinco pasos se realizan por software implementado para un algoritmo particular de decodificación.

Un lector de código de barras puede considerarse como dos elementos separados: el dispositivo de entrada y el decodificador. Estos dos elementos pueden estar separados y conectados por un cable o pueden estar en una única unidad.

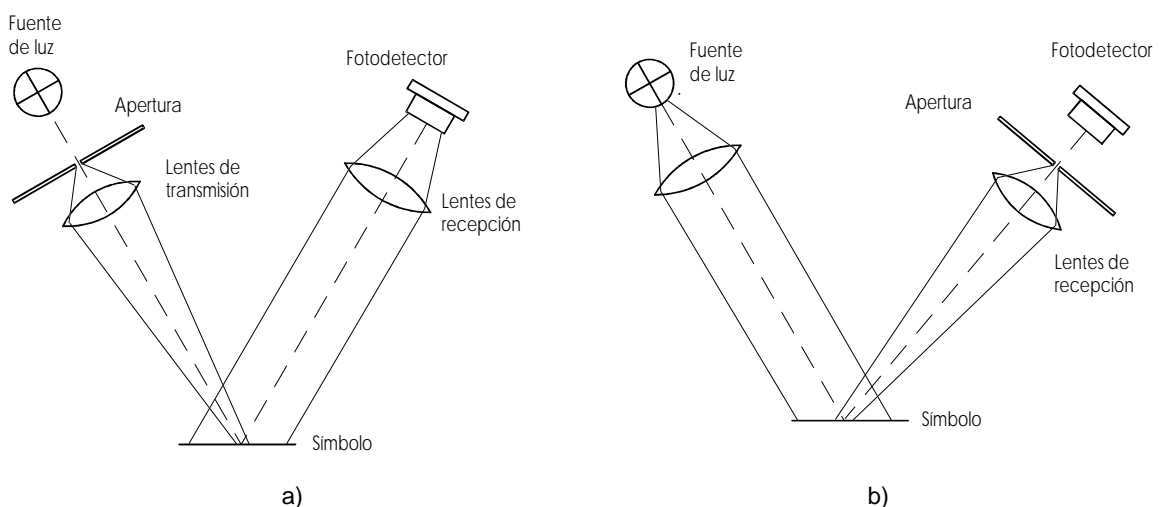


Fig. 4.17 Focalización en emisión (a) y en recepción (b)

El dispositivo de entrada emplea técnicas electro-ópticas para barrer el código de barras. El dispositivo ilumina el símbolo con energía luminosa y examina la cantidad de luz reflejada por un área localizada del símbolo. Normalmente se emplea un laser de semiconductor en los más económicos o los de tipo manual, o de gas en los estáticos y de elevadas prestaciones. Nos referiremos a esta pequeña zona o área como punto y no tiene que ser necesariamente circular. Las dimensiones del punto en un eje perpendicular al eje de la longitud de las barras debe ser consistente con el ancho del elemento más estrecho que sea barrido. El punto se forma bien a partir de un amplio conjunto de luz reflejada desde un haz focalizado (figura 4.17 a) o iluminando el símbolo con luz y usando una apertura focalizada para recoger la luz (figura (4.17 b)). Por supuesto se puede focalizar en emisión y recepción, pero en este caso se necesitaría un gran alineamiento y el código siempre tendría que colocarse en la focal de los dos sistemas para poder ser leído.

La luz reflejada del punto del símbolo es dirigida a un detector (fotodiodo o algún dispositivo equivalente) que genera una pequeña corriente proporcional a la luz recibida. Un amplificador en la entrada del dispositivo amplifica la señal desde el fotodiodo hasta un nivel adecuado de modo que pueda utilizarse. El voltaje analógico recogido por el detector es proporcional a la reflectividad observada por el punto de barrido ya que el haz de barrido cruza el patrón del código de barras. Nótese como la forma de los bordes del patrón del código de barras en la señal a la salida del detector es redondeada (figura 4.18.a). Esto es consecuencia del ancho finito del punto que hace que al pasar de una barra (oscura) a un espacio (claro) el punto no lo hace de forma instantánea sino que habrá un espacio de tiempo durante el cual, el punto luminoso incide sobre la barra y sobre el hueco simultáneamente. Por este motivo se debe conseguir un haz estrecho; cuanto más estrecho sea el punto más abrupta será la transición. Con objeto de diferenciar entre barras y espacios la señal analógica es convertida en una señal digital por medio de un circuito conocido como 'wavershaper'. Esta operación se ilustra en la figura 4.18.b.

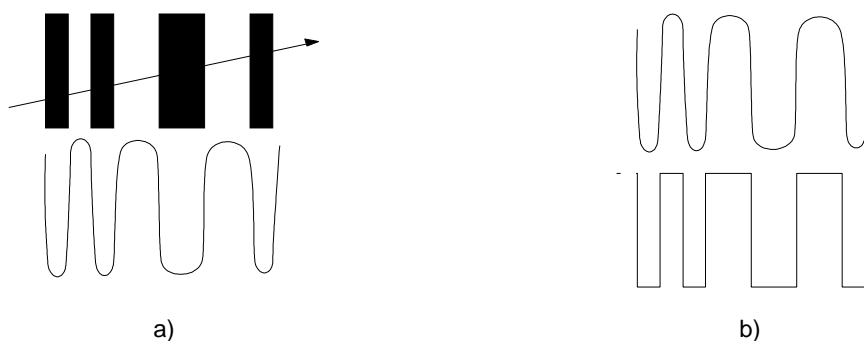


Fig. 4.18 a) Barrido de un símbolo (y representación simplificada de la salida del detector)  
b) Operación de regeneración de la señal para obtener una secuencia digital (0 y 1)

El voltaje de salida de un dispositivo de entrada puede ser analógica o digital. Si la salida es analógica debe incorporarse un circuito 'wavershaper' en la unidad del decodificador. La figura (4.19) muestra el diagrama de bloques completo del dispositivo de entrada.

Existen muchos tipos de lectores de códigos de barras. El primero, descrito como un lápiz óptico activo ('active light pen' o 'wand'), está hecho con forma de un lápiz y tiene un único fotodetector junto con una fuente de luz LED en su punta. Ésta está conectada a través de un cable a la parte estática del dispositivo. El operador pasa la punta a través del código de barras y el dispositivo detecta las barras claras y oscuras y las traduce en caracteres. El código incluye también un carácter de chequeo para comprobar que el código se ha leído correctamente y se lo indica al operador usualmente con un tono audible. No se necesita mover el lápiz a velocidad constante y siempre que el movimiento no sea demasiado irregular, puede reconocerse el patrón de barras. Además, el código puede barrerse en cualquier dirección, siempre que se recorran todas las barras (fig. 4.16).

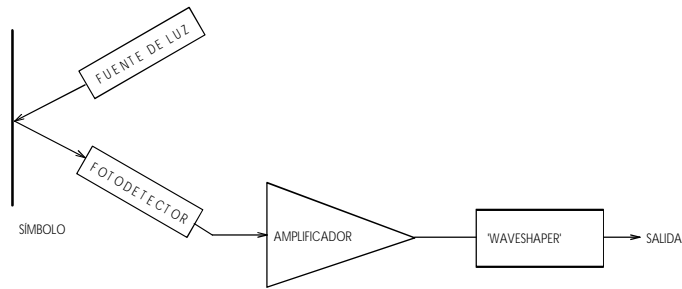


Fig. 4.19 Diagrama de bloques del dispositivo de entrada

El segundo tipo de lector es también bastante común. En este caso, el rayo del LED (o un láser semiconductor) está diseñado para barrer repetidamente a lo largo de una línea en lugar de ser un único punto. El operador sólo tiene que situar el escáner enfrente del código de barras y aproximadamente paralelo a éste y no necesariamente en contacto. El barrido del haz se consigue con la vibración de un espejo que refleja el haz.

El tercer tipo de lector de código de barras es estacionario, usualmente está construido dentro de una caja de chequeo y debajo de un panel de cristal. De nuevo se barre el haz pero esta vez sigue un curso complejo trazando líneas en cuatro direcciones usualmente, situadas entre ellas a 45 grados para lo que se emplea un prisma giratorio. El código de barras no necesita estar en una superficie paralela a la parte superior de la caja ya que siempre que pueda 'verse' por el escáner, puede leerse. Si varios códigos de barras están impresos en el paquete, de modo que siempre hay uno que puede verse por el escáner, el código puede pasarse sobre el escáner en cualquier orientación, aunque esto no es frecuente. De nuevo, el dispositivo indica al operador si el código se ha leído correctamente. Este es el tipo de lector que podemos encontrar habitualmente en los supermercados.