

B.3

Números
flotantes

En el sistema numérico decimal, los números grandes como 120 0000 a menudo se escriben en **notación científica** como $1.2 * 10^5$ o tal vez $120 * 10^3$ y los números pequeños como 0.000120 como $1.2 * 10^{-4}$ en lugar de un número con una ubicación fija para el punto decimal. Los números en esta forma de notación se escriben en términos de 10 elevados a alguna potencia. Asimismo, se puede usar esta función para números binarios pero escritos en términos de 2 elevados a alguna potencia. Por ejemplo, se podría tener 1010 escrito como $1.010 * 2^3$ o tal vez $10.10 * 2^2$. Debido a que el punto binario se puede mover a diferentes ubicaciones mediante una elección de la potencia a la cual el 2 esté elevado, esta notación se denomina como **punto flotante**.

Un número de punto flotante está en la forma $a * r^e$, donde a es denominado **mantisa**, r el **radio** o **base** y e el **exponente** o **potencia**. Con números binarios la base se entendió como 2, es decir se tiene $a * 2^e$. La ventaja del uso de números con puntos flotante es que, comparados con la representación del punto fijo, un rango mucho más amplio de números se puede representar por un número dado de dígitos.

Puesto que con números de punto flotante es posible almacenar un número en una cantidad de formas diferentes, por ejemplo $0.1 * 10^2$ y $0.01 * 10^3$, con sistemas de computación esos números están **normalizados**; es decir, todos ellos están puestos en la forma de $0.1 * r^e$. Por lo tanto, con números binarios se tiene $0.1 * 2^e$ y así, si se tuvo 0.00001001, éste se podría convertir en $0.1001 * 2^{-4}$. Para tomar en cuenta el signo de un número binario entonces se agrega un bit con signo de 0 para un número positivo y 1 para un número negativo. De esta forma, el número $0.1001 * 2^{-4}$ se convierte en $1.1001 * 2^{-4}$ si es negativo y $0.1001 * 2^{-4}$ si es positivo.

Si se quiere añadir $2.01 * 10^3$ y $10.2 * 10^2$ se tiene que crear la potencia (por lo general se usa el término exponente), la misma para cada uno. Así, se puede escribir $2.01 * 10^3 + 1.02 * 10^3$. Entonces se puede añadir dígito por dígito, tomando en cuenta a cualquier portador, para dar $2.03 * 10^3$. Se adopta un procedimiento similar para números binarios de punto flotante. De esta forma, si se desea agregar $0.101100 * 2^4$ y $0.111100 * 2^2$ primero se ajustan para tener los mismos exponentes, es decir $0.101100 * 2^4$ y $0.001111 * 2^4$, y luego se suman dígito por dígito para dar $0.111011 * 2^4$.

Asimismo, para la resta dígito por dígito de números de punto flotante sólo puede ocurrir entre dos números cuando tienen el mismo exponente. Así, $0.1101100 * 2^{-4}$ menos $0.1010100 * 2^{-5}$ se puede escribir como $0.01010100 * 2^{-4} - 0.101010 * 2^{-4}$ y el resultado sería $0.1000010 * 2^{-4}$.

B.4

Código Gray

Considere dos números sucesivos en un código binario 0001 y 0010 (denario 2 y 3); los 2 bits han cambiado en el grupo del código en ir desde un número al siguiente. Así, si se tenía un decodificador absoluto (sección 2.3.7) y posiciones sucesivas asignadas para números binarios sucesivos, entonces se tienen que realizar dos cambios en este caso, lo que puede significar problemas en esos dos cambios, deben hacerse exactamente en el mismo instante; si uno ocurre en una fracción de tiempo antes que el otro, entonces puede haber momentáneamente otro número indicado. Por lo tanto, al ir desde 0001 a 0010 se podría tener de manera momentánea 0011 o 0000. Así es probable que se use un método alternativo de codificar.

El **código Gray** es un código: sólo 1 bit en el grupo de código que va de un número al siguiente. El código Gray no tiene un valor específico de acuerdo

Tabla B.3 Código Gray.

Número decimal	Código binario	Código Gray	Número decimal	Código binario	Código Gray
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

con su posición de bit en el grupo de código. Por lo tanto, no es apropiado para operaciones aritméticas, pero se usa bastante en dispositivos de entrada/salida tal y como los decodificadores absolutos. En la Tabla B.3 se hace una lista de los números decimales y sus valores en el código binario y en el código Gray.

Problemas

- B.1 ¿Cuál es el número decimal más grande que se puede representar al usarse un número binario de 8 bits?
- B.2 Convierta los siguientes números binarios a números decimales: a) 1011, b) 10 0001 0001.
- B.3 Convierta los siguientes números decimales a hex: a) 423, b) 529.
- B.4 Convierta los siguientes números BCD a decimales: a) 0111 1000 0001, b) 0001 0101 0111.
- B.5 ¿Cuáles son las representaciones de los complementos a dos de los siguientes números decimales: a) -90, b) -35?
- B.6 ¿Qué bits de paridad par deben agregarse a: a) 100 1000, b) 100 1111?
- B.7 Realice las siguientes restas utilizando el complemento a dos: a) 21 - 13, b) 15 - 3.



Apéndice C: Álgebra booleana

C.1 Leyes del álgebra booleana

El álgebra booleana implica los dígitos binarios 1 y 0 y las operaciones \dagger , + y a la inversa. Las leyes de esta álgebra son:

- 1 Cualquier elemento de la operación OR consigo mismo es igual a sí mismo (idempotencia): $A + A = A$.
- 2 Cualquier elemento de la operación AND consigo mismo es igual a sí mismo (idempotencia): $A \dagger A = A$.
- 3 No importa el orden en el que se consideren las entradas para las compuertas OR y AND (conmutatividad):

$$A + B = B + A \quad \text{y} \quad A \dagger B = B \dagger A$$

- 4 La siguiente tabla de verdad indica la ley de asociatividad:

$$A + (B \dagger C) = (A + B) \dagger (A + C)$$

A	B	C	$B \dagger C$	$A + B \dagger C$	$A + B$	$A + C$	$(A + B) \dagger (A + C)$
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

- 5 De la misma manera se puede utilizar una tabla de verdad para mostrar que se pueden manejar términos entre paréntesis de la misma manera que en el álgebra ordinaria:

$$A \dagger (B + C) = A \dagger B + A \dagger C$$

- 6 Cualquier elemento de la operación OR con su propio inverso es igual a 1:

$$A + \overline{A} = 1$$

- 7 Cualquier elemento de la operación AND con su propio inverso es igual a 0:

$$A \dagger \overline{A} = 0$$

- 8 Cualquier elemento de la operación OR con un 0 es igual a sí mismo; cualquier elemento de la operación OR con un 1 es igual a 1. Así, $A + 0 = A$ y $A + 1 = 1$.

- 9 Cualquier elemento de la operación AND con un 0 es igual a 0; cualquier elemento de la operación AND con un 1 es igual a sí mismo. Así, $A \dagger 0 = 0$ y $A \dagger 1 = A$.

Como una ilustración del uso de lo anterior para simplificar las expresiones booleanas, considere la simplificación

$$(A + B) \uparrow C + A \uparrow C$$

Al utilizar la ley 5 para el primer término da

$$A \uparrow C + B \uparrow C + A \uparrow C$$

Se reagrupa esto y se utiliza la ley 6 para dar

$$A \uparrow (C + C) + B \uparrow C = A \uparrow 1 + B \uparrow C$$

Por lo tanto, al utilizar la ley 9 la expresión simplificada se convierte en

$$A + B \uparrow C$$

C.2

Leyes de De Morgan

Como se mostró antes, las leyes del álgebra booleana se pueden emplear para simplificar expresiones booleanas. Además, se tiene lo que se conoce como las **leyes de De Morgan**:

- 1 La suma de A y B globalmente negados o invertidos es igual al producto de los elementos A y B negados o invertidos de forma individual. La siguiente tabla de verdad muestra la validez de esto:

$$A + B = A \uparrow B$$

A	B	A + B	$\overline{A + B}$	\overline{A}	\overline{B}	$\overline{A} \uparrow \overline{B}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

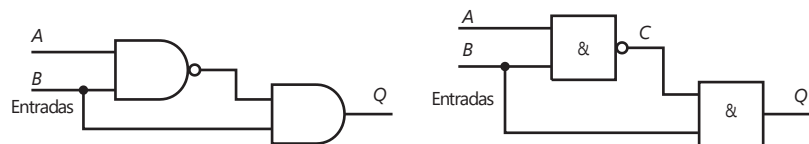
- 2 El producto de los elementos A y B globalmente negados o invertidos es igual a la suma de A y B negados o invertidos en forma individual. La siguiente tabla de verdad muestra la validez de esto

$$\overline{A \uparrow B} = \overline{A} + \overline{B}$$

A	B	$A \uparrow B$	$\overline{A \uparrow B}$	\overline{A}	\overline{B}	$\overline{A} + \overline{B}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

Como ejemplo del uso de las leyes de De Morgan, considere la simplificación del circuito lógico que se muestra en la Figura C.1.

Figura C.1 Simplificación de un circuito.



La ecuación booleana para la salida en términos de la entrada es

$$Q = \overline{A+B} B$$

Al aplicar la segunda ley de De Morgan da

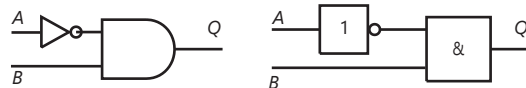
$$Q = (\overline{A} + \overline{B}) B$$

Esto se puede escribir como

$$Q = \overline{A} B + \overline{B} B = \overline{A} B + 0 = \overline{A} B$$

Por lo tanto, el circuito simplificado es como se muestra en la Figura C.2.

Figura C.2 Simplificación del circuito de la figura C.1.



C.3 Generación de la función booleana a partir de las tablas de verdad

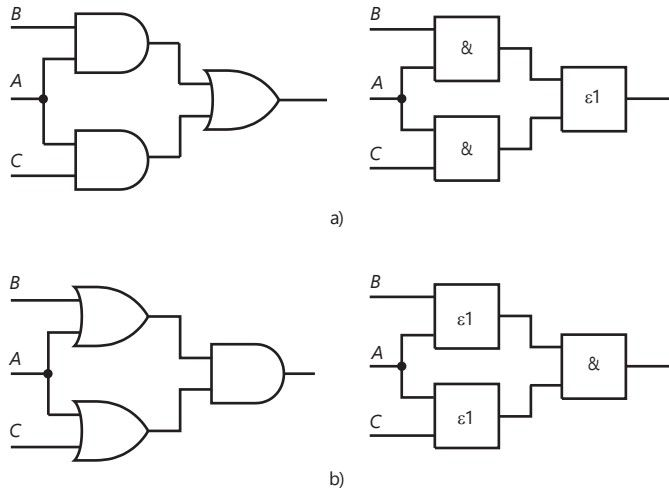
Dada una situación donde los requerimientos de un sistema se pueden especificar en términos de una tabla de verdad, ¿cómo se crea un sistema de compuerta lógica que utilice la mínima cantidad de compuertas para dar esa tabla de verdad?

El álgebra booleana se puede utilizar para manipular funciones de interruptores en muchas formas equivalentes, algunas de las cuales toman muchas más compuertas lógicas que otras; sin embargo, la forma en la que se minimizan la mayoría es con compuertas AND que dirigen una compuerta OR sencilla o viceversa. Dos compuertas AND que dirigen una compuerta OR sencilla (Figura C.3a) dan

$$A B + A C$$

A esto se le denomina como la forma de **suma de productos**.

Figura C.3 a) Suma de productos, b) producto de sumas.



Para dos compuertas OR que dirigen una compuerta AND sencilla (Figura C.3b) se tiene

$$(A + B)(A + C)$$

A esto se le conoce como la forma del **producto de sumas**. Así, al considerar qué forma mínima puede encajar en una tabla de verdad dada, el procedi-

miento usual es encontrar la suma de productos o los productos de sumas que encajan con los datos. Por lo general se usa la forma de la suma de productos. El procedimiento que se utiliza es considerar cada fila de la tabla de verdad en turno y encontrar el producto que encajaría en la fila. El resultado general es entonces la suma de todos esos productos.

Suponga que se tiene una fila en una tabla de verdad de

$$A = 1, B = 0 \text{ y salida } Q = 1$$

Cuando A es 1 y B no es 1, entonces la salida es 1, por lo que el producto que encaja es éste

$$Q = A \uparrow B$$

Se puede repetir esta operación para cada fila de una tabla de verdad como indica la siguiente tabla.

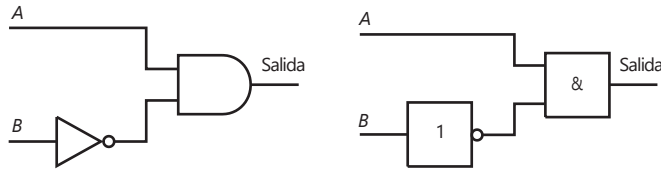
A	B	Salida	Productos
0	0	0	--
0	1	0	--
1	0	1	$A \uparrow B$

Sin embargo, sólo la fila de la tabla de verdad que tiene una salida de 1 necesita ser considerada, ya que las filas con salida 0 no contribuyen a la expresión final; el resultado entonces es

$$Q = A \uparrow B$$

El sistema de compuerta lógica que dará esta tabla de verdad es como se muestra en la Figura C.4.

Figura C.4 Compuertas lógicas para la tabla de verdad.



Otro ejemplo más, considere la siguiente tabla de verdad en la que sólo se incluyen los términos de los productos que dan una salida 1:

A	B	C	Salida	Productos
0	0	0	1	$\bar{A} \uparrow \bar{B} \uparrow \bar{C}$
0	0	1	0	
0	1	0	1	$\bar{A} \uparrow B \uparrow \bar{C}$
0	1	1	0	
1	0	0	0	
1	0	1	0	
1	1	0	0	
1	1	1	0	

Así, la suma de productos que encaja con esta tabla es

$$Q = \bar{A} \bar{B} \bar{C} + \bar{A} \bar{B} C$$

Esto se puede simplificar para dar

$$Q = \bar{A} \bar{C} (\bar{B} + B) = \bar{A} \bar{C}$$

De esta manera, la tabla de verdad se puede generar con sólo una compuerta NAND.

C.4

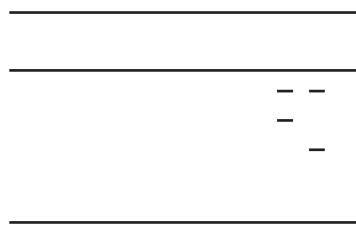
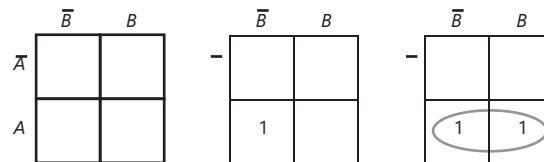
Mapas de Karnaugh

El **mapa de Karnaugh** es un método gráfico que se puede utilizar para producir expresiones booleanas simplificadas de sumas de productos obtenidos de las tablas de verdad. La tabla de verdad tiene una fila para el valor de la salida para cada combinación de valores de entrada. Con dos variables de entrada hay cuatro líneas en la tabla de verdad, con tres variables de entrada hay seis líneas y con cuatro variables de entrada hay dieciséis líneas. Así, con dos variables de entrada hay cuatro términos de producto, con tres variables de entrada hay seis y con cuatro variables de entrada hay dieciséis. El mapa de Karnaugh está trazado como un arreglo rectangular de celdas; cada celda corresponde al valor del producto en particular. De esta manera, con dos variables de entrada hay cuatro celdas, con tres variables de entrada hay seis celdas y con cuatro variables de entrada hay dieciséis celdas. Los valores de salida para las filas están colocados en sus celdas en el mapa de Karnaugh, a pesar de que es usual indicar sólo el 1 de los valores de salida y dejar las celdas con salida 0 como vacías.

La Figura C.5a) muestra el mapa para dos variables de entrada. Las celdas se dan con los valores de salida para los productos siguientes:

- Celda izquierda superior $\bar{A} \bar{B}$,
- Celda izquierda inferior $\bar{A} B$,
- Celda derecha superior $A \bar{B}$,
- Celda derecha inferior $A B$

Figura C.5 Mapa de dos variables de entrada.



a)

b)

c)

El arreglo de los cuadros del mapa es tal que los cuadros adyacentes horizontalmente difieren sólo en una variable y, asimismo, los cuadros adyacentes en forma vertical difieren sólo en una variable. De esta manera, de forma horizontal con el mapa de dos variables; las variables difieren sólo en A y verticalmente sólo en B .

Para la siguiente tabla de verdad, si se colocan los valores dados para los productos en el mapa de Karnaugh, sólo indicando donde una celda tiene valor de 1 y dejando en blanco aquellas celdas con un valor 0, entonces se obtiene el mapa que se muestra en la Figura C.5b):

A	B	Salida	Productos
0	0	0	$A\bar{B}$
0	1	0	$A\bar{B}$
1	0	1	

A

A

A

A

Debido a que sólo la entrada 1 se encuentra en el cuadro derecho inferior, la tabla de verdad se puede representar por la expresión booleana

$$\text{salida} = A \uparrow B$$

Como un ejemplo extra, considere la siguiente tabla de verdad:

A	B	Salida	Productos
0	0	0	--
0	1	0	--
1	0	1	--

Ésta da el mapa de Karnaugh que se muestra en la Figura C.5c), que tiene una salida dada por

$$\text{salida} = A \uparrow B + A \uparrow B$$

Esto se puede simplificar:

$$A \uparrow B + A \uparrow B = A \uparrow (B + B) = A$$

Cuando dos celdas que contienen un 1 con un borde vertical en común, se puede simplificar la expresión booleana para ajustar la variable común. Se puede hacer esto a través de la inspección de un mapa, que indique cuáles entradas de celdas se pueden simplificar mediante aros alrededor de ellos, como en la Figura C.5c).

La Figura C.6a) muestra el mapa de Karnaugh para la siguiente tabla de verdad que cuenta con tres variables de entrada:

A	B	C	Salida	Productos
0	0	0	1	$\bar{A} \bar{B} \bar{C}$
0	0	1	0	$\bar{A} \bar{B} C$
0	1	0	1	$\bar{A} B \bar{C}$
0	1	1	0	$\bar{A} B C$
1	0	0	0	$A \bar{B} \bar{C}$
1	0	1	0	$A \bar{B} C$
1	1	0	0	$AB \bar{C}$
1	1	1	0	ABC

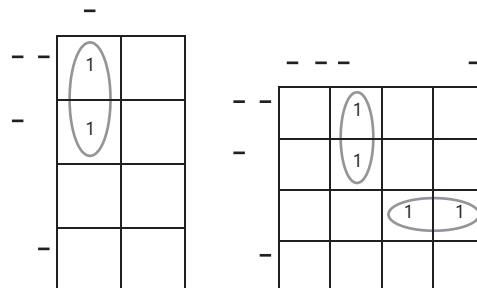


Figura C.6 a) Mapa de tres variables de entrada, b) mapa de cuatro variables de entrada.

a)

b)

Alfaomega

MECATRÓNICA. SISTEMAS DE CONTROL ELECTRÓNICO EN LA INGENIERÍA MECÁNICA Y ELÉCTRICA/BOLTON

C C

A • B

C • D C • D C • D C • D

A • B

A • B

A • B

A • B

A • B

A • B

Como se hizo antes, se puede utilizar un aro para simplificar el resultado de la expresión booleana para sólo la variable común. El resultado es

$$\text{salida} = \overline{A} \overline{C}$$

La Figura C.6b) muestra el mapa de Karnaugh para la siguiente tabla de verdad que tiene cuatro variables de entrada. El aro simplifica la expresión booleana resultante para dar

$$\text{salida} = \overline{A} \overline{C} D + A \overline{B} C$$

A	B	C	D	Salida	Productos
0	0	0	0	0	
0	0	0	1	1	$A \overline{B} \overline{C} D$
0	0	1	0	0	
0	0	1	1	0	
0	1	0	0	0	
0	1	0	1	1	$A \overline{B} C \overline{D}$
0	1	1	0	0	
0	1	1	1	0	
1	0	0	0	0	
1	0	0	1	0	
1	0	1	0	0	
1	0	1	1	0	
1	1	0	0	0	
1	1	0	1	0	
1	1	1	0	1	—

Lo anterior representa sólo algunos ejemplos simples de mapas de Karnaugh y el uso del aro. Observe que, al hacer el aro, las celdas adyacentes se pueden considerar para estar en las filas superior e inferior y en las columnas derecha e izquierda. Piense en los extremos opuestos del mapa juntos. Al encerrar un par de unos adyacentes en un mapa se elimina la variable de que aparezca en forma complementada o sin complementar. Al encerrar en un aro un cuarteto de unos adyacentes se eliminan las dos variables que aparecen en las formas complementada y sin complementar. Al encerrar en un aro un octeto de unos adyacentes se eliminan las tres variables que aparecen en las formas complementada y sin complementar.

Como otro ejemplo, considere una máquina automatizada que sólo empezará cuando dos de los tres sensores *A*, *B* y *C* den señales. La siguiente tabla de verdad encaja en este requerimiento y la Figura C.7a) muestra el resultado de las tres variables en el diagrama de Karnaugh. La expresión booleana que encaja en el mapa y de esta manera describe el resultado de la máquina es

$$\text{resultado} = A \overline{B} + B \overline{C} + A \overline{C}$$

La Figura C.7b) muestra las compuertas lógicas que podrían utilizarse para generar esta expresión booleana. $A \overline{B}$ describe una compuerta AND para las entradas *A* y *B*. Asimismo, $B \overline{C}$ y $A \overline{C}$ son dos compuertas AND más. Los signos + indican que las salidas de las tres compuertas AND son las entradas para una compuerta OR.

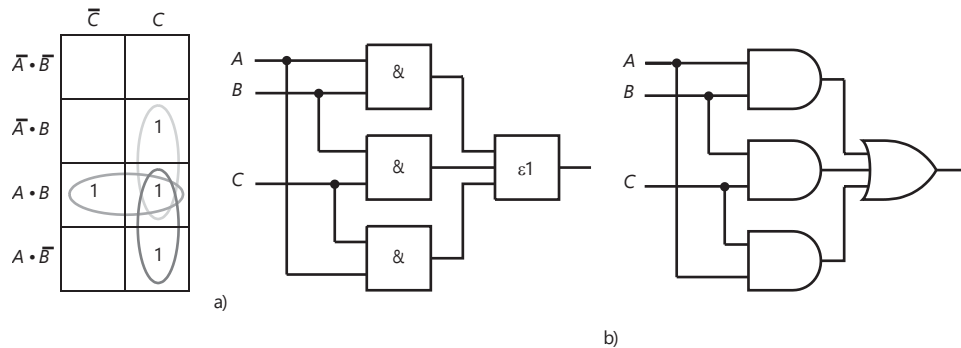


Figura C.7 Máquina automatizada.

A	B	C	Salida	Productos
0	0	0	0	
0	0	1	0	
0	1	0	0	$\bar{A} \bar{B} C$
0	1	1	1	$\bar{A} B C$
1	0	0	0	
1	0	1	1	$A \bar{B} C$
1	1	0	1	$A B \bar{C}$

En algunos sistemas lógicos hay algunas combinaciones variables de entrada para las cuales las salidas no están especificadas. Se les denomina como "estados sin importancia". Al entrar éstos en un mapa de Karnaugh, las celdas se pueden establecer ya sea en 1 o en 0, de tal forma que las ecuaciones de salida se pueden simplificar.

Problemas

- C.1 Establezca las funciones booleanas que se pueden emplear para describir las siguientes situaciones:
- Hay una salida cuando el interruptor A está cerrado y cualquiera de los interruptores B o C está cerrado.
 - Hay una salida cuando cualquiera de los interruptores A o B está cerrado y cualquiera de los interruptores C o D está cerrado.
 - Hay una salida si el interruptor A está abierto o el interruptor B está cerrado.
 - Hay una salida cuando el interruptor A está abierto y el interruptor B está cerrado.
- C.2 Establezca las funciones booleanas para cada uno de los circuitos lógicos que se muestran en la Figura C.8.
- C.3 Para la ecuación booleana $Q = (A \dagger C + B \dagger C) \dagger (A + C)$ construya una tabla de verdad.
- C.4 Simplifique las siguientes ecuaciones booleanas:
- $Q = A \dagger \bar{C} + A \dagger C \dagger D + \bar{C} \dagger D$
 - $Q = A \dagger B \dagger D + A \dagger \bar{B} \dagger D$
 - $Q = A \dagger B \dagger C + C \dagger D + C \dagger D \dagger E$

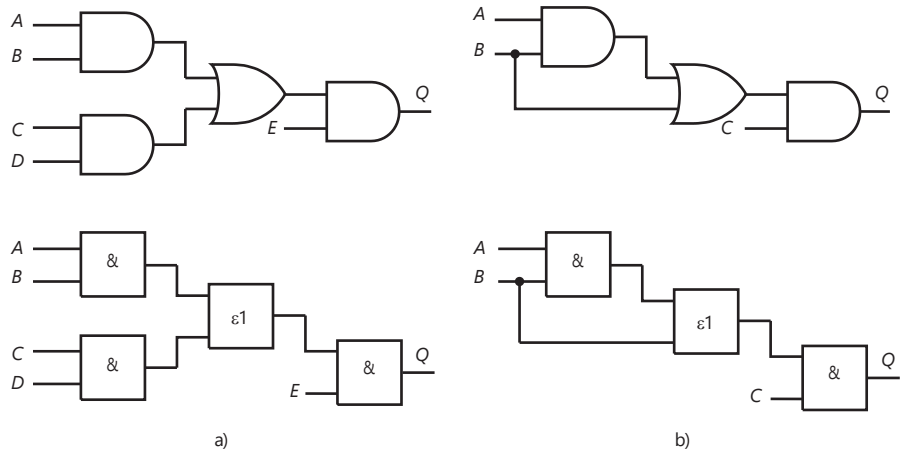


Figura C.8 Problema C.2.

- C.5 Utilice las leyes de De Morgan para demostrar que una compuerta NOR con entradas invertidas es equivalente a una compuerta AND.
- C.6 Dibuje los mapas de Karnaugh para las siguientes tablas de verdad y luego determine la ecuación booleana simplificada para las salidas:

a)

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	1

b)

A	B	C	Q
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

- C.7 Simplifique las siguientes ecuaciones booleanas con la ayuda de los mapas de Karnaugh:

a) $Q = \bar{A} \bar{B} C + \bar{A} \bar{B} \bar{C} + A \bar{B} \bar{C}$

b) $Q = \bar{A} \bar{B} \bar{C} D + A \bar{B} \bar{C} D + A \bar{B} C \bar{D} + A \bar{B} C D + A \bar{B} \bar{C} \bar{D} + A \bar{B} \bar{C} D + A \bar{B} C \bar{D} + A \bar{B} C D$

- C.8 Invente un sistema que permitirá que una puerta se abra sólo cuando se presionen correctamente cuatro botones combinados, con cualquier combinación incorrecta sonará una alarma.



Apéndice D: Conjuntos de instrucciones

A continuación se presentan las instrucciones que se usan con el M68HC11 de Motorola, con el 8051 de Intel y con los microcontroladores PIC16Cxx.

M68HC11

Instrucción	Mnemónico	Instrucción	Mnemónico
<i>Carga</i>		<i>Rotar/acarrear</i>	
Cargar un dato en el acumulador A	LDAA	Rotar a la izquierda los bits de la memoria	ROL
Cargar un dato en el acumulador B	LDAB	Rotar a la izquierda los bits del acumulador A	ROLA
Cargar un dato en el acumulador doble	LDD	Rotar a la izquierda los bits del acumulador B	ROLB
Cargar un dato en el apuntador de pila	LDS	Rotar a la derecha los bits de la memoria	ROR
Cargar un dato en el registro de índice X	LDX	Rotar a la derecha los bits del acumulador A	RORA
Cargar un dato en el registro de índice Y	LDY	Rotar a la derecha los bits del acumulador B	RORB
Leer el dato de la pila y cargarlo al acumulador A	PULA	Corrimiento aritmético a la izquierda de bits de la memoria	ASL
Leer el dato de la pila y cargarlo al acumulador B	PULB	Corrimiento aritmético a la izquierda de bits del acumulador A	ASLA
Leer el registro de índice X de la pila	PULX	Corrimiento aritmético a la izquierda de bits del acumulador B	ASLB
Leer el registro de índice Y de la pila	PULY	Corrimiento aritmético a la derecha de bits de la memoria	ASR
<i>Registros de transferencia</i>		Corrimiento aritmético a la derecha de bits del acumulador A	ASRA
Transferir un dato del acumulador A al acumulador B	TAB	Corrimiento aritmético a la derecha de bits del acumulador B	ASRB
Transferir un dato del acumulador B al acumulador A	TBA	Corrimiento lógico a la izquierda de bits de la memoria	LSL
Transferir un dato del apuntador de pila al registro de índice X	TSX	Corrimiento lógico a la izquierda de bits del acumulador A	LSLA
Transferir un dato del apuntador de pila al registro de índice Y	TSY	Corrimiento lógico a la izquierda de bits del acumulador B	LSLB
Transferir un dato del registro de índice X al apuntador de pila	TXS	Corrimiento lógico a la izquierda de bits del acumulador D	LSLD
Transferir un dato del registro de índice Y al apuntador de pila	TYS	Corrimiento lógico a la derecha de bits de la memoria	LSR
Intercambiar datos entre el acumulador doble y el registro de índice X	XGDX	Corrimiento lógico a la derecha de bits del acumulador A	LSRA
Intercambiar datos entre el acumulador doble y el registro de índice Y	XGDY	Corrimiento lógico a la derecha de bits del acumulador B	LSRB
<i>Decremento/Incremento</i>		Corrimiento lógico a la derecha de bits del acumulador C	LSRD
Restar 1 al contenido de la memoria	DEC	<i>Prueba de datos con ajuste de códigos de condiciones</i>	
Restar 1 al contenido del acumulador A	DECA	Prueba lógica AND entre el acumulador	BITA
Restar 1 al contenido del acumulador B	DECB	A y la memoria	
Restar 1 al apuntador de pila	DES	Prueba lógica AND entre el acumulador B	
Restar 1 al registro de índice X	DEX	y la memoria	BITB
Restar 1 al registro de índice Y	DEY	Comparar el acumulador A con el acumulador B	CBA
Sumar 1 al contenido de la memoria	INC	Comparar el acumulador A con la memoria	CMPA
Sumar 1 al contenido del acumulador A	INCA	Comparar el acumulador B con la memoria	CMPB
Sumar 1 al contenido del acumulador B	INCB	Comparar el acumulador doble con la memoria	CPD
Sumar 1 al apuntador de pila	INS	Comparar el registro de índice X con la memoria	CPX
Sumar 1 al registro de índice X	INX	Comparar el registro de índice Y con la memoria	CPY
Sumar 1 al registro de índice Y	INY	Restar \$00 de la memoria	TST

(Continúa)

Instrucción	Mnemónico	Instrucción	Mnemónico
Restar \$00 del acumulador A	TSTA	Restar la memoria del acumulador B con acarreo	SBCB
Restar \$00 del acumulador B	TSTB	Restar el contenido de la memoria del acumulador A	SUBA
<i>Interrupción</i>		Restar el contenido de la memoria del acumulador B	SUBB
Limpiar máscara de interrupción	CLI	Restar el contenido de la memoria del acumulador doble	SUBD
Establecer una máscara de interrupción	SEI	Reemplazar el acumulador A por su complemento a 2s	NEGA
Interrupción de software	SWI	Reemplazar el acumulador B por su complemento a 2s	NEGB
Regresar de la interrupción	RTI	Multiplicar el acumulador A por el acumulador B	MUL
Esperar interrupción	WAI	Dividir un entero D sin signo entre el registro de índice X	IDIV
<i>Complementar y limpiar</i>		Dividir una fracción D sin signo entre el registro de índice X	FDIV
Limpiar memoria	CLR	<i>Bifurcación condicional</i>	
Limpiar el acumulador A	CLRA	Bifurcación si el signo es negativo	BMI
Limpiar el acumulador B	CLRB	Bifurcación si el signo es positivo	BPL
Borrar los bits de la memoria	BCLR	Bifurcación si está definido un desborde	BVS
Fijar los bits en la memoria	BSET	Bifurcación si está limpio el desborde	BVC
<i>Registros de almacenamiento</i>		Bifurcación si es menor que cero	BLT
Almacenar el contenido del acumulador A	STAA	Bifurcación si es mayor que o igual a cero	BGE
Almacenar el contenido del acumulador B	STAB	Bifurcación si es menor que o igual a cero	BLE
Almacenar el contenido del acumulador doble	STD	Bifurcación si es mayor que cero	BGT
Almacenar el apuntador de pila	STS	Bifurcación si es igual	BEQ
Almacenar el registro de índice X	STX	Bifurcación si no es igual	BNE
Almacenar el registro de índice Y	STY	Bifurcación si es mayor	BHI
Introducir en la pila los datos del acumulador A	PSHA	Bifurcación si es menor o igual	BLS
Introducir en la pila los datos del acumulador B	PSHB	Bifurcación si es mayor o igual	BHS
Introducir en la pila el contenido del registro de índice X	PSHX	Bifurcación si es menor	BLO
Introducir en la pila el contenido del registro de índice Y	PSHY	Bifurcación si el acarreo es 0	BCC
<i>Lógica</i>		Bifurcación si el acarreo es 1	BCS
Aplicar el operador AND al contenido del acumulador A	ANDA	<i>Saltar y bifurcar</i>	
Aplicar el operador AND al contenido del acumulador B	ANDB	Saltar a la dirección	JMP
Aplicar el operador EXCLUSIVE -XOR al contenido del acumulador A	EORA	Saltar a la subrutina	JSR
Aplicar el operador EXCLUSIVE -XOR al contenido del acumulador B	EORB	Regresar de la subrutina	RTS
Aplicar el operador OR al contenido del acumulador A	ORAA	Bifurcar a la subrutina	BSR
Aplicar el operador OR al contenido del acumulador B	ORAB	Bifurcar siempre	BRA
Reemplazar la memoria por su complemento a 1	COM	Nunca bifurcar	BRN
Reemplazar el acumulador A por su complemento a 1	COMA	Establecer bits de bifurcación	BRSET
Reemplazar el acumulador B por su complemento a 1	COMB	Limpiar bits de bifurcación	BRCLR
<i>Aritmética</i>		<i>Código de condición</i>	
Sumar el contenido del acumulador A al del acumulador B	ABA	Limpiar acarreo	CLC
Sumar el contenido del acumulador B al del registro de índice X	ABX	Limpiar desborde	CLV
Sumar el contenido del acumulador B al del registro de índice Y	ABY	Establecer acarreo	SEC
Sumar la memoria al acumulador A sin acarreo	ADDA	Establecer desborde	SEV
Sumar la memoria al acumulador B sin acarreo	ADDB	Transferir un dato del acumulador A al registro de código de condición	TAP
Sumar la memoria al acumulador doble sin acarreo	ADDD	Transferir un dato del registro de código de condición al acumulador A	TPA
Sumar la memoria al acumulador A con acarreo	ADCA	<i>Diversos</i>	
Sumar la memoria al acumulador B con acarreo	ADCB	No opera	NOP
Ajuste decimal	DAA	Detener procesamiento	STOP
Restar el contenido del acumulador B al contenido del acumulador A	SBA	Modo de prueba especial	TEST
Restar la memoria del acumulador A con acarreo	SBCA		

Nota: La cantidad de bits en un registro depende del procesador. En un microprocesador de 8 bits por lo general hay registros de 8 bits. Algunas veces es posible utilizar juntos dos de los registros de datos para duplicar la cantidad de bits. Dicha combinación se denomina como registro doble.

Intel 8051

Instrucción	Mnemónico	Instrucción	Mnemónico
<i>Transferencia de datos</i>			
Mover datos al acumulador	MOV A, #data	Salto si el acarreo está puesto	JC rel
Mover el registro al acumulador	MOV A, Rn	Salto si el acarreo no está puesto	JNC rel
Mover el byte directo al acumulador	MOV A, direct	Salto si el bit directo está puesto	JB bit, rel
Mover la RAM indirecta al acumulador	MOV A, @Ri	Salto si el bit directo no está puesto	JNB bit, rel
Mover el acumulador al byte directo	MOV direct, A	Salto si el bit directo está puesto y borrarlo	JBC bit, rel
Mover el acumulador a la RAM externa	MOVX @Ri, A	<i>Llamada a subrutina</i>	
Mover el acumulador al registro	MOV Rn, A	Llamada absoluta a subrutina	ACALL addr 11
Mover el byte directo a la RAM indirecta	MOV @Ri, direct	Llamada larga a subrutina	LCALL addr 16
Mover el dato inmediato al registro	MOV Rn, #data	Regreso de subrutina	RET
Mover el byte directo al byte directo	MOV direct, direct	Regreso de interrupción	RETI
Mover la RAM indirecta al byte directo	MOV direct, @Ri	<i>Manipulación de bit</i>	
Mover el registro al byte directo	MOV direct, Rn	Borra el acarreo	CLR C
Mover el dato inmediato al byte directo	MOV direct, #data	Borra el bit	CLR bit
Mover el dato inmediato a la RAM indirecta	MOV @Ri, #data	Fija el acarreo	SETB C
Cargar apuntador de datos con constante de 16 bits	MOV DPTR, #data16	Fija el bit	SETB bit
Mover el código relativo al DPTR, al acumulador	MOV A, @A+DPTR	Complementa el acarreo	CPL C
Mover la RAM externa, dirección 16 bits, al acumulador	MOVX A, @DPTR	Un bit AND al bit de acarreo	ANL C,bit
Mover el acumulador a la RAM externa, dirección 16 bits	MOVX @DPTR, A	Complemento AND del bit al bit de acarreo	ANL C,/bit
Intercambia byte directo con el acumulador	XCH A, direct	Un bit OR al bit de acarreo ORL C,bit	
Intercambia la RAM indirecta con el acumulador	XCH A, @Ri	Complemento OR del bit al bit de acarreo ORL C,/bit	
Intercambia el registro con el acumulador	XCH A, Rn	Mueve un bit al acarreo MOV C,bit	
Empuja el byte directo a la pila	PUSH direct	Mueve el acarreo a un bit MOV bit,C	
Extrae el byte directo de la pila	POP direct	<i>Operaciones lógicas</i>	
<i>Bifurcación</i>			
Salto absoluto	AJMP addr 11	Acumulador AND al byte directo ANL direct, A	
Salto largo	LJMP addr 16	Dato inmediato AND al byte directo ANL direct, #data	
Salto corto, dirección relativa	SJMP rel	Dato inmediato AND al acumulador ANL A, #data	
Salto indirecto relacionado al DPTR	JMP @A+DPTR	Byte directo AND al acumulador ANL A, direct	
Salto si el acumulador es cero	JZ rel	RAM indirecta AND al acumulador ANL A, @Ri	
Salto si el acumulador no es cero	JNZ rel	Registro AND al acumulador ANL A, Rn	
Comparar el byte directo con el acumulador y salta si no son iguales	CJNE A, direct, rel	Acumulador OR al byte directo ORL direct, A	
Comparar el dato inmediato con el acumulador y salta si no son iguales	CJNE A, #data, rel	Dato inmediato OR al byte directo ORL direct, #data	
Comparar el dato inmediato con el registro y salta si no son iguales	CJNE Rn, #data, rel	Dato inmediato OR al acumulador ORL A, #data	
Comparar el dato inmediato con el indirecto y salta si no son iguales	CJNE @Ri, #data, rel	Byte directo OR al acumulador ORL A, direct	
Decrementa el registro y salta si no es cero	DJNZ Rn, rel	RAM indirecta OR al acumulador ORL A, @Ri	
Decrementa el byte directo y salta si no es cero	DJNZ A, direct, rel	Registro OR al acumulador ORL A, Rn	
<i>Sumar</i>			
		Acumulador XOR al byte directo XRL direct, A	
		Dato inmediato XOR al acumulador XRL direct, #data	
		Dato inmediato XOR al acumulador XRL A, #data	
		Byte directo XOR al acumulador XRL A, direct	
		RAM indirecta XOR al acumulador XRL A, @Ri	
		Registro XOR al acumulador XRL A, Rn	
		Sumar datos inmediatos al acumulador	ADD A, #data

(Continúa)

Instrucción	Mnemónico	Instrucción	Mnemónico
Sumar byte directo al acumulador	ADD A, direct	Cambia medio byte con el acumulador	SWAP A
Sumar RAM indirecta al acumulador	ADD A, @Ri	Ajuste decimal del acumulador	DA A
Sumar registro al acumulador	ADD A, Rn	<i>Incrementar y decrementar</i>	
Sumar datos inmediatos al acumulador	ADDC A, #data	Incrementa el acumulador	INC A
con acarreo		Incrementa el byte directo	INC direct
Sumar datos de byte directo al acumulador	ADDC A, direct	Incrementa la RAM indirecta	INC @Ri
con acarreo			
Sumar RAM indirecta al acumulador	ADDC A, @Ri	Incrementa el registro	INC Rn
con acarreo		Decrementa el acumulador	DEC A
Sumar registro al acumulador con acarreo	ADDC A, Rn	Decrementa el byte directo	DEC direct
<i>Restar</i>		Decremento la RAM indirecta	DEC @Ri
Restar datos inmediatos del	SUBB A, #data	Decremento el registro	DEC Rn
acumulador con préstamo		Incrementa el apuntador de datos	INC DPTR
Restar byte directo del acumulador	SUBB A, 29	<i>Operaciones de borrado y complemento</i>	
con préstamo		Complementa el acumulador	CPL A
Restar RAM indirecta del	SUBB A, @Ri	Borra el acumulador	CLR A
acumulador con préstamo		<i>Operaciones de rotación</i>	
<i>Multiplicación y división</i>		Rota el acumulador a la derecha	RR A
Multiplicar A por B	MUL AB	Rota el acumulador a la derecha hasta C	RRC A
Dividir A entre B	DIV AB	Rota el acumulador a la izquierda	RL A
<i>Operaciones matemáticas decimales</i>		Rota el acumulador a la izquierda hasta C	RLC A
Intercambiar el dígito de orden bajo		<i>No operación</i>	
de la RAM indirecta con el acumulador	XCHD A, @Ri	No operación	NOP

Nota: Un valor precedido por # es un número, #data16 es una constante de 16 bits; Rn se refiere a contenidos de un registro; @ Ri se refiere al valor en la memoria donde el registro apunta; DPTR es el apuntador de datos; direct es la ubicación de la memoria donde los datos utilizados en una instrucción se pueden encontrar.

PIC16Cxx

Instrucción	Mnemónico	Instrucción	Mnemónico
Sumar un número al número en el registro de trabajo	addlw number	Mueve (copia) el número en un registro	movf FileReg,w
Sumar el número en el registro de trabajo al número en el registro de archivo y guardarlo en el registro de archivo	addwf FileReg,f	de archivo al registro de trabajo	
Sumar el número en el registro de trabajo al número en el registro de archivo y guardarlo en el registro de trabajo	addwf FileReg,w	Mueve (copia) un número en el registro de trabajo	movlw number
AND un número al número en el registro de trabajo y guardar el resultado en el registro de trabajo	andlw number	Mueve (copia) el número en un registro de archivo al registro de trabajo	mowf FileReg
AND un número en el registro de trabajo al número en el registro de archivo y guardar el resultado en el registro de archivo	andwf FileReg,f	No operación	nop
Borra un bit en un registro de archivo: hacerlo 0	bcf FileReg,bit	Regresa de una subrutina y habilita el bit de habilitación de interrupción global	refie
Pon un bit en un registro de archivo: hacerlo 1	bsf FileReg,bit	Regresa de una subrutina con un número en el registro de trabajo	retlw number
Prueba un bit en un registro de archivo y salta a la siguiente instrucción si el bit es 0	btfsc FileReg,bit	Regresa de una subrutina	return
Prueba un bit en un registro de archivo y salta a la siguiente instrucción si el bit es 1	btfss FileReg,bit	Rota los bits del registro de archivo a la izquierda hasta el bit de acarreo	rlf FileReg,f
Llama a una subrutina, después de la cual regresar a donde se partió	call AnySub	Rota los bits del registro de archivo a la derecha hasta el bit de acarreo	rrf FileReg,f
Borrar: hacer 0 el número en el registro de archivo	cirf FileReg	Manda el PIC a dormir, un modo de bajo consumo de potencia	sleep
Borrar: hacer 0 el número en el registro de trabajo	clrw	Resta el número en el registro de trabajo de un número	sublw number
Borrar el número en el temporizador vigilante	clrwdt		
Complementar el número en un registro de archivo y dejar el resultado en el registro de archivo	comf FileReg,f	Resta el número en el registro de trabajo del número en el registro de archivo, poner resultado en el registro de archivo	subwf FileReg,f
Decrementar un registro de archivo, resultado en registro de archivo	decf FileReg,f	Intercambia las dos mitades de un número de 8 bits en el registro de archivo, dejar el resultado en registro de archivo	swapf FileReg,f
Decrementar un registro de archivo, si el resultado es cero saltar a la siguiente instrucción	decfsz FileReg,f	Usar el número en el registro de trabajo especificar qué bits son entrada o salida	tris PORTX
Salta al punto del programa etiquetado	gotot label		
Incrementa un registro de archivo, resultado en registro de archivo	incf FileReg,f	XOR un número con el número en el registro de trabajo	xorlf number
OR un número con el número en el registro de trabajo	iorlw number		
OR un número en el registro de trabajo con el número en un registro de archivo y guardar el resultado en el registro de archivo	iorwf FileReg,f	XOR el número en el registro de trabajo con el número en el registro de archivo y coloca el resultado en el registro de archivo	xorwf FileReg,f

Nota: f se usa para el registro de archivo, w para el registro de trabajo y b por bit. Los mnemónicos indican los tipos de operandos involucrados, por ejemplo: movlw indica la operación de mover con lw indicando un valor de literal, o sea, un número, en el registro de trabajo w; mowf indica una operación de mover donde el registro de trabajo y el de archivo están involucrados.



Apéndice E: Funciones en biblioteca de C

Las siguientes son algunas funciones comunes de bibliotecas en C. No es una lista completa de todas las funciones dentro de alguna biblioteca o una lista completa de todas las bibliotecas disponibles en el compilador.

<ctype.h>

isalnum	int isalnum(int ch)	Hace la prueba por caracteres alfanuméricos, regresa valor diferente de cero si el argumento es una letra o un dígito o un cero si no es alfanumérico.
isalpha	int isalpha(int ch)	Hace la prueba por caracteres alfabéticos, regresa diferente de cero si es una letra del alfabeto, de otra forma regresa a cero.
iscntrl	int iscntrl(int ch)	Hace la prueba por carácter de control, regresa diferente de cero si está entre 0 y 0x1F o es igual a 0x7F (DEL), de otra forma cero.
isdigit	int isdigit(int ch)	Prueba por un carácter de dígito decimal regresa diferente de cero si es un dígito (0 a 9), cero de otra manera.
isgraph	int isgraph(int ch)	Prueba por un carácter imprimible (excepto espacio), regresa diferente de cero si es imprimible, cero de otra manera.
islower	int islower(int ch)	Prueba por un carácter en minúscula, regresa diferente de cero si es minúscula, de otra forma cero.
isprint	int isprint(int ch)	Prueba por un carácter imprimible (incluyendo espacio) regresa diferente de cero si es imprimible, de otra forma cero.
ispunct	int ispunct(int ch)	Prueba por un carácter de puntuación, regresa diferente de cero si es un carácter de puntuación o cero en otro caso.
isspace	int isspace(int ch)	Prueba por un carácter espacio, regresa diferente de cero si es un espacio, tabulador, forma de alimentación regreso de carro o carácter de nueva línea, de otra forma cero.
isupper	int isupper(int ch)	Prueba por mayúsculas, regresa diferente de cero si es mayúscula, de otra forma cero.
isxdigit	int isxdigit(int ch)	Prueba por un carácter hexadecimal, regresa diferente de cero si es hexadecimal, de otra forma cero.

<math.h>

acos	double acos(double arg)	Regresa el arco coseno del argumento.
asin	double asin(double arg)	Regresa el arco seno del argumento.
atan	double atan(double arg)	Regresa el arco tangente del argumento. Requieren un argumento.

atan2	double atan2(double y, double x)	Regresa el arco tangente de y/x.
ceil	double ceil(double num)	Regresa el entero más pequeño que no es menor que num.
cos	double cos(double arg)	Regresa el coseno de arg. El valor de arg debe estar en radianes.
cosh	double cosh(double arg)	Regresa el coseno hiperbólico de arg.
exp	double exp(double arg)	Regresa e^x donde x es arg.
fabs	double fabs(double num)	Regresa el valor absoluto de num.
floor	double floor(double num)	Regresa el entero más grande no mayor que num.
fmod	double fmod(double x, double y)	Regresa el residuo en punto flotante de x/y.
ldexp	double ldexp(double x, int y)	Regresa x veces 2^y .
log	double log(double num)	Regresa el logaritmo natural de num.
log10	double log10(double num)	Regresa el logaritmo en base 10 de num.
pow	double pow(double base, double exp)	Regresa la base elevada a la potencia exp.
sin	double sin(double arg)	Regresa el seno de arg.
sinh	double sinh(double arg)	Regresa el seno hiperbólico de arg.
sqrt	double sqrt(double num)	Regresa la raíz cuadrada de num.
tan	double tan(double arg)	Regresa la tangente de arg.
tanh	double tanh(double arg)	Regresa la tangente hiperbólica de arg.

<stdio.h>

getchar	int getchar(void)	Regresa el siguiente carácter tecleado.
gets	char gets(char *str)	Lee los caracteres introducidos por el teclado hasta leer una vuelta de carro y los guarda en un arreglo apuntado por str.
printf	int printf(char *str, ...)	Imprime la cadena señalada por str.
puts	int puts(char *str)	Imprime la cadena señalada por str.
scanf	int scanf(char *str, ...)	Lee información hacia las variables señaladas por los argumentos siguientes a la cadena de control.

<stdlib.h>

abort	void abort(void)	Causa la terminación inmediata del programa.
abs	int abs(int num)	Regresa el valor absoluto del entero num.
bsearch	void bsearch(const void *key, const void *base, size_t num, size_t size, int(*compare)(const void *, const void *))	Realiza una búsqueda binaria en el arreglo señalado por base y regresa el apuntador al primer miembro que coincida con la llave apuntada por key. El número de elementos en el arreglo es especificado por num y el tamaño en bytes de cada elemento por size.

calloc	void *calloc(size_t num, size_t size)	Reserva memoria suficiente para un arreglo de objetos num de tamaño dado por size, regresando un apuntador al primer byte de la memoria reservada.
exit	void exit(int status)	Causa terminación normal inmediata de un programa. El valor del estatus se pasa al proceso llamado.
free	void free(void *ptr)	Libera la memoria reservada apuntada por ptr.
labs	long labs(long num)	Regresa el valor absoluto del entero largo num.
malloc	void *malloc(size_t size)	Regresa un apuntador al primer byte de memoria del tamaño dado por size que fue reservado.
qsort	void qsort(void *base, size_t num, size_t size, int(*compare)(const void*, const void*))	Acomoda el arreglo apuntado por base. El número de elementos en el arreglo está dado por num y el tamaño en bytes de cada elemento por size.
realloc	void *realloc(void *ptr, size_t size)	Cambia el tamaño de la memoria reservada apuntada por ptr a aquella especificada por size.

Nota: size_t es el tipo de variable 'size of' y usualmente representa el tamaño de otro parámetro u objeto.

<time.h>

asctime	char *asctime(const struct tm *ptr)	Convierte tiempo de una forma de estructura a una cadena de caracteres apropiados para ser desplegados, regresando un apuntador a la cadena.
clock	clock_t clock(void)	Regresa el número de ciclos de reloj que han transcurrido desde que el programa empezó su ejecución.
ctime	char *ctime(const time_t *time)	Regresa un apuntador a una cadena de la forma día mes fecha horas:minutos:segundos año\n\n0 dando un apuntador al número de segundos transcurrido desde 00:00:00 Tiempo de Greenwich.
difftime	double difftime(time_t time 2, time_t time 1)	Regresa la diferencia en segundos entre el tiempo 1 y tiempo 2.
gmtime	struct tm *gmtime (const time_t *time)	Regresa un apuntador al tiempo convertido de forma long inter a forma de estructura.
localtime	struct tm *localtime (const time_t *time)	Regresa un puntero al tiempo convertido de forma long inter a una estructura en tiempo local. time.
time	time_t time(time_t *system)	Regresa el tiempo del calendario del sistema.

Nota: time_t y clock_t se utilizan como el tipo para variables 'time of' y 'number of cycles'.



Apéndice F: MATLAB Y SIMULINK

F.1

MATLAB

Un software de computación ayuda a calcular y modelar sistemas; un programa que a menudo se utiliza es MATLAB (marca registrada de Mathworks Inc.) versión 4.0 o posterior. Para información adicional consulte la guía del usuario u obras como Hahn, B, *Essential MATLAB for Engineers and Scientists*, 5a. ed., Elsevier, 2012 o Moore, H., *MATLAB for Engineers*, Pearson, 2013.

Los comandos se introducen escribiéndolos enseguida del indicador (`>`) y oprimiendo la tecla enter o return para ejecutar el comando. En la siguiente explicación de los comandos no se repetirá que se debe oprimir la tecla enter o return, se dará por supuesta en todos los casos. Para iniciar MATLAB en los sistemas Windows o Macintosh, haga clic en el icono de MATLAB o escriba matlab. En la pantalla aparecerá el indicador de MATLAB, `>`. Para salir de MATLAB escriba quit o exit después del indicador. Puesto que MATLAB es un programa que distingue entre mayúsculas y minúsculas, para escribir los comandos debe utilizar letras minúsculas.

Al escribir help después del indicador, o seleccionar help de la barra del menú en la parte superior de la ventana de MATLAB, se despliega una lista de temas de ayuda. Para obtener ayuda sobre un tema en particular en la lista, por ejemplo exponentes, escriba help exp. Si escribe lookfor y algún tema, indicará a MATLAB que busque información sobre dicho tema, por ejemplo, lookfor integ desplegará varios comandos que sirven para integrar.

Por lo general, las operaciones matemáticas se introducen en MATLAB de la misma forma como se escribirían en papel. Por ejemplo,

`> a = 4/2`

da como respuesta

`a =`

2

y

`> a = 3*2`

da como respuesta

`a =`

6

Las operaciones se llevan a cabo en el siguiente orden: ^ potenciación, * multiplicación, / división, + suma, - resta. El orden de precedencia de los operadores es de izquierda a derecha, pero los paréntesis () se pueden utilizar para modificar el orden. Por ejemplo,

$$\bullet a = 1 + 2^3 > 4 * 5$$

da como respuesta

$$a = \\ 11$$

porque se tiene $2^3 > 4$ multiplicado por 5 y sumado con 1, donde

$$\bullet a = 1 + 2^3 > (4 * 5)$$

da como respuesta

$$a = \\ 1.4$$

porque se tiene 2^3 dividido entre el producto de 4 por 5, y luego sumado a 1.

Las siguientes son algunas funciones matemáticas disponibles en MATLAB:

abs(x)	Da el valor absoluto de x , es decir, $ x $
exp(x)	Da la exponencial de x , es decir, e^x
log(x)	Da el logaritmo natural de x , es decir, $\ln x$
log10(x)	Da el logaritmo base 10 de x , es decir, $\log_{10} x = \log x$
sqrt(x)	Da la raíz cuadrada de x , es decir, \sqrt{x}
sin(x)	Da el $\sin x$, donde x está en radianes
cos(x)	Da el $\cos x$, donde x está en radianes
tan(x)	Da la $\tan x$, donde x está en radianes
asin(x)	Da el arccsen x , es decir, $\sin^{-1} x$
acos(x)	Da el arccos x , es decir, $\cos^{-1} x$
atan(x)	Da el arctan x , es decir, $\tan^{-1} x$
csc(x)	Da $1/\sin x$
sec(x)	Da $1/\cos x$
cot(x)	Da $1/\tan x$

para introducir π se escribe pi.

En lugar de escribir una serie de comandos después del indicador, se puede preparar un archivo de texto y después ejecutarlos al hacer que MATLAB se refiera a ese archivo. El término archivo-M se utiliza en estos archivos de texto, dado que contienen comandos MATLAB consecutivos con el sufijo .m. Al escribir este tipo de archivos, la primera línea debe iniciar con la palabra function seguida por una declaración que identifique el nombre de la función (function name) y la entrada y la salida en la forma

$$\text{function [salida] = function name [entrada]}$$

Por ejemplo, function y = cotan(x) es el archivo utilizado para determinar el valor de y dado por $\cotan x$. Este archivo se puede invocar con una secuencia de comandos MATLAB al escribir el nombre seguido de la entrada, por ejemplo cotan(x). De hecho, ya está incluida en MATLAB y se utiliza cuando se necesita la cotangente de x . Sin embargo, el archivo podría estar escrito por el usuario. Una función que tiene varias entradas debe especificarlas todas en

el enunciado de la función. Por otra parte, cuando una función produce varios valores, será necesario especificar todas las salidas posibles.

Las líneas que inician con % son líneas de comentarios; MATLAB no las interpreta como comandos. Por ejemplo, suponga que se prepara un programa para calcular los valores de la raíz cuadrática media de una columna de datos, el programa se vería como

```
function y=rms(x)
% rms Raíz cuadrática media
% rms(x) da el valor de la raíz cuadrática media de los
% elementos del vector columna x.
xs=x^2;
s=size(x);
y=sqrt(sum(xs)/s);
```

Se definió xs como el cuadrado de cada valor x . Con el comando $s=size(x)$ se obtiene la magnitud; es decir, la cantidad de entradas en la columna de datos. El comando $y=sqrt(sum(xs)/s)$ obtiene la raíz cuadrada de la suma de todos los valores xs divididos entre s . El comando ; se coloca al final de cada línea del programa.

MATLAB ofrece una cantidad de cajas de herramientas que contienen colecciones de archivos-M. Es de particular relevancia para este libro Toolbox Control System, que permite obtener respuestas en tiempo de un sistema que tiene entradas tipo impulso, escalón, rampa, etc., junto con el análisis de Bode, Nyquist, lugar geométrico de las raíces, etc. Por ejemplo, para llevar a cabo un diagrama de Bode de un sistema descrito por una función de transferencia $4/(s^2 + 2s + 3)$, el programa es

```
%Genera diagrama de Bode para G(s)=4/(s^2 + 2s + 3)
num=4
den=[1 2 3];
bode(num,den)
```

El comando `bode(num,den)` produce el diagrama de Bode de ganancia en dB contra la frecuencia en rad/s en una escala logarítmica y fase en grados contra la frecuencia en rad/s en una escala logarítmica.

F.1.1 Graficación

Para producir gráficas lineales de dos dimensiones se utiliza el comando `plot(x,y)`, el cual permite graficar los valores de x y y . Por ejemplo,

```
x=[0 1 2 3 4 5];
y=[0 1 4 9 16 25];
plot(x,y)
```

Para graficar una función, ya sea estándar o definida por el usuario, se utiliza el comando `fplot(function name,lim)`, donde `lim` define el intervalo de graficación, es decir, los valores mínimo y máximo de x .

El comando `semilogx(x,y)` genera una gráfica de los valores de x y y utilizando una escala logarítmica para x y una escala lineal para y . El comando `semilogy(x,y)` genera una gráfica de los valores de x y y utilizando una escala lineal para x y una escala logarítmica para y . El comando `loglog(x,y)` genera una gráfica de los valores de x y y mediante escalas logarítmicas para ambas. El comando `polar(theta,r)` produce una gráfica en coordenadas polares, donde θ es el argumento en radianes y r la magnitud.

El comando subplot permite dividir la ventana de gráficas y colocar las gráficas en cada subdivisión. Por ejemplo,

```
x=(0 1 2 3 4 5 6 7);
y=exp(x);
subplot(2,1,1);plot(x,y);
subplot(2,1,2);semilogy(x,y);
```

En el comando subplot(m,n,p), los dígitos m y n indican que la ventana de gráficas se dividirá en una cuadrícula o malla de $m * n$ de ventanas más pequeñas, donde m es el número de renglones, n es el número de columnas y el dígito p especifica la ventana que se utilizará para la gráfica. Las subventanas se numeran por renglón, de izquierda a derecha y de arriba hacia abajo. Así, la secuencia de comandos anterior divide la ventana en dos, con una gráfica arriba de la otra; la gráfica de arriba es una gráfica lineal y la de abajo es una gráfica semilogarítmica.

Se pueden seleccionar el número y tipo de líneas de la cuadrícula, el color de la gráfica y la inclusión de texto en una gráfica. El comando print se utiliza para imprimir una copia de una gráfica ya sea en un archivo o en una impresora. Esto se puede hacer al seleccionar el archivo en la barra de menú en la ventana y luego seleccionar la opción print.

F.1.2 Funciones de transferencia

Las siguientes líneas de un programa MATLAB ilustran cómo especificar una función de transferencia y mostrarla en la pantalla:

```
%G(s)=4(s+10)/(s+5)(s+15)
num=*4[1 10];
den = conv ([1 5], [1 15]);
printsys(num,den,'s')
```

El comando num se utiliza para indicar el numerador de la función de transferencia en potencias decrecientes de s . El comando den se utiliza para indicar el denominador en potencias decrecientes de s para cada uno de los dos polinomios del denominador. El comando conv multiplica dos polinomios, en este caso son $(s + 5)$ y $(s + 15)$. El comando printsys despliega la función de transferencia en la pantalla, con numerador y denominador especificados y escritos en el dominio de s .

Algunas veces la función de transferencia se presenta como el cociente de dos polinomios y es necesario determinar los polos y ceros. Para esto se puede utilizar

```
% Encontrar los polos y ceros de la función de transferencia
% G(s)=(5s^2 + 3s + 4)/(s^3 + 2s^2 + 4s + 7)
num=[5 3 4];
den=[1 2 4 7];
[z,p,k]=tf2zp(num,den)
```

[z,p,k]=tf2zp(num,den) es el comando para determinar y desplegar los ceros (z), los polos (p) y la ganancia (k) de la función de transferencia introducida.

MATLAB se puede utilizar para obtener gráficas que muestren la respuesta de un sistema para diferentes entradas. Por ejemplo, con el siguiente programa se obtiene la respuesta del sistema a una entrada tipo escalón unitario, $u(t)$, que tiene una función de transferencia especificada:

```
% Despliegue la respuesta a una entrada de escalón para un sistema con
% función de transferencia G(s)=5/(s^2 + 3s + 12)
```

```
num=5;
den=[1 3 12];
step(num,den)
```

F.1.3 Diagramas de bloques

A menudo los sistemas de control se representan como una serie de bloques interconectados, cada bloque con una característica específica. MATLAB permite que se construyan sistemas en bloques interconectados. Los comandos que se utilizan son `doop` cuando un bloque con una función de transferencia en lazo abierto tiene realimentación unitaria. Si la realimentación no es unitaria, se utiliza el comando `feedback`. Por ejemplo, a la Figura F.1 le corresponde el programa:

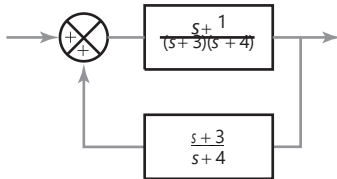


Figura F.1 Diagrama de bloques.

```
% Sistema con lazo de realimentación
ngo=[1 1];
dgo=conv([1 3],[1 4]);
nh=[1 3];
dh=[1 4];
[ngc2,dgc2]=feedback(ngo,dgo,nh,dh)
printsys(ngc2,dgc2,'s')
```

`ngo` y `dgo` indican el numerador y denominador de una función de transferencia de lazo abierto $G_0(s)$, `nh` y `dh` son el numerador y denominador respectivamente de la función de transferencia del lazo de realimentación $H(s)$. El programa da como resultado un despliegue de la función de transferencia del sistema completo.

El comando `series` indica que dos bloques están en serie en una trayectoria en particular; el comando `parallel` indica que dichos bloques están en paralelo.

F.2

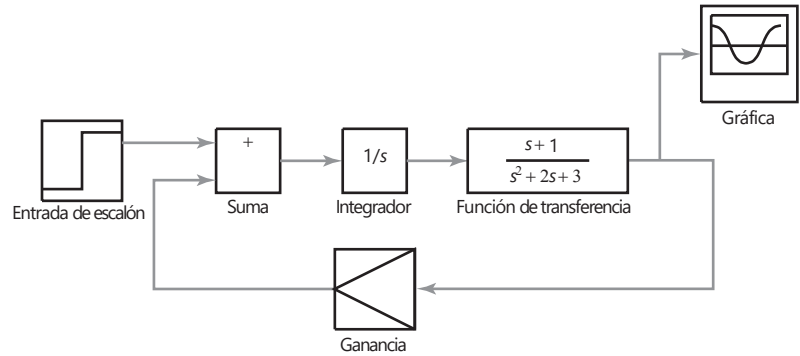
SIMULINK

SIMULINK se utiliza con MATLAB para especificar sistemas mediante cajas o bloques de "conexión" en pantalla en lugar de, como lo anterior, la escritura de una serie de comandos para generar la descripción del diagrama de bloques. Una vez iniciado MATLAB, con el comando `simulink` se ejecuta SIMULINK, lo que abre la ventana de control SIMULINK con sus iconos y menús desplegables en su barra superior. Haga clic en `File`, luego en `New` en el menú desplegable, esto abre una ventana en la cual se puede ensamblar un sistema.

Para iniciar el ensamble de los bloques requeridos, regrese a la ventana de control y haga doble clic en el icono `linear`. Haga clic y arrastre el icono `transfer Fcn` hacia la nueva ventana aún sin título. Si requiere un bloque para la ganancia, haga clic y arrastre el icono `gain` hacia la ventana sin título. Haga esto mismo con el icono `sum` y quizá también con el icono `integrator`. De esta manera, arrastre todos los iconos requeridos dentro de la ventana sin título. Luego haga doble clic en el icono `Sources` y elija la fuente apropiada del menú desplegable, por ejemplo la entrada `step`, y arrástrelo a la ventana sin título. Ahora haga doble clic en el icono `sinks` (descarga o exhibición) y arrastre el icono `Graph` hacia la ventana sin título. Para conectar los iconos, oprima el botón del ratón mientras la flecha del ratón está en el símbolo de salida de un icono y arrástrelo hasta el símbolo de entrada del icono que se desea conectar. Repita lo anterior con todos los iconos hasta terminar de armar todo el diagrama de bloques.

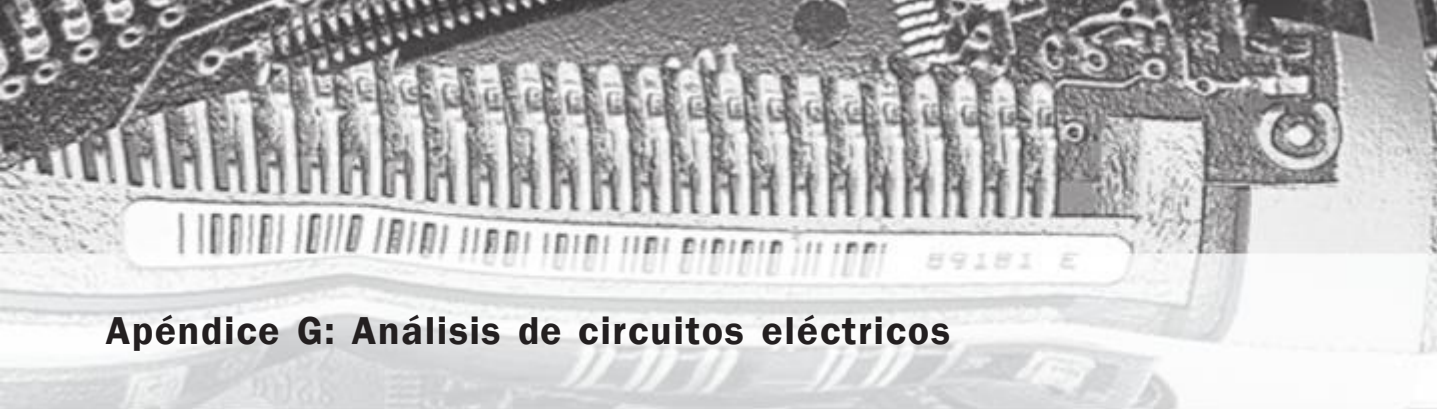
Para asignar a la caja de transfer `Fcn` una función de transferencia, haga doble clic en la caja y aparecerá un cuadro de diálogo en la que puede introdu-

Figura F.2 Ejemplo del uso de SIMULINK.



cir comandos MATLAB para numerador y denominador. Si requiere $(s + 1)$, haga clic en el numerador y escriba [1 1]. Haga clic en el denominador y si requiere $(s^2 + 2s + 3)$ escriba [1 2 3], luego haga clic en el icono Done. Haga doble clic en el icono Gain y escriba el valor de la ganancia. Haga doble clic en el icono Sum y ponga los signos + o - dependiendo si se requiere realimentación positiva o negativa. Haga doble clic en el icono Graph y defina los parámetros de la gráfica. Ahora ya tiene todo el diagrama de la simulación en la pantalla. En la Figura F.2 se muestra la forma que puede tomar. Para eliminar cualquier bloque o conexión, selecciónelos haciendo clic y luego presione la tecla .

Para simular el comportamiento del sistema, haga clic en Simulation para desplegar su menú. Seleccione Parameters y defina los momentos de inicio y término para la simulación. En el menú Simulation seleccione Start. SIMULINK entonces creará una ventana para graficar y desplegará la salida correspondiente del sistema. Para guardar el archivo seleccione File y haga clic en SAVE AS en el menú desplegable. Escriba el nombre del archivo en el cuadro de diálogo y haga clic en Done.



Apéndice G: Análisis de circuitos eléctricos

G.1 Circuitos de corriente continua

En el análisis de circuitos se aplican las leyes básicas comprendidas en las leyes de Kirchhoff.

- 1 **La ley de Kirchhoff de la corriente** establece que en cualquier nodo en un circuito eléctrico, la corriente que entra es igual a la corriente que sale de él.
- 2 **La ley de Kirchhoff del voltaje** establece que en torno a un circuito cerrado o malla, la suma de voltaje que fluye por todos los componentes es igual a la suma de los voltajes aplicados.

Aun cuando circuitos que contengan combinaciones de resistores conectados en serie y en paralelo, suelen verse reducidos a un circuito sencillo al determinar sistemáticamente la resistencia equivalente de los resistores y los reductores conectados en serie o en paralelo y al reducir el problema del análisis a un circuito muy sencillo, podrían requerirse las técnicas siguientes cuando se traten circuitos más complejos.

G.1.1 Análisis de nodos

Un **nodo** es un punto en un circuito donde se unen dos o más dispositivos; es decir, es aquella conjunción en la cual se tiene una corriente que entra y una corriente que sale. Un **nodo principal** es un punto donde se conectan entre sí tres o más elementos. Precisamente, en la Figura G.1, b) y d) son los nodos principales. Se elige uno de los nodos principales para que sea un nodo de referencia de modo que las diferencias de potencia en los demás nodos se consideren en referencia con dicho nodo. Para el análisis siguiente en que se aplica la Figura G.1, se ha tomado d como el nodo de referencia. Entonces se aplica la ley de Kirchhoff de la corriente a cada nodo no referente. El procedimiento es el siguiente.

- 1 Se traza un diagrama de circuito con etiquetas y se marcan en él los nodos principales.
- 2 Se selecciona uno de los nodos principales como un nodo de referencia.
- 3 Se aplica la ley de Kirchhoff de la corriente a cada uno de los nodos no referentes, aplicando la ley de Ohm para expresar las corrientes que pasan a través de los resistores respecto a los voltajes de nodos.
- 4 Se resuelven las ecuaciones simultáneas resultantes. Si hay n nodos principales habrá ecuaciones $(n - 1)$.
- 5 Se utilizan los valores derivados de los voltajes de los nodos para determinar las corrientes en cada bifurcación del circuito.

Como muestra, considere la Figura G.1. Los nodos son a, b, c y d con b y d como nodos principales. Tomo el nodo d como el nodo de referencia.

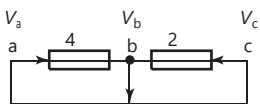


Figura G.1 Circuito para análisis de nodos.

d

Si V_a , V_b y V_c son los voltajes de nodo relativos al nodo d, entonces la d.p. a través del resistor de $4\ \Omega$ es $(V_a - V_b)$, que es V_b a través del resistor de $3\ \Omega$ y que al pasar por el resistor de $2\ \Omega$ es $(V_c - V_b)$. De este modo, la corriente a través del resistor de $4\ \Omega$ es $(V_a - V_b)/4$, la que pasa por el resistor de $3\ \Omega$ es $V_b/3$ es $V_b/3$, y la que pasa por el resistor de $2\ \Omega$ es $(V_c - V_b)/2$. Por lo tanto, al aplicar la ley de Kirchhoff de la corriente al nodo b resulta:

$$\frac{V_a - V_b}{4} + \frac{V_c - V_b}{2} = \frac{V_b}{3}$$

Sin embargo, $V_a = 10\ \text{V}$ y $V_c = 5\ \text{V}$ y por lo tanto:

$$\frac{10 - V_b}{4} + \frac{5 - V_b}{2} = \frac{V_b}{3}$$

De este modo $V_b = 4.62\ \text{V}$. La d.p. a través del resistor de $4\ \Omega$ es entonces $10 - 4.62 = 5.38\ \text{V}$ y por eso la corriente que pasa a través de él es $5.38/4 = 1.35\ \text{A}$. La d.p. a través del resistor de $3\ \Omega$ es $4.62\ \text{V}$ y así la corriente a través de él es $4.62/3 = 1.54\ \text{A}$. La d.p. a través del resistor de $2\ \Omega$ es $5 - 4.62 = 0.38\ \text{V}$ y así la corriente a través de él es de $0.38/2 = 0.19\ \text{A}$.

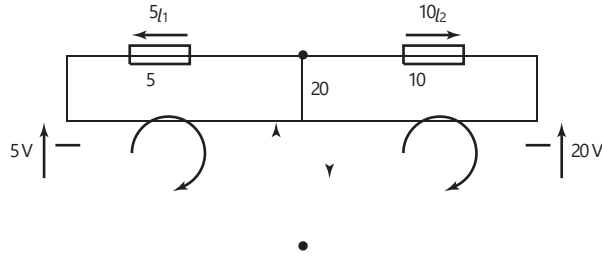
G.1.2 Análisis de mallas

El término **ciclo** se aplica a una secuencia de elementos de circuito que conforma una ruta cerrada. Una **malla** es un ciclo de circuitos el cual no contiene ningún otro ciclo en su conjunto. El análisis de mallas implica la definición de una corriente que circula alrededor de cada malla. Se debe escoger la misma dirección para cada corriente de malla y la convención usual es hacer que todas las corrientes de malla circulen en el sentido de las manecillas del reloj. Una vez especificadas las corrientes de malla, se aplica la ley de Kirchhoff del voltaje a cada malla. Éste es el procedimiento.

- 1 Se etiqueta cada una de las mallas con corrientes de malla en sentido de las manecillas del reloj.
- 2 Se aplica la ley de Kirchhoff del voltaje a cada malla, se dan las diferencias de potencia a través de cada resistor de la ley de Ohm respecto de las corrientes que fluirán a través de ellas y en la dirección opuesta a la corriente. La corriente a través de un resistor que bordea precisamente una malla es la corriente de malla; la corriente que pasa por un resistor que bordea dos mallas es la suma algebraica de las corrientes de malla a través de las dos mallas.
- 3 Se resuelven las ecuaciones simultáneas resultantes para obtener las corrientes de malla. Si hay n mallas habrá n ecuaciones.
- 4 Utilice los resultados para las corrientes de malla para determinar las corrientes en cada bifurcación del circuito.

Como ejemplo, para el circuito que se muestra en la Figura G.2 hay tres ciclos, ABCF, CDEF y ABCDEF, pero sólo los dos primeros son mallas. En estas mallas se pueden definir las corrientes I_1 e I_2 como que circulan en la misma dirección que las manecillas del reloj.

Figura G.2 Circuito que ilustra el análisis de mallas.



Para la malla 1, aplicando la ley de Kirchhoff del voltaje resulta $5 - 5I_1 - 20(I_1 - I_2) = 0$. Lo cual se puede reescribir como:

$$5 = 25I_1 - 20I_2$$

Para la malla 2, aplicando la ley de Kirchhoff del voltaje resulta $-10I_2 - 20(I_2 - I_1) = 0$. Lo cual se puede reescribir como:

$$20 = 20I_1 - 30I_2$$

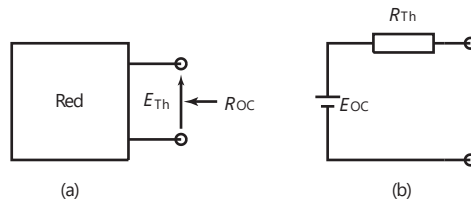
Ahora tenemos un par de ecuaciones simultáneas y por lo tanto $I_2 = -1.14$ A e $I_1 = -0.71$ A. El signo de resta indica que las corrientes están en direcciones opuestas a las que se indica en la Figura. De modo que la corriente a través del resistor de $20\ \Omega$ está en la dirección de I_1 y $-0.71 + 1.14 = 0.43$ A.

G.1.3 El teorema de Thévenin

El circuito equivalente para cualquier red de dos terminales que contenga una fuente de voltaje o de corriente lo proporciona el **teorema de Thévenin**:

Toda red de dos terminales (Figura G.3a) que contenga fuentes de voltaje o de corriente puede ser reemplazado por un circuito equivalente que contenga un voltaje igual al voltaje del circuito abierto del circuito original en serie, con la resistencia medida entre las terminales cuando no hay carga conectada entre ellos y todas las fuentes independientes en la red están establecidas en igual a cero (Figura G.3b).

Figura G.3 a) La red, b) su equivalente.



Si tenemos un circuito lineal para utilizar el teorema de Thévenin hay que dividirlo en dos circuitos, A y B, conectados a un par de terminales. Así ya se puede aplicar el teorema de Thévenin para reemplazar, digamos, el circuito A por su circuito equivalente. El circuito abierto de voltaje de Thévenin para el circuito A es el que resulta cuando el circuito B está desconectado y la resistencia de Thévenin para A es el análisis de resistencia en las terminales de A con todas sus fuentes independientes establecidas igual a cero. La Figura G.4 ilustra esta secuencia de pasos.

Figura G.4 Método paso por paso para el análisis de circuitos.

1. Identificar las dos partes A y B del circuito y separarlas por terminales.
2. Separar la parte A de la B.
3. Reemplazar A por su equivalente de Thévenin, es decir una fuente de voltaje con una resistencia en serie.
4. Reconectar el circuito B y efectuar el análisis.

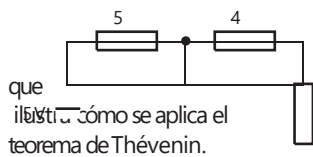
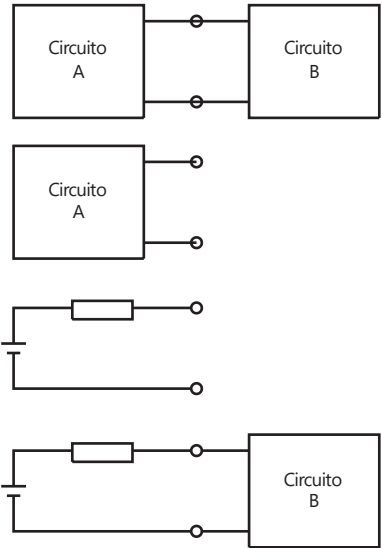


Figura G.5 Circuito de ejemplo

A guisa de ejemplo, considere aplicar el teorema de Thévenin para determinar la corriente a través del resistor de 10 Ω en el circuito dado en la Figura G.5.

Dado que el mayor interés está en la corriente a través del resistor de 10 Ω lo identificamos como la red B y el resto del circuito como la red A, conectándolas por terminales (Figura G.6a). Luego hay que separar A de B (Figura G.6b) y determinar su equivalente de Thévenin.

El voltaje del circuito abierto es el que cruza el resistor de 20 Ω; es decir, la fracción del voltaje total que pasa por el resistor de 20 Ω:

$$E_{Th} = 5 \frac{20}{20 + 5} = 4V$$

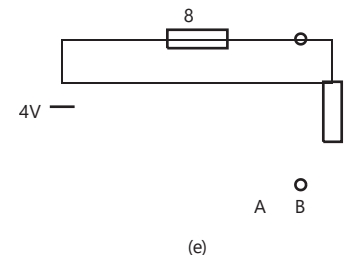
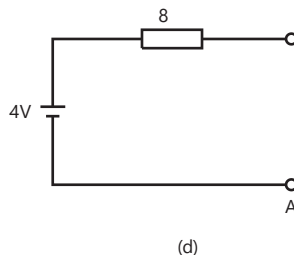
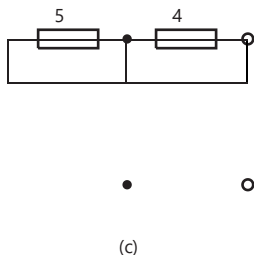
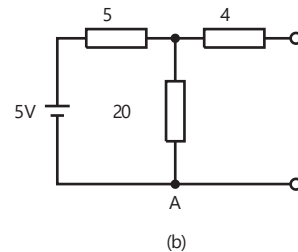
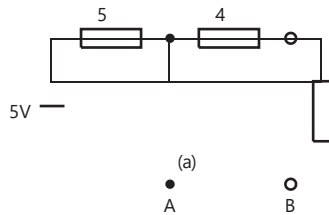


Figura G.6 El análisis de Thévenin: a) toma de los puntos de terminal; b) separación de los elementos del circuito; c) análisis de resistencia en las terminales; d) circuitos equivalentes, y e) el circuito completo.

El análisis de resistencia en las terminales cuando la fuente de voltaje es igual a cero es que el resistor en serie de $4\ \Omega$ con un arreglo en paralelo de $5\ \Omega$ y $20\ \Omega$ (según la Figura G.6c) es:

$$R_{Th} = 4 + \frac{20 \times 5}{20 + 5} = 8\ \Omega$$

Por tanto el circuito equivalente de Thévenin es como se muestra en la Figura G.6d), y cuando la red B se conecta a él tenemos el circuito que se muestra en la Figura G.6e). De donde la corriente a través del resistor de $10\ \Omega$ es $I_{10} = 4 / (8 + 10) = 0.22\ \text{A}$.

G.1.4 El teorema de Norton

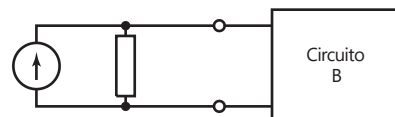
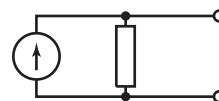
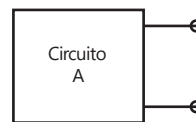
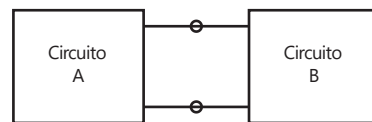
De igual manera que con el teorema de Thévenin, podemos tener un circuito equivalente en cualquier red de dos terminales que cuente con fuentes de voltaje o de corriente, y ese es el **teorema de Norton**:

Cualquier red de dos terminales que contenga fuentes de voltaje o de corriente puede ser reemplazada por una red equivalente que conste de una fuente de corriente, igual a la corriente entre las terminales cuando están en cortocircuito, en paralelo con la resistencia medida entre las terminales cuando no hay una carga entre ellas y cuando todas las fuentes independientes en la red están establecidas igual a cero.

Si tenemos un circuito lineal habrá que dividirlo en dos circuitos, el A y el B, conectados en un par de terminales (Figura G.7). Entonces se puede utilizar ya el teorema de Norton para reemplazar, digamos, el circuito A por su circuito equivalente. La corriente del cortocircuito de Norton para el circuito A es la que se da cuando el circuito B está desconectado y la resistencia de Norton para el circuito A es el análisis de la resistencia en las terminales de A con todas sus fuentes independientes establecidas igual a cero.

Figura G.7 Método paso por paso para el análisis de circuitos aplicando el teorema de Norton.

1. Identificar las dos partes A y B del circuito y separar por terminales.
2. Separar A de B.
3. Reemplazar A por su equivalente de Norton.
4. Reconectar el circuito B y efectuar el análisis.



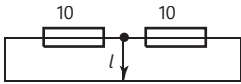


Figura G.8 Circuito de análisis

Como ejemplo de la aplicación del teorema de Norton, considere la determinación de la corriente I a través del resistor de 20Ω en la Figura G.8.

El circuito se puede trazar de nuevo en la forma que se muestra en la Figura G.9a) como dos redes A y B conectadas y la red B elegida para que sea el resistor de 20Ω a través del cual se requiere que fluya la corriente. Luego se determina el circuito equivalente de Norton para la red A (Figura G.9b). Al poner las terminales de la red A en cortocircuito se da el circuito que se muestra en la Figura G.9c). La corriente del cortocircuito será la suma, teniendo en cuenta las direcciones, de las corrientes a partir de las dos bifurcaciones de los circuitos que contengan fuentes de voltaje, es decir $I_{sc} = I_1 - I_2$. La corriente $I_1 = 15/10 = 1.5 \text{ A}$, dado que la otra parte de la red está en cortocircuito, el $I_2 = 10/10 = 1.0 \text{ A}$. Por lo tanto $I_{sc} = 0.5 \text{ A}$. La resistencia de Norton está dada por la que cruza las terminales cuando todas las fuentes están establecidas en cero (Figura G.9d). En consecuencia:

$$R_N = \frac{10 \times 10}{10 + 10} = 5 \Omega$$

Por lo tanto el circuito equivalente de Norton es el que se muestra en la Figura G.9e). De donde resulta que cuando lo ponemos con la red B (Figura G.9f), obtenemos la corriente I . La d.p. que cruza los resistores es $0.5 \times R_{total}$ y por consiguiente la corriente I es esta d.p. dividida entre 20 . Así:

$$I = 0.5 \times \frac{5}{5 + 20} = 0.1 \text{ A}$$

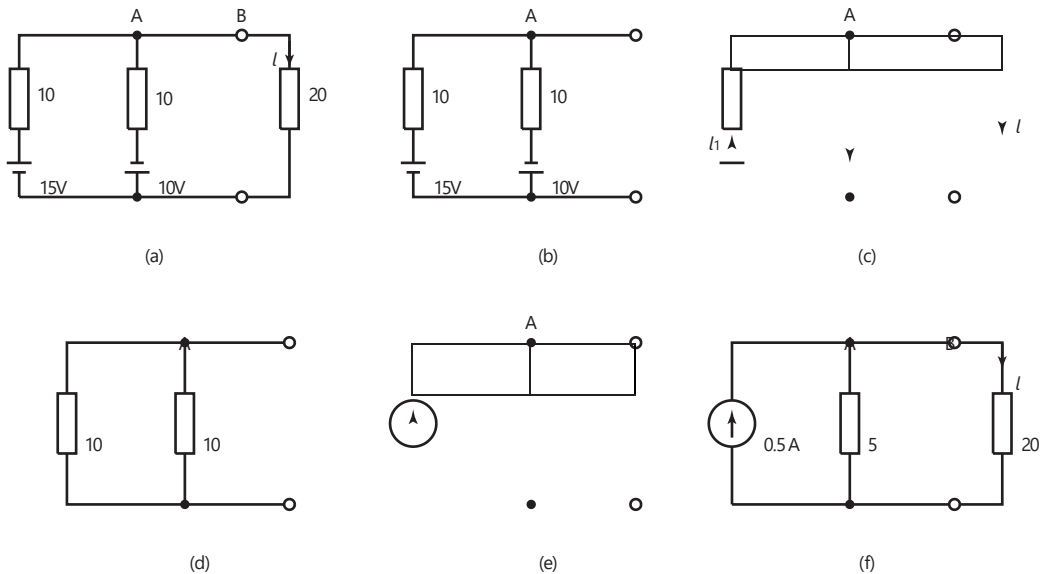


Figura G.9 Análisis de Norton: a) nuevo trazo del circuito; b) red A; c) terminales en cortocircuito; d) fuentes establecidas en cero; e) equivalente de Norton, y f) las partes de los circuitos, combinadas.

G.2

Se puede generar una forma de onda senoidal si se gira una línea de radio OA a una velocidad angular ω constante (Figura G.10a), variando la proyección vertical de la línea AB con el tiempo de manera senoidal. El ángulo ωt de la

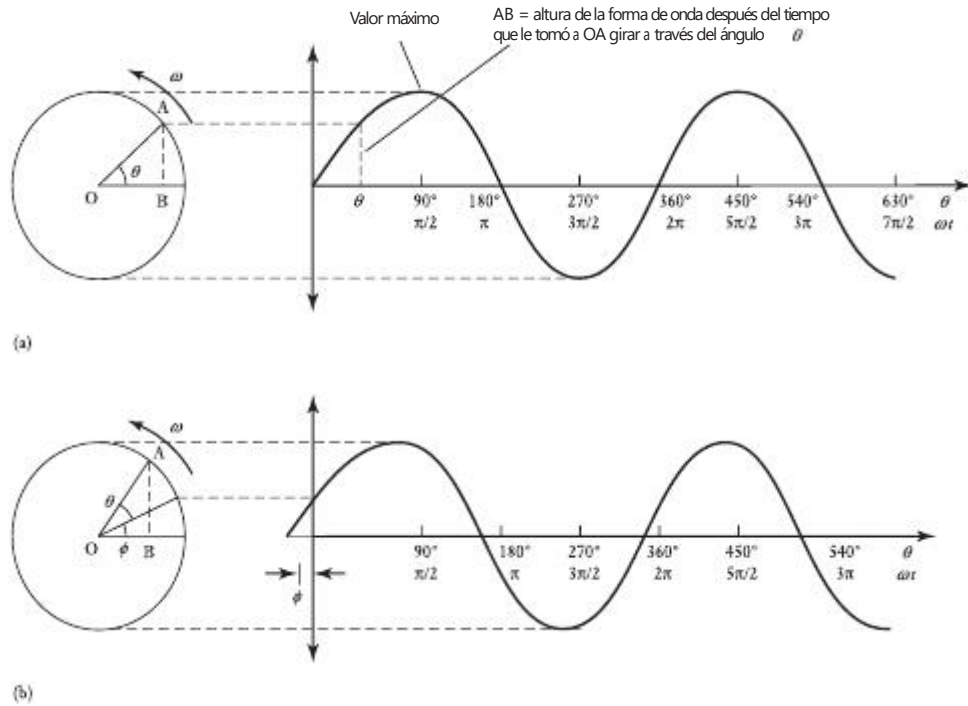


Figura G.10 Generación de ondas senoidales: a) valor cero en el tiempo $t = 0$, (b) un valor inicial en $t = 0$.

línea Ab en un tiempo t es vt . La frecuencia f de rotación es $1/T$, donde T es el tiempo que toma dar un giro (rotación) completo, y por lo tanto $v = 2\pi f$. En la Figura G.10a) la línea OA que gira se muestra como si comenzara desde una posición horizontal en tiempo $t = 0$. La Figura 10b) muestra que la línea OA en $t = 0$ ya está en algún ángulo ϕ . Como la línea AO gira con una velocidad angular v , entonces en un tiempo t el ángulo barrió su vt y por lo tanto en el tiempo t el ángulo con respecto a la horizontal es $vt + \phi$. Las corrientes alternas y voltajes senoidales se pueden describir por dichas líneas que rotan y de las que salen las ecuaciones $i = I_m \sin vt$ y $v = V_m \sin vt$ para voltajes con valores cero en el tiempo $t = 0$ y para los que arrancan en algún ángulo inicial ϕ por $i = I_m \sin (vt + \phi)$ y $v = V_m \sin (vt + \phi)$. Los símbolos en minúscula se utilizan para los términos de corriente y voltaje que se modifican con el tiempo; las mayúsculas se reservan para términos invariables.

Con circuitos de corriente alterna surge la necesidad de considerar la relación entre una corriente alterna que fluye a través de un componente y el voltaje alterno que lo atraviesa. Si tomamos la corriente alterna como la referencia para un circuito en serie y la consideramos para que se represente con $i = I_m \sin vt$, entonces el voltaje se puede representar como $v = V_m \sin (vt + \phi)$. Se dice que hay una **diferencia de fase** de ϕ entre la corriente y el voltaje. Si ϕ tiene un valor positivo entonces se dice que el voltaje está **conduciendo** la corriente (como en la Figura G.10 si a) representa la corriente y b) el voltaje); pero si el valor es negativo, entonces se dice que el voltaje está **retardando** la corriente.

Se puede describir una corriente alterna senoidal con sólo especificar la línea que gira con respecto a su longitud y su ángulo inicial en relación con una

línea horizontal de referencia. Para este tipo de líneas se utiliza el término **fasor**, abreviatura del concepto vector de fase. La longitud del fasor puede representar el valor máximo de la forma de onda senoidal o valor del cuadrado de la raíz media (r.m.s), puesto que el valor máximo es proporcional al valor del rms. Como las corrientes y los voltajes en el mismo circuito tendrán la misma frecuencia, los fasores que los representen girarán con la misma velocidad angular y mantendrán los mismos ángulos de fase entre ellos todas las veces, no debemos preocuparnos de trazar los efectos de su rotación sino sólo de trazar los diagramas de fasores proporcionando las posiciones angulares relativas de los fasores y pasar por alto sus rotaciones.

Lo siguiente resume los principales puntos de los fasores:

- 1 Un fasor tiene una longitud que es directamente proporcional al valor máximo de la cantidad senoidal alterna o porque el valor máximo es proporcional al valor del r.m.s., una longitud proporcional al valor del r.m.s.
- 2 Los fasores se toman para que giren en sentido contrario al de las manecillas del reloj y tienen una cabeza de flecha al final de la cual giran.
- 3 El ángulo entre dos fasores muestra el ángulo de fase entre sus formas de onda. El fasor que está en un ángulo mayor que el sentido contrario al de las manecillas del reloj se dice que va hacia adelante, en tanto que el ángulo menor contrario al de las manecillas del reloj se dice que se retrasa.
- 4 La línea horizontal se toma como el eje de referencia y a uno de los fasores se le da esa dirección; los demás tienen sus ángulos de fase proporcionados en relación con este eje de referencia.

G.2.1 Resistencia, inductancia y capacitancia en circuitos de a.c.

Considere una corriente senoidal $i = I_m \sin vt$ que pasa a través de una **resistencia pura**. Una resistencia pura es aquella que sólo tiene resistencia pero no inductancia ni capacitancia. Desde el momento en que se supone que hay que aplicar la ley de Ohm, el voltaje v que cruza por la resistencia debe ser $v = Ri$ y por lo tanto $v = RI_m \sin vt$. De este modo, la corriente y el voltaje están en fase. El voltaje máximo se dará cuando $\sin vt = 1$ y por consiguiente $V_m = RI_m$.

Ahora considere una corriente senoidal $i = I_m \sin vt$ que pasa a través de una **inductancia pura**. Una inductancia pura es aquella que sólo tiene inductancia pero no resistencia ni capacitancia. Con una inductancia, una corriente cambiante produce una fuerza contraelectromotriz $L \frac{di}{dt}$, donde L es la inductancia. La fuerza electromotriz aplicada debe sobrepasar a esta fuerza contraelectromotriz para que haya un flujo de corriente. Por lo tanto el voltaje v que pasa a través de la inductancia es $L \frac{di}{dt}$ y por consiguiente

$$v = L \frac{di}{dt} = L \frac{d}{dt}(I_m \sin vt) = vLI_m \cos vt$$

Dado que $\cos vt = \sin(vt - 90^\circ)$, la corriente y el voltaje están desfasados y el voltaje está conduciendo la corriente a 90° . El voltaje máximo se presenta cuando $\cos vt = 1$ y por lo tanto tenemos que $V_m = vLI_m$. A la fórmula V_m/I_m se le denomina **reactancia inductiva** X_L . Entonces $X_L = V_m/I_m = vL$. Puesto que $v = 2\pi f$, entonces $X_L = 2\pi fL$ y por lo tanto la reactancia es proporcional a la frecuencia f . A mayor frecuencia, mayor oposición a la corriente.

Considere un circuito que sólo tiene **capacitancia pura** con un voltaje senoidal $v = V_m \sin vt$ que se ha de aplicar a través de él. Una capacitancia pura es aquella que sólo tiene capacitancia pero no resistencia ni inductancia. La

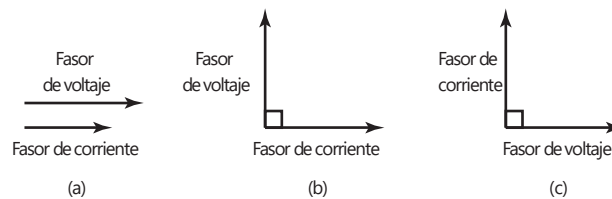
carga q sobre las placas de un capacitor está en relación con el voltaje v por $q = C v$. Entonces, puesto que la corriente es la relación del movimiento de la carga dq/dt , tenemos que $i =$ velocidad del cambio de $q =$ velocidad de cambio de $(C v) = C \frac{dv}{dt}$ (velocidad de cambio de v), es decir $i = C \frac{dv}{dt}$.

$$i = C \frac{dv}{dt} = C \frac{d}{dt} (V_m \sin vt) = v C V_m \cos vt$$

Como $\cos vt = \sin(vt + 90^\circ)$, la corriente y el voltaje están desfasados, la corriente está conduciendo el voltaje a 90° . La corriente máxima se presenta cuando $\cos vt = 1$ y por lo tanto $I_m = v C V_m$. A la fórmula V_m / I_m se le denomina **reactancia capacitiva** X_C . Por lo tanto $X_C = V_m / I_m = 1/vC$. La reactancia tiene la unidad de ohms y es una medida de la oposición a la corriente. Cuanto mayor sea la reactancia mayor debe ser el voltaje que la corriente debe conducir a través de ella. Dado que $v = 2\pi f$, la reactancia es inversamente proporcional a la frecuencia f y por lo tanto cuanto mayor sea la frecuencia, menor será la oposición a la corriente. Con c.d., es decir frecuencia cero; la reactancia es infinita y en consecuencia la corriente no fluye.

En resumen, la Figura G.11 muestra los fasores de voltaje y corriente para a) resistencia pura, b) inductancia pura, y c) capacitancia pura.

Figura G.11 Fasores con a) resistencia pura; b) inductancia pura, y c) capacitancia pura.

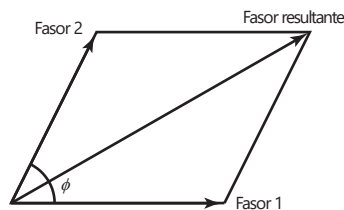


G.2.2 Circuitos de c.a. en serie

En un circuito en serie el voltaje total es la suma de la diferencia de potencial en serie (p.d.s.) que cruza los componentes en serie, aunque la p.d.s. puede diferir en fase. Esto significa que si consideramos los factores, rotarán con la misma velocidad angular pero pueden tener longitudes diferentes y arrancar con un ángulo de fase entre ellos. Se puede obtener la suma de los voltajes de dos series si se aplica la **ley del paralelogramo** de vectores para agregar dos fasores:

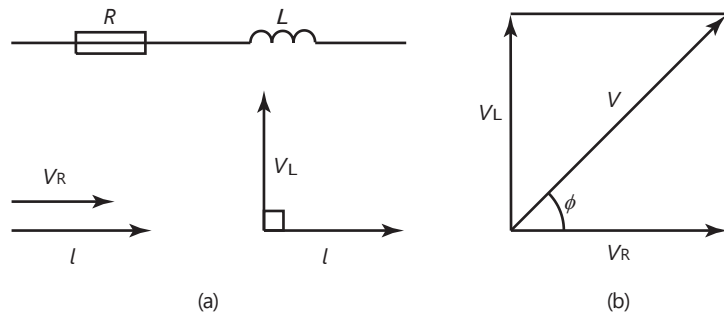
Si dos fasores están representados en tamaño y dirección por los lados adyacentes de un paralelogramo, entonces la diagonal de ese paralelogramo es la suma de los dos (Figura G.12).

Figura G.12 Suma de los fasores 1 y 2 los cuales tienen un ángulo ϕ de fase entre ellos.



Si el ángulo de fase entre los dos fasores de tamaños V_1 y V_2 es de 90° , entonces se puede calcular el resultante aplicando el teorema de Pitágoras con un tamaño V dado por $V^2 = V_1^2 + V_2^2$ y un ángulo de fase ϕ en relación con el fasor de V_1 dado por $\tan \phi = V_2 / V_1$.

Figura G.13 Circuito en serie RL.



Como ejemplo ilustrativo de lo anterior, considere un circuito de corriente alterna que tiene una resistencia en serie con inductancia (Figura G.13 a). Para un circuito de este tipo el voltaje de la resistencia está en fase con la corriente y el voltaje del inductor conduce la corriente a 90°. Por lo tanto el fasor para la suma de voltaje disminuye a través de dos componentes en serie dado por la Figura G.13b) como un fasor de voltaje con un ángulo de fase ϕ . Se puede aplicar el teorema de Pitágoras para dar la magnitud V del voltaje, es decir $V^2 = V_R^2 + V_L^2$, y trigonometría para dar el ángulo de fase ϕ ; es decir, el ángulo por el cual el voltaje conduce la corriente como $\tan \phi = V_L/V_R$ o $\cos \phi = V_R/V$.

Puesto que $V_R \leq IR$ y $V_L \leq IX_L$ entonces $V^2 = (IR)^2 + (IX_L)^2 = I^2(R^2 + X_L^2)$. El término **impedancia** Z se utiliza para la oposición de un circuito al flujo de corriente, que se ha de definir como $Z \leq V/I$ con la unidad de ohms. En consecuencia, para la resistencia e inductancia en serie, la impedancia del circuito resulta de

$$Z = \sqrt{R^2 + X_L^2} = \sqrt{R^2 + (\omega L)^2}$$



Información adicional

La siguiente es una breve lista de textos los cuales pueden ser útiles al proporcionar más información de importancia en el estudio de la mecatrónica.

Sensores y acondicionamiento de señales

- Boyes, W., *Instrumentation Reference Book*, Newnes, 2002.
- Clayton, G.B. y Winder, S., *Operational Amplifiers*, Newnes, 2003.
- Figliola, R.S., y Beasley, D.E., *Theory and Design for Mechanical Measurements*, John Wiley, 2000, 2005, 2011.
- Fraden, J., *Handbook of Modern Sensors*, Springer, 2001, 2004, 2010.
- Gray, P.R., Hurst, P.J., Lewis S.H., y Meyer, R.G., *Analysis and Design of Analog Integrated Circuits*, Wiley, 2009.
- Holdsworth, B., *Digital Logic Design*, Newnes 2000.
- Johnson, G.W. y Jennings R., *LabVIEW Grapical Programming*, McGraw-Hill, 2006.
- Morris, A.S., *Measurement and Instrumentation Principles*, 3a. ed., Newnes, 2001.
- Park J. y Mackay, S., *Practical Data Acquisition for Instrumentation and Control Systems*, Elsevier, 2003.
- Travis, J. y Kring, J., *LabVIEW for Everyone*, Prentice Hall, 2006.

Actuación

- Bolton, W., *Mechanical Science*, Blackwell Scientific Publications, 1993, 1998, 2006.
- Gottlieb, I.M., *Electric Motors and Control Techniques*, TAB Books, McGraw-Hill, 1994.
- Kenjo, T. y Sugawara, A., *Stepping Motors and their Microprocessor Controls*, Clarendon Press, 1995.
- Manning, N., *Hydraulic Control Systems*, Wiley, 2005
- Norton, R.L., *Design of Machinery*, McGraw-Hill, 2003.
- Pinches, M.J. y Callear, B.J., *Power Pneumatics*, Prentice-Hall, 1996.
- Wildi, T., *Electrical Machines, Drives and Power Systems*, Pearson, 2005.

Modelado de sistemas

- Astrom, K.J. y Wittenmark, B., *Adaptive Control*, Dover, 1994.
- Attaway, S., *Matlab: A Practical Introduction to Programming and Problem Solving*, Butterworth-Heinemann, 2009.
- Bennet, A., *Real-time Computer Control*, Prentice-Hall, 1993
- Bolton, W., *Laplace and z-Transforms*, Longman, 1994.
- Bolton, W., *Control Engineering*, Longman, 1992, 1998.
- Bolton, W., *Control Systems*, Newnes, 2002.
- D'Azzo J.J., Houpis C.H., y Sheldon, N., *Linear Control System Analysis and Design with Matlab*, CRC Press, 2003.
- Dorf, R.C. y Bishop, H., *Modern Control Systems*, Pearson, 2007.
- Fox, H., y Bolton, W., *Mathematics for Engineers and Technologists*, Butterworth-Heinemann, 2002.
- Hahn, B., *Essential MATLAB for Engineers and Scientists*, 5a. ed., Elsevier, 2012.
- Moore, H., *MATLAB for Engineers*, Pearson, 2013.

Sistemas de microprocesadores

- Arduino, www.arduino.cc
- Barnet, R.H., *The 8051 Family of Microcontrollers*, Prentice-Hall, 1994.
- Barrett, S.F., *Arduino Microcontrollers Processing for Everyone!* Morgan & Chaypool; 2013.
- Bates, M., *PIC Microcontrollers*, Newnes, 2000, 2004.
- Blum, J., *Exploring Arduino: Tools and Techniques for Engineering Wizardry*, Wiley, 2003.
- Bolton, W., *Microprocessor Systems*, Longman, 2000.
- Bolton, W., *Programmable Logic Controllers*, Newnes, 1996, 2003, 2006, 2009.
- Cady, F.M., *Software and Hardware Engineering: Motorola M68HC11*, OUP 2000.
- Calcutt, D., Cowan F. y Parchizadeh H., *8051 Microcontrollers: An Application Based Introduction*, Newnes, 2004.
- Ibrahim, D., *PIC Basic: Programming and Projects*, Newnes 2001
- Johnsonbauhg, R., y Kalinn, M., *C para científicos e ingenieros*, Prentice-Hall, 1996.
- Lewis, R.W., *Programming Industrial Control Systems using IEC 1131-3*, The Institution of Electrical Engineers, 1998.
- Morton, J., *PIC: Your Personal Introductory Course*, Newnes, 2001, 2005
- Parr, E.A., *Programmable Controllers*, Newnes, 1993, 1999, 2003.
- Pont, M.J., *Embedded C*, Addison-Wesley, 2002
- Predko., *Programming and Customizing the PIC Microcontroller*, Tab Electronics, 2007.
- Rohner, P., *Automation with Programmable Logic Controller*, Macmillan, 1996.
- Spasov, P., *Microcontroller Technology: The 68HC11*, Prentice-Hall, 1992, 1996, 2001.
- Vahid, F., y Givargis, T., *Embedded System Design*, Wiley, 2002.
- Van Sickle, T., *Programming Microcontrollers in C*, Newnes, 2001.
- Yeralan, S., y Ahluwalia, A., *Programming and Interfacing the 8051 Microcontroller*, Addison-Wesley, 1995.
- Zurell, K., *C Programming for Embedded Systems*, Kindle Edition, 2000.

Sistemas electrónicos

- Storey, N., *Electronics A Systems Approach* 5a. ed., Pearson, 2013.
- Este texto consiste de dos partes: Circuitos y componentes eléctricos y sistemas electrónicos.

En www.pearsoned.co.uk/storey-elec pueden verse recursos para los estudiantes escritos específicamente para complementar el texto. Se puede tener acceso a videos tutoriales haciendo clic en los títulos, que incluyen algunos que son especialmente importantes para la mecatrónica.

- 3A: Las leyes de Kirchhoff
- 3B: Análisis nodal
- 3C: Análisis de mallas
- 3D: Selección de las técnicas de análisis de circuitos
- 4A: Condensadores en serie y en paralelo
- 5A: Transformadores
- 5B: Aplicaciones a sensores inductivos
- 6A: Voltajes y corrientes alternos
- 6B: Uso de la impedancia compleja
- 7A: Energía en circuitos con resistencia y reactancia
- 7B: Corrección del factor de potencia para un motor eléctrico
- 8A: Diagramas de cuerpo
- 9A: Teoremas de valor inicial y final
- 9B: Determinación de la constante de tiempo de un circuito
- 10A: Selección de un motor para una aplicación dada
- 11A: Diseño de sistemas de arriba abajo
- 11B: Identificación de las entradas y salidas de un sistema
- 12A: Sensores de posición óptica
- 12B: Selección de un sensor para una aplicación dada
- 13A: Una comparación de las técnicas de exhibición
- 13B: Selección de un actuador para una aplicación dada
- 14A: La modelación de las características de un amplificador
- 14B: Ganancia de potencia
- 14C: Amplificadores diferenciales
- 14D: Especificación de un amplificador de audio
- 15A: Sistemas de retroalimentación
- 15B: Retroalimentación negativa
- 15C: Sistemas de bucle abierto y de bucle cerrado
- 16A: Circuitos básicos op-amp
- 16B: Algunos circuitos útiles adicionales op-amp
- 16C: Respuesta de frecuencia de los circuitos op-amp
- 16D: Resistencia de entrada y salida de los circuitos op-amp
- 16E: Análisis de los circuitos op-amp
- 18A: Amplificadores simples FET
- 18B: Circuito equivalente de señal pequeña de un amplificador FET
- 18C: Un amplificador de retroalimentación negativa basado en un DE MOSFET
- 18D: Un amplificador de ganancia con interruptor
- 19A: Un amplificador bipolar simple de transistores
- 19B: Análisis de un amplificador bipolar simple de transistores
- 19C: Análisis de un amplificador de retroalimentación basado en un transistor bipolar
- 19D: Un amplificador colector común
- 19E: Diseño de un divisor de fases
- 20A: Amplificadores de push-pull
- 20B: Clases de amplificadores
- 20C: Amplificadores de energía
- 20D: Diseño de un interruptor operado eléctricamente
- 21A: Circuito simplificado de un amplificador operacional bipolar
- 21B: Circuito simplificado de un amplificador operacional CMOS

21C: Circuito de un amplificador operacional 741
22A: Ruido en los sistemas electrónicos
24A: Cantidades y variables binarias
24B: Compuertas lógicas
24C: Lógica combinatoria
24D: Simplificación algebraica booleana
24E: Mapas de Karnaugh
24F: Aritmética binaria
24G: Códigos numéricos y alfabéticos
24H: Un arreglo de votación binaria para un sistema tolerante a las fallas
25A: Lógica secuencial
25B: Registradores de turnos
25C: Contadores simples
25D: Un cronómetro digital
26A: Los transistores como interruptores
26B: Familias lógicas
26C: Lógica transistor-transistor (TTL)
26D: Lógica complementaria de semiconductores de óxido de metal (CMOS)
26E: Implementación de compuertas complejas en CMOS
26F: Las interfases de compuertas lógicas de familias diferentes
26G: La disipación de energía en los sistemas digitales
26H: Cómo enfrentar problemas relacionados con el ruido en sistemas industriales
27A: Implementación de sistemas digitales
27B: Un arreglo lógico programable (PLA)
27C: Lógica de arreglos programables (PAL)
27D: Lógica de arreglos complejos
27E: Sistemas de microcomputadora
27F: Técnicas de entrada/salida
27G: Entrada/salida impulsada por interruptores
27H: Control del motor en una lavadora
28A: Muestreo
28B: Conversión digital-analógica
28C: Multiplexación
28D: Adquisición y conversión de datos en un teléfono inteligente (*smartphone*)
29A: Sistemas de comunicación
29B: Modulación analógica
29C: Modulación digital y por pulsos
30B: Un sistema de control para un brazo robótico

Respuestas

Las siguientes son las respuestas de los problemas numéricos y guías breves para las posibles respuestas de los problemas descriptivos.

Capítulo 1

- 1.1 a) Sensor: mercurio, acondicionador de señal: el conducto delgado donde se mueve el mercurio, display: las marcas en la barra, b) sensor: tubo curvado, acondicionador de señal: engranes, display: aguja moviéndose sobre la escala.
- 1.2 Vea el texto.
- 1.3 Comparación/controlador: termostato, corrección: quizás un relevador, proceso: calor, variable: temperatura, medición: dispositivo sensible a la temperatura como bandas bimetálicas, por ejemplo.
- 1.4 Vea la Figura P.1.

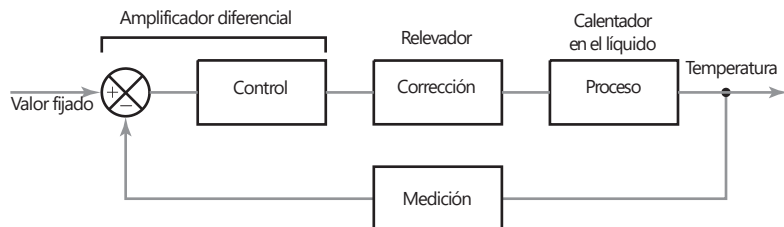


Figura P.1 Problema 1.4.

- 1.5 Vea el texto.
- 1.6 Vea el texto.
- 1.7 Por ejemplo: entra agua, enjuaga, sale agua, entra agua, calienta agua, enjuaga, sale agua, entra agua, enjuaga, sale agua.
- 1.8 Tradicional: voluminoso, funciones limitadas, requiere darle cuerda. Mecatrónico: compacto, muchas funciones, no requiere cuerda, menos costoso.
- 1.9 Termostato bimetalico: lento, precisión limitada, funciones simples, barato. Mecatrónico: rápido, preciso, muchas funciones, tiende a bajar de precio.

Capítulo 2

- 2.1 Consulte el texto para la explicación de los términos.
- 2.2 3.9%.

- 2.3 76.4 s.
 2.4 0.73%.
 2.5 0.105 &.
 2.6 Incremental: ángulo no absoluto, se mide desde una referencia; absoluto: identificación completa del ángulo.
 2.7 162.
 2.8 a) $\pm 1.2^\circ$, b) 3.3 mV.
 2.9 Vea el texto.
 2.10 2.8 kPa.
 2.11 19.6 kPa.
 2.12 0.89%.
 2.13 $+1.54^\circ \text{ C}$.
 2.14 Sí.
 2.15 9.81 N, 19.62 N, por ejemplo, un deformímetro.
 2.16 Por ejemplo: una placa de orificio con celda de presión diferencial.
 2.17 Por ejemplo: celda de presión diferencial.
 2.18 Por ejemplo: sensor de desplazamiento LVDT.

Capítulo 3

- 3.1 Como en la Figura 3.2 con $R_2/R_1 = 50$, por ejemplo, $R_1 = 1 \text{ k}\Omega$, $R_2 = 50 \text{ k}\Omega$.
 3.2 $200 \text{ k}\Omega$.
 3.3 Figura 3.5 con dos entradas, por ejemplo, $V_A = 1 \text{ V}$, $V_B = 0$ a 100 mV , $R_A = R_2 = 40 \text{ k}\Omega$, $R_B = 1 \text{ k}\Omega$.
 3.4 Figura 3.11 con $R_1 = 1 \text{ k}\Omega$ y $R_2 = 2.32 \text{ k}\Omega$.
 3.5 $V = K\sqrt{I}$.
 3.6 $100 \text{ k}\Omega$.
 3.7 80 dB.
 3.8 Fusible para protección de alta corriente, resistencia limitante para reducir corrientes, diodo para rectificar la c.a., circuito con diodo Zener para protección de voltaje y polaridad, filtro pasabajas para eliminar ruido e interferencia, optoaislador para aislar altos voltajes del microprocesador.
 3.9 0.234 V.
 3.10 $2.1 \cdot 10^{-4} \text{ V}$.
 3.11 Como se dan en el problema.

Capítulo 4

- 4.1 24.4 mV.
 4.2 9.
 4.3 0.625 V.
 4.4 1, 2, 4, 8.
 4.5 12 ms.
 4.6 Vea el texto.
 4.7 Buffer, convertidor digital-analógico, protección.
 4.8 0.33 V, 0.67 V, 1.33 V, 2.67 V.
 4.9 $32 \text{ } 768R$.
 4.10 15.35 ms.
 4.11 Factor de 315.

Capítulo 5

- 5.1 Por ejemplo: a) selección de boleto AND dinero correcto ingresado, el dinero correcto es decidido por compuerta OR entre las posibilidades, b) AND con protecciones de seguridad, lubricante, refrigerante, pieza de trabajo, potencia, etc., todo operando o en su lugar, c) Figura P.2, d) AND.
- 5.2 a) Q, b) P.
- 5.3 AND.
- 5.4 A como 1, B como 0.
- 5.5 Vea la Figura P.3.
- 5.6 Vea la Figura P.4.
- 5.7 Como en el texto, sección 5.3.1, para compuertas NOR con acoplamiento de cruz.

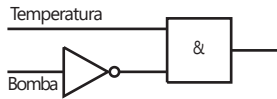


Figura P.2 Problema 5.1(c).

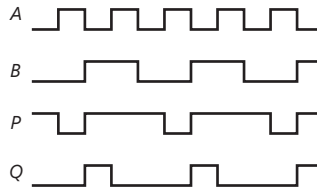


Figura P.3 Problema 5.5.

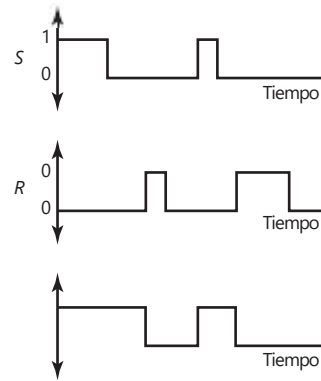


Figura P.4 Problema 5.6.

Capítulo 6

- 6.1 Vea el texto.
- 6.2 Vea la sección 6.1.
- 6.3 Por ejemplo: a) una grabadora, b) un medidor de bobina móvil, c) un disco duro o un CD, d) un osciloscopio con memoria, un disco duro o un CD.
- 6.4 Puede ser un puente de cuatro brazos, un amplificador operacional diferencial y un voltímetro como display. Los valores de las componentes dependerán del grueso elegido para el acero y del diámetro de la celda de carga. Puede elegir que el tanque se monte en tres celdas.
- 6.5 Puede ser como el mostrado en la Figura 3.8 con compensación por unión fría por un puente (sección 3.5.2). La linealidad se puede alcanzar con la selección de los materiales del termopar.
- 6.6 Pueden ser termistores con un elemento de muestreo y retención seguidos por ADC para cada sensor. Con esto la señal será digital evitando problemas de interferencia en la transmisión. Se pueden usar optoaisladores para aislar altos voltajes/corrientes, seguidos de un multiplexor que alimente a los medidores digitales.
- 6.7 Se basa en el principio de Arquímedes, el empuje hacia arriba del cuerpo flotante es igual al peso del fluido desplazado.
- 6.8 Se puede usar un LVDT o deformímetros con un puente de Wheatstone.
- 6.9 Por ejemplo: a) tubo de Bourdon, b) termistores, graficador galvanométrico, c) celdas de deformímetros, puente de Wheatstone, amplificador diferencial, voltímetro digital, d) tacogenerador, acondicionador de señal para formar pulsos, contador.

Capítulo 7

- | | | | |
|-----|---|------|---|
| 7.1 | Vea la sección 7.3. | 7.7 | 0.0057 m ² . |
| 7.2 | Vea la sección 7.3.2. | 7.8 | 124 mm. |
| 7.3 | Vea la sección 7.4. | 7.9 | 1.27 MPa, 3.9 · 10 ⁻⁵ m ³ /s. |
| 7.4 | Vea las Figuras: a) 7.14, b) 7.8b,
c) 7.10, d) 7.13. | 7.10 | a) 0.05 m ³ /s, b) 0.10 m ³ /s. |
| 7.5 | A+, B+, A, B. | 7.11 | a) 0.42 m ³ /s, b) 0.89 m ³ /s. |
| 7.6 | Vea la Figura P.5. | 7.12 | 960 mm. |

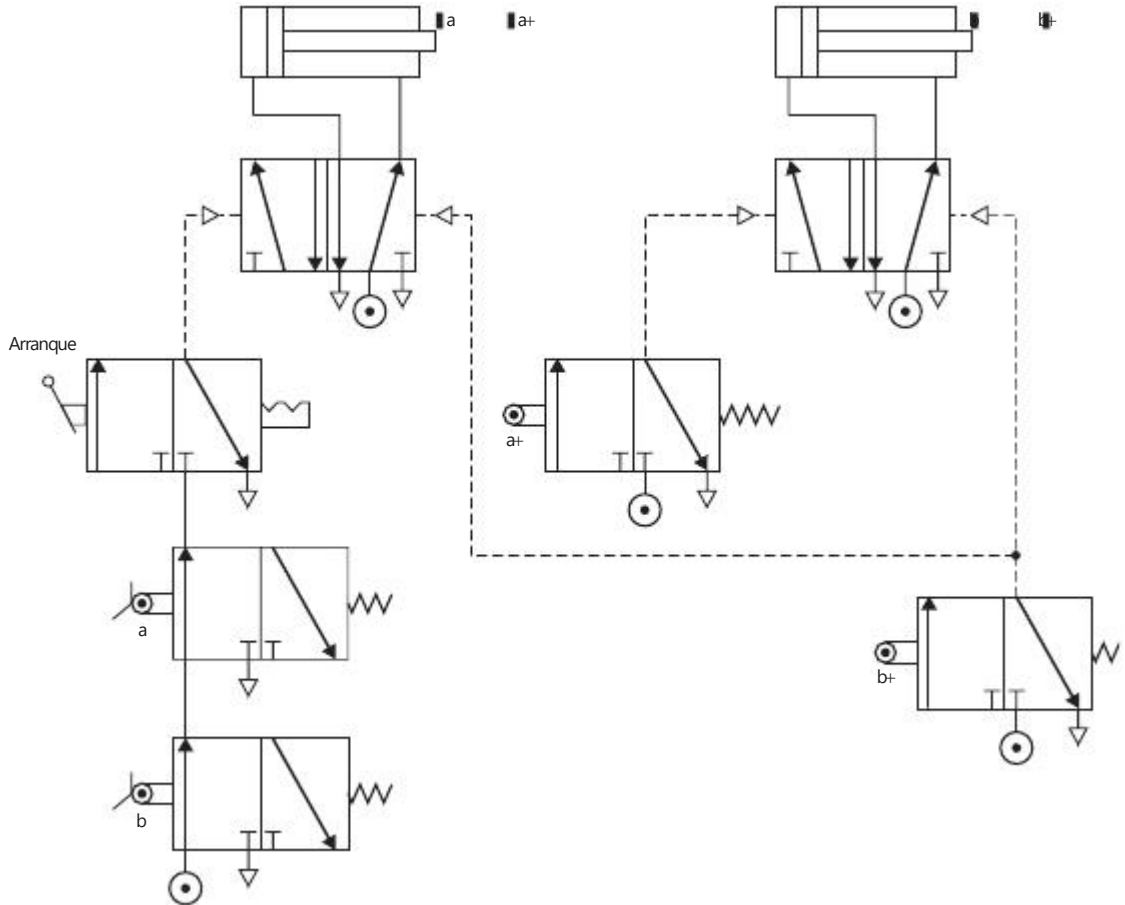


Figura P.5 Problema 7.6.

Capítulo 8

- 8.1 a) Un sistema de elementos arreglados para transmitir movimiento de una forma a otra. b) Una secuencia de eslabones y articulaciones que proporcionan una respuesta controlada a un movimiento de entrada.
- 8.2 Vea la sección 8.3.1.
- 8.3 a) 1, b) 2, c) 1, d) 1, e) 3.
- 8.4 a) Traslación pura, b) traslación pura, c) rotación pura, d) rotación pura, e) traslación más rotación.

- 8.5 Retroceso rápido.
- 8.6 Caída rápida en el desplazamiento seguida por un regreso en forma gradual.
- 8.7 60 mm.
- 8.8 Leva en forma de corazón, con distancia del eje a la parte alta del corazón de 40 mm y de 100 mm del eje a la base del corazón (Figura 8.13a).
- 8.9 Por ejemplo: a) levas en un eje, b) mecanismo de retroceso rápido, c) leva excéntrica, d) piñón y cremallera, e) transmisión de correa, f) engranes cónicos.
- 8.10 1/24.

Capítulo 9

- 9.1 Actúa como un flip-flop.
- 9.2 Vea el texto y la Figura 9.7.
- 9.3 a) Motor en serie, b) motor en derivación (en paralelo).
- 9.4 a) Corriente directa en derivación, b) motor de inducción o sincrónico con inversor, c) c.d., d) c.a.
- 9.5 Vea la sección 9.5.4.
- 9.6 Vea la sección 9.7.
- 9.7 480 pulsos/s.
- 9.8 9°.
- 9.9 a) 4 kW, b) 800 W, c) 31.8 N m.
- 9.10 0.65 N m.
- 9.11 2.
- 9.12 3.6 N m.

Capítulo 10

- 10.1 Vea la sección 10.2.
- 10.2 256.
- 10.3 64K - 8.
- 10.4 Vea la sección 10.3.
- 10.5 Vea la Figura 10.9 y el texto asociado.
- 10.6 a) E, b) C, c) D, d) B.
- 10.7 256.
- 10.8 a) 0, b) 1.
- 10.9 Vea la sección 10.3.1, inciso 6.
- 10.10 Vea la sección 10.3.2, inciso 5.
- 10.11 Alta para reiniciar la terminal.
- 10.12 a) IF A
THEN
BEGIN B
END B
ELSE
BEGIN C
END C
ENDIF A
b) WHILE A
BEGIN B
END B
ENDWHILE A

Capítulo 11

- 11.1 a) 89, b) 99.
- 11.2 No se tiene que especificar la dirección puesto que está implícita en el mnemónico.
- 11.3 a) CLRA, b) STAA, c) LDAA, d) CBA, e) LDX.
- 11.4 a) LDAA \$20, b) DECA, c) CLR \$0020, d) ADDA \$0020.
- 11.5 a) Guarda el valor del acumulador B en la dirección 0035, b) carga el acumulador A con el dato F2, c) apaga la bandera de acarreo, d) suma 1 al valor en el acumulador A, e) compara C5 con el valor en el acumulador A, f) borrar la dirección 2000, g) saltar a la dirección dada por el registro de índice más 05.
- 11.6 a)
- | | | | |
|-------|------|--------|------------------------|
| DATA1 | EQU | \$0050 | |
| DATA2 | EQU | \$0060 | |
| DIFF | EQU | \$0070 | |
| | ORG | \$0010 | |
| | LDAA | DATA1 | ; Toma el minuendo |
| | SUBA | DATA2 | ; Resta el sustraendo |
| | STAA | DIFF | ; Guarda la diferencia |
| | SWI | | ; Termina el programa |
- b)
- | | | | |
|-------|------|--------|----------------------------------|
| MULT1 | EQU | \$0020 | |
| MULT2 | EQU | \$0021 | |
| PROD | EQU | \$0022 | |
| | ORG | \$0010 | |
| | CLR | PROD | ; Borra la dirección producto |
| | LDAB | MULT1 | ; Toma el primer número |
| SUM | LDAA | MULT2 | ; Toma el multiplicando |
| | ADDA | PROD | ; Suma el multiplicando |
| | STAA | PROD | ; Guarda el resultado |
| | DECB | | ; Decrementa el acumulador B |
| | BNE | SUM | ; Salto si la suma no se termina |
| | WAI | | ; Termina el programa |
- c)
- | | | | |
|-------|------|--------|---------------------------------|
| FIRST | EQU | \$0020 | |
| | ORG | \$0000 | |
| | CLRA | | ; Limpia el acumulador |
| | LDX | #0 | |
| MORE | STAA | \$20,X | |
| | INX | | ; Incrementa el registro índice |
| | INCA | | ; Incrementa el acumulador |
| | CMPA | #\$10 | ; Compara con número 10 |
| | BNE | MORE | ; Salta si no es cero |
| | WAI | | ; Termina el programa |
- d)
- | | | | |
|------|-------|---------|---------------------------------|
| | ORG | \$0100 | |
| | LDX | #\$2000 | ; Coloca el apuntador |
| LOOP | LDA A | \$00,X | ; Carga el dato |
| | STA A | \$50,X | ; Almacena el dato |
| | INX | | ; Incrementa el registro índice |
| | CPX | \$3000 | ; Compara |
| | BNE | LOOP | ; Salta |
| | SWI | | ; Termina el programa |
- 11.7
- | | | | |
|-------|-----|--------|--|
| YY | EQU | ??? | ; Valor seleccionado para el retardo necesario |
| SAVEX | EQU | \$0100 | |
| | ORG | \$0010 | |

	STA	SAVEX	; Guarda el acumulador A
	LDAA	YY	; Carga el acumulador A
	LOOP	DECA	; Decrementa el acumulador A
		BNE	LOOP ; Salta si no es cero
		LDA	SAVEX ; Restablece el acumulador
11.8		RTS	; Regresa al programa que invocó
	LDA	\$2000	; Lee el dato de entrada
	AND A	#\$01	; Borra todos los bits excepto el bit 0
	BEQ	\$03	; Si el interruptor es bajo, salta a JMP que tiene 3 líneas de programa
	JMP	\$3000	; Si el interruptor es alto, no salta y ejecuta JMP
	Continúa		

Capítulo 12

- 12.1 a) La variable contador es un entero, b) a la variable num se le asigna el valor 10, c) la palabra name será presentada en el display, d) el display es Number 12, e) incluir el archivo stdio.h
- 12.2 a) Llama las librerías necesarias para ejecutar la función printf(), b) indica el principio y el final de un grupo de instrucciones, c) inicia una nueva línea, d) problema 3.
- 12.3 El número es 12.
- 12.4 # include <stdio.h>

```
int main(void);
{
    int len, width;
    printf("ingrese longitud: ");
    scanf("%d", &len);
    printf("ingrese ancho: ");
    scanf("%d", &width);
    printf("Area es %d, len * width);
    return 0;
}
```

- 12.5 Similar al programa dado en la sección 12.3, inciso 4.
- 12.6 Divide el primer número entre el segundo a menos que éste sea cero.

Capítulo 13

- 13.1 Vea la sección 13.3.
- 13.2 Vea la sección 13.3. Una interfaz en paralelo tiene el mismo número de líneas de entrada/salida que el microprocesador. La interfaz en serie tiene sólo una línea de entrada/salida.
- 13.3 Vea la sección 13.2.
- 13.4 Vea la sección 13.4.
- 13.5 Vea la sección 13.4 y la Figura 13.10.
- 13.6 Vea la sección 13.4.1.
- 13.7 Vea la sección 13.3.3. El poleo requiere la interrogación de todos los periféricos a intervalos frecuentes, aun cuando algunos no estén activados, por lo que

es una pérdida de tiempo. Interrupt se inicia a solicitud del periférico y por lo tanto es más eficiente.

- 13.8 CRA 00110100, CRB 00101111.
- 13.9 Como en el programa en la sección 13.4.2, con LDAA #\$05 reemplazado por LDAA #\$34 y LDAA #\$34 reemplazado por LDAA #\$2F.
- 13.10 Como en el programa en la sección 13.4.2 seguido de READ LDAA \$2000 ; Lee el puerto A
Quizá después de un retardo el programa debería ser BRA READ.

Capítulo 14

- 14.1 a) AND, b) OR
- 14.2 a) Figura 14.9b, b) Figura 14.10b, c) un circuito de enclavamiento, Figura 14.16, con los interruptores Entrada 1 para inicio y Entrada 2 para alto.
- 14.3 0 LD X400, 1 LD Y430, 2 ORB, 3 ANI X401, 4 OUT Y430.
- 14.4 0 LD X400, 1 OR Y430, 3 OUT Y430, 4 OUT T450, 5 K 50; temporizador de retardo.
- 14.5 0 LD X400, 1 OR Y430, 2 ANI M100, 3 OUT Y430, 4 LD X401, 5 OUT M100; restablece el enclavamiento.
- 14.6 Como en la Figura 14.28 con el temporizador 1 con $K = 1$ para 1 s y el temporizador 2 con $K = 20$ para 20 s.
- 14.7 Figura P.6.
- 14.8 Figura P.7.
- 14.9 Figura P.8.

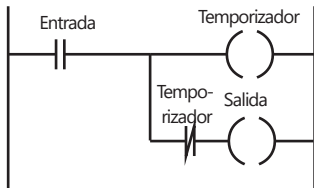


Figura P.6 Problema 14.7.

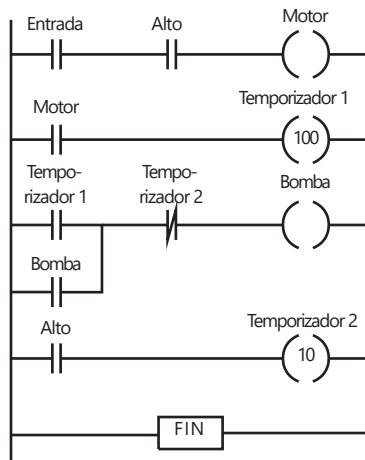


Figura P.7 Problema 14.8.

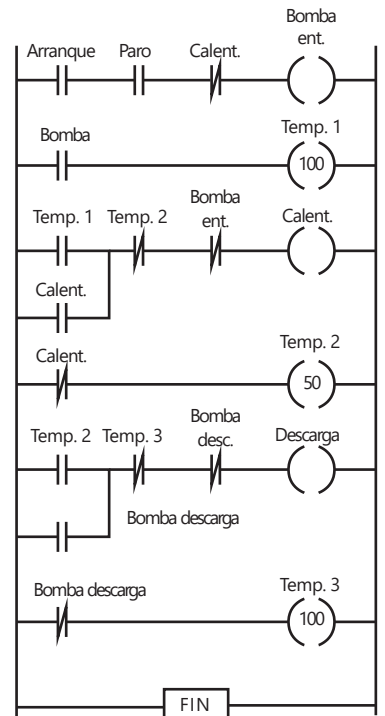


Figura P.8 Problema 14.9.

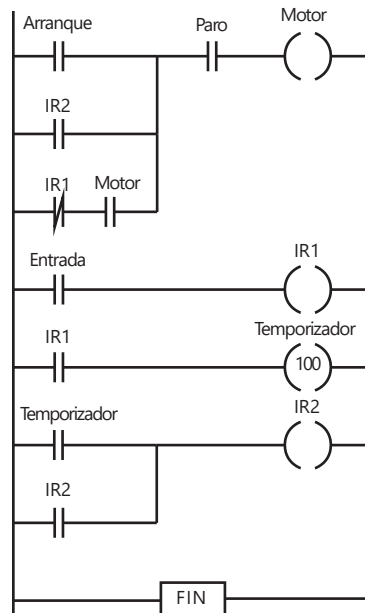


Figura P.9 Problema 14.10.

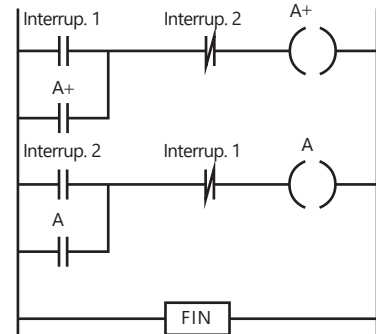


Figura P.10 Problema 14.13.

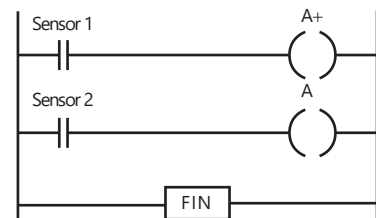


Figura P.11 Problema 14.14.

- 14.10 Figura P.10.
 14.11 Una salida comenzaría, como antes, pero se apaga al presentarse la siguiente entrada.
 14.12 Vea la sección 14.10.
 14.13 Dos circuitos latch, como en la Figura P.10.
 14.14 Figura P.11.

Capítulo 15

- 15.1 Vea la sección 15.2.
 15.2 Vea la sección 15.3.
 15.3 Bus.
 15.4 Banda ancha.
 15.5 Vea la sección 15.5.1.
 15.6 Vea la sección 15.4.
 15.7 Vea la sección 15.3.1.
 15.8 NRFD a PD0, DAV a STRA e IRQ, NDAC a STRB, datos al puerto C.
 15.9 TTL a conversión del nivel de señal RS-232C.
 15.10 Vea la sección 15.7.1.

Capítulo 16

- 16.1 Vea la sección 16.1.
 16.2 Vea la sección 16.2.
 16.3 Vea la sección 16.2.
 16.4 Vea la sección 16.5.3 para autoprueba y prueba de la suma de la verificación programables y la sección 16.2 para un temporizador vigilante.
 16.5 Vea la sección 16.5.3.

Capítulo 17

17.1 (a) $m \frac{d^2x}{dt^2} + \frac{dx}{dt} + (k_1 + k_2)x = F$ (b) $\frac{dx}{dt} + (k_1 + k_2)x = F$

17.2 Como en la Figura 17.3a).

17.3 $c \frac{du}{dt} + ku_b$

17.4 Dos resortes torsionales con un bloque para momento de inercia,

$$T = I \frac{d^2u}{dt^2} + (k_1 + k_2)u$$

17.5 $v = v_R + \frac{1}{RC} \int v_R dt$

17.6 $v = \frac{L}{R} \frac{dv_R}{dt} + \frac{1}{CR} \int v_R dt + v_R$

17.7 $v = R_1 C \frac{dv_C}{dt} + \frac{R_2}{R_1} v_C$

17.8 $RA_2 \frac{dh_2}{dt} + h_2 rg = h_1$

17.9 $RC \frac{dT}{dt} + T = T_r$. Capacitor cargado descargando a través de una resistencia.

17.10 $RC \frac{dT_1}{dt} = T_1 - 2T_2 + T_3$

17.11 $pA = m \frac{d^2x}{dt^2} + \frac{1}{C} x$, $R =$ resistencia al movimiento del vástago,
 $c =$ capacitancia del resorte.

17.12 $T = a \frac{I_1}{d^2u} + \frac{nc_2b}{c_1} \frac{du}{dt} + \frac{nk_2bu}{k_1}$

Capítulo 18

18.1 $\frac{IR}{k_1 k_2} \frac{dv}{dt} + v = \frac{1}{k_2} v$

18.2 $(L_a + L_L) \frac{di_a}{dt} + (R_a + R_L)i_a - k_1 \frac{du}{dt} + k_2 i_a = T$

18.3 Igual que un motor controlado por armadura.

Capítulo 19

- 19.1 $4 \frac{dx}{dt} + x = 6y$
- 19.2 a) 59.9° C, b) 71.9° C.
- 19.3 a) $i = \frac{V}{R} (1 - e^{-Rt/L})$, b) L/R , c) V/R .
- 19.4 a) Oscilaciones continuas, b) subamortiguada, c) críticamente amortiguada, d) sobreamortiguada.
- 19.5 a) 4 Hz, b) 1.25, c) $i = I_{A1} e^{-8t} - \frac{4}{3} e^{-2t} + I_0$.
- 19.6 a) 5 Hz, b) 1.0, c) $x = (-32 + 6t)e^{-5t} + 6$.
- 19.7 a) 9.5%, b) 0.020 s.
- 19.8 a) 4 Hz, b) 0.625, c) 1.45 Hz, d) 0.5 s, e) 8.1%, f) 1.4 s.
- 19.9 a) 0.59, b) 0.87.
- 19.10 2.4.
- 19.11 0.09.
- 19.12 3.93 rad/s, 0.63 Hz.

Capítulo 20

- 20.1 (a) $\frac{1}{ms^2 + cs + k}$, (b) $\frac{1}{ms^2 + cs + k}$, (c) $\frac{1}{LCs^2 + RCs + 1}$
- 20.2 a) 3 s, (b) 0.67 s
- 20.3 (a) $1 + e^{-2t}$, (b) $2 + 2e^{-5t}$
- 20.4 a) Sobreamortiguada, b) subamortiguada, c) críticamente amortiguada, d) subamortiguada.
- 20.5 $t e^{-3t}$
- 20.6 $2e^{-4t} - 2e^{-3t}$
- 20.7 (a) $\frac{4s}{s^2(s+1)+4}$, (b) $\frac{2(s+2)}{(s+1)(s+2)+2}$,
 (c) $\frac{4}{(s+2)(s+3)+20}$, (d) $\frac{2}{s(s+2)+20}$

>

$$(s + 2)(s + 3) + 20$$

- 20.8 $5s(s + 53)$
 20.9 $5s(s^2 + s + 10)$
 20.10 $2s(3s + 1)$
 20.11 $-1, -2$
 20.12 a) Estable, b) inestable, c) inestable, d) estable, e) inestable.

Capítulo 21

- 21.1 a) $\frac{5}{2v^2 + 4} \frac{v}{2}$, b) $\frac{2}{22v^4 + v}$
 c) $\frac{1}{24v^6 - 3v^4 + 3v^2 + 1} \frac{v(3 - 2v^2)}{1 - 3v^2}$
 21.2 $0.56 \sin(5t - 38^\circ)$
 21.3 $1.18 \sin(2t + 25^\circ)$
 21.4 a) i) $1, 90^\circ$, ii) $0.44, 450^\circ$, iii) $0.12, 26.6^\circ$, iv) $0, 0^\circ$,
 b) i) $1, 0^\circ$, ii) $0.32, -71.6^\circ$, iii) $0.16, -80.5^\circ$, iv) $0, -90^\circ$.
 21.5 Veá la Figura P.12.
 21.6 a) $1, s$, b) $3.2s(1 + s)$, c) $2.0s(s^2 + 2zs + 1)$,
 d) $3.2s[(1 + s)(0.01s^2 + 0.2zs + 1)]$

Capítulo 22

- 22.1 Veá la sección 22.3.
 22.2 a) 8 minutos, b) 20 minutos.
 22.3 a) 12 s, b) 24 s.
 22.4 5.
 22.5 Veá el texto. En particular P offset, PI y PID no offset.
 22.6 3, 666 s, 100 s.
 22.7 3, 100 s, 25 s.
 22.8 Veá las secciones a) 22.12.1, b) 22.12.2, c) 22.12.3.
 22.9 1.6.
 22.10 Respuesta de primer orden con una constante de tiempo c/K_P .

Capítulo 23

- 23.1 Por ejemplo, intente el diámetro y grado de rojo. También puede considerar el peso. Sus resultados necesitan poder distinguir claramente entre denominaciones de monedas, cualquiera que sea su condición.

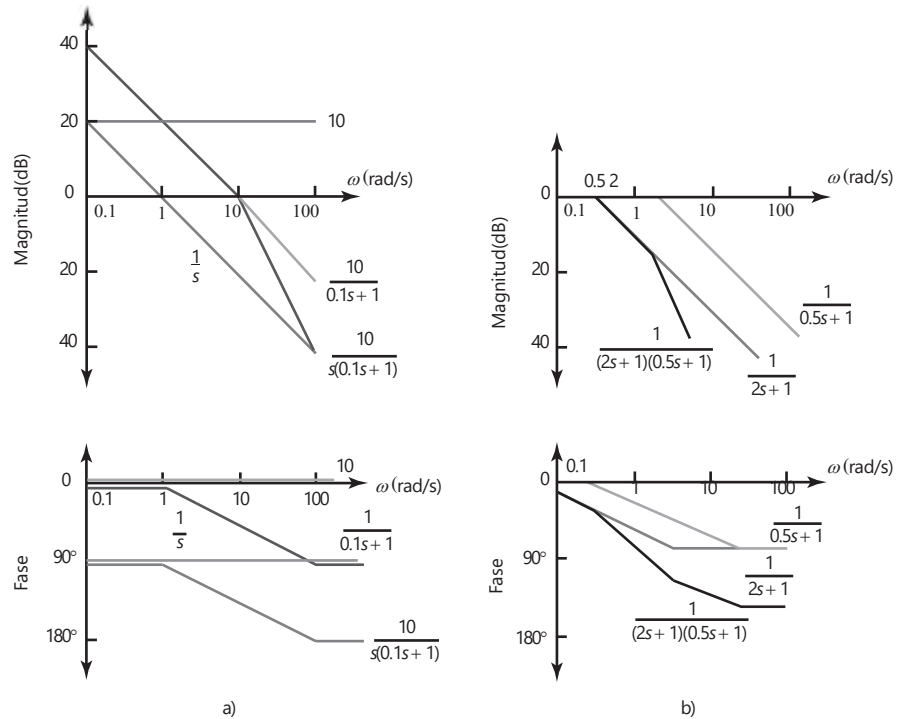


Figura P.12 Problema 21.5.

- 23.2 a) 1/6, b) 1/36, c) 1/10.
 23.3 0.99.
 23.4 0.002.
 23.5 0.625.
 23.6 0.761.
 23.7 Por ejemplo, si la temperatura en una habitación es $< 20^\circ\text{C}$ y el temporizador está en ON, el calefactor está encendido; si el calentador está en ON, la bomba está encendida; si la bomba está en ON y la temperatura de la habitación es $< 20^\circ\text{C}$, entonces la válvula está abierta, pero si el temporizador no está en ON el calentador tampoco lo está. Si la temperatura en la habitación no es $< 20^\circ\text{C}$ es que la válvula no está en ON, y si el calentador tampoco está en ON, la bomba no está en ON. Esto también se puede precisar si se considera que debe haber una limitante de que el calentador esté restringido a funcionar por debajo de 60°C .

Capítulo 24

- 24.1 Las posibles soluciones son: a) termopar, compensación de unión fría, amplificador, ADC, PIA, microprocesador, DAC, unidad termistora para controlar el elemento de calor del horno, b) sensores de los haces de luz, PLC, deflectores

para clasificador operados por solenoides, c) control de lazo cerrado con un motor de c.d. para el movimiento en cada dirección, un motor de c.d. como actuador del movimiento de la pluma, microprocesador como comparador y controlador, y realimentación de un codificador óptico.

Tareas de investigación

Las siguientes son descripciones breves del tipo de información que debe contener una respuesta.

- 24.2 Un sistema de frenos ABS tiene sensores de tipo inductor detectando la velocidad de cada una de las ruedas del vehículo, acondicionadores de señal para convertir las señales de los sensores en pulsos de 5 V, un microcontrolador con un programa para calcular la velocidad de las ruedas y la tasa de desaceleración durante el frenado para que cuando se exceda un límite prefijado, el microcontrolador dé una salida a las válvulas solenoide en el modulador hidráulico para prevenir un incremento en la fuerza de frenado o para reducirla en caso necesario.
- 24.3 El motor del carro mueve la cabeza de impresión en forma lateral, mientras imprime los caracteres. Después de imprimir una línea, el motor de alimentación del papel lo corre. La cabeza de impresión consiste en agujas movidas por solenoides, por lo general un renglón de 9, que impactan en la cinta de tinta. Un microcontrolador se puede utilizar para controlar las salidas. Para mayores detalles, vea *Microcontroller Technology: The 68HC11* de P. Spasov (Prentice Hall, 1992, 1996).
- 24.4 El bus CAN opera con señales que tienen un bit de inicio seguido por el nombre que indica el destino del mensaje y su prioridad, seguido por bits de control, seguidos por los datos que se envían, seguidos por bits de CRC, seguidos por bits de confirmación de la recepción y concluye con los bits de final.

Tareas de diseño

Las siguientes son descripciones breves de las posibles soluciones.

- 24.5 Un termómetro digital que usa un microprocesador tiene un sensor de temperatura como el LM35, un ADC, un ROM como el Motorola MCM6830 o el Intel 8355, una RAM como la Motorola MCM6810 o la Intel 8156, un microprocesador M6800 o Intel 8085A y un display de LED. Con un microcontrolador como el Motorola MC68HC11 o el Intel 8051 puede tenerse un sólo sensor de temperatura, quizá con un acondicionador de señal y un display de LED.
- 24.6 Un óhmmetro digital involucra un multivibrador monoestable que proporciona un pulso con una duración de 0.7RC. Una variedad de capacitores fijos se usan para proporcionar diferentes rangos de resistencias. El intervalo puede determinarse usando un microcontrolador o un microprocesador, más memoria y enviarlos a un display de LED apropiado.
- 24.7 Esto puede requerir un sensor de presión, por ejemplo, el semiconductor transductor Motorola MPX2100AP, un acondicionador de señal para convertir la pequeña señal del sensor al nivel apropiado, por ejemplo, un amplificador instrumental, usando amplificadores operacionales, un microcontrolador, por ejemplo el MC68HC11, un controlador de LCD como el MC145453 y un display LCD de cuatro dígitos.
- 24.8 Esto puede resolverse utilizando el M68HC11EVM con una salida con modulación por ancho de pulso (PWM) al motor. Cuando se desea realimentación se puede utilizar un codificador óptico.
- 24.9 Se puede hacer que cada caja sea cargada utilizando una corriente que se aplica a una válvula solenoide, la cual opera un cilindro neumático que a su vez

mueve una placa que empuja la caja a la tolva o canal de descarga. La caja permanece en el canal que está cerrado por otra tolva. La presencia de la caja es detectada por un sensor que indica que la siguiente caja puede ser admitida en el canal. Esto continúa hasta que se acumulan cuatro cajas en el canal. La placa al final del canal se abre, activada por otra válvula solenoide, para permitir la salida de las cajas a la banda. La llegada de las cajas a la banda puede detectarse con un sensor montado en el extremo del escape. Esto permite que el proceso completo se pueda repetir.

Apéndice A

- A.1 a) $2/s^2$, b) $2(s^2 + 4)$, c) e^{2s} , d) $sX(s) - 2$, e) $3s^2X(s)$, f) $1/[s(s + 1)]$.
- A.2 a) t , b) $5 \cos 3t$, c) $1 + 2e^t$, d) e^{3t} .
- A.3 5.

Apéndice B

- B.1 255.
- B.2 a) 11, b) 529.
- B.3 a) 1A7, b) 211.
- B.4 a) 781, b) 157.
- B.5 a) 1010 0110, b) 1101 1101.
- B.6 a) 0, b) 1.
- B.7 a) 8, b) 12.

Apéndice C

- C.1 a) (a) $A(B)$, (b) $A(B)$, c) $A(B)$, d) $A(B)$
- C.3 $A(B)$

A	B	C	Q
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

- C.4 a) (a) $Q = C(A + D)$, (b) $Q = C(A + D)$
- C.5 Como se da en el problema.
- C.6 a) (a) $Q = A + B$, (b) $Q = C$
- C.7 $Q = A + B$
- C.8 Cuatro compuertas de entrada AND con dos compuertas NOT si la combinación correcta es 1, 1, 0, 0: $Q = A \bar{B} \bar{C} \bar{D}$



Índice

A

acceso a redes por paso por ranura 380
acceso a redes por paso por token 380
acceso indirecto en C 315
acceso múltiple con detección de portador con detección de colisión (CSMA/CD) 380
acceso múltiple con detección de portador impedimento de colisión (CSMA/CA) 394
acción de surgimiento de corriente en compuertas lógicas 120
 acondicionamiento de señal 69-70
acoplamiento de inercia en motores 234-5
actuador de diafragma 180
actuadores giratorios 185
actuadores semigiratorios 185
adaptador de Motorola MC6850 341, 406
adaptador de Motorola MC6852 382
adaptador síncrono de datos en serie (SSDA) 382
adaptadores de interfaz para comunicación asíncrona (ACIA) 341-2
adaptadores de interfaz periféricos (PIA) 336-41
en el movimiento de limpiaparabrisas 541
inicialización 339
señales de interrupción 339-40
adquisición de datos (DAQ) 106-8
detección de errores 108
exactitud 107-8
registradores de datos 154
tarjetas con clavijas 151-3
álgebra Booleana 587-95
leyes de De Morgan 588-9
leyes del 587-8
mapas de Karnaugh 591-4
tablas de verdad 589-91
algoritmo 274
Ambiente de desarrollo Arduino 322
American National Standards Institute (ANSI) 302

American Standard Code for Information Interchange (ASCII) 140
amortiguador de entrada 251
amortiguador rotacional 416
ecuaciones 417
amortiguadores 414-16
ecuaciones de bloques funcionales 417
amperímetros, calibración de 159
amplificador de instrumentación 76
amplificador diferencial 74-8
amplificador integrador 73-4
amplificador inversor 71-2
amplificador logarítmico 78
amplificador no inversor 72
amplificador operacional 70-81
amplificador operacional en circuitos PID 517
amplificador sumador 72-3
amplificadores de muestreo y retención 105
amplificadores integradores y diferenciadores 73-4
amplificadores reales 80-1
análisis de circuitos de c.a. 615-19
circuitos en serie 618-19
resistencia, inductancia y capacitancia en 617-18
análisis de circuitos eléctricos 610-15
análisis de mallas 611-12
análisis de nodos 610-11
teorema de Norton 614-15
teorema de Thévenin 612-14
circuitos de c.a. 615-19
resistencia, inductancia y capacitancia en 617-18
análisis de malla de circuitos 611-12
análisis de mallas 611
analizador de firma para localización de fallas 404
analizador lógico de fallas 404
ancho de banda 501
ANSI (American National Standards Institute) 302
aprendizaje 533-4
aproximación de Tustin 110
aproximaciones sucesivas en ADC 101
apuntadores en C 315-16

aritmética con 315
y arreglos 315
árboles de probabilidad condicionada, 531
archivos de biblioteca estándar en C 304
archivos de encabezado en C 317
Arduino
programas 319-3
robots 566-7
tarjetas 270-271, 319
amadura en motores de cd 218
arquitectura de Harvard 263-4
arquitectura de microprocesadores 244, 246
Arquitectura de redes de sistemas (SNA) 384
arquitectura de von Neumann 242, 264
arquitectura del sistema Motorola 6800 247, 252-3
arquitectura Intel 8085A 248
arreglos en C 313-14
y apuntadores 315
arreglos multidimensionales en C 314
articulaciones 191
asignaciones en C 306
atenuación 83
aterrizado de señales 89-91
autoenfoco en cámaras digitales 22-3
autopueba para la localización de fallas 405
autorregulación 528

B

banda de paso 83
banda muerta de los transductores 32
bandas de transmisión 200-1
tipos de 201
bandas en V 201
bandas planas 201
bandas redondas 201
bandas reguladoras de tiempos 201
barreras (plumas) de estacionamiento 551
básculas de baño 545-8

- BASIC 278
bit de paridad 108
bit más significativo (MSB) 95, 581
bit menos significativo (LSB) 95, 581
Bit User Fast Friendly Aid to Logical Operations (Buffalo) 406
bits 95
bits de verificación de redundancia cíclica 381
bloques funcionales de sistemas hidráulicos 426-8
ecuaciones 430
bloques funcionales de sistemas mecánicos 414-22
ecuaciones 417
sistemas rotacionales 416-17
y analogías eléctricas 425-6
bloques funcionales de sistemas térmicos 433-5
construcción de 434-5
ecuaciones 434
bloques funcionales en sistemas de fluidos 426-33
construcción de los 430-3
ecuaciones de los 430
bloques funcionales en sistemas eléctricos 422-6
construcción 424-5
ecuaciones 423
y analogías mecánicas 425-6
bomba de engranajes 166, 167
bomba de pistón 166
bomba de pistón axial 166
bomba de pistón radial 166, 167
bomba de vástago 166, 167
borrador maestro y restablecimientos en los PIC 268
brazo de robot, funciones de transferencia 478
búferes 329, 330-1
búferes de tres estados 330-1
bus CAN 389
bus de arquitectura de microcanal (MCA) 393
bus de arquitectura estándar con norma industrial extendida (EISA) 393
bus de computadora AT 393
bus de computadora XT 393
bus de control 243
bus de datos 242-3
bus de direcciones 243
bus de instrumentación de Hewlett Packard 391
bus I²C 387-8
bus LAN Ethernet 380
bus S 393
bus TURBOcanal 393
bus VME 393-4
bus VXI 394
buses
en sistemas de microprocesadores 241-3
buses de la matriz de conectores 393-4
- C**
- cadena cinemática de cuatro barras 192-3
cadena de palancas 193
cadenas cinemáticas 191-4
de cuatro barras 192-3
mecanismo de biela-corredera-cigüeñal 193-4
cadenas en C 307-8
cadenas 202
cámara digital 552-7
enfoque automático 23
camino cerrado de tierra 90-1
campo de dirección 286
campo en lenguaje ensamblador 285-86
capa de aplicación en OSI 383
capa de enlace de datos en OSI 82
capa de presentación en OSI 383
capa de red en OSI 383
capa de sesión en OSI 383
capa física en OSI 382
capa transportadora en OSI 383
capacidad de rango 183
capacitancia hidráulica 426
ecuaciones 430
capacitancia neumática 428
ecuaciones 430
capacitancia pura en circuitos de a.c. 617
capacitancia térmica
ecuaciones 434
capacitores 422-3
ecuaciones 423
características estáticas de los transductores 33-5
características principales 303-8
asignaciones 306
cadenas 307-8
comentarios 305
función principal 305
funciones 303
funciones de biblioteca estándar 304
instrucciones 303
operaciones sobre bits 307
operadores aritméticos 306
operadores de relación 306
operadores lógicos 307
palabras clave 303
preprocesador 304-5
retorno 304
secuencias de escape 308
variables 305-6
carga de sistemas mecánicos 191
carga, en pantallas 136-7
carretera de datos de Allen-Bradley 384
celda de bits 143
celda de carga con extensómetro 49-50
celdas de carga
calibración de 159
en báscula de baño 545
en sistemas de medición 155-6
ciclo de programa en los P.L.C. 353
ciclo de trae-ejecuta 246
cicloconvertidor en motores de corriente alterna 227
Ciclos
en C 310-13
instrucción *for* 310-11
instrucción *switch* 312-13
instrucción *while* 311-12
ciclos 309-13
instrucción *for* 309-10
instrucción *switch* 311-13
instrucción *while* 311
ciclos de retardo 291
cilindro hidráulico 175
cilindro neumático 175
cilindros 175-8
secuenciación 177-8
cinemática 189
circuito amortiguador 212
circuito amortiguador o de frenado 212
circuito del interruptor de temperatura 79
circuito en escalera R-2R 99
circuitos de enclavamiento en los PLC 361
circuitos en serie 618-19
circuitos de c.d. 610-15
análisis de mallas 611-12
análisis de nodos 610-11
teorema de Norton 614-15
teorema de Thévenin 612-14
circuitos en serie de ca 618-19
circuitos H 222-3, 229-30
circuitos integrados 118
y compuertas lógicas 119-20
circuitos PID 517
circuitos RLC 462
codificación de fase (PE) 143-4
codificación de longitud de corrida limitada (RLL) 144-5, 146
codificación de modulación de frecuencia modificada (MFM) 144
codificación de no regreso a cero (NRZ) 143
codificador 122
codificadores absolutos 42

- codificadores absolutos de 3 bits 43
- codificadores incrementales 42, 46
- codificadores ópticos 42-3
- código binario decimal (BCD) 122-4
- código de máquina 278
- código de operación 284
- código Gray 43, 585-6
- código mnemónico 278
- coeficiente de Hall 45
- cojinetes 202-4
 - selección de los rodamientos 204
- cojinetes autoalineables 203
- cojinetes con lubricación hidrodinámica 202
- cojinetes con lubricación hidrostática 202
- cojinetes de bola de contacto angular 203
- cojinetes de bolas de empuje de ranura de carrera 203
- cojinetes de bolas de ranura de relleno 203
- cojinetes de bolas y rodillos 203
- cojinetes de deslizamiento de capa límite 203
- cojinetes de deslizamientos 202-3
- cojinetes de doble hilera 203
- cojinetes de empuje 202
- cojinetes de rodamiento 202-3
- cojinetes de rodamiento de película sólida 203
- cojinetes de rodillo cilíndrico 204
- cojinetes de rodillo cónico 204
- cojinetes de rodillos 204
- comparador 79-80
- comparador digital 121-2
- compensación de temperatura en el puente de Wheatstone 85-7
- compensación del termopar en el puente de Wheatstone 88
- compensación en las funciones de transferencia 482-3
- complemento a dos en matemáticas binarias 583
- complemento a uno en matemáticas binarias 583
- compresor de acción simple 168
- compresor de doble acción 168
- compresor de una etapa 168
- compresor rotativo 168, 169
- compresor rotativo helicoidal 169
- compresor simple de acción 168
- compresores 169
- compuerta AND 112, 113
- compuerta NAND 115-16, 118
- compuerta NOR 116-17, 118
- compuerta NOT 115
- compuerta OR 114-15
- compuerta XOR 117
- compuertas lógicas 112, 113-20
 - aplicaciones 120-6
 - convertidor de código 122-6
 - codificador 122
 - comparador digital 121-2
 - generadores de paridad de bit 120-1
 - compuerta AND 112, 113-14
 - combinación 117-19
 - compuerta NAND 115-16, 118
 - compuerta NOR 116-17, 118
 - compuerta NOT 115
 - compuerta OR 114-15
 - compuerta XOR 117
 - familias de 119-20
- computadora que opera apropiadamente (COP) 259, 398
- comunicaciones digitales 376
- condición de Grashof 193
- condicionador de señal 8
- condiciones en estado estable
 - en sistemas de segundo orden 462
 - y respuesta transitoria 480
- conexiones de ánodo y cátodo común para LED 148-9
- conjuntos de instrucciones 598-9
- conmutador en motores de cd 217
- constante de tiempo de los transductores 34
- constante de tiempo en sistemas de primer orden 456-8
- constante integral de tiempo 515
- consumo de potencia de las compuertas lógicas 120
- contadores en los PLC 365
- contadores en los PLC 365-7
- contadores hacia arriba en los PLC 366
- control autosintonizable 525
- control autosintonizable en control adaptativo 525
- control calendarizado de ganancia 524-5
- control de acceso a redes 380
- control de computadora centralizado 376
- control de enlace de datos de alto nivel (HDLC) 382
- control de enlace lógico 383
- control de flujo y ciclos 309-13
 - instrucción *If/else* 310
 - instrucción *If* 309-10
- control de la interrupción 333
- control de la relación aire-combustible (AFR) 558
- control de paridad 146
- control de realimentación 10
- control de temperatura vía PLC 372-3
- control de velocidad 523
- control de voltaje 211
- control del motor de paso 537
- control del robot 549
- control del tren de potencia 557
- control del voltaje de cd por 211
- control digital 112
- control digital 505
- control en cascada
 - de los cilindros, 178, 179
 - en los PLC 364
- control lambda 558-9
- controlador de interfaz periférico (PIC) 263
 - arquitectura de Harvard 263-4
 - conjunto de instrucciones 600
 - direccionamiento 282
- familia 272
 - Intel 8051 272
 - entrada de cristal 268
 - borrador maestro y restablecimientos 268
 - puerto esclavo paralelo 268
 - entrada/salida serial 268
 - puertos de entrada/salida 265-6
 - entradas análogas 266-7
 - temporizadores 267-8
 - M68HC11 273
- programas 288-9
- registros de propósito especial 268-9
- controlador lógico programable (PLC) 21-2, 349
- circuitos enclavados 361
 - como monitor de sistemas 408-9
 - contadores 365-7
 - detección de fallas en el 397
 - diagramas de escalera 355
 - entrada/salida análoga 371-3
 - en barreras (plumas) de estacionamiento 551
 - manejo de datos 369-71
 - comparación de datos 370-1
 - conversiones de código 371
 - movimiento de datos 370
 - operaciones aritméticas 371
 - procesamiento de entrada/salida 353-4
- direccionamiento 354
- actualización continua 353
- copiado masivo de entrada/salida 353-4
 - en interruptor temporizado 537-8
 - estructura 349-53
- formas de 352-3
- entrada/salida 350-2
- programas de entrada 352
 - funciones lógicas 356-8
 - listas de instrucciones 358-60
 - programas en escalera 354-8
- relevadores internos 362
- relevadores maestros 368-9
- saltos 369

- secuenciación 363-4
 registros de corriente 367-8
 temporizadores 364-5
 controlador proporcional electrónico 511-2
 controladores de tres términos (PID) 509, 516-17
 controladores digitales 519-17
 modos de control 518-19
 controladores en lazo cerrado 505, 506-7
 control adaptable 523-6
 control de velocidad 523
 controladores digitales 517-19
 desempeño 520-1
 error en estado estable en los 507-8
 modo derivativo 509, 512-14
 modo en dos posiciones 509-10
 modo integral 509, 514-16
 modo PID 509, 516-17
 modo proporcional 509, 510-12
 modos de control 509
 para lector de disco duro 562
 respuesta del sistema 512
 retardadores 507
 sintonización 521-3
 controladores secuenciales 19-21
 conversión analógica a digital (ADC) 16, 95, 98, 101, 104
 conversión digital a analógica (DAC) 98
 interfaz, 347
 conversiones de código en los PLC 371
 convertidor 122-4
 en los LED 149
 convertidor analógico a digital (ADC) en controles digitales 517
 tasa de muestreo 519
 teorema de muestreo 97
 convertidor de código 122-6
 convertidor de rampa doble 103
 convertidor en motores de c.a. 227
 convertidores digitales a analógicos (DAC) 16, 99-101
 copiado masivo de entradas/salidas en los PLC 353-4
 corriente de desbalance de entrada 81
 corriente polarizada de entrada 81
 corrientes de vuelta a tierra 90
 criterio de Nyquist 97
 cronómetro 555 132-3, 540
 curva de reacción de proceso 521-2
- D**
- datos de protocolo 381
 datos enclavados (latched) 130
 datos retenidos 130
 decodificación de datos 22
 decodificador de 3 líneas a 8 líneas 124-5
 decodificador de BCD a decimal 122-4
 decodificador de dirección 250
 derivadas en la transformada de Laplace 577
 desarrollo de un programa 316-17
 archivos de encabezado 316
 destello ADC 104
 detección de errores
 método de la paridad para 108
 paridad par en 108
 paridad impar en 108
 verificación de suma en 108
 detectores de la temperatura de la resistencia (RTD) 56-7
 devanado de campo en motores de c.d. 218
 diagrama de cuerpo libre 417, 419
 diagrama de flujo 274-5
 diagramas de bloque en MATLAB 608
 diagramas de Bode 492-500
 construcción de los 496-8
 identificación del sistema 498-500
 para $G(s) = 1/s$ 493
 para $G(s) = K$ 492-3
 para sistemas de segundo orden 494-6
 para sistemas de primer orden 493-4
 diagramas de Bode para 493-4
 como sistemas dinámicos 452-8
 constante de tiempo 456-8
 constante de tiempo en 456-8
 ejemplos de 454-6
 entrada escalón en los 474
 funciones de transferencia en 474-6
 ejemplos de 475-6
 entrada escalón 474
 ganancia en estado estable en 457-8
 respuesta de frecuencia en 489-91
 respuesta forzada 453-4
 respuesta forzada en los 453-4
 respuesta natural 452-3
 respuesta natural en 452-3
 transductores como 454-5
 transformada de Laplace en los 474
 diagramas de escalera en los PLC 355
 dientes de engrane helicoidal 197
 dientes helicoidales 197
 diferencia de fase en circuitos de c.a. 616
 diferencia en matemáticas binarias 583
 diodo de protección 214
 diodo Zener 81
 diodos 209-10
 de protección o flyback 208
 diodos de rueda libre 208
 diodos emisores de luz (LED) 147, 148-9
 diodos emisores de puntos 148, 150
 direccionamiento absoluto 283
 direccionamiento de página cero 283
 direccionamiento del registro 283
 direccionamiento directo 283, 284
 direccionamiento directo de memoria (DMA) 107
 direccionamiento extendido 283
 direccionamiento implicado o inherente 283
 direccionamiento indexado 283, 284
 direccionamiento indirecto 283
 direccionamiento inherente o implicado 283, 284
 direccionamiento inmediato 282-3, 284
 direccionamiento relativo 283
 discos duros 145-6
 discos formateados 146
 discos magnéticos 145-6
 diseño cinemático 190
 disparador Schmitt 64
 display de siete segmentos, decodificación 344-7
 displays
 efecto de la carga 136-7
 poner en blanco 345
 para presentación de datos 136
 displays de lámpara de neón 147
 displays de lámpara incandescente 147
 dispositivo acoplado por carga (CCD) 62, 552
 dispositivos de interrupción como actuadores 207
 dispositivos de tipo solenoide como actuadores 207
 dispositivos periféricos 249
 dominio de s en la transformada de Laplace 571
 propiedad de corrimiento 576
 dominio del tiempo en la transformada de Laplace 571
 drenado de corriente (*sinking*) en los PLC 351
 duplicaciones 97
- E**
- ECL (emisor acoplado lógico) 119
 ecuación Booleana 113
 ecuación en diferencias en el procesamiento de señales digitales 109
 ecuaciones auxiliares 460
 ecuaciones diferenciales 418, función forzada 449
 EEPROM (PROM eléctricamente borrrable) 249
 programas embebidos 297

efecto Hall 45
 ejes de engranes paralelos 197
 elemento correcto en sistemas en lazo cerrado 13
 elemento de comparación de los sistemas en lazo cerrado 12
 elemento de proceso de los sistemas en lazo cerrado 13
 elementos de control de los sistemas en lazo cerrado 12-13
 elementos de medición de los sistemas de lazo cerrado 13
 emisor acoplado lógico (ECL) 119
 emulación en la localización de fallas 405-6
 en sistemas de primer orden 489-91
 en sistemas de segundo orden 491-2
 especificaciones de desempeño 501-2
 estabilidad 502-3
 para entrada sinusoidal 486
 enfoque secuencial del proceso de diseño 6
 engranaje de corona 196
 engranes 196-9
 engranes con doble hilera de dientes helicoidales 196
 engranes cónicos 196
 engranes helicoidales 197
 entrada de cristal en los PIC 267-8
 entrada de impulso en sistemas dinámicos 451-2
 entrada de paso
 en sistemas de primer orden 474
 en sistemas de segundo orden 477
 en sistemas dinámicos 451-2, 464
 entrada senoidal
 fasores en 487-9
 ecuaciones 488-9
 en sistemas dinámicos 451-2
 respuesta de frecuencia para 486
 entrada/salida
 direccionamiento de 326-9
 en los PLC 350-2
 actualización continua 353
 copiado masivo de entradas/salidas 353-4
 direcciones 354
 entrada/salida analoga 371-3
 procesamiento 353-4
 pruebas en los PLC 407-8
 puertos en los PIC 264-6
 registros 327-9
 entrada/salida aislada 326
 entrada/salida analógica
 de los PIC 266, 268
 de los PLC 371-373
 entrada/salida de actualización continua en los PLC 353

entrada/salida de memoria mapeada 327
 entrada/salida de reconocimiento completo 332
 entrada/salida en serie de los PIC 268
 entrada/salida serial 262
 EPROM (ROM borrable y programable) 248-9, 250
 programas embebidos en 297
 error de cuantización 97
 error de histéresis de los transductores 31
 error de linealidad 104
 error de no linealidad de los transductores 31
 error en estado estable en controladores en lazo cerrado 507-8
 corrimiento 508
 errores
 en controladores de lazo cerrado 507-8
 en terminología del funcionamiento 30
 esbozo en programas Arduino 320
 escaneo de no enlazado en tubos de rayos catódicos 139-40
 estabilidad de los transductores 32
 estructura del programa 302-9
 estructura del programa en C 302-8
 etapa conceptual del proceso de diseño 5
 etiquetas 286
 exactitud de transductores 30
 extensómetro patrón inactivo 86
 extensómetros, calibración del factor de medición 159

F

factor de amortiguamiento relativo 460
 falla de un componente pasivo 402
 falla del chip 402
 fallas de hardware 400-1
 fallas de software 402
 fallas por corto circuito 402
 fasores
 en entrada sinusoidal 487-8
 ecuaciones 488-9
 filtrado 83
 filtro antiduplicación 98
 filtro de paso alto 83
 filtro de paso bajo 83
 filtro de paso de banda 83
 filtro pasivo 83-4
 filtros 83-4
 firmware 249
 flip-flop 126-8
 flip-flop D 129-31
 flip-flop de compuerta SR 128
 flip-flop JK 128-9

flip-flop SR 127-8
 flotadores 55
 forma de rampa del ADC 104
 formato JPEG 554
 FORTRAN 278
 fotodiodos 61
 fotorresistor 61
 fototransistores 61
 fototransistores de Darlington 61
 fracciones parciales en la transformada de Laplace 578-9
 frecuencia de corte 83
 frecuencia de esquina en diagramas de Bode 494
 frecuencia modulada (FM)
 codificación 144
 fuelle 431
 sensores de presión absoluta 51
 fuente de señal aterrizada 90
 fuente de señal flotante 90
 fuerza contraelectromotriz en motores de cd 219
 función de muestreo en los microcontroladores 261
 función de paso unitario en la transformada de Laplace 572-3
 función de transferencia 512
 función de pertenencia 532-3
 función de pulso en la transformada de Laplace 573-4
 función de pulso unitario en la transformada de Laplace 573
 función de respuesta de frecuencia 489
 función de transferencia de frecuencia 489
 función delta de Dirac en la transformada de Laplace 573
 función Heaviside en la transformada de Laplace 572
 función lógica AND en los PLC 356, 360
 función lógica NOR en los PLC 356, 360
 función lógica OR en los PLC 356, 360
 función lógica XOR en los PLC 357, 360
 funciones de biblioteca 601-3
 funciones de transferencia 471-4
 compensación 482-3
 diagramas de Bode para 492-6, 500
 en controladores digitales 518
 en sistemas de primer orden 474-6
 ejemplos de 475-6
 entrada de paso 474
 en sistemas de segundo orden 476-8
 ejemplos de 478
 entrada de paso 477
 polos 481

plano de s 482
 sistemas en serie 478-9
 ejemplos de 479
 transformada de Laplace
 471, 472-4
 funciones en lenguaje C 303
 funciones lógicas en los PLC 356-8
 y listas de instrucciones en los PLC
 359
 funciones periódicas en la transformada
 de Laplace 576-7

G

ganancia de corriente en transistores
 bipolares 213
 ganancia de respuesta de frecuencia 490
 ganancia de voltaje 70
 ganancia en estado estable en sistemas
 de primer orden 457-8
 generador de c.a. 47
 generadores de bit de paridad 120-1
 grabación magnética 142-6
 códigos para 143-5
 grabación óptica 146-7
 grabadoras analógicas registradoras 138
 grabadores 136
 grados de libertad 190

I

identificación del sistema 498-500
 identificación del sistema en sistemas
 dinámicos 467
 impedancia de entrada 70
 impedancia de salida 70
 impedancia de salida de los
 transductores 33
 impedancia en circuitos de c.a. 619
 impresoras 141-2
 impresoras de inyección de tinta 141
 impresoras de matriz de puntos 141
 indicador de presiones 49
 indicadores 136
 inductancia en circuitos de c.a. 617-18
 inductancia pura en circuitos de a.c. 617
 inductores 422-3
 ecuaciones 423
 inercia hidráulica 427
 ecuaciones 430
 inercia neumática 429
 ecuaciones 430
 inercia, momentos de 234-5, 416-17
 inspección para localización de fallas
 402-3

inspección por generador de pulsos
 lógicos para localización de fallas 403
 inspección por multímetro para
 localización de fallas 402
 inspección por osciloscopio para
 localización de fallas 402
 inspección visual para localización
 de fallas 402
 instrucción for
 en lenguaje C 310-11
 instrucción *if* en C 309-10
 instrucción *if/else* en C 310
 instrucciones de computadora 278
 instrucciones en C 303
 instrumentos virtuales 152
 integrales en la transformada de
 Laplace 577
 Inteligencia artificial
 autorregulación 528
 Intercambio de paquete de interred
 (IPX) 383
 Intercambio de paquete secuenciado
 (SPX) 383
 Interconexión de sistemas abiertos
 (OSI) 382-3
 interfaces 326
 análoga 347
 búfers 330-1
 interrupciones 332-5
 pantalla de siete segmentos 344-6
 PIA 336-39
 ejemplo 341
 inicialización 339
 señales de interrupción 339-40
 requerimientos 329-30
 interfaces de comunicación paralela
 391-4
 interfaz GPIB 391, 392-3
 interfaces de comunicación serial
 341-4, 385-90
 interfaces de entrada 241
 interfaces de salida 241
 interfaz de bus de instrumentos de
 propósito general (GPIB) 391,
 392-3, 394
 interfaz 385-7
 interfaz de bus I²C 387-8
 interfaz de ciclo de corriente de
 20 mA 387
 interfaz firewire 390
 interfaz serial 336
 interfaz serial de los microcontroladores
 343-4
 interfaz USB 390
 interferencia electromagnética en las
 señales 91-2
 interrupciones 250, 332-5
 interrupciones 263

interruptor accionado con leva 20, 537
 interruptor de lengüeta 44
 interruptor de proximidad inductivo 42
 interruptor límite 63
 en barreras (plumas) de
 estacionamiento 551
 interruptor temporizador 537-40
 interruptores
 de estado sólido 209-15
 fallas en los 401
 introducción de datos con 63-5
 mecánicos 207-9
 relevadores 208-9
 interruptores de estado sólido 209-15
 diodos 209-10
 MOFSET 215
 tiristores y triacs 210-12
 transistores bipolares 212-15
 interruptores de proximidad 44-5
 interruptores enmascarables 334
 interruptores mecánicos 63, 207-9
 de estado sólido 209-15
 relevadores 208-9
 intervalo de cuantización 97
 intervalo de transductores 30
 IP (Protocolo de Internet) 383
 IPX (Intercambio de paquete de
 interred) 383

J

juegos de instrucciones 278, 279-85
 aritmética 279-80
 direccionamiento 282-5
 Intel 8051, 598-9
 lógicas 280-1
 Motorola M68HC11 596-7
 movimiento de datos 284-5
 PIC16Cxx 600
 transferencia y movimiento
 de datos 279

L

LabVIEW 152
 lanzamientos de interruptores 63
 lavadora doméstica 273-4
 lazo de corriente de 20 mA 387
 lazos de realimentación 12, 479-80
 lector de código de barras 559-61
 lectores de código de barras 561
 lectura de un canal ADC 318-20
 cómo encender y apagar un motor
 317-18
 lenguaje C 278

arreglos 313-14
 y apuntadores 314
 ramificaciones 309-10
 instrucción *if/else* 310
 instrucción *if* 309-10
 ventajas de 302
 lenguaje ensamblador 278
 ejemplos de 287-90
 programas 285-90
 subrutinas 290-3
 tablas de consulta 293-6
 lenguajes 278
 levas 194-6
 seguidores 194, 196
 levas 195
 levas en forma de corazón 195-6
 levas en forma de pera 195-6
 ley de Kirchhoff de la corriente 609
 ley de Kirchhoff del voltaje 611
 ley de metales intermedios 60
 ley de temperaturas intermedias 60
 ley del paralelogramo de vectores en circuitos de c.a. 618
 leyes de De Morgan 588-9
 leyes de Kirchhoff 424-5
 libertad, grados de 190
 linealidad en sistemas de ingeniería 443-5
 líneas de control en las PIA 337
 listas de instrucciones en los PLC 358-60
 y funciones lógicas 359
 y ramificaciones 359-60
 localización de fallas 397
 localización de fallas 405
 localización de fallas de la entrada a la salida 405
 localización de fallas de la salida a la entrada 405
 localización de fallas por división a la mitad 405
 lógica CMOS (semiconductor complementario de metal-óxido) 119, 120
 lógica combinacional 112
 lógica del semiconductor complementario metal-óxido (CMOS) 119, 120
 lógica difusa 532-3
 lógica digital 112
 lógica secuencial 112, 126-33
 flip-flop 126-8
 flip-flop D 129-31
 flip-flop SR 127-8
 registros 131
 sistemas síncronos 128
 temporizador 555, 132-3

M

magnitud o ganancia 490
 manejo de datos en los PLC 369-70
 comparación de datos 370-1
 conversiones de código 371
 movimiento de datos 70
 operaciones aritméticas 371
 manipulación de señal 70
 mantisa en números de punto flotante 585
 mapas de Karnaugh 591-4
 máquinas 533-4
 máquinas inteligentes 528
 margen de fase 503
 margen de ganancia 503
 masas 414, 415
 ecuaciones 417
 matemáticas binarias 582-4
 MATLAB 604-8
 diagramas de bloque 608
 funciones de transferencia 607-8
 graficación 606-7
 mecanismo de doble cigüeñal 193
 mecanismo de doble palanca 193
 mecanismo de palanca-cigüeñal 193-4
 mecanismo de retroceso rápido 194
 mecanismos 188-9
 mecatrónica
 diseños 537-48
 orígenes 3
 proceso de diseño 5, 6
 básculas de baño 545-8
 interruptor temporizado 537-40
 movimiento del limpiaparabrisas 541-5
 sistemas
 barreras (plumas) de estacionamiento 551-2
 cámara digital 552-7
 ejemplos de 4, 22-5
 lector de código de barras 559-61
 robot para levantar y depositar objetos 548-51
 sistema de control automotriz 557-9
 unidad de disco duro 561-3
 y diseños tradicionales 6
 medición de temperatura, salida binaria de la 157-8
 medidor de movimiento de bobina 137
 medidores análogos en presentación de datos 137, 138
 medidores análogos y digitales en presentación de datos 137-8
 Melsec-Net 384
 memoria 241

en microprocesador 246-9
 EEPROM 249
 EPROM 248-9
 PROM 248
 RAM 249
 ROM 247-8
 MEMS (sistemas microelectromecánicos) 25
 método de paridad para detección de error 108
 método de reacción de proceso de la sintonización 521-2
 método del último ciclo de sintonización 522-3
 microcontrolador 16C74A 263-5
 microcontrolador 16F84 263-5
 microcontrolador Atmel VR 270-1
 microcontrolador Intel 8051, 260-3
 ALE 261
 direccionamiento 282
 entrada externa 261
 puertos de entrada/salida 260-1
 microcontroladores 4, 242, 253-54
 aplicaciones 272-4
 lavadora doméstica 273-4
 sistema de medición de temperatura 272-3
 Intel 8051 260-3
 ALE 261
 entrada/salida serial 262
 interrupción 263
 PSEN 261
 puertos de entrada/salida 260-1
 RESET 262
 temporizador 262-3
 XTAL 262
 interfaces en serie de 343-4
 Motorola M68HC11 254-60
 controlador de interrupción 258
 COP 259
 modos 257-60
 puertos 255-7
 PWM 259-60
 temporizador 258-9
 terminales del oscilador 258
 selección 271-2
 sistemas embebidos 296-9
 término de selección 261
 microcontroladores de microchip 263-69
 microinterruptor 44
 microprocesadores 4, 241, 243-6
 arquitectura 244-6
 registros 244-6
 unidad aritmética y lógica (ALU) 244
 unidad de control 244
 microprocesadores embebidos 241
 microsensores en MEMS 25

- minuendo en matemáticas binarias 582
- modelos matemáticos 413-14
- modo de comunicación duplex
 - completo 379
- modo de comunicación semidúplex 379
- modo de comunicación simplex 378
- modo de control de dos pasos 509-10
- modo de control integral 509, 514-16
 - con modo proporcional 515-16
- modo de control PID 509, 516-17
- modo de control proporcional
 - 509, 510-12
 - con control derivativo 513-14
 - con modo integral 515-16
 - con respuesta de sistema 512
- modos de comunicación de datos
 - en serie 378-9
 - modo dúplex completo
 - o bidireccional 379
 - modo semidúplex o unidireccional 379
 - modo simplex, 378
- modos de puerto en serie 344
- modulación de amplitud de pulso 89
- modulación de ancho de pulso (PWM) 89, 221-2
- modulación de pulso 88-9
- modulación por amplitud de pulso 89
- MOFSET 215
- momentos de inercia 416-17
 - ecuaciones 417
 - en sistema de piñón y cremallera 439-40
- mosaico de Bayer 554
- motor bipolar 229
- motor de vástago 185
- motor paso a paso de imán permanente 228
- motor paso a paso de reluctancia variable 227-8
- motor paso a paso híbrido 228
- motor unipolar 230-1
- motores
 - fallas en los 400
 - selección de 234-7
 - acoplamiento de inercia 234-5
 - requerimientos de par motor 235-6
 - requerimientos de potencia 236-7
- motores de inducción de c.a.
 - monofásicos de jaula de ardilla 225
- motores bobinados en serie 220
- motores compuestos de c.d. 221
- motores controlados por armadura 441-2
 - realimentación negativa en 480
- motores controlados por campo 443
- motores de arrollamiento en derivación 220-1
- motores de c.a. 225
- motores de c.a. de inducción
 - de tres fases 226
- motores de c.d. 217-25
 - como sistemas electromecánicos 440-3
 - funciones de transferencia en 479
- motores de c.d. de excitación independiente 221
- motores de cd con escobillas 217-19
 - control de 221-3
 - con devanado de campo 220-1
- motores de cd de imán permanente y sin escobillas 224-5
- motores de minipasos 232
- motores paso a paso 227-34
 - características 229-30
 - control 229-32
 - en cámaras digitales 555
 - en limpiaparabrisas 541-3
 - especificaciones 229
 - minipasos 232
 - retardo con 295-6
 - selección de 233-4
- motores síncronos 226
- Motorola 6800 286
- Motorola Freescale M68HC11
 - conjuntos de instrucciones 596-7
 - controlador de interrupción 258
 - COP 259
 - interfaz de comunicación serial 336
 - interrupciones 335
 - microcontrolador 254-60
 - modos 257-8
 - modos de direccionamiento 282
 - programa 287
 - puertos 255-7
 - PWM 259-60
 - reconocimiento 331-2
 - temporizador 258-9
 - terminales del oscilador 258
- movimiento de datos en lenguaje ensamblador 284-5
- movimiento de los limpiaparabrisas 541-5
 - adaptador de interfaz periférico en el 541
 - motor paso a paso en 541-3
- movimiento de traslación 189
- señales hidráulicas del 445
- movimiento rotacional 189
 - señales hidráulicas de 445
- movimiento rotacional a traslacional 199
- movimiento, tipos de 189
- multiplexión de división de tiempo 106
- multiplexores 105-6
- multiplexores digitales 105-6
- multivibrador monoestable 132-3

N

- neumática 165
- nivel de señal 70
- nivel lógico 119
- niveles de cuantización 97
- nodo principal 610
- nodos 191
 - análisis de circuitos de nodos 610-11
 - en redes 379
- notación científica 585
- Novell Netware 383
- NuBus 393
- números binarios 95, 581-2
- números binarios con signo 583
- números binarios sin signo 583
- números de punto flotante 585
- números normalizados de punto flotante 585

O

- operación unipolar de DAC 100
- operaciones aritméticas en los PLC 371
- operaciones basadas en el reloj 505
- operadores aritméticos en C 306
- operadores de relación en C 306-7
- operadores lógicos en C 307
- operadores sobre bits en C 307
- operando 282
- optoaislador Darlington 82
- optoaislador transistor 82
- optoaisladores 82
- osciloscopio de rayos catódicos 138-9

P

- palabra 95
- palabras clave en C 303
- pantallas alfanuméricas 147
- pantallas de cristal líquido (LCD) 149-51
- paridad impar 108
- paridad par en la detección de errores 108
- PASCAL 278
- película de fluoruro de polivinilideno (PVDF) 53-4
- percepción 528-9
- pila 290
- piñón de engranes 196
- pixeles 140
- placa oscilante 167

- plano de s en funciones de transferencia 482
- poleo 19, 332
- poleo 332
- polos de campo en motores de c.d. 218
- polos de interruptores 63
- polos en funciones de transferencia 481
- posición angular de la rueda de la polea 157
- posiciones de palanca 193
- potencia en números de punto flotante 585
- potenciómetro rotacional 440
- precisión de los transductores 30
- preprocesadores en C 304-5
- presentación de datos
 - elementos 137-42
 - medidores análogos y digitales 137-8
 - registros en tablas análogas 138
 - osciloscopio de rayos catódicos 138-9
 - impresoras 141-2
 - unidad de exhibición visual 139-41
 - exhibidores para 136-7, 147-51
 - grabación magnética 142-6
 - discos 145-6
 - grabación óptica 146-7
 - sistemas de medición
- presión diferencial 55-6
- presión manométrica 180
- primera propiedad de corrimiento de la transformada de Laplace 576
- principio de la restricción mínima 190
- principio de superposición 77, 443
- procesamiento de señales digitales 109-10
- ecuación de diferencias en 109
- procesamiento de señales en tiempo discreto 109
- procesos basados en eventos 505
- producto de sumas en álgebra booleana 589
- programa 20, 278
- programa de ejemplo 308-9
- programación 274-6
- programación en escalera en los PLC 354-8
 - funciones lógicas 356-8
- programas 288
- PROM (memoria programable de solo lectura) 248
- propiedad de corrimiento de la transformada de Laplace 576
- propiedad de corrimiento del dominio del tiempo en la transformada de Laplace 576
- propiedad de linealidad de la transformada de Laplace 576
- protección 69, 81-2
- protocolo Bisync 382
- Protocolo de automatización de la manufactura (MAP) 383-4
- Protocolo de control de transmisión de internet (TCP) 383
- Protocolo de Internet (IP) 383
- Protocolo técnico y de oficina (TOP) 384
- protocolos inalámbricos 394-5
- prueba 158
- prueba de preinstalación 158
- prueba de tuberías 158
- prueba de tuberías y cable 158
- prueba de verificación de la suma 405
- prueba del programa en los PLC 407
- PSEN 261
- punto de Wheatstone 84-7
 - calibración del 159-60
 - compensación de temperatura 85-7
 - compensación del termopar 88
 - en básculas de baño 545-6
- punto de Wheatstone balanceado 84-5
- puerto paralelo esclavo IN PICS 268
- punto de rompimiento en los diagramas de Bode 494

R

- radio en números de punto flotante 585
- RAM (memoria de acceso aleatorio) 249, 251, 253
- ramificación y listas de instrucción en los PLC 359-60
- rampa 451-2
- razón de decaimiento o decremento en sistemas dinámicos 465
- razón de rechazo al modo común (CMRR) 76, 90
- razón de transferencia 82
- razonamiento 530-3
 - basado en la regla 531-3
 - mecanismo para el 530-1
- razonamiento basado en la regla 531-3
- razonamiento determinístico 530
- razonamiento no determinístico 530
- reactancia capacitiva en c.a.
 - circuitos 618
- reactancia inductiva en circuitos de ca 617
- reajuste en controladores en lazo cerrado 508
- realimentación 9-11
- realimentación de la unidad 508
- realimentación de velocidad 523
- realimentación negativa 12, 479
- ejemplos de 480
- rebotes de interruptores 63-4
- receptor/transmisor asincrónico universal (UART) 341
- reconocimiento (handshaking) 329, 331-2
- reconocimiento de muestreo y aceptación 331
- reconocimiento de patrones 529
- rectificador controlado por silicio (SCR) 210
- red de área ancha (WAN) 380
- red de área de controlador red (CAN) 558
- red de área local (LAN) 380
- red de resistores ponderados 99
- redes 379-81
 - normas 383-4
 - topología 379-80
- redes neurales 529-30
- registro acumulador 244
- registro de control en las PIA 337
- registro de apuntador de instrucciones 245
- registro de bandera 245
- registro de código de condición 245
- registro de corrimiento 367
- registro de direccionamiento de memoria 246
- registro de estado 245
- registro de instrucciones 246
- registro de interface periférica en PIA 336
- periféricos 326
- registro de interfaz para periféricos PIA 336
- registro de salida 252
- registro del apuntador de pila 246
- registro del contador de programa 245
- registros de corrimiento en los PLC 367-8
- registros de datos 131
- registros de función especial 263
- registros de propósito especial 269-70
- registros de propósito general 246
- registros en sistemas de microprocesador 244-6
- regla de Bayes 531, 533
- regla de Ziegler-Nichols 525
- relación aire-combustible (AFR) 558
- relevadores 208
- relevadores 208-9
 - fallas en 400
- relevadores auxiliares en los PLC 362
- relevadores de tiempo de retardo 209
- relevadores internos en los PLC 362
- relevadores maestros en los PLC 368-9
- relevadores marcadores en los PLC 362
- repetibilidad de transductores 31
- reproducibilidad de los transductores 31-2

- requerimientos del par motor de los motores 235-6
 - requerimientos 329-30
 - buffering/aislamiento 329
 - control del temporización 329
 - conversión de código 329
 - serial a paralelo 329-30
 - interfaz en serie 336
 - para comunicaciones en serie 341-3
 - requerimientos de potencia de los motores 236-7
 - RESET 262
 - resistencia en circuitos de c.a. 617-18
 - resistencia hidráulica 426
 - ecuaciones 430
 - resistencia neumática 428
 - ecuaciones 430
 - resistencia pura en circuitos de a.c. 617
 - resistores 423
 - ecuaciones 423
 - resolución 97
 - de los transductores 32
 - en los ADC 104
 - resonancia pico 501
 - resortes 414-15
 - ecuaciones 417
 - resortes torsionales 416
 - respuesta
 - en sistemas de primer orden 452-3
 - respuesta libre 450-1
 - respuesta de frecuencia 489-92
 - diagramas de Bode 492-500
 - construcción 496-8
 - para sistemas de primer orden 493-4
 - para $G(s) = 1/s$ 492
 - para $G(s) = K$ 492-3
 - para sistemas de segundo orden 494-6
 - respuesta forzada
 - en sistemas de primer orden 453-4
 - en sistemas dinámicos 450-1
 - respuesta transitoria y localización de polo 480-3
 - respuestas en estado estable en sistemas dinámicos 451
 - respuestas transitorias en sistemas dinámicos 451
 - restricción mínima, principio de 190
 - retardo con motores paso a paso 295-6
 - retornos en C 304
 - retrasos 507
 - robot para levantar y depositar objetos 548-51
 - robótica 563-7
 - aplicaciones 566
 - componentes 565-6
 - robot Arduino 566-7
 - tres leyes de 564-5
 - ROM (memoria de solo lectura) 247-8, 253
 - RS-232 385-7
 - rueda loca o guía 198
 - ruido 70
 - inmunidad en las compuertas lógicas 119
 - ruta de realimentación 479
- S**
- salida binaria de medición de temperatura 157-8
 - saltos condicionales en los PLC 369
 - saltos en los PLC 369
 - saturación en transistores bipolares 213
 - Schottky TTL 120
 - SCR (rectificador controlado por silicio) 210
 - secuenciación en los PLC 363-4
 - secuencias de escape en C 308
 - seguidor de leva de punta 196
 - seguidor de leva de rodillo 196
 - seguidor de leva deslizante 196
 - seguidor de leva en cuchillo 196
 - seguidor de leva en forma de hongo 196
 - seguidor de leva oscilatorio 196
 - seguidor de leva plana 196
 - seguidor de voltaje 72
 - segundo teorema de corrimiento en la transformada de Laplace 576
 - semántica en los protocolos para la transmisión de datos 381
 - sensibilidad a la carga 52
 - sensibilidad de los transductores 30-1
 - sensibilidad de voltaje 53
 - sensor 8, 29-30
 - fallas en los 400-1
 - selección de los 62-3
 - terminología de funcionamiento 30-5
 - sensor de placa de orificio 54-5
 - sensor inductivo 551
 - sensor medidor de turbina 55
 - sensor táctil 53-4
 - sensores de capacitancia 38-9
 - sensores de desplazamiento 35
 - sensores de desplazamiento asimétrico 39
 - sensores de flujo de líquido 54
 - sensores de fuerza 49-50
 - sensores de galga extensométrica 37-8
 - sensores de luz, 61-2
 - sensores de movimiento 45
 - sensores de nivel de líquido 55-6
 - sensores de posición 35
 - sensores de presión 50-5
 - calibración de los 158
 - sensores de presión de fluido 50-4
 - sensores de presión de fluido de la cápsula 50
 - sensores de proximidad 35
 - sensores de proximidad de corrientes parásitas 41
 - sensores de temperatura 56-61
 - calibración de los 159
 - sensores de toque 565-6
 - sensores de velocidad 46-9
 - sensores del potenciómetro 35-7
 - sensores fotosensitivos 44
 - sensores Hall 224
 - sensores inteligentes 29-30
 - sensores neumáticos 43-4
 - sensores piezoeléctricos 52-3
 - sensores piroeléctricos 48-9
 - sensores de efecto Hall 45-6
 - señales
 - filtrado 83-4
 - problemas con 89-92
 - interferencia electromagnética 91-2
 - puesta a tierra 89-91
 - transferencia de potencia 92
 - tipo adecuado de 69-70
 - señales de interrupción a través de PIA 339-40
 - señales digitales 95
 - señales síncronos 128
 - seudocódigo 274, 275-6
 - seudoperaciones 286
 - simulación en la localización de fallas 406-7
 - SIMULINK 608-9
 - sintaxis en los protocolos para la transmisión de datos 381
 - sintonización de controladores 521-3
 - método de la curva de reacción del proceso 521-2
 - método de la última ganancia 522-3
 - sistema basado en la regla 531
 - sistema de alarma 131
 - sistema de alarma de temperatura 156-7
 - sistema de comunicación jerárquica 377
 - sistema de control automatiz 557-9
 - sistema de control de computadora 519
 - sistema de control de personas 558
 - sistema de control vehicular 558
 - sistema de mando motor 23-4
 - sistema de medición de temperatura 272-3
 - sistema de parámetros concentrados 414
 - sistema de piñón y cremallera 199, 439-40
 - sistema de tornillo y tuerca 199
 - sistema decimal codificado en binario (BCD) 585
 - sistema decimal codificado en código binario (BCD) 581-2

- sistema numérico hexadecimal 581-2
- sistema numérico octal 581-2
- sistema torsional 463
- sistemas 6-8
 - con ciclos de realimentación 479-80
 - en serie 478-9
 - ejemplos de 479
- sistemas adaptables basados en un modelo de referencia 526
- sistemas alambrados 13
- sistemas asíncronos 128
- sistemas conectados 8
- sistemas críticamente amortiguados 460, 463-4
- sistemas de actuación 165
- sistemas de actuación mecánica 188-9
 - movimiento, tipos de 189-91
 - carga 191
 - libertades y restricciones 190-1
- sistemas de control 9-21
 - analógicos 16-19
 - digitales 16-19
 - en lazo abierto 11-12
 - en lazo cerrado 11-12
 - elementos de 12-15
 - retroalimentación 9-11
 - secuencial 19-21
- sistemas de control adaptables 523-6
 - control de ganancia preprogramada 524-5
 - de control autosintonizable 525
 - sistemas de modelo de referencia 526
- sistemas de control adaptables programado 524
- sistemas de control análogo 16-19
- sistemas de control continuo 505
- sistemas de control de fluido 184-5
- sistemas de control digital 16-19
- sistemas de frenado anti-amarre 559
- sistemas de medición 7, 8-9
 - calibración de los 158-60
 - pantallas para los 136
 - medición de temperatura 157-8
- sistemas de microprocesadores 241-42
 - en básculas de baño 545
 - buses 242-3
 - control 241
 - ejemplos 250-3
 - fallas en 402-4
 - técnicas para localización 402-4
 - entrada/salida 249-50, 253
 - memoria 241, 246-9
- sistemas de modelado 7-8
- sistemas de resorte-amortiguador-masa 417
 - como sistemas de segundo orden 458-9
- sistemas de segundo orden como sistemas de segundo orden 458-64
 - ejemplos de 462-4
 - respuesta forzada 461-2
 - respuesta libre 459-61
 - medidas de desempeño 464-7
 - condiciones de estado estable en 462
 - diagramas de Bode para 494-6
 - entrada de paso en 477
 - respuesta de frecuencia en 491-2
 - sistema resorte-amortiguador-masa como 458-9
 - transformadas de Laplace en 476
- sistemas diferenciales 90
- sistemas dinámicos
 - ecuaciones diferenciales en los 449-50
 - entradas, formas de 451-2
 - identificación del sistema 467
 - respuestas libre y forzada 450-1
 - respuestas transitoria y permanente 451
- sistemas de primer orden 452-8
 - constante de tiempo 456-8
 - ejemplos de 454-6
 - respuesta forzada 453-4
 - respuesta libre 452-3
- sistemas de segundo orden 458-44
 - ejemplos de 462-4
 - medidas de desempeño 464-7
 - respuesta forzada 461-2
 - respuesta libre 459-61
- sistemas discretos 505
- sistemas distribuidos 377
- sistemas eléctricos de actuación 207
 - motores de c.a. 225-7
 - motores de c.d. 217-25
 - motores paso a paso 227-34
 - interruptores mecánicos 207
 - de estado sólido 209-15
 - relevadores 208-9
 - solenoides 215-17
- sistemas electromecánicos 440-3
 - motor de c.d. 440-3
 - potenciómetro rotatorio 440
- sistemas embebidos 296-99
- sistemas en ingeniería 439
 - linealidad 443-5
 - no linealidad en 444
 - sistemas electromecánicos 440-3
 - sistemas rotacional-traslacional 439-40
- sistemas en lazo abierto 11-12, 506-7
 - perturbación en 506
- sistemas en lazo cerrado 11-12, 506-7
 - elementos de los 12-15
 - funciones de transferencia en los 479
- perturbación en 506
- sistemas estables 480, 482
- sistemas hidráulicos 165-7
 - en robots para levantar y depositar objetos 550
 - fallas en los 401
 - hidráulica 165
- sistemas hidráulicos-mecánicos 445-8
- sistemas inestables 480-3
- sistemas microelectromecánicos (MEMS) 25
- sistemas neumáticos 167-9
 - fallas en los 401
 - robots 550
- sistemas numéricos 581-6
- sistemas programables 13
- sistemas rotacionales 416-17
- sistemas rotacional-traslacional 439-40
- sistemas síncronos 128
- sistemas sobreamortiguados 460
- sistemas subamortiguados 461
- sobrepaso en los sistemas dinámicos 465
- software 249, 278
- solenoides 215-17
- SPX (Intercambio de paquete secuenciado) 383
- subrutinas 290-3
 - subrutinas de retardo 291-3
 - subrutinas de retardo 291-3
- suma de productos en álgebra booleana 589
- suma en matemáticas binarias 582
- suministro de corriente (*sourcing*) en los PLC 351
- superposición, principio de 77, 443
- sustrando en matemáticas binarias 583

T

- tabla de vectores 333
- tablas de consulta 293-6
- tablas de verdad 113, 114, 115, 116, 117, 118
 - en álgebra booleana 589-91
- tableros DAQ 152
- tabuladores electrónicos de datos 54
- tacogenerador 47-8
- tacogenerador de reluctancia variable 47
- tarjetas 152
- tarjetas de clavijas para DAQ 151-3
- tarjetas emuladoras 405-6
 - autoprueba 405
 - forzado 407-8
 - métodos sistemáticos para la 405

- simulación 406-7
- técnicas de detección de fallas 397-8
- tasa de gran volumen 81
- TCP (Protocolo de control de transmisión del Internet) 383
- teclados 64-5
- técnicas para buscar fallas 403
- temporización en los protocolos para la transmisión de datos 381
- temporizador 262-3
 - XTAL1, XTAL12, 262
- temporizador vigilante 259
- temporizadores
 - en los PIC 267-8
 - en los PLC 364-5
- teorema de muestreo 97-8
- teorema de muestreo de Shannon 97
- teorema de Norton 614-15
- teorema de Thévenin 612-14
- teoremas de valor final 508
- teoremas de valor inicial en la transformada de Laplace 577
- termistores 57-8
 - en el control del motor de un automóvil 551
- termiodiodos 58
- tempopares 58-61
- termopilas 61
- tiempo de asentamiento
 - de los transductores 34
 - en sistemas dinámicos 466
- tiempo de búsqueda 146
- tiempo de conversión 96, 104
- tiempo de levantamiento
 - de los transductores 34
 - en sistemas dinámicos 464
- tiempo de respuesta de los transductores 33-4
- tiempo de retardo de propagación en compuertas lógicas 120
- tierra virtual 71
- tiras bimetálicas 56
- tiristores 210-12
- TIWAY 384
- topología de la red 379, 380
- topología de la red de anillo 379, 380
- topología de la red de árbol 379
- topología de la red de bus de datos 379
- topología de la red de estrella 379
- topología de la red jerárquica 379
- topología de las redes 379-80
- tornillo de avance 199
- transductor y sensor digital 29
- transductores 29-30
 - como sistemas de primer orden 454-5
- transferencia de datos en paralelo 329 y reconocimiento 331
- transferencia de datos en serie 330
- transferencia de potencia en señales 92
- transformada de Laplace 471, 472-4, 578-9
 - a partir de principios básicos 571-2
 - en sistemas de primer orden 474
 - en sistemas de segundo orden 476
 - fracciones parciales 578-9
 - función de escalón unitario 572-3
 - función pulso 573-4
 - propiedades de la 575-7
 - reglas 473
 - transformada estándar 574-5
 - transformada inversa 578-9
- transformada inversa de Laplace 578-9
- transformador diferencial de variación lineal (LVDT) 40-1
 - con fuelles 51
- transformadores diferenciales 40-1
- transistor de efecto de campo de óxido metálico (MOSFET) 215
- transistores 58
- transistores bipolares 212-15
- transistores búfer 214
- transistores en par de Darlington 214
- transistor-transistor lógico (TTL) 119, 120
- transmisión asíncrona 336, 378
- transmisión de banda ancha 380-1
- transmisión de datos
 - control de acceso a la red 380
 - protocolos 381-2
 - redes 379-81
 - transmisión de datos en serie 378-9
 - transmisión en paralelo 377-8
 - velocidad de la 378
- transmisión de datos en paralelo 377-8
- transmisión de datos en serie 378
- transmisión en banda base 381
- transmisión síncrona 336, 378
- trayectoria hacia adelante 479
- tren de engrane simple 197
- trenes de engranes 196, 197-9, 421-2
- trenes de engranes compuestos 198
- triacs 210-12
 - característica del triac 211
 - control del voltaje por 211
 - con unidad de cruzamiento cero 82
 - optoaisladores del triac 82
- trinquete 200
- TTL (transistor-transistor lógico) 119, 120
- tubo de Bourdon 51

U

- ubicación de los polos y respuesta transitoria 480-3
- unidad aritmética y lógica (ALU) 244
- unidad central de procesamiento (CPU) 241, 243-6
- unidad de disco duro 561-3
- unidad de exhibición visual (VDU) 139-41
- unidad de muestra y retención 96

V

- válvula de carrete 169-70, 445-6
- válvula de carrete giratorio 170
- válvula de doble asiento 170
- válvula de un asiento 181
- válvula de un solenoide 172
- válvula de vástago 170
- válvula limitadora de presión 173
- válvulas 169
 - símbolos 170-2
- válvulas de control de presión 173-4
- válvulas de control de proceso 180-5
 - cuerpo de las válvulas 181-3
 - dimensión 183-4
- válvulas de control proporcional 178-9
- válvulas de posición finita 169
- válvulas de posición infinita 169
- válvulas de presión limitada 173-4
- válvulas de regulación de presión 173
- válvulas de secuencia de presión 174
- válvulas de servocontrol 178-9
- válvulas direccionales 173
- válvulas operadas por piloto 172-3
- válvulas para dirección de control 169-73
- variables en C 305-6
- vástagos 181
- vástagos de contorno lineal 181
- vástagos de porcentaje igual 182