También es posible obtener una expresión simplificada de producto de sumas para la función de la figura 3-17. En este caso, la única forma de combinar los ceros es incluyendo los minitérminos indiferentes 0 y 2 con los ceros, para dar una función complementada simplificada:

$$F' = z' + wy'$$

Al calcular el complemento de F' obtenemos la expresión simplificada como producto de sumas:

$$F(w, x, y, z) = z(w' + y) = \Sigma(1, 3, 5, 7, 11, 15)$$

En este caso, hemos incluido los minitérminos 0 y 2 con los ceros, y el minitérmino 5 con los unos.

3-6 IMPLEMENTACIÓN CON NAND Y NOR

Muchos circuitos digitales se construyen con compuertas NAND y NOR en lugar de con compuertas AND y OR. Las primeras son más fáciles de fabricar con componentes electrónicos y son las compuertas básicas empleadas en todas las familias de lógica de CI digitales. En virtud del destacado papel que las compuertas NAND y NOR desempeñan en el diseño de circuitos digitales, se han desarrollado reglas y procedimientos para convertir funciones booleanas expresadas en términos de AND, OR y NOT en diagramas lógicos NAND y NOR equivalentes.

Circuitos NAND

Se dice que la compuerta NAND es una compuerta universal porque cualquier sistema digital puede implementarse con ella. Para demostrar que cualquier función booleana se puede implementar con compuertas NAND, basta con demostrar que las operaciones lógicas AND, OR y complemento se pueden obtener exclusivamente con compuertas NAND. Esto se aprecia en la figura 3-18. La operación complemento se obtiene con una compuerta NAND de una sola entrada que se comporta exactamente como un inversor. La operación AND requiere dos compuertas NAND. La primera produce la operación NAND y la segunda invierte el sentido lógico de la señal. La operación OR se logra con una compuerta NAND que lleva inversores en cada entrada.

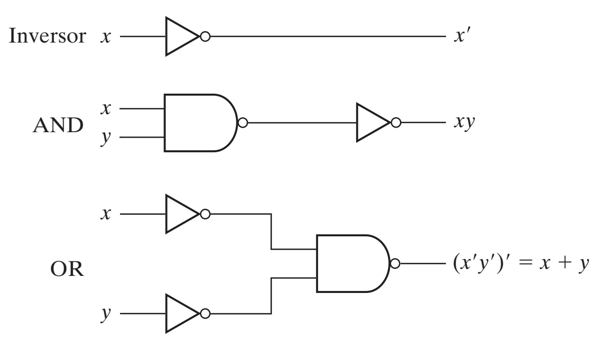
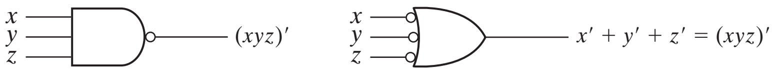


FIGURA 3-18
Operaciones lógicas con compuertas NAND



a) AND–invertir b) Invertir–OR

FIGURA 3-19
Dos símbolos gráficos para la compuerta NAND

Una forma conveniente de implementar una función booleana con compuertas NAND consiste en obtener la función booleana simplificada en términos de operadores booleanos y luego convertir la función a lógica NAND. La conversión de una expresión algebraica de AND, OR y complemento a NAND se efectúa aplicando sencillas técnicas de manipulación de circuitos que convierten los diagramas AND-OR en diagramas NAND.

Para facilitar la conversión a lógica NAND, es conveniente definir un símbolo gráfico alterno para la compuerta. En la figura 3-19 se presentan dos símbolos gráficos equivalentes para la compuerta NAND. El símbolo AND-invertir ya se definió antes y consiste en un símbolo gráfico AND seguido de un pequeño indicador circular de negación llamado burbuja. Como alternativa, podemos representar una compuerta NAND con un símbolo gráfico OR precedido por una burbuja en cada entrada. El símbolo invertir-OR para la compuerta NAND es consecuencia del teorema de DeMorgan y de la convención de que el indicador de negación denota complementación. Los dos símbolos gráficos son útiles en el análisis y diseño de circuitos NAND. Si se usan ambos símbolos en el mismo diagrama, decimos que el circuito está en notación mixta.

Implementación de dos niveles

La implementación de funciones booleanas con compuertas NAND requiere expresar la función en forma de suma de productos. Para ver la relación entre una expresión de suma de productos y su implementación NAND equivalente, consideremos los diagramas lógicos de la figura 3-20. Los tres diagramas son equivalentes e implementan la función

$$F = AB + CD$$

En a), la función se implementa con compuertas AND y OR. En (b), las compuertas AND se han sustituido por compuertas NAND y la compuerta OR se sustituyó por una compuerta NAND representada por un símbolo gráfico invertir-OR. Recuerde que una burbuja denota

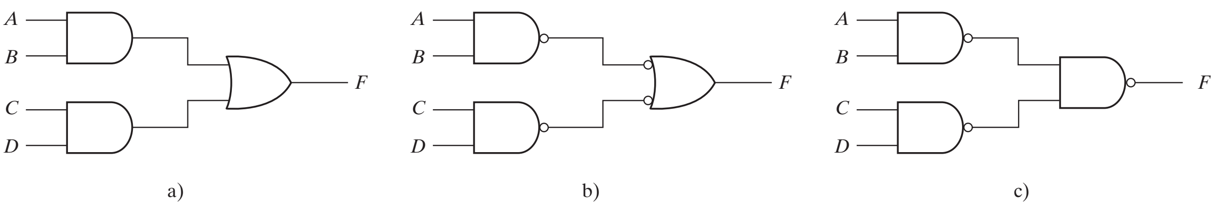


FIGURA 3-20
Tres formas de implementar $F = AB + CD$

complementación y que dos burbujas en una misma línea representan doble complementación, y pueden eliminarse. Si se quitan las burbujas de las compuertas de b) se obtiene el circuito de a). Por tanto, los dos diagramas implementan la misma función y son equivalentes.

En la figura 3-20c), se ha representado la compuerta NAND final con un símbolo gráfico AND-invertir. Al dibujar diagramas lógicos NAND, los circuitos que se muestran en b) o c) son aceptables. El de b) usa notación mixta y representa una relación más directa con la expresión booleana que implementa. La implementación NAND de la figura 3-20c) se puede verificar algebraicamente. La función que implementa se convierte fácilmente a suma de productos aplicando el teorema de DeMorgan:

$$F = \big((AB)'(CD)'\big)' = AB + CD$$

EJEMPLO 3-10

Implemente la función booleana siguiente con compuertas NAND:

$$F(x, y, z) = (1, 2, 3, 4, 5, 7)$$

El primer paso es simplificar la función como suma de productos. Esto se hace con el mapa de la figura 3-21a), del cual se obtiene la función simplificada

$$F = xy' + x'y + z$$

La implementación NAND de dos niveles se presenta en la figura 3-21b) en notación mixta. Observe que la entrada z necesita una compuerta NAND de una sola entrada (inversor) para compensar la burbuja de la compuerta del segundo nivel. En la figura 3-21c) se presenta otra forma de dibujar el diagrama lógico. Aquí todas las compuertas NAND se representan con el

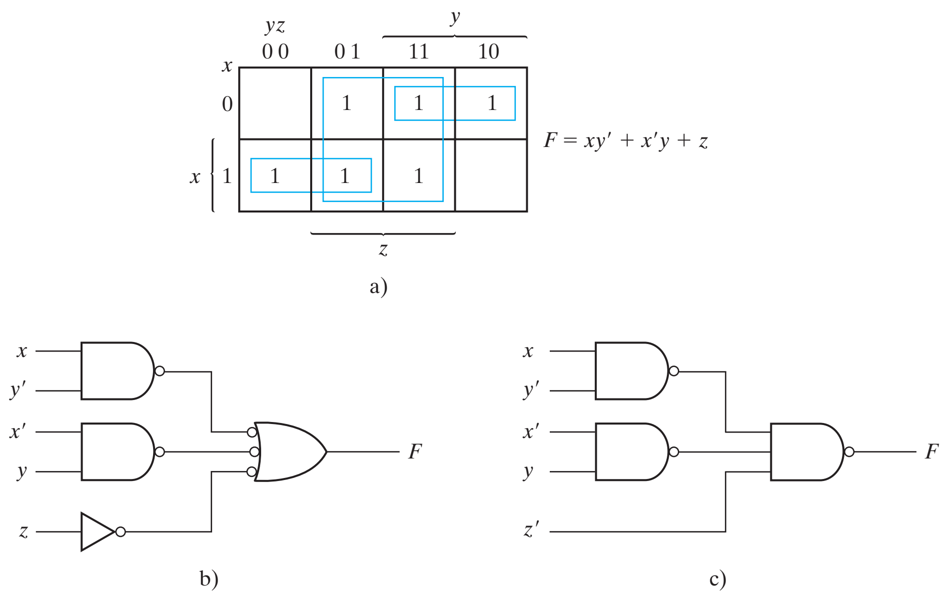


FIGURA 3-21
Solución del ejemplo 3-10

mismo símbolo gráfico. El inversor con entrada z se ha eliminado, pero la variable de entrada se ha complementado y se denota con z'.

■

El procedimiento descrito en el ejemplo anterior sugiere que es factible implementar una función booleana con dos niveles de compuertas NAND. El procedimiento para obtener el diagrama lógico a partir de una función booleana es el siguiente:

1. Simplificar la función y expresarla como suma de productos.
2. Incluir una compuerta NAND por cada término de producto que tenga por lo menos dos literales. Las entradas de cada compuerta NAND serán las literales del término. Esto constituye un grupo de compuertas de primer nivel.
3. Incluir una sola compuerta en el segundo nivel, empleando el símbolo gráfico AND-invertir o invertir-OR, cuyas entradas provienen de las salidas de las compuertas de primer nivel.
4. Los términos con una sola literal requerirán un inversor en el primer nivel, pero si esa literal solitaria está complementada, se le podrá conectar directamente a una entrada de la compuerta NAND del segundo nivel.

Circuitos NAND multinivel

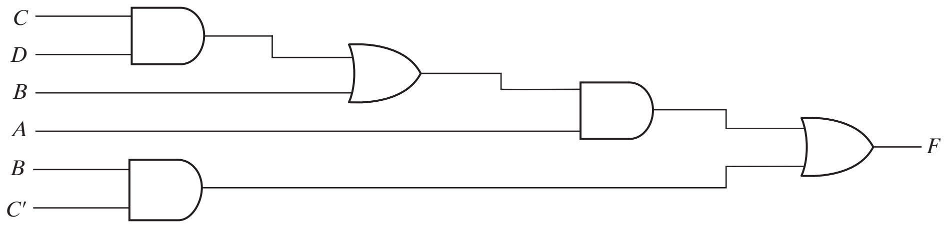
La forma estándar de expresar funciones booleanas da pie a una implementación de dos niveles. Hay ocasiones en que el diseño de sistemas digitales produce estructuras con tres o más niveles de compuertas. El procedimiento más común para diseñar circuitos multinivel es expresar la función booleana en términos de operaciones AND, OR y complemento. Entonces, la función podrá implementarse con compuertas AND y OR. Luego, si es necesario, se le puede convertir en un circuito con puras compuertas NAND. Considere, por ejemplo, la función booleana:

$$F = A(CD + B) + BC'$$

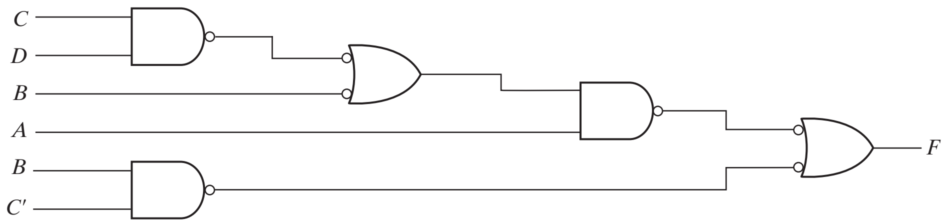
Aunque es posible quitar los paréntesis y reducir la expresión a una forma estándar de suma de productos, preferimos implementarla como circuito multinivel como ilustración. La implementación AND-OR se ilustra en la figura 3-22a). El circuito tiene cuatro niveles de compuertas. El primer nivel posee dos compuertas AND. El segundo tiene una compuerta OR seguida de una compuerta AND en el tercer nivel y una compuerta OR en el cuarto nivel. Los diagramas lógicos con un patrón de niveles alternos de compuertas AND y OR se pueden convertir fácilmente en un circuito NAND utilizando la notación mixta. Esto se muestra en la figura 3-22b). El procedimiento consiste en sustituir cada compuerta AND por un símbolo gráfico AND-invertir, y cada compuerta OR, por un símbolo gráfico invertir-OR. El circuito NAND realizará la misma lógica que el diagrama AND-OR siempre que haya dos burbujas sobre la misma línea. La burbuja asociada a la entrada B produce una complementación adicional, que debe compensarse cambiando la literal de entrada a B'.

El procedimiento general para convertir un diagrama AND-OR multinivel en un diagrama NAND con notación mixta es el siguiente:

1. Convertir todas las compuertas AND en compuertas NAND con símbolos gráficos AND-invertir.
2. Convertir todas las compuertas OR en compuertas NAND con símbolos gráficos invertir-OR.



a) Compuertas AND-OR



b) Compuertas NAND

FIGURA 3-22
Implementación de $F = A(CD + B) + BC'$

3. Revisar todas las burbujas del diagrama. Por cada burbuja que no esté compensada por otra sobre la misma línea, hay que insertar un inversor (compuerta NAND de una sola entrada) o complementar la literal de entrada.

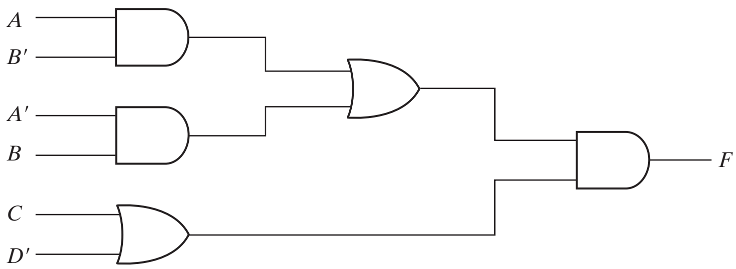
Como ejemplo adicional, consideremos la función booleana multinivel

$$F = (AB' + A'B)(C + D')$$

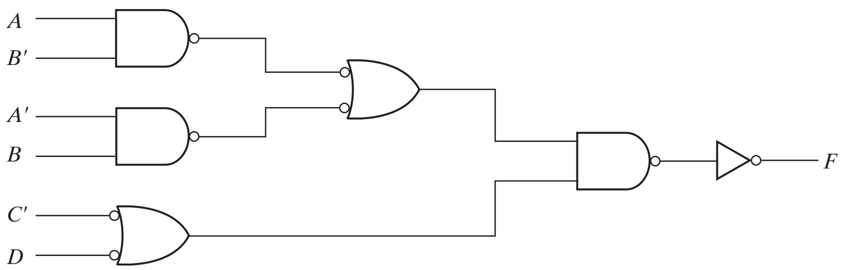
La implementación AND-OR se presenta en la figura 3-23a) con tres niveles de compuertas. La conversión a NAND con notación mixta aparece en la parte b) del diagrama. Las dos burbujas adicionales asociadas a las entradas C y D' hacen que estas dos literales se complementen a C' y D. La burbuja en la compuerta NAND de salida complementa el valor de salida, por lo que necesitamos insertar una compuerta inversora en la salida para complementar otra vez la señal y obtener el valor original.

Implementación NOR

La operación NOR es el dual de la operación NAND. Por tanto, todos los procedimientos y reglas para la lógica NOR son el dual de los procedimientos y reglas correspondientes que se han desarrollado para la lógica NAND. La compuerta NOR es otra compuerta universal que sirve para implementar cualquier función booleana. La implementación de las operaciones de complemento, OR y AND con compuertas NOR se aprecia en la figura 3-24. La operación de complemento se obtiene de una compuerta NOR con una sola entrada que se comporta exac-



a) Compuertas AND-OR



b) Compuertas NAND

FIGURA 3-23
Implementación de $F = (AB' + A'B)(C + D')$

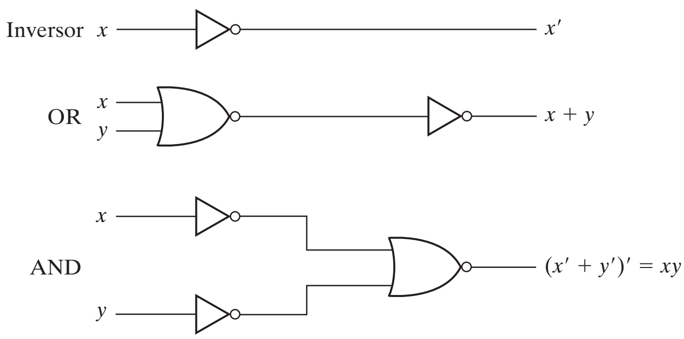
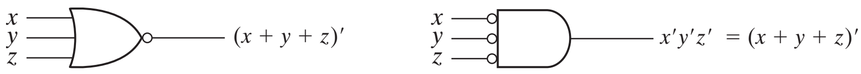


FIGURA 3-24
Operaciones lógicas con compuertas NOR

tamente como un inversor. La operación OR requiere dos compuertas NOR y la AND, una compuerta NOR con inversores en cada entrada.

Los dos símbolos gráficos de la notación mixta se muestran en la figura 3-25. El símbolo OR-invertir define la operación NOR como un OR seguido de un complemento. El símbolo invertir-AND complementa cada una de las entradas y luego realiza una operación AND. Los dos símbolos designan la misma operación NOR y son lógicamente idénticos por el teorema de DeMorgan.



a) OR-invertir b) invertir-AND

FIGURA 3-25
Dos símbolos gráficos para la compuerta NOR

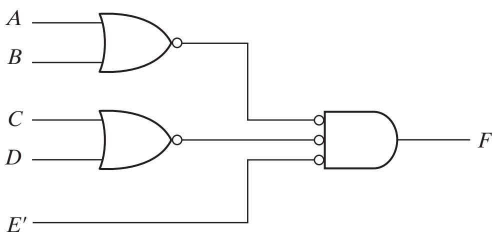


FIGURA 3-26
Implementación de $F = (A + B)(C + D)E$

Una implementación de dos niveles con compuertas NOR requiere simplificar la función en forma de producto de sumas. Recuerde que la expresión simplificada de producto de sumas se obtiene del mapa combinando los ceros y complementando. Una expresión en producto de sumas se implementa con un primer nivel de compuertas OR que produce los términos de suma, seguido de una compuerta AND de segundo nivel que da el producto. La transformación del diagrama OR-AND en un diagrama NOR se logra cambiando las compuertas OR por compuertas NOR representadas por símbolos gráficos OR-invertir, y la compuerta AND, por una compuerta NOR representada por un símbolo gráfico invertir-AND. Si la compuerta de segundo nivel tiene como entrada un término de una sola literal, ésta deberá complementarse. La figura 3-26 muestra la implementación NOR de una función expresada como producto de sumas:

$$F = (A + B)(C + D)E$$

El patrón OR-AND se detecta fácilmente quitando los pares de burbujas que estén sobre la misma línea. La variable E se complementa para compensar la tercera burbuja en la entrada de la compuerta de segundo nivel.

El procedimiento para convertir un diagrama AND-OR multinivel en un diagrama sólo NOR es similar al que se utilizó para las compuertas NAND. En el caso de NOR, hay que sustituir cada compuerta OR por un símbolo OR-invertir, y cada compuerta AND, por un símbolo invertir-AND. Toda burbuja que no esté compensada por otra burbuja en la misma línea necesitará un inversor, o se deberá complementar la literal de entrada.

La transformación del diagrama AND-OR de la figura 3-23a) en un diagrama NOR se ilustra en la figura 3-27. La función booleana para este circuito es

$$F = (AB' + A'B)(C + D')$$

El diagrama AND-OR equivalente se deduce del diagrama NOR quitando todas las burbujas. Para compensar las burbujas en cuatro entradas, es preciso complementar las literales de entrada correspondientes.

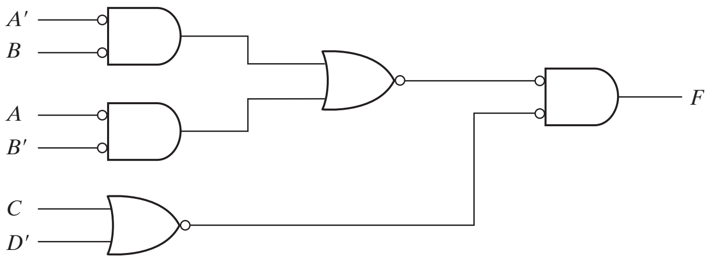


FIGURA 3-27
Implementación de $F = (AB' + A'B)(C + D')$ con compuertas NOR

3-7 OTRAS IMPLEMENTACIONES DE DOS NIVELES

Los tipos de compuertas que más comúnmente se encuentran en los circuitos integrados son las NAND y NOR. Por ello, las implementaciones de lógica NAND y NOR son las más importantes en la práctica. Algunas compuertas NAND o NOR (pero no todas) contemplan la posibilidad de una conexión con alambre entre las salidas de dos compuertas para formar una función lógica específica. Este tipo de lógica se llama *lógica alambrada* (*wired*, en inglés). Por ejemplo, si se conectan entre sí compuertas NAND TTL de colector abierto, efectúan la lógica AND alambrada. (La compuerta TTL de colector abierto se representa en la figura 10-11 del capítulo 10.) La lógica AND alambrada efectuada con dos compuertas NAND se muestra en la figura 3-28a). La compuerta AND se dibuja con líneas que pasan por el centro de la compuerta, para distinguirla de una compuerta convencional. La compuerta AND alambrada no es una compuerta física, sino un símbolo que designa la función obtenida de la conexión alambrada que se indica. La función lógica implementada por el circuito de la figura 3-28a) es

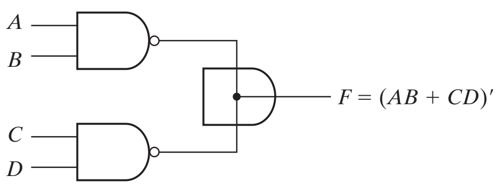
$$F = (AB)' \cdot (CD)' = (AB + CD)'$$

y se denomina función AND-OR-INVERT.

Asimismo, la salida NOR de compuertas ECL se puede vincular para desempeñar una función OR alambrada. La función lógica implementada por el circuito de la figura 3-28b) es

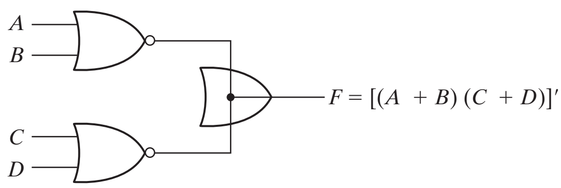
$$F = (A + B)' + (C + D)' = \left[(A + B)(C + D)\right]'$$

y se denomina función OR-AND-INVERT.



a) AND alambrado en compuertas NAND TTL de colector abierto

(AND-OR-INVERT)



b) OR alambrado en compuertas ECL

(OR-AND-INVERT)

FIGURA 3-28
Lógica alambrada