

Tabla 4.2 Código BCD.

Dígito decimal	BCD 8421
0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001

Utilizando el código ASCII de 7 bits, representaremos la palabra bit.

Letra	B	I	T
Código decimal	66	73	84
Código binario	1000010	1001001	1010100

## ● CÓDIGOS

Un código es un conjunto de símbolos y normas que permiten la representación de información. Gracias a los códigos es posible que las computadoras (que sólo trabajan con datos binarios) puedan procesar información que no es numérica. Los códigos digitales permiten la representación de números, letras y señales de control usando únicamente bits.

Seguramente el lector debe conocer algunos códigos, como la clave Morse, donde cada letra se representa mediante una secuencia de puntos y rayas. De modo similar, las antiguas tarjetas perforadas podían contener información gracias al código Hollerith, el cual asociaba la posición de las perforaciones con símbolos alfabéticos específicos.

### ● BCD

El código binario decimal o BCD (del inglés *Binary-Coded Decimal*), también llamado código 8421, representa cada dígito decimal por medio de cuatro dígitos binarios. El BCD no tiene equivalencia para letras. Este código asigna una representación binaria de 4 bits, entre 1010 y 1111, a cada dígito entre 0 y 9 sin signo. La conversión entre las representaciones BCD y decimal se puede llevar a cabo simplemente sustituyendo 4 dígitos BCD por cada dígito decimal y viceversa. Sin embargo, en la práctica se suelen agrupar dos dígitos BCD en un byte de 8 bits, que por tanto puede representar cualquier valor comprendido entre 0 y 99. El código BCD se puede utilizar en operaciones aritméticas. La suma es la más importante de estas operaciones, ya que las otras tres se pueden llevar a cabo utilizando la suma.

### ● EBCDIC

Este código diseñado por la IBM es una versión ampliada del BCD y requiere de 8 bits, con los cuales puede representar letras y símbolos, además de los números.

En las transmisiones de datos es necesario utilizar un gran número de caracteres de control para la manipulación de los mensajes y realización de otras funciones, de ahí que el código BCD se extendiera a una representación utilizando 8 bits, que dio origen al código EBCDIC (del inglés *Extended Binary Coded Decimal Interchange Code* o código de intercambio del código binario decimal extendido).

### ● ASCII

El código ASCII (*American Standard Code for Information Interchange* o Código Estadounidense Estándar para el Intercambio de Información) agrupa 7 bits con los que se representan 96 caracteres y 32 símbolos de control. Se utiliza para el intercambio de información

entre dispositivos fabricados por diferentes empresas y para transmisión telefónica de datos. La versión extendida del código ASCII utiliza 8 bits para manejar 255 caracteres.

Los códigos del 0 al 31 no se utilizan para caracteres. Éstos se denominan caracteres de control ya que se utilizan para acciones como:

- Retorno de carro (CR).
- Timbre (BELL).
- Los códigos 65 al 90 representan las letras mayúsculas.
- Los códigos 97 al 122 representan las letras minúsculas (si cambiamos el sexto bit, se pasa de mayúscula a minúscula; esto equivale a agregar 32 al código ASCII en base decimal).
- No se representa ninguna letra con tilde.

Su uso primordial es facilitar el intercambio de información entre sistemas de procesamiento de datos y equipos asociados y dentro de sistemas de comunicación de datos.

## ● Unicode

Unicode es un sistema para representar caracteres de todos los diferentes idiomas en el mundo. Unicode es un sistema de codificación de caracteres de 16 bits desarrollado en 1991. Unicode puede representar cualquier carácter a través de un código de 16 bits, independientemente del operativo o el idioma de programación utilizado. Incluye casi todos los alfabetos actuales (incluido árabe, armenio, cirílico, griego, hebreo y latín) y es compatible con el código ASCII.

Una nueva norma, llamada Estándar Unicode (Unicode Worldwide Code Standard o Estándar del Código Universal Unicode), se usa en las computadoras con Windows NT. IE4 y Netscape 4 entienden al Unicode y su uso se está extendiendo. El Unicode versión 3.0 contiene 49,194 caracteres de los utilizados en los idiomas más importantes del mundo. Además de letras y números, el Unicode incluye puntuación, símbolos matemáticos y técnicos, formas geométricas, caracteres gráficos y modelos del Braille. Unicode contiene el código ASCII de norma como un subconjunto.

## ● REPRESENTACIÓN DE IMÁGENES

Hoy en día las imágenes se representan en una computadora mediante uno de dos métodos: gráfico de mapa de bits o gráfico de vectores.

### ● Mapa de bits

En este método, una imagen se divide en una matriz de píxeles (picture elements: elementos de imagen), donde cada píxel es un pequeño punto. El tamaño del píxel depende de lo que se conoce como resolución. Por ejemplo, una imagen puede dividirse en 1 000 píxeles o 10 000 píxeles. En el segundo caso, aunque hay una mejor representación de la imagen (mejor resolución), se necesita más memoria para almacenarla.

Después de dividir una imagen en píxeles, a cada píxel se le asigna un patrón de bits. El tamaño y el valor del patrón dependen de la imagen. Para una imagen formada sólo por puntos blancos y negros (por ejemplo, un tablero de ajedrez), un patrón de bits es suficiente para representar un píxel. Un patrón de 0 representa un píxel negro y uno de 1 re-

presenta un pixel blanco. Luego los patrones se registran uno tras otro y se almacenan en la computadora.

A continuación se muestra un fragmento de la imagen ampliado en donde se pueden observar los pixeles y la coloración de los mismos. En la vista en miniatura se observa con claridad el objeto, representado por  $256 \times 256$  pixeles (65 536 pixeles totales) y una profundidad de color de 24 bits en formato RGB (Red, Green y Blue) en donde se tienen 8 bits para representar cada uno de los colores. En la vista ampliada sólo vemos el perfil izquierdo del tigre mostrando el ojo, y podemos observar los pixeles.

### ● Gráficos vectoriales



**Figura 4.7** Imagen de tigre blanco.

El problema con el método de los gráficos de mapa de bit es que los patrones de bits exactos para representar una imagen particular deben guardarse en una computadora. Posteriormente, si se desea cambiar el tamaño de la imagen debe cambiarse el tamaño de los pixeles, lo cual crea una apariencia difusa y granulada. No obstante, el método de gráficos vectoriales no guarda los patrones de bits. Una imagen se descompone en una combinación de curvas y líneas. Cada curva o línea se representan por medio de una fórmula matemática. Por ejemplo, una línea puede describirse mediante las coordenadas de sus puntos extremos y un círculo puede describirse mediante las coordenadas de su centro y la longitud de su radio. La combi-

nación de estas fórmulas se almacena en una computadora. Cuando la imagen se va a desplegar o imprimir, el tamaño de la imagen se proporciona al sistema como una entrada. El sistema rediseña la imagen con el nuevo tamaño y usa la misma fórmula para dibujar la imagen. En este caso, cada vez que una imagen se dibuja, la fórmula se vuelve a evaluar.

Los gráficos vectoriales, también conocidos como gráficos orientados a objetos, son el segundo gran grupo de imágenes digitales. Son más simples que los gráficos de mapas de bits, ya que en ellos las imágenes se almacenan y representan por medio de trazos geométricos controlados por cálculos y fórmulas matemáticas, tomando algunos puntos de la imagen como referencia para construir el resto.

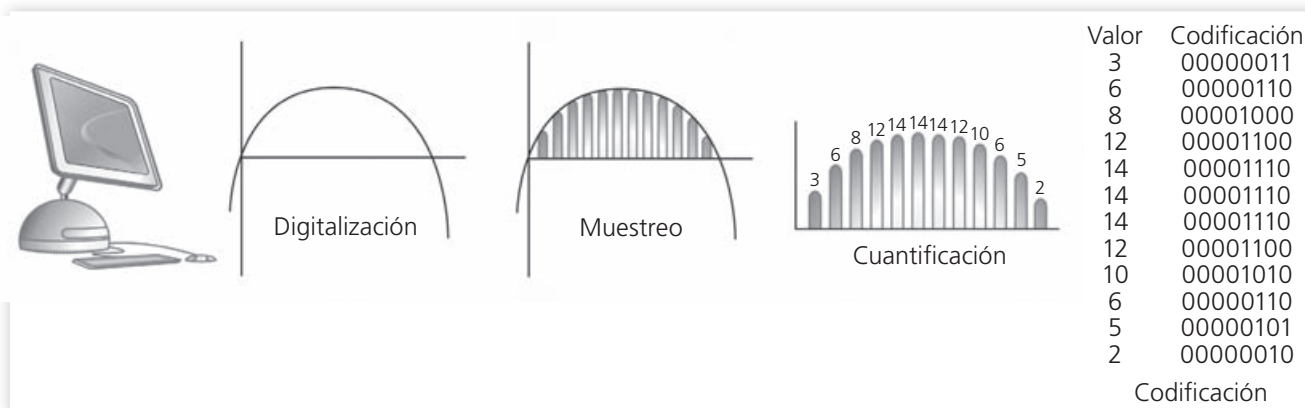
Por tanto, las imágenes en los gráficos vectoriales no se construyen pixel a pixel, sino que se forman a partir de vectores, que son objetos formados por una serie de puntos y líneas rectas o curvas definidas matemáticamente.

Por ejemplo, hemos visto que una línea se define en un gráfico de mapa de bits mediante las propiedades de cada uno de los pixeles que la forman, mientras que en un gráfico vectorial se hace por la posición de sus puntos inicial y final y por una función que describe el camino entre ellos. Análogamente, un círculo se define vectorialmente por la posición de su punto central (coordenadas  $x, y$ ) y por su radio ( $r$ ).

### ● REPRESENTACIÓN DE AUDIO

Para digitalizar el sonido se pueden utilizar diversos dispositivos; sin embargo, la forma más simple de hacerlo es utilizar la tarjeta de sonido y cualquier otro dispositivo que podamos usar para generar sonido; puede ser un micrófono, un reproductor de discos, etcétera.

El primer paso para convertir (digitalizar) el sonido será el muestreo o *sampling*, el cual consiste en tomar muestras a intervalos de tiempo regulares; las muestras se cuantifican. La cuantificación significa asignar un valor (de un conjunto) a una muestra. Los valores cuantificados se cambian a patrones binarios y se almacenan en un patrón de bits para cada muestra.



**Figura 4.8** Digitalización de audio.

Debemos definir la frecuencia de muestreo, ya que ésta nos indica la cantidad de muestras que se toman en determinado intervalo de tiempo, por ejemplo, en un segundo. Por lo general, la frecuencia se da en hercios (ciclos por segundo), y podemos variar desde 8000 hasta 48000.

Al igual que en las imágenes, se debe definir la resolución o profundidad del sonido, que es un número que indica cuántos bits (longitud del patrón de bits) se utilizan para representar cada muestra. Por razones obvias, a mayor resolución y frecuencia de muestreo mejor será la calidad del sonido almacenado.

Supongamos que queremos almacenar un sonido con dos canales, es decir, sonido estéreo con una frecuencia de muestreo de 44 100 hercios y 16 bits. Esto significa que se deben tomar unas 44 100 muestras por segundo y cada una se representará con 16 bits.

Por tanto, si queremos saber cuántos ceros y unos se necesitan para almacenar 3.5 minutos de sonido digital con calidad de CD, debemos realizar el siguiente cálculo:

3.5 minutos × 60 segundos = 210 segundos.
210 segundos × 44 100 muestras por segundo = 9 261 000 muestras
9 261 000 muestras × 16 bits = 148 176 000 bits
148 176 000 bits × 2 canales = 296 352 000 bits
296 352 000 bits/8 bits por byte = 37 044 000 bytes
37 044 000 bytes/1 024 000 bytes por MB = 36.17578125 MB

Entonces, para almacenar una canción de 3.5 minutos con calidad de CD se requieren 296 352 000 bits, que son aproximadamente 36.2 MB.

Como podemos ver, el tamaño de almacenamiento requerido para un audio de estas características es demasiado. Es por eso que hoy en día existen numerosas cantidades de formatos de audio utilizados para representar el sonido, los cuales nos permiten comprimir el tamaño de los archivos. En seguida se presentan algunos de estos formatos con sus descripciones.

### Advanced Audio Coding (Codificación de Audio Avanzada)

Codificación estándar para audio reconocida por ISO en el patrón MPG-2. En teoría, almacena más que el MP3 en menos espacio. Es el formato de audio que utiliza Apple para los archivos de audio que reproduce el iPod y que pueden comprarse a través de Internet. Extensión: aac.

### WAV

Fue desarrollado por Microsoft e IBM y apareció por primera vez para el ambiente Windows en 1995. Los archivos de audio guardados en el formato de sonido Microsoft tienen esta extensión. Con el tiempo se convirtió en un estándar de grabación para música de CD. Su soporte de reproducción es uno de los más importantes pues funciona en cualquier aplicación de Windows y en equipos domésticos comunes con reproductor de CD. Extensión: wav.

### MPEG

El MP3 (MPEG-1 Audio Layer 3) es un códec de audio muy extendido. Los archivos creados con este códec tienen la extensión .mp3, por lo que también se le llama formato MP3. Su peculiaridad es su tamaño de compresión: 11 a 1, lo que quiere decir que si un CD de música normal contiene unas 13 canciones, en un CD con MP3 tendríamos 143. Aparte del ahorro de espacio hay que añadir que no se pierde calidad de sonido en tasas de bits (*bit rates*) normales o incluso, en un archivo MP3 de la tasa de bits máxima sacado de un disco de vinilo, puede tener mayor calidad de sonido que un archivo de CD. Extensión: mp3.

### AU (Audio for Unix)

Se utiliza en archivos de sonido con sistema Unix de Sun™ Microsystems and NeXT™, la extensión AU viene de Audio y también funciona como estándar acústico para el lenguaje de programación JAVA. Extensión: au.

### WMA (Windows Media Audio)

Es la abreviatura de Windows Media Audio. Es la versión de Windows para comprimir audio. Es muy parecido al MP3. No sólo reduce el tamaño de archivos grandes, sino que también se adapta a diferentes velocidades de conexión en caso de que se necesite reproducir en Internet en tiempo real. Extensión: wma.

### Ogg Vorbis

El funcionamiento de este formato de compresión es similar al de los anteriores, pues también se utiliza para guardar y reproducir música digital. Lo que diferencia a Ogg Vorbis del resto del grupo es que es gratuito, abierto y no está patentado. Su principal atractivo es la importante reducción que hace de un archivo de audio sin restarle calidad. Asimismo, se distingue por su versatilidad para reproducirse en prácticamente cualquier dispositivo y por ocupar muy poco espacio. Extensión: ogg.

### Real Networks Real Audio

Soporte multimedia creado por la empresa Real Networks, con una alta tasa de compresión y algoritmos especiales que reducen considerablemente el tamaño de los archivos de