

Circuitos y Sistemas Digitales

Funciones Lógicas y Tablas de verdad

Clase 10



La excelencia no se improvisa



1. Introducción

Los mapas de Karnaugh son una herramienta fundamental en el diseño de sistemas digitales, ya que permiten simplificar expresiones booleanas de manera visual y eficiente. Desarrollados por Maurice Karnaugh, estos diagramas bidimensionales organizan las combinaciones de variables lógicas para identificar patrones y agrupar términos, facilitando la reducción de ecuaciones complejas a formas mínimas. Esta técnica no solo optimiza el uso de componentes en circuitos integrados, sino que también mejora su rendimiento al reducir retardos y costos. En esta unidad, exploraremos su estructura, reglas de agrupación y cómo aprovechar su potencial para transformar tablas de verdad en diseños lógicos simplificados, sentando las bases para aplicaciones prácticas en electrónica digital.

Aplicaremos los mapas de Karnaugh al diseño de circuitos lógicos específicos. Primero, analizaremos el sumador, un bloque esencial en sistemas aritméticos, donde utilizaremos los mapas para minimizar la lógica de acarreo y suma en operaciones binarias. Posteriormente, abordaremos el detector de números primos, un circuito combinacional que identifica si un número binario dado es primo, empleando agrupaciones estratégicas en el mapa para optimizar las compuertas lógicas requeridas. Estos ejemplos ilustrarán cómo los mapas de Karnaugh no solo son teóricamente valiosos, sino también indispensables en la implementación de sistemas digitales eficientes y confiables.

Clase 10: Funciones Lógicas y Tablas de verdad.

Resultado o resultados de aprendizaje que será abordado con el contenido de la clase. **(No agregue los criterios)**

Diseñar circuitos digitales y sistemas digitales complejos, empleando las herramientas adecuadas.

3.6. Mapas de Karnaugh

Los Mapas de Karnaugh (también conocidos como mapas K) son una herramienta útil en el diseño simplificado de circuitos lógicos digitales. Proporcionan una forma visual para minimizar expresiones booleanas, facilitando la implementación de funciones lógicas mediante compuertas electrónicas.

Fue enunciado en 1952 por E.W. Veitch y modificado, presentando la forma actual del mapa, por el ingeniero de IBM Maurice Karnaugh en 1953 (Alegre & García Sevilla, 2010).

3.6.1. Código Gray

El código Gray (o código reflejado binario) es un sistema de numeración binario en el que dos números consecutivos difieren en un solo bit (dígito binario). Esta característica lo hace especialmente útil en aplicaciones donde los cambios simultáneos de múltiples bits pueden generar errores, como en sistemas electrónicos, sensores rotatorios, y mapas de Karnaugh (donde se usa para organizar las variables).

En el código Gray, al pasar de un número al siguiente, solo un bit cambia de valor. Por ejemplo, en una secuencia de 3 bits:

Decimal 0 $\rightarrow$ 1: 000 $\rightarrow$ 001 (cambia el último bit).

Decimal 1 $\rightarrow$ 3: 001 $\rightarrow$ 011 (cambia el segundo bit).

Esto evita errores en sistemas donde múltiples bits no pueden actualizarse al mismo tiempo.

A diferencia del binario estándar, los bits en el código Gray no representan valores posicionales (como 1, 2, 4, 8...). Su valor se define por su secuencia única, la cual se construye "reflejando" los bits anteriores y cambiando el valor de un solo bit.

En los Mapas de Karnaugh las variables en los K-maps se ordenan siguiendo el código Gray para garantizar que las celdas adyacentes sean lógicamente contiguas (solo difieran en una variable). Esto permite agrupar términos simplificados.

Los mapas de Karnaugh (K-map) son una técnica gráfica para simplificar expresiones booleanas, esencial en el diseño de circuitos digitales. Su estructura se basa en una representación tabular donde cada celda corresponde a una combinación única de variables de entrada, organizadas de manera que celdas adyacentes difieran en un solo bit (código Gray). Esta disposición permite identificar visualmente términos lógicos que pueden combinarse para reducir la expresión final.

3.6.2. Características:

Los mapas de Karnaugh permiten:

1. Visualización de Combinaciones: Representa todas las posibles combinaciones de entrada en una cuadrícula.
2. Identificación de agrupamientos: Permite identificar grupos de celdas adyacentes con valores idénticos, lo que facilita la simplificación.
3. Minimización de términos: Ayuda a reducir el número de términos y variables en una expresión booleana.

3.6.3 Metodología para usar los mapas de Karnaugh

A. Identificación de Variables:

Primero se debe determinar cuántas variables lógicas tiene la función booleana. Dependiendo del número de variables, se construirá un mapa de Karnaugh que tendrá el mismo número de celdas que minterms o maxterms posibles.

B. Construcción del Mapa:

- Para funciones con dos variables, el mapa será una matriz 2×2 .
- Para tres variables, se utiliza un mapa 2×4 .
- Para cuatro variables, se usa un mapa 4×4 .

C. Etiquetado de las Casillas:

Las casillas del mapa representan todas las combinaciones posibles de los valores lógicos (0 y 1) de las variables. Estas combinaciones se distribuyen de manera que los adyacentes solo difieran en una variable.

D. Inserción de Valores:

Se coloca un '1' en las casillas correspondientes a las combinaciones de entradas que hacen que la salida de la función sea 1. Si el resultado es 0, se puede dejar en blanco o marcar con un cero.

E. Agrupación de Celdas:

A continuación se deben agrupar las casillas contiguas que contienen '1'. Las reglas para agrupar son:

- Las agrupaciones deben ser potencias de dos (1, 2, 4, 8, etc.).
- Las agrupaciones pueden envolver cualquier número de dimensiones del mapa.
- Los grupos pueden superponerse y cada '1' puede formar parte de más de un grupo.

F. Identificación de Patrones:

Al identificar los patrones o grupos, se reducen las variables lógicas en la expresión final. Por ejemplo, si dos celdas adyacentes contienen '1', entonces una variable es irrelevante para esa parte de la función.

G. Simplificación de la Expresión:

Para cada grupo identificado, se simplifica eliminando las variables que varían dentro del grupo. La expresión final será la suma (OR lógico) de las expresiones simplificadas de cada grupo.

La Figura 1 muestra un mapa de Karnaugh para dos variables (a,b) que ha sido llenado con los minterms de la función $f(a,b)$, cuya tabla de verdad se muestra en la Tabla 1, donde se puede observar a la izquierda de la diagonal la variable a y en vertical sus valores 0 y 1, mientras que a la derecha de la diagonal se tiene la variable b con sus valores 0 y 1 en horizontal. La proyección de los valores de las variables identifican la celda del mapa de Karnaugh, así en la esquina superior izquierda se proyectan el 0 de a y el 0 de b, por lo que esa celda es 00, mientras que en la esquina inferior derecha se proyectan los valores 1 de a y 1 de b, por lo que la celda es 11. Por lo general la

identificación de las celdas se las realiza en decimal. Nótese además que el subíndice de los min-terms corresponde al valor de la celda, lo que facilitará su llenado.

Tabla 1

Tabla de verdad de la función $f(a,b)$

a	b	$f(a,b)$
0	0	m_0
0	1	m_1
1	0	m_2
1	1	m_3

Nota: Elaborado por Escobar C. (2025)

Figura 1

Mapa de karnaugh para las variables a,b.

		b	
		0	1
a	0	00 m_0	01 m_1
	1	10 m_2	11 m_3

Nota: Elaborado por Escobar C. (2025)

La Figura 2 muestra un mapa de Karnaugh para tres variables (a,b,c) que ha sido llenado con los min-terms de la función $f(a,b,c)$, cuya tabla de verdad se muestra en la Tabla 2, donde se puede observar a la izquierda de la diagonal la variable a y en vertical sus valores 0 y 1, mientras que a la derecha de la diagonal se tiene las variables bc con sus valores 00, 01, 10 y 11 en horizontal. La proyección de los valores de las variables identifican la celda del mapa de Karnaugh, así en la esquina superior izquierda se proyectan el 0 de a, el 0 de b y el 0 de c por lo que esa celda es 000,

mientras que en la esquina inferior derecha se proyectan los valores 1 de a, 1 de b y 0 de c, por lo la celda es 110. Por lo general la identificación de las celdas se las realiza en decimal. Nótese además que el subíndice de los minterms corresponde al valor de la celda, lo que facilitará su llenado.

Tabla 2

Tabla de verdad de la función $f(a,b,c)$

a	b	c	$f(a,b,c)$
0	0	0	m_0
0	0	1	m_1
0	1	0	m_2
0	1	1	m_3
1	0	0	m_4
1	0	1	m_5
1	1	0	m_6
1	1	1	m_7

Nota: Elaborado por Escobar C. (2025)

Figura 2

Mapa de karnaugh para las variables a,b,c.

		<u>bc</u>			
		00	01	11	10
a	0	000 m_0	001 m_1	011 m_3	010 m_2
	1	100 m_4	101 m_5	111 m_7	110 m_6

Nota: Elaborado por Escobar C. (2025)

Tabla 3

Tabla de verdad de la función $f(a,b,c,d)$

a	b	c	d	$f(a,b,c,d)$
0	0	0	0	m_0
0	0	0	1	m_1
0	0	1	0	m_2
0	0	1	1	m_3
0	1	0	0	m_4
0	1	0	1	m_5
0	1	1	0	m_6
0	1	1	1	m_7
1	0	0	0	m_8
1	0	0	1	m_9
1	0	1	0	m_{10}
1	0	1	1	m_{11}
1	1	0	0	m_{12}
1	1	0	1	m_{13}
1	1	1	0	m_{14}
1	1	1	1	m_{15}

Nota: Elaborado por Escobar C. (2025)

La Figura 3 muestra un mapa de Karnaugh para cuatro variables (a,b,c,d) que ha sido llenado con los minterm de la función $f(a,b,c)$, cuya tabla de verdad se muestra en la Tabla 3, dónde se puede observar a la izquierda de la diagonal la variable ab y en vertical sus valores 00, 01, 11 y 10, mientras que a la derecha de la diagonal se tiene las variables cd con sus valores 00, 01, 11 y 10 en horizontal. La proyección de los valores de las variables identifican la celda del mapa de Karnaugh, así en la esquina superior izquierda se proyectan el 0 de a, el 0 de b, el 0 de c y el 0 de d por lo que esa celda es 0000, mientras que en la equina inferior derecha se proyectan los valores 1 de a, 0 de b, 1 de c y 0 de d, por lo la celda es 1010. Por lo general la identificación de las celdas se las realiza en decimal. Nótese además que el subíndice de los minterms corresponde al valor de la celda, lo que facilitará su llenado.

Figura 3

Mapa de karnaugh para las variables a,b,c,d

		cd			
		00	01	11	10
ab	00	0000 m_0	0001 m_1	0011 m_3	0010 m_2
	01	0100 m_4	0101 m_5	0111 m_7	0110 m_6
	11	1100 m_{12}	1101 m_{13}	1111 m_{15}	1110 m_{14}
	10	1000 m_8	1001 m_9	1011 m_{11}	1010 m_{10}

Nota: Elaborado por Escobar C. (2025)

[En este enlace puede ver un video explicativo del uso de mapas de Karnaugh aplicado a una función de 2 variables.](#)

3.7 Circuitos Lógicos con Mapas de Karnaugh

3.7.1 Sumador Binario

Un sumador es un circuito que realiza operaciones aritméticas en binario. Se divide en dos tipos:

- **Medio Sumador (Half Adder)**
 - Entradas: Operador1, Operador2

- Salidas: Acarreo de salida, Suma
- **Sumador Completo (Full Adder)**
 - Entradas: Operador1, Operador2, Acarreo de entrada
 - Salidas: Acarreo de salida, Suma

Para este curso implementaremos el sumador completo (Full Adder), que considera los acarreo de entrada y de salida y que tiene la ventaja que puede ser implementado en cascada para permitir la suma de números de varios bits.

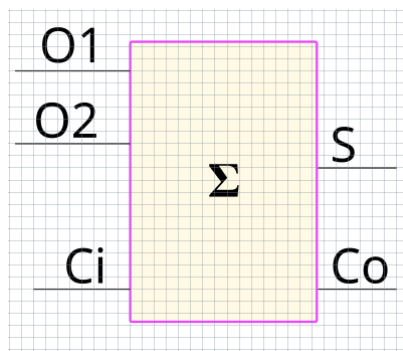
La **¡Error! No se encuentra el origen de la referencia.** muestra el diagrama de bloques de un sumador completo dónde se puede identificar las variables de entrada y salida, mientras que la

Tabla 4, dónde se resalta en gris las columnas de salida, muestra la tabla de verdad del sumador

- Entradas:
 - O1: Operador 1
 - O2: Operador 2
 - Ci: Acarreo de entrada
- Salidas:
 - Co: Acarreo de salida
 - S: Suma

Figura 4

Diagrama de bloques de un sumador completo.



Nota: Elaborado por Escobar C. (2025)

Tabla 4

Tabla de verdad del sumador completo.

	O ₁	O ₂	C _i	C _o	S
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	1	0
6	1	1	0	1	0
7	1	1	1	1	1

Nota: Elaborado por Escobar C. (2025)

Para el llenado de la tabla de ha considerado la operación suma entre las variables O₁, O₂ y C_i, donde la operación suma de estas tres variables determinan el valor de las variables C_o y S. Así por ejemplo para el valor 0, 0, 0 tenemos;

$$0 + 0 + 0$$

$$C_o = 0$$

$$S = 0$$

Mientras que para los valores 1, 1, 1 para O₁, O₂ y C_i, tenemos;

$$1 + 1 + 1$$

$$C_o = 1$$

$$S = 1$$

Tal como se muestra en la tabla de verdad.

Una vez que la tabla de verdad se ha llenado, se procede con la simplificación de cada una de las variables de salida, lo que a partir de ahora lo haremos usando los mapas de Karnaugh. Así;

A. Mapa de Karnaugh para la variable Co

$$C_o = \{m_3, m_5, m_6, m_7\}$$

Con los minterm m3, m5, m6, m7, tal como se muestra en la Figura 5, se procede al llenado del mapa de Karnaugh los cuales corresponden a la celdas cuyo contenido es uno. Las demás celdas tendrán cero.

Figura 5

Mapa de Karnaugh para la variable Co.

$O_1 \backslash O_2 C_i$					
		00	01	11	10
O_1	0	000 0	001 0	011 1	010 0
	1	100 0	101 1	111 1	110 1

Nota: Elaborado por Escobar C. (2025)

La Figura 5 muestra el mapa de Karnaugh llenado para la variable Co, dónde se puede notar que cumpliendo el criterio de que las agrupaciones que deben ser adyacentes y potencias de dos se puede obtener 3 grupos de 2 elementos, el primero entre las celdas 3 y 7, el segundo entre las celdas 5 y 7 y el tercero entre las celdas 7 y 6.

En el primer agrupamiento se simplifica la variable O_1 , ya que tiene los valores (0+1) y nos queda el término $O_2 C_i$

En el segundo agrupamiento se simplifica la variable O_2 y nos queda el término $O_1 C_i$

En el tercer agrupamiento se simplifica la variable C_i , por lo que nos queda el término $O_1 O_2$

Con lo que tenemos:

$$Co = O_2 C_i + O_1 C_i + O_1 O_2$$

Agrupando términos:

$$Co = O_1 (C_i + O_2) + O_2 C_i$$

Esto con la finalidad de facilitar la implementación.

B. Mapa de Karnaugh para la variable S.

$$S = \{m1, m2, m4, m7\}$$

Con los minterm $m1, m2, m4, m7$, tal como se muestra en la Figura 6, se procede al llenado del mapa de Karnaugh los cuales corresponden a la celdas cuyo contenido es uno. Las demás celdas tendrán cero.

Figura 6

Mapa de Karnaugh para la variable S.

		O ₂ C _i			
		00	01	11	10
O ₁	0	000 0	001 1	011 0	010 1
	1	100 1	101 0	111 1	110 0

Nota: Elaborado por Escobar C. (2025)

En este caso no es posible ningún agrupamiento, por lo que se leen directamente la proyección de las variables de cada una de las celdas que contienen 1.

$$S = O_1O_2'C_i' + O_1'O_2'C_i + O_1O_2C_i + O_1'O_2C_i'$$

Aquí podemos agrupar los términos comunes:

$$S = O_1(O_2'C_i' + O_2C_i) + O_1'(O_2'C_i + O_2C_i')$$

Aquí podemos recordar las tablas XOR (Tabla 5) y XNOR (Tabla 6) que se vieron en la clase anterior, donde se puede identificar que nuestra salida de S dentro de los paréntesis tiene justo esas combinaciones pero con las variables O_2C_i .

- $(O_2'C_i' + O_2C_i) = O_2 \odot C_i$
- $(O_2'C_i + O_2C_i') = O_2 \oplus C_i$

Tabla 5

Tabla de verdad del operador XOR.

A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

$$A \oplus B = A'B + AB'$$

Nota: Elaborado por Escobar C. (2025)

Tabla 6

Tabla de verdad del operador XNOR.

A	B	$A \odot B$
0	0	1
0	1	0
1	0	0
1	1	1

$$A \odot B = A'B' + AB$$

Nota: Elaborado por Escobar C. (2025)

Por lo que nuestra variables S será:

$$S = O_1(O_2 \odot C_i) + O_1'(O_2 \oplus C_i)$$

Y dado que $\odot$ es la negación de $\oplus$, la función S puede ser escrito como:

$$S = O_1(O_2 \oplus C_i)' + O_1'(O_2 \oplus C_i)$$

Dónde se puede observar una nueva compuerta XOR, con lo que finalmente nuestra función S queda de la siguiente manera:

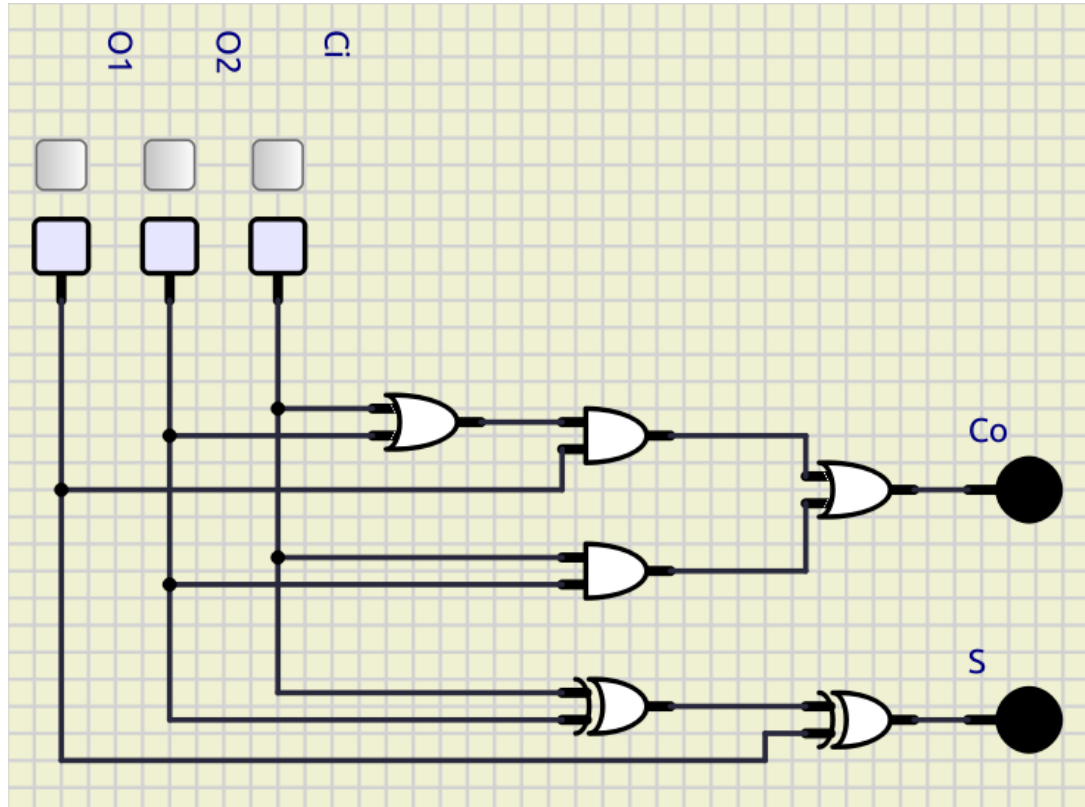
$$S = O_1 \oplus (O_2 \oplus C_i)$$

La Figura 7 muestra la implementación en SimulIDE (González et al., 2025) de la simulación del circuito SumadorCompleto.simul1 en el que podrá probar todas las combinaciones posibles de entrada y validar las salidas de acuerdo a la

Tabla 4.

Figura 7

Simulación SimulIDE del circuito sumador completo.



Nota: Elaborado por Escobar C. (2025)

[En este enlace puede ver como un sumador completo para dos números de un bit se puede convertir en un sumador dos números de n-bits al conectar varias veces en cascada este circuito.](#)

3.7.2. Detector números primos

El siguiente diseño considera tres entradas a, b, c dónde la salida se activa si la combinación de entrada corresponde a un número primo, considerándose como tal a un número natural positivo mayor que 1 que es divisible solo para él mismo y el 1, la

Tabla 7 muestra la tabla de verdad del detector de números primos donde se ha resaltado la salida P.

Tabla 7

Tabla de verdad del detector de números primos.

	a	b	c	P
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

Nota: Elaborado por Escobar C. (2025)

Aquí podemos extraer la función P expresada en función de los minterm:

$$P = \{m_2, m_3, m_5, m_7\}$$

Con estos minterm procedemos a llenar el mapa de Karnaugh, como se muestra en la Figura 8 dónde se muestran las agrupaciones posibles. Nótese que la celda 011 y la 111 contienen unos que ya han sido agrupados, por tanto no es necesario realizar un grupo entre ellos.

Figura 8

Mapa de Karnaugh de la salida P.

		bc			
a		00	01	11	10
		000	001	011	010
0		0	0	1	1
1		0	1	1	0

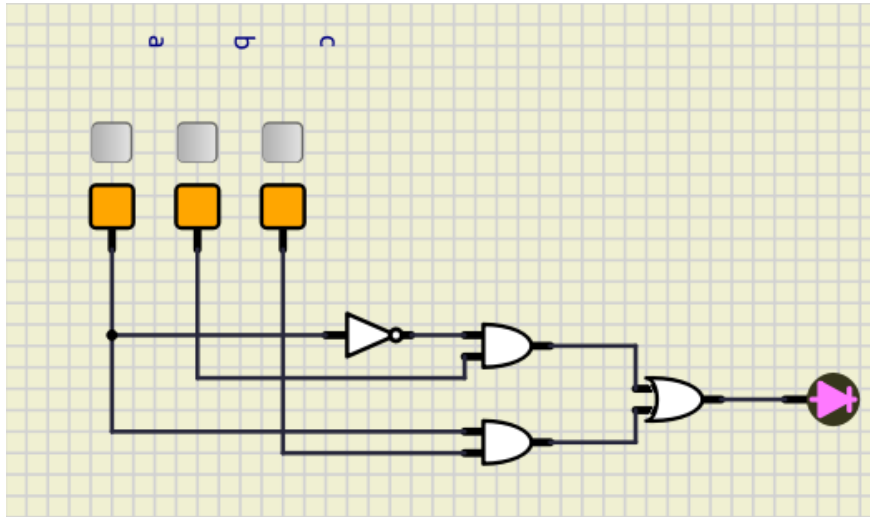
Nota: Elaborado por Escobar C. (2025)

La función de salida P extraída del mapa:

$$P = a'b + ac$$

Figura 9

Simulación SimulIDE del circuito detector de números primos.



Nota: Elaborado por Escobar C. (2025)

La Figura 9 muestra la implementación de la simulación del circuito DetectorPrimos.simul1 en el que puede probar todas las combinaciones posibles de entrada y validar las salidas de acuerdo a la

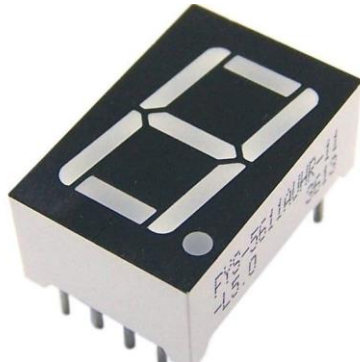
Tabla 7.

A. Display de 7 segmentos.

Un display de 7 segmentos es un componente electrónico usado para mostrar números decimales (0-9) y algunos caracteres (como A, b, C, E, F, etc.). Consiste en siete barras LED (segmentos) dispuestas en forma de "8", donde cada segmento se enciende o apaga para formar los caracteres deseados. La Figura 10 muestra una versión comercial de este dispositivo, dónde se puede apreciar los 7 segmentos y el punto, mientras que la Figura 11 describe el nombre y ubicación de cada uno de los segmentos y cuales se encenderían para expresar los números del 0 al 9.

Figura 10

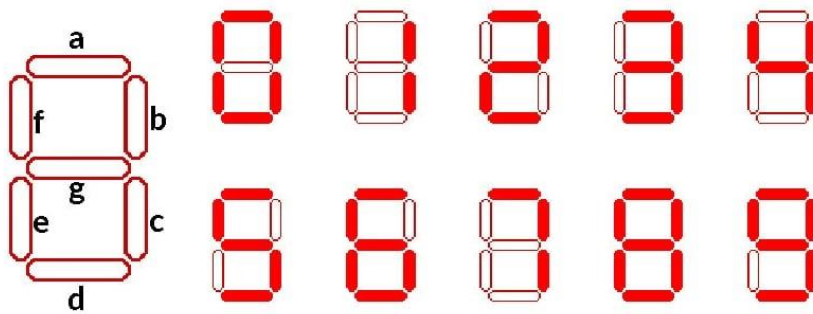
Display de 7 segmentos



Nota: Recuperado de (Isaac et al., 2025)

Figura 11

Disposición de los segmentos a, b, c, d, e, f y g en el display de 7 segmentos



Nota: Recuperado de (Isaac et al., 2025)

El display de 7 segmentos también puede ser usado para representar letras, según como se enciendan los segmentos, así por ejemplo podríamos expresar “HoLA”



Referencias citadas en la Clase 10.

Alegre, S. E. A., & García Sevilla, F. (2010). Electrónica digital: lógica digital integrada : teoría, problemas y simulación (2a. ed.). Ra-Ma.

González, S., De Francisco, P., Padrah, Z., Vrana, J., Kellner, K., Salishhchev, A., & Otros. (2025). Simulador de circuitos SimulIDE. Simulador de Circuitos SimulIDE.
<https://github.com/KDE/ktechlab>

Isaac, P. E., Arboledas, J., García, J., & De Frutos, T. (2025). Hardware Libre. Hardware Libre.
<https://www.hwlibre.com/>

Definición de los términos citados en la Clase 10.

Mapas de Karnaugh: (también conocidos como mapas K) son una herramienta útil en el diseño simplificado de circuitos lógicos digitales. Proporcionan una forma visual para minimizar expresiones booleanas, facilitando la implementación de funciones lógicas mediante compuertas electrónicas.

Código Gray: (o código reflejado binario) es un sistema de numeración binario en el que dos números consecutivos difieren en un solo bit (dígito binario)



La excelencia no se improvisa

síguenos

