

# Capítulo 1

---

## Sistemas Numéricos

### Temario

- 1.1 Representación de los sistemas numéricos
- 1.2 Conversión entre bases
- 1.3 Aritmética
- 1.4 Complementos
- 1.5 Nomenclatura para representar números con signo
- 1.6 Complemento disminuido
- 1.7 Códigos numéricos.

## Objetivos

*Al concluir este capítulo el lector estará en capacidad de:*

- 1.- Representar los sistemas numéricos decimal, binario, octal y hexadecimal empleando la notación posicional y polinomial.
- 2.- Representar números en base  $n$  empleando las expresiones posicional y polinomial.
- 3.- Convertir números de base  $n$  a un equivalente en base  $m$  empleando las operaciones de división, para los enteros, y multiplicación, para las fracciones.
- 4.- Convertir en forma directa números cuya base origen  $n$  es potencia de la base destino  $m$ .
- 5.- Resolver operaciones aritméticas de suma, resta, multiplicación y división ente números de cualquier base  $n$ , empleando los principios básicos de la aritmética decimal.
- 6.- Representar números con signo empleando el concepto de *convención de signo*.
- 7.- Resolver operaciones de suma y resta con números negativos de base  $n$ , empleando los conceptos de complemento y complemento disminuido.
- 8.- Explicar el concepto de código, empleando ejemplos de aplicación de la vida cotidiana.
- 9.- Representar números decimales empleando el código BCD.
- 10.- Resolver operaciones de suma empleando el código BCD.
- 11.- Explicar el código Gray, y el concepto de precisión, empleando una aplicación industrial.
- 12.- Conocer el código ASCII e identificar la representación correspondiente a un caracter alfanumérico.

## Introducción.

Las matemáticas, así como los sistemas numéricos, son tan antiguos como la propia humanidad. Por ejemplo, los sistemas de cálculo primitivos estaban basados en sistemas numéricos de base diez o base cinco, debido a cuestiones naturales relacionadas por la cantidad de dedos de nuestras manos.

Por evidencias encontradas en registros y libros, se ha determinado que el sistema numérico decimal era utilizado ya hace 4000 años, aproximadamente, por los egipcios. Con ciertas características particulares, este sistema fue empleado siglos después por los romanos.

Por el mismo tiempo al cual hace referencia el párrafo anterior, los babilonios utilizaban muescas en forma de cuña (cuneiforme) para representar a su sistema numérico, las cuales se marcaban en tablillas. Una cuña representaba al número uno, mientras que una flecha representaba al número diez. Por otro lado, para el número sesenta se volvía a emplear el mismo símbolo que se asignaba al número uno, pero se le daba importancia a la posición que guardaba en el número completo, esto es, se maneja el criterio posicional de “peso” del dígito. Así, los babilonios ya empleaban el sistema numérico sexagesimal, el cual seguimos empleando hasta nuestros días para el manejo de segundos y minutos de nuestro uso horario. Ver figura 1.1

Número	1	2	3	5	10	20	50
Babilonia	▼	▼ ▼	▼ ▼ ▼	▼ ▼ ▼ ▼ ▼	<	< <	< < < < <
Egipto (Jeroglíficos)	I	II	III	IIII	A	A A	A A A A A

Figura 1.1 Símbolos empleados por las culturas Babilónica y Egipcia.

Al paso del tiempo y muchos siglos después, la evolución de las culturas y sus correspondientes necesidades motivaron la aparición y uso de otros sistemas numéricos. Ya en el siglo XIX los matemáticos George Boole y Cantor trabajaron sobre la teoría de conjuntos la cual es empleada hoy

día para el análisis y síntesis de sistemas digitales, altamente empleada en computadoras, calculadoras, aparatos electrónicos y juguetes, entre otras muchas aplicaciones.

En resumen se puede establecer que de todos los sistemas numéricos que han sido empleados por la humanidad, algunos llegan a tener mayor éxito que otros, esto debido a la facilidad en el manejo de las operaciones y en el impacto positivo cuando se trata de hacer una implantación de dicho sistema. Definitivamente, el sistema numérico base 10 ha sido el más exitoso, por cuestiones naturales. El sexagesimal ha perdurado por más de 4000 años, aunque su impacto se limita al uso horario, pero empleado en forma masiva por el hombre.

Por otro lado, con el advenimiento y desarrollo de la tecnología electrónica de estado sólido, la cual apareció a mediados del siglo XX, el sistema binario (digitales) y sus equivalentes; cuaternario, octal y Hexadecimal, adquirieron una relevancia fundamental. Sería casi imposible imaginar el mundo moderno sin computadoras, calculadoras programables, hornos de microondas, lavadoras inteligentes, naves espaciales y televisores a control remoto y con funciones programables sin el recurso del sistema numérico binario. Así pues, el resto de este capítulo estará dedicado en específico al estudio, conversión, operaciones aritméticas y manejo de códigos empleando esos sistemas numéricos.

## 1.1 Representación de los Sistemas Numéricos

Un sistema numérico consiste de un conjunto de símbolos denominados “dígitos”, con relaciones definidas por la adición (+), resta, multiplicación y división. Los sistemas numéricos más usados (más no los únicos) son:

- Decimal
- Binario
- Octal
- Hexadecimal

En general se emplean dos posibles opciones para representar a cualquier número en cualquier base, *posicional* y *polinomial*. A continuación se revisan ambas opciones.

**1.1.1.- Notación Posicional.** La posición de cada dígito indica su peso o relevancia, esto es, un número  $N$  se puede representar de la siguiente forma:

$$N = (a_{n-1}a_{n-2}...a_0a_{-1}...a_{-m})_r$$

$r = \text{Base (Radix)}$

**Ejemplo 1.1** Identificar las posiciones del número decimal 3845.43

$$\begin{array}{cccccc} ( & 3 & 8 & 4 & 5. & 4 & 3 )_{10} & r = 10 \\ \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \uparrow & \\ a_3 & a_2 & a_1 & a_0 & a_{-1} & a_{-2} & \end{array}$$

**1.1.2.- Notación polinomial.** En términos generales cualquier número  $N$  de rádix  $r$  puede ser escrito de acuerdo a la siguiente fórmula:

$$N = \sum_{i=-m}^{n-1} a_i r^i$$

**Ejemplo 1.2.** Representar en forma polinomial al número decimal 3845.43.

$(3845.43)_{10}$

$$N = 3 \times (10)^3 + 8 \times (10)^2 + 4 \times (10)^1 + 5 \times (10)^0 + 4 \times (10)^{-1} + 3 \times (10)^{-2}$$

## 1.2 Conversión entre bases

En principio es razonable entender que exista una cantidad casi infinita de representaciones numéricas dado que la única restricción que se tiene es con respecto al término  $r > 1$ , el cual representa a la base del sistema numérico correspondiente. Sin embargo siempre resulta importante cuestionarse sobre la utilidad práctica de todas esas posibilidades. Aún más, una vez establecido el criterio de utilidad es conveniente establecer su equivalente a un sistema numérico más familiar para el usuario. Por ejemplo, por cuestiones obvias en una cabina de mando será conveniente desplegar los dígitos de la lectura de una variable de control en términos del sistema base diez. Por otro lado, al especialista en diseño de sistemas digitales de automatización para máquinas lavadoras de ropa le resultará altamente práctico el manejo de los sistemas numéricos binarios y hexadecimal, los cuales no son útiles para el cliente que adquiere ese producto ya que él requiere que se le muestren todas las indicaciones numéricas en el sistema numérico base 10. En la tabla 1.1 se muestra una relación entre las bases más comunes.

Decimal	Binario	Octal	Hexadecimal	Decimal	Binario	Octal	Hexadecimal	Decimal	Binario	Octal	Hexadecimal
0	0	0	0	11	1011	13	B	22	10110	26	16
1	1	1	1	12	1100	14	C	23	10111	27	17
2	10	2	2	13	1101	15	D	24	11000	30	18
3	11	3	3	14	1110	16	E	25	11001	31	19
4	100	4	4	15	1111	17	F	26	11010	32	1A
5	101	5	5	16	10000	20	10	27	11011	33	1B
6	110	6	6	17	10001	21	11	28	11100	34	1C
7	111	7	7	18	10010	22	12	29	11101	35	1D
8	1000	10	8	19	10011	23	13	30	11110	36	1E
9	1001	11	9	20	10100	24	14	31	11111	37	1F
10	1010	12	A	21	10101	25	15	32	100000	40	20

Tabla 1.1 Relación entre las bases decimal, binaria, octal y hexadecimal.

Para lograr la conversión entre diferentes bases es necesario aplicar procedimientos específicos. En concreto en este capítulo se revisarán dos de ellos, el método de la división y el método de la multiplicación. Ambos se detallan a continuación.

### 1.2.1 Conversión mediante división por base.

Se emplea para convertir un entero, base A, a un equivalente en base B.

**Ejemplo 1.3:** Convertir  $(248)_{10}$  a su equivalente

a) Binario

b) Hexadecimal.

a)  $(248)_{10} = (N)_2$

Primer división:  $248/2$  Cociente 124; Residuo 0

Segunda división:  $124/2$  Cociente 62; Residuo 0

Tercer división:  $62/2$  Cociente 31; Residuo 0

Cuarta división  $31/2$  Cociente 15; Residuo 1

Quinta división  $15/2$  Cociente 7; Residuo 1

Sexta división  $7/2$  Cociente 3; Residuo 1

Séptima división  $3/2$  Cociente 1; Residuo 1

Octava división  $1/2$  Cociente 0; Residuo 1

$$(248)_{10} = (11111000)_2$$

$(248)_{10} = (N)_{16}$

Primer división:  $248/16$  Cociente 15; Residuo 8

Segunda división:  $15/16$  Cociente 0; Residuo 15

$$(248)_{10} = (F8)_{16}$$

### 1.2.2. Conversión mediante multiplicación por base.

Se emplea para conversión de fracciones de una base A, a otra en base B.

**Ejemplo 1.4.** Realizar la siguiente conversión  $(0.1285)_{10}$  a base 8 empleando hasta tres dígitos significativos.

Primer multiplicación:  $0.1285 \times 8 = 1.028$

Segunda multiplicación:  $0.028 \times 8 = 0.224$

Tercera multiplicación:  $0.224 \times 8 = 1.792$

$$(0.1285)_{10} = (0.101)_8$$

**Ejemplo 1.5:** Convertir  $(18.6)_9$  a su equivalente en base 11.

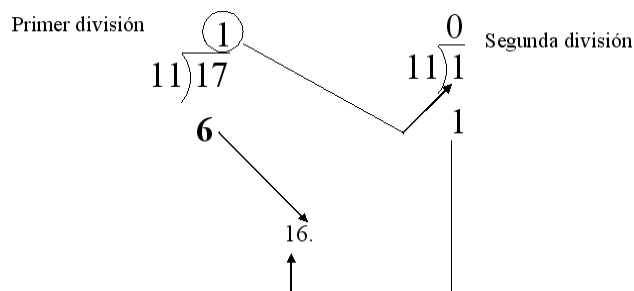
Procedimiento: primero convertimos el número de su base origen, en este caso es base 9, a base 10.

Posteriormente convertiremos el número obtenido en base 10 a base 11 empleando los métodos de la división y multiplicación, para las partes enteras y decimales, respectivamente.

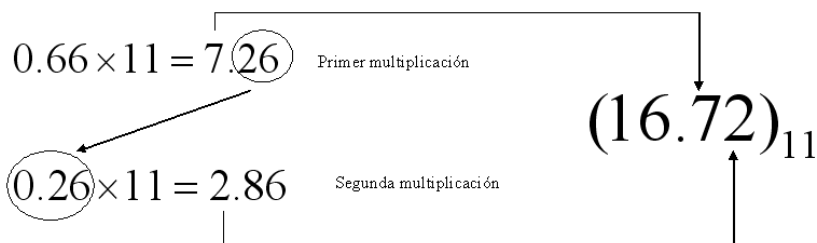
Primero se realiza la conversión a base 10.

$$N_{10} = 1 \times (9)^1 + 8 \times (9)^0 + 6 \times (9)^{-1} = (17.66)_{10}$$

Ahora se realizan las operaciones de división



Ahora, para la parte decimal se realizan las operaciones de multiplicación.



### 1.2.3 Conversión entre bases que son potencia una de otra

Algunos procedimientos de conversión pueden resultar relativamente sencillos siempre y cuando se cumpla con algunos requisitos mínimos, tales como la existencia de una relación de potencia entera y positiva entre las bases origen  $B_1$ , y destino  $B_2$ . A continuación se explican esos procedimientos.

Para convertir un número  $N$  de una base origen  $B_1$ , a una destino  $B_2$ , en donde se cumple la relación:

$$B_2 = B_1^k \text{ siendo } k \text{ un número natural}$$

Se procede a agrupar el número  $N$  en grupos de tamaño  $k$ , tanto hacia la izquierda como hacia la derecha del punto decimal. Una vez generadas estas agrupaciones se procede a sustituir cada una de ellas por su representación equivalente, según la base destino  $B_2$ .

En el caso de conversión de  $B_2$  a  $B_1$  se procede a expresar cada dígito de  $B_2$  por su equivalente en  $B_1$ , utilizando  $k$  dígitos para cada uno de ellos.

**Ejemplo 1.6** Realizar la conversión del siguiente número binario a su equivalente en base 8

$$N = (1011011.1010111)_2$$

Las bases que se plantean en este ejercicio son la base dos, que es la base origen, y la base ocho, que es la destino. La relación en potencia entre ambas es 3, ya que  $2^3 = 8$

Así, los bits del número original se agrupan de tres en tres (la relación de potencia), y se expresan en su equivalente base ocho.

$$(001)(011)(011).(101)(011)(100)$$

Por lo tanto:  $(1011011.1010111)_2 = (133.534)_8$

**Ejemplo 1.7** Convertir número  $(1011011.1010111)_2$  a su equivalente en base 16.

Como la base de origen es 2 y la base destino es 16, entonces la potencia que los relaciona es 4.

Por lo tanto el número base dos se agrupará en elementos de 4 dígitos.

$$(0101)1011).(1010)(1110)$$

Cuyo equivalente en base 16 es:  $(5B.AE)_{16}$

**Ejemplo 1.8.** Convertir el número  $(AF.16C)_{16}$  a base 8

Como la base de origen es de mayor magnitud que la magnitud de la base destino, además de que no existe relación de potencia entre ambas, lo recomendable es convertir primero a binario y posteriormente convertir es binario a su equivalente en base ocho.

$$(AF.16C)_{16} = (1010)(1111).(0001)(0110)(1100)$$

Ahora se agrupa el número en binario en conjuntos de tres bits, ya que la potencia que relaciona a la base 2 (origen), con la base 8 (destino) es tres.

$$(010)(101)(111).(000)(101)(101)(100)$$

$$(257.0554)_8$$

Por lo tanto el resultado es:  $(257.0554)_8$

**Ejemplo 1.9** Convertir  $(AF.15C)_{16}$  a su equivalente en base 8

Primera opción. Se empleará el procedimiento de división y multiplicación.

$$(AF.16C)_{16} = (N)_{10} = 10 \times 16^1 + 15 \times 16^0 + 1 \times 16^{-1} + 5 \times 16^{-2} + 12 \times 16^{-3} = 175.089_{10}$$

Primer división:  $175/8$  Cociente 21; Residuo 7

Segunda división:  $21/8$  Cociente 2; Residuo 5

Tercer división  $2/8$  Cociente 0; Residuo 2

Primera multiplicación:  $0.089 \times 8 = 0.712$

Segunda multiplicación:  $0.712 \times 8 = 5.696$

$$(AF.16C)_{16} = (257.05)_{8}$$

Segunda opción. Convertir el número original a su equivalente en binario, para posteriormente convertir directamente esta última expresión a su representación final, base ocho.

$$(AF.16C)_{16} = (1010\ 1111.0001\ 0110\ 1100)_2$$

Ahora, sabiendo que la relación entre base dos y base ocho está dada por  $2^3 = 8$ , se determina que  $(1010\ 1111.0001\ 0110\ 1100)_2 = (010\ 101\ 111.000\ 101\ 101\ 100)_2 = (257.0554)_8$

Resultado que concuerda con el procedimiento anterior.

**Ejemplo 1.10** Convertir  $(3231)_4$  a su equivalente en base 16

El factor de potencia es:  $4^n = 16$ , siendo  $n = 2$ .

Siguiendo la relación entre base cuatro y hexadecimal

0	0
1	1
2	2
3	3
10	4
11	5
12	6
13	7
20	8
21	9
22	A
23	B
30	C
31	D
32	E

Se obtiene que:  $\underline{32}\ \underline{31} = ED$ .



### 1.3 Aritmética.

En general las operaciones de aritmética son las mismas no importando el sistema numérico en el cual se esté trabajando. Obviamente, por cuestiones naturales e históricas, la aritmética en base diez ha sido la más empleada y por lo tanto la más sencilla de aplicar. En todo caso las tablas de suma y multiplicación serían lo único que cambiaría cuando se trabaja con otras bases. La resta y la división son el proceso inverso a las dos operaciones anteriores.

Al trabajar con una base nueva sería conveniente iniciar con una tabla que relacione los resultados más sencillos y aplicarlos en la solución de ejercicios. Algunas de estos resultados se pueden ver en las tablas 1.2 a) y 1.2b .

suma	0	1	10	11	100	101	110	111	1000
0	0	1	10	11	100	101	110	111	1000
1	1	10	11	100	101	110	111	1000	1001
10	10	11	100	101	110	111	1000	1001	1010
11	11	100	101	110	111	1000	1001	1010	1011
100	100	101	110	111	1000	1001	1010	1011	1100
101	101	110	111	1000	1001	1010	1011	1100	1101
110	110	111	1000	1001	1010	1011	1100	1101	1110
111	111	1000	1001	1010	1011	1100	1101	1110	1111
1000	1000	1001	1010	1011	1100	1101	1110	1111	10000

Tabla 1.2 a) Suma en binario.

Suma	0	1	2	3	10	11	12	13	20
0	0	1	2	3	10	11	12	13	20
1	1	2	3	10	11	12	13	20	21
2	2	3	10	11	12	13	20	21	22
3	3	10	11	12	13	20	21	22	23
10	10	11	12	13	20	21	22	23	30
11	11	12	13	20	21	22	23	30	31
12	12	13	20	21	22	23	30	31	32
13	13	20	21	22	23	30	31	32	33
20	20	21	22	23	30	31	32	33	100

Tabla 1.2 b) Suma en cuaternario

#### 1.3.1 Aritmética binaria.

Como se mencionó anteriormente, las operaciones aritméticas siguen las mismas reglas de ejecución, no importando la base en la que se esté trabajando. Como principio el lector puede seguir las reglas de operación básicas, empleando la base binaria. Para otras bases puede hacer extensivo el mismo principio que se muestra en la siguiente tabla 1.3.

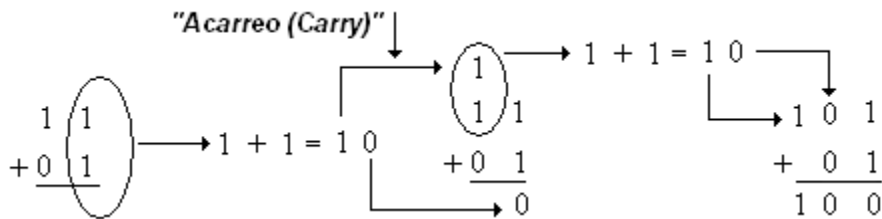
suma	resta	multiplicación	división
$0 + 0 = 0$	$0 - 0 = 0$	$0 \times 0 = 0$	$0/0 =$ indeterminado
$0 + 1 = 1$	$1 - 0 = 1$	$0 \times 1 = 0$	$0/1 = 0$
$1 + 0 = 1$	$1 - 1 = 0$	$1 \times 0 = 0$	$1/0 =$ indeterminado
$1 + 1 = 10$ . El resultado se expresaría como cero y un acarreo de uno.	$0 - 1 = 1$ Como el uno es de mayor magnitud al cero, se hace un préstamo al cero: $10 - 1 =$ dos menos uno. El resultado es uno y se envía el préstamo a operar en suma con el sustraendo de la siguiente columna	$1 \times 1 = 1$	$1/1 = 1$

Tabla 1.3 Operaciones aritméticas básicas en binario.

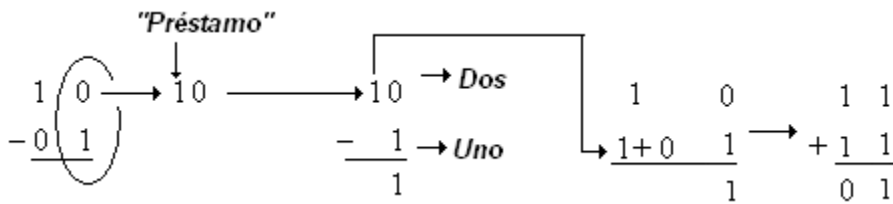
Para operaciones complejas la multiplicación se hace bit a bit y por último se realiza la suma de cada uno de los resultados. En el caso de la división se sigue aplicando la metodología de prueba y error para ajustar el cociente.

**Ejemplo 1.11** Suma  $a + b$ , en donde  $a = (11)_2$  y  $b = (01)_2$

Suma



Resta  $a - b$ , en donde  $a = (10)_2$  y  $b = (01)_2$



Para comprobar la operación, se puede realizar la suma del resultado obtenido más el sustraendo para obtener el minuendo.

**Ejemplo 1.12** Suma  $a + b$ ; Resta  $a - b$ , en donde  $a = (1110101)_2$  y  $b = (101001)_2$

Suma

$$\begin{array}{r} 1110101 \\ + 101001 \\ \hline 10011110 \end{array}$$

Resta

$$\begin{array}{r} 1110101 \\ - 101001 \\ \hline 10011100 \end{array}$$

Para comprobar la operación, se puede realizar la suma del resultado obtenido más el sustraendo para obtener el minuendo:

$$\begin{array}{r} 10011100 \\ + 101001 \\ \hline 1110101 \end{array}$$

**Ejemplo 1.13** Suma  $a + b$ ; Resta  $a - b$ , en donde  $a = 110110$ ;  $b = 101111$ .

Suma

$$\begin{array}{r} 1\ 1\ 0\ 1\ 1\ 0 \\ +\ 1\ 0\ 1\ 1\ 1\ 1 \\ \hline 1\ 1\ 0\ 0\ 1\ 0\ 1 \end{array}$$

Resta

$$\begin{array}{r} 1\ 1\ 0\ 1\ 1\ 0 \\ -\ 1\ 0\ 1\ 1\ 1\ 1 \\ \hline 0\ 0\ 0\ 1\ 1\ 1 \end{array}$$

Comprobación de la operación de resta

$$\begin{array}{r} 1\ 0\ 1\ 1\ 1\ 1 \\ +\ 0\ 0\ 0\ 1\ 1\ 1 \\ \hline 1\ 1\ 0\ 1\ 1\ 0 \end{array}$$

**Ejemplo 1.14**

Suma  $a + b + c$ ; Resta  $a - b - c$ , en donde  $a = 111011$ ;  $b = 11110$ ;  $c = 1101$

Suma de  $a + b$

$$\begin{array}{r} 1\ 1\ 1\ 0\ 1\ 1 \\ +\ 1\ 1\ 1\ 1\ 0 \\ \hline 1\ 0\ 1\ 1\ 0\ 0\ 1 \end{array}$$

Suma de  $a + b + c$

$$\begin{array}{r} 1\ 0\ 1\ 1\ 0\ 0\ 1 \\ +\ 1\ 1\ 0\ 1 \\ \hline 1\ 1\ 0\ 0\ 1\ 1\ 0 \end{array}$$

Resta de  $a - b$

$$\begin{array}{r} 1\ 1\ 1\ 0\ 1\ 1 \\ -\ 1\ 1\ 1\ 1\ 0 \\ \hline 0\ 1\ 1\ 1\ 0\ 1 \end{array}$$

Resta de  $a - b - c$

$$\begin{array}{r} 1\ 1\ 1\ 0\ 1 \\ -\ 1\ 1\ 0\ 1 \\ \hline 1\ 0\ 0\ 0\ 0 \end{array}$$

Como ejercicio adicional se deja la comprobación de los resultados, empleando la base diez.

## Multiplicación y división.

**Ejemplo 1.15.** Multiplicar  $a \times b$ ; Dividir  $a/b$ , en donde  $a = 111011$  y  $b = 1110$ .

Multiplicación.

$$\begin{array}{r}
 \phantom{\times} 1 \ 1 \ 1 \ 0 \ 1 \ 1 \\
 \times \phantom{00000} 1 \ 1 \ 0 \\
 \hline
 0 \ 0 \ 0 \ 0 \ 0 \ 0 \\
 1 \ 1 \ 1 \ 0 \ 1 \ 1 \\
 1 \ 1 \ 1 \ 0 \ 1 \ 1 \\
 \hline
 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0
 \end{array}$$

División

$$\begin{array}{r}
 \phantom{1110} 1 \ 0 \ 0 \ . \ 001 \\
 1110 \overline{) 111 \ 0 \ 1 \ 1} \\
 \underline{111 \ 0} \phantom{00} \\
 0 \ 000 \ 1 \ 1 \ 000 \\
 \underline{1 \ 110} \phantom{00} \\
 01010
 \end{array}$$

### 1.3.2 Aritmética Octal

Algunos resultados básicos de las operaciones de suma y multiplicación que se muestran en las tablas 1.4 a y 1.4 b, respectivamente, y pueden ser empleados como punto de referencia para realizar ejercicios más complejos.

Suma	0	1	2	3	4	5	6	7	10	11	12
0	0	1	2	3	4	5	6	7	10	11	12
1	1	2	3	4	5	6	7	10	11	12	13
2	2	3	4	5	6	7	10	11	12	13	14
3	3	4	5	6	7	10	11	12	13	14	15
4	4	5	6	7	10	11	12	13	14	15	16
5	5	6	7	10	11	12	13	14	15	16	17
6	6	7	10	11	12	13	14	15	16	17	20
7	7	10	11	12	13	14	15	16	17	20	21
10	10	11	12	13	14	15	16	17	20	21	22
11	11	12	13	14	15	16	17	20	21	22	23
12	12	13	14	15	16	17	20	21	22	23	24

Tabla 1.4 a) Suma en octal

Multiplicación	0	1	2	3	4	5	6	7	10	11	12
0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	10	11	12
2	0	2	4	6	10	12	14	16	20	22	24
3	0	3	6	11	14	17	22	25	30	33	36
4	0	4	10	14	20	24	30	34	40	44	50
5	0	5	12	17	24	31	36	43	50	55	62
6	0	6	14	22	30	36	44	52	60	66	74
7	0	7	16	25	34	43	52	61	70	77	106
10	0	10	20	30	40	50	60	70	100	110	120
11	0	11	22	33	44	55	66	77	110	121	132
12	0	12	24	36	50	62	74	106	120	132	144

b) Multiplicación en octal

**Ejemplo 1.16.** Suma  $a + b$ ; Resta  $a - b$ , en donde  $a = (3546)_8$ ;  $b = (377)_8$

Suma

$$\begin{array}{r} 3 \ 5 \ 4 \ 6 \\ + \quad 3 \ 7 \ 7 \\ \hline (4 \ 1 \ 4 \ 5)_8 \end{array}$$

Resta

$$\begin{array}{r} 3 \ 5 \ 4 \ 6 \\ - \quad 3 \ 7 \ 7 \\ \hline (3 \ 1 \ 4 \ 7)_8 \end{array}$$

**Ejemplo 1.17** Suma  $a + b + c$ ; Resta  $a - b - c$ , en donde  $a = (6214)_8$ ;  $b = (5143)_8$ ;  $c = (615)_8$

Suma de  $a + b + c$

$$\begin{array}{r} 6 \ 2 \ 1 \ 4 \\ + \quad 5 \ 1 \ 4 \ 3 \\ \hline 6 \ 1 \ 5 \\ \hline (1 \ 4 \ 1 \ 7 \ 4)_8 \end{array}$$

Resta  $a - b$

$$\begin{array}{r} 6 \ 2 \ 1 \ 4 \\ - \quad 5 \ 1 \ 4 \ 3 \\ \hline (1 \ 0 \ 5 \ 1)_8 \end{array}$$

Resta  $a - b - c$

$$\begin{array}{r} 1 \ 0 \ 5 \ 1 \\ - \quad 6 \ 1 \ 5 \\ \hline (0 \ 2 \ 3 \ 4)_8 \end{array}$$

**Ejemplo 1.18** Multiplicación  $a \times b$ ; División  $a/b$ .  $a = (540)_8$ ;  $b = (13)_8$

Multiplicación

$$\begin{array}{r} 5 \ 4 \ 0 \\ \times \quad 1 \ 3 \\ \hline 2 \ 0 \ 4 \ 0 \\ 5 \ 4 \ 0 \\ \hline 7 \ 4 \ 4 \ 0 \end{array}$$

División

$$\begin{array}{r} 4 \ 0 \\ 13 \overline{) 5 \ 4 \ 0} \\ \underline{0 \ 0 \ 0} \end{array}$$

### 1.3. 3 Aritmética Hexadecimal

Tal y como se vio con anterioridad, el sistema hexadecimal es relativamente más complejo que el binario, cuaternario, octal y decimal debido a que se requieren 16 símbolos para representar a su base completa. Por eso mismo se recomienda que en la medida de lo posible se utilicen tablas de operaciones algebraicas como apoyo para resolver ejercicios, sobre todo cuando se emplean dos o más dígitos en las expresiones numéricas.

Para la operación de suma, puede tomarse como referencia la tabla 1.5.

+	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
1	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
2	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12
3	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13
4	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14
5	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15
6	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16
7	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17
8	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18
9	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19
A	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A
B	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B
C	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
D	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
E	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E
F	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
10	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20

Tabla 1.5 Suma en Hexadecimal

**Ejemplo 1.19** Suma  $a + b$ ; Resta  $a - b$  en donde  $a = (71D0)_{16}$  y  $b = (2A54)_{16}$ ; R

Suma

$$\begin{array}{r}
 7 \ 1 \ D \ 0 \\
 + \ 2 \ A \ 5 \ 4 \\
 \hline
 (9 \ C \ 2 \ 4)_H
 \end{array}$$

Resta

$$\begin{array}{r}
 7 \ 1 \ D \ 0 \\
 - \ 2 \ A \ 5 \ 4 \\
 \hline
 (4 \ 7 \ 7 \ C)_H
 \end{array}$$

**Ejemplo 1.20** Suma  $a + b + c$ ; Resta  $a - b - c$  en donde  $a = (F53E)_{16}$ ,  $b = (CE1)_{16}$  y  $c = (100A)_{16}$

Suma  $a + b$ 

$$\begin{array}{r}
 F \ 5 \ E \ E \\
 + \ C \ E \ 1 \\
 \hline
 (1 \ 0 \ 2 \ 1 \ F)_{16}
 \end{array}$$

Suma  $a + b + c$ 

$$\begin{array}{r}
 1 \ 0 \ 2 \ 1 \ F \\
 + \ 1 \ 0 \ 0 \ A \\
 \hline
 (1 \ 1 \ 2 \ 2 \ 9)_{16}
 \end{array}$$

Resta  $a - b$ 

$$\begin{array}{r}
 F \ 5 \ E \ E \\
 - \ C \ E \ 1 \\
 \hline
 E \ 8 \ 5 \ D
 \end{array}$$

Resta  $a - b - c$ 

$$\begin{array}{r}
 E \ 8 \ 5 \ D \\
 - \ 1 \ 0 \ 0 \ A \\
 \hline
 D \ 8 \ 5 \ 3
 \end{array}$$

**Ejemplo 1.21** Multiplicación  $a \times b$ ; División  $a/b$ .  $a = (160)_{16}$ ;  $b = (B)_{16}$

Multiplicación

$$\begin{array}{r} 160 \\ \times B \\ \hline F20 \end{array}$$

División

$$\begin{array}{r} 20 \\ B \overline{)160} \\ 00 \end{array}$$

## 1.4 Complementos

La aritmética complementaria se emplea como herramienta de aplicación para la parte operativa de números negativos. Complemento a base  $r$  y complemento disminuido,  $(r-1)$  son las opciones más empleadas, sobre todo en su aplicación en computadoras, cuando se trata de manejar números negativos y las correspondientes operaciones aritméticas.

A continuación se muestra la ecuación para el complemento y posteriormente se tratarán algunos ejemplos. Por último se revisarán ejemplos empleando el complemento disminuido.

### 1.4.1 Definición de complemento.

$[N]_r = \text{Complemento.}$

$$[N]_r = (r^n)_r - (N)_r$$

En donde:

$n$  es el número de dígitos a emplear en la operación.

$r$  es la base.

$(N)_r$  es el número al cual se le calculará su complemento.

Como casos específicos se tienen el complemento a uno y el complemento a dos ya que ambos se aplican principalmente en calculadoras y computadoras.

**Ejemplo 1.22** Calcular el complemento a 2 del número  $(N)_2 = (01100101)_2$

$$[N]_2 = (2^8)_2 - (01100101)_2 = (10011011)_2$$

**Ejemplo 1.23.** Empleando el ejercicio anterior, verificar que el complemento de un número es igual al negativo del mismo.

Haciendo la operación:  $[N]_2 + (N)_2 = 00000000$ , por lo tanto se comprueba que:

$$[N]_2 = -(N)_2$$

**Ejemplo 1.24.** Obtener el complemento del número  $(N)_{10} = (40960)_{10}$

$$1.- [N]_{10} = ((10)^5)_{10} - (40960)_{10} = (59040)_{10}$$

$$2.- \text{Verificando: } [N]_{10} + (N)_{10} = 59040 + 40960 = 100000$$

Siendo los primeros cinco bits los correspondientes al resultado

**Ejemplo 1.25.** Determine el complemento a 2 de  $(N)_2 = (11001)_2$ , empleando 8 dígitos.

$$[N]_2 = ((2)^8)_2 - (00011001)_2$$

$$[N]_2 = 100000000 - (00011001)_2$$

$$[N]_2 = (011100111)_2$$

Nota: sólo los primeros ocho dígitos forman parte de la solución, ya que  $n = 8$

#### 1.4.2 Algoritmo para obtener el complemento sin aplicar la definición.

Sea el número  $(N)_r = (a_{n-1}a_{n-2} \dots a_0 \cdot a_{-1}a_{-2} \dots a_{-m})_r$ .

- 1.- Reste una unidad a la base:  $r-1$
- 2.- Réstele cada dígito del número  $(N)_r$  al resultado del punto anterior.
- 3.- Sume una unidad al resultado obtenido en el punto 2.

**Ejemplo 1.26.** Calcule el complemento a dos del número  $110110_2$

Paso 1

$$r-1 = 1$$

Paso 2

$$\begin{array}{r} 111111 \\ -110110 \\ \hline 001001 \end{array}$$

Paso 3

$$\begin{array}{r} 001001 \\ + \quad \quad 1 \\ \hline 001010 \end{array}$$



**Ejemplo 1.27.** Calcule el complemento a diez del número  $2154_{10}$

Paso 1:  $r - 1 = 9$

Paso 2

Paso 3

9999

7845

$- \underline{2154}$

$+ \underline{\quad 1}$

7845

7846

Complemento

**Ejemplo 1.28.** Calcule el complemento a nueve del número  $5878_9$

Paso 1:  $r - 1 = 8$

Paso 2

Paso 3

8888

3010

$- \underline{5878}$

$+ \underline{\quad 1}$

3010

3011

Complemento

## 1.5 Nomenclatura para representar números con signo.

En la gran mayoría de las aplicaciones numéricas de los sistemas digitales se tiene la necesidad de aplicar tanto números positivos como negativos. Calculadoras, computadoras, teléfonos celulares, y Controladores lógicos programables, PLC por sus siglas en inglés, son sólo algunos ejemplos. Por lo general se maneja un bit adicional ubicado en la extrema izquierda de los dígitos que representa a la magnitud. En general se emplean el número cero para representar el signo positivo, no importando cuál sea la base. Para la representación negativa se emplea siempre el dígito más alto de la base. Así, para la base binaria se tendría que el número uno representa el signo negativo. En el caso de la base diez, el número nueve estaría representado dicho signo. A esta convención se le denomina sistema de signo – magnitud. En general, esta representación se emplea sólo para el sistema binario, por ejemplo, sea  $(N)_2 = (11011)_2$ , se empleará un bit extra en la posición extrema izquierda para representar el signo del número en cuestión, esto es

$+(N)_2 = (0,11011)$  por lo que  $-(N)_2 = (1,11011)$

Aunque como se puede ver, la aplicación de esta convención es sencilla, resulta poco viable en la práctica, ya que su implantación electrónica es relativamente compleja, además de que pudiera generar algunas confusiones aritméticas. Esto último se puede ilustrar con una simple operación de suma entre un número  $a$  y su negativo. Por ejemplo, sea la suma  $6 + (-6)$ , resuelta mediante la aplicación de la convención signo – magnitud.

$+6_{10} = +(N)_2 = (0,110)$  ;  $-6_{10} = -(N)_2 = (1,110)$ , por lo tanto  $+(N)_2 + (-N)_2 = 10,100 = 1 + 4!$ , lo cual es imposible. Nótese que existen los tres dígitos que representan a la magnitud, ubicados a la derecha de la coma. Así mismo se cuenta el bit inmediatamente a la izquierda de la coma, el cual representa el signo del resultado, que en este caso es positivo. Por último existe un bit de acarreo, el cual carece de relevancia y por lo tanto se descarta del resultado.

Por todas las razones de desventajas mencionadas en la aplicación de la convención signo – magnitud, es que se opta por implantar la metodología de los complementos, que en todo caso para

sistemas digitales serían el complemento a dos y el complemento a uno. Por lo tanto, para el caso del número seis positivo se tendrían las representaciones siguientes

$$+6_{10} = +(N)_2 = (0,110)_2; \quad -6_{10} = -(N)_2 = [0,110]_2 = (1,010)_2$$

**Ejemplo 1.29.** Obtenga el equivalente en decimal del número  $[N]_2 = (1,1111010)$

$$[N]_2 = (1,1111010)$$

$$\text{Ya que: } [N] + (N) = 0$$

$$\text{Entonces: } [N] = -(N)$$

$$\text{Por lo tanto: } (N) = -[1,1111010]_2 = [[1,1111010]]_2 = (0,0000110)_2$$

$$\text{Obteniéndose como magnitud: } (N) = (0,0000110) = 6_{10}$$

$$\text{Así entonces, el equivalente en base diez es: } [N]_2 = (1,1111010) = -6_{10}$$

**Ejemplo 1.30.** Resolver en binario las siguientes operaciones:

a) Sin emplear complemento:  $5_{10} - 3_{10}$

b) Con complemento a 2:  $5_{10} - 3_{10}$

c)  $3_{10} - 5_{10}$ , con complemento a 2.

d)  $5_{10} - 12_{10}$  con complemento a 2.

e)  $-3_{10} - 4_{10}$  con complemento a 2.

f)  $-3_{10} - 5_{10}$  con complemento a 2.

a)  $5_{10} - 3_{10} = (101)_2 - (011)_2 = (010)_2$

b)  $5_{10} = (0,101)_2; 3_{10} = (0,011)_2; -3 = -(0,011)_2$

$$-3 = -(0,011)_2 = [0,011]_2 = (1,101)_2$$

$$5_{10} + (-3)_{10} = (0,101)_2 + (1,101)_2 = (10,010)_2$$

Obsérvense los cuatro dígitos menos significativos. El cuarto de ellos representa el signo, que en este caso es positivo. Los restantes tres representan la magnitud, por lo tanto, el resultado final en base diez es:  $2_{10}$

c)  $3_{10} - 5_{10} = (0,011)_2 - (0,101)_2 = (0,011)_2 + (1,011) = (1,110)_2$

De donde se observa que el resultado es un número negativo, ya que el cuarto bit es igual a uno, por lo tanto, el complemento de él es: 0010, el cual representa al número dos en base decimal.

Así, el resultado en decimal es  $-2_{10}$

d)  $5_{10} - 12_{10}$

$$12_{10} = (0,1100); [0,1100]_2 = (1,0100); 5_{10} = (0,0101)$$

Entonces:

$$\begin{array}{r} 0,0101 \\ + \quad 1,0100 \\ \hline 1,1001 \end{array} \longrightarrow \text{es número negativo.}$$

La magnitud y equivalencia en base diez se determina a partir de su complemento a dos:

$$[1,1001] = (0,0111)_2 = -7_{10}$$

$$e) -3_{10} - 4_{10} = -(0,011)_2 - (0,100)_2 = (1,101)_2 + (1,100)_2 = (1,001)_2$$

El cual se observa que es un número negativo, tal y como se esperaba. Por lo tanto, para conocer la magnitud del mismo, en decimal, empleamos el complemento a 2: 0111, el cual es  $7_{10}$ , por lo que el resultado total será  $-7_{10}$

Queda como ejercicio verificar el resultado si se procede a la operación  $-(3 + 4)$ .

$$f) -3_{10} - 5_{10} = -(0,011)_2 - (0,101)_2 = (1,101)_2 + (1,011)_2 = (1,000)_2$$

El cual se observa que efectivamente es negativo. Para conocer su magnitud calculamos su complemento:  $1000 = \mathbf{;010;}$  El error se identifica si revisamos la operación desde el inicio, ya que el resultado esperado,  $-8$ , requiere de al menos cinco dígitos, cuatro para la magnitud y uno para especificar el signo, por lo tanto la operación correcta debe de ser:

$$-3_{10} - 5_{10} = -(0,0011)_2 - (0,0101)_2 = (1,1101)_2 + (1,1011)_2 = (1,1000)_2$$

Evidentemente se observa que es un número negativo, cuya magnitud se calcula con su complemento:  $01000 = 8_{10}$ , por lo tanto el resultado final es  $-8_{10}$

---



---

**Ejemplo 1.31.** Empleando el complemento a dieciséis, realice la siguiente operación:

$$(FACB)_{16} - (ABCD)_{16}$$

$$ABCD = (0,ABCD); \quad [0,ABCD] = (F,5433)_{16}$$

$$FACB = (0,FACB)$$

$$\begin{array}{r} 0, FAC B \\ + F, \underline{54\ 3\ 3} \\ \hline 0,4\ EFE \end{array}$$

---

**Ejemplo 1.32.** Empleando el complemento correspondiente, realizar las siguientes operaciones y expresar el resultado en base diez

a) Complemento a 4

$$(132)_4 - (231)_4 ; (132)_4 = (0,132)_4$$

$$(231)_4 = (0,231)_4$$

$$[0,231]_4 = (3,103)_4$$

$$\begin{array}{r} 0 \ 1 \ 3 \ 2 \\ + 3 \ 1 \ 0 \ 3 \\ \hline (3 \ 3 \ 0 \ 1)_4 \end{array}$$

Observe que el bit de signo indica que el resultado es negativo, por lo tanto, al obtener el complemento de la suma, se obtiene que la magnitud del resultado es:

$$[3,301]_4 = (0,033) = 15_{10}, \text{ por lo tanto:}$$

$$(3,301)_4 = -15_{10}$$

b)  $(246)_6 - (315)_6$  Complemento a 6

No se puede por que tenemos un seis en el primer dígito y se supone que estamos trabajando en base 6

c)  $(131)_7 - (230)_7$  Complemento a 7

$$(131)_7 = (0,131)_7$$

$$(230)_7 = (0, 230)_7$$

$$[0,230]_7 = (6,440)_7$$

$$0,131$$

$$\underline{6,440}$$

$$6,601$$

Se complementa para conocer la magnitud  $[6,601]_7 = (0,066)_7 = 48_{10}$

Por lo tanto:

$$(6,601)_7 = -48_{10}$$

d)  $(245)_8 - (314)_8$  Complemento a 8

$$(245)_8 = (0,245)_8$$

$$(314)_8 = (0, 314)_8$$

$$[0,314]_8 = (7,464)_8$$

Por lo tanto, la resta será la suma de  $(0,245)_8$  y  $(7,464)_8$

$$\begin{array}{r} 0,245 \\ 7,464 \\ \hline 7,731 \end{array}$$

Se complementa para conocer la magnitud  $[7,731]_8 = (0,047)_8 = 39_{10}$

Por lo tanto:

$$(7,731)_8 = -39_{10}$$

**Ejemplo 1.33.** Empleando el complemento a dos, realice la siguiente operación:

$$-25_{10} - 46_{10}$$

Considerando que la suma generará desbordamiento, se asignará desde el principio un bit adicional al número 46 para contrarrestar ese problema, entonces:

$$\begin{array}{rcl} 25 = (0,0011001)_2 ; [0,0011001]_2 & = & 1,1010010 \\ 46 = (0,0101110)_2 ; [0,0101110]_2 & = & + \underline{1,1100111} \\ & & 1,0111001 \end{array}$$

El cual claramente se observa que es negativo. Para determinar su magnitud y equivalencia en base diez, se obtiene su complemento a dos:  $[1,0111001] = (0,1000111) = 71_{10}$

Por lo tanto:

$$1,0111001 = -71_{10}$$

## 1.6 Complemento disminuido.

El complemento disminuido de un número N sigue representando al negativo de ese número, sólo que el procedimiento de complementaridad se modifica un poco, facilitando el proceso. En particular, el complemento a uno es un recurso altamente empleado, tal y como se mencionó con anterioridad, en calculadoras, computadoras y cualquier dispositivo digital electrónico que realice operaciones aritméticas.

La representación del complemento disminuido viene expresado por:

$$[\overline{N}]_r = ((r)^n)_r - (N)_r - 1$$

$$\text{Por lo tanto } [N]_r = [\overline{N}]_r + 1$$

Ejemplos de diferentes complementos disminuidos se muestran a continuación.

**Ejemplo 1.34.** Del término  $01100101_2$  obtenga su complemento a uno y relaciónelo con el complemento a dos.

La operación  $r-1-a_i$  es:

$$\begin{array}{r} 11111111 \\ 01100101 \\ \hline 10011010 \end{array}$$

Por lo tanto:  $\overline{[01100101]}_2 = 10011010$

La relación entre complemento a uno y complemento a dos está dada por:  $\overline{[N]}_r + 1 = [N]_r$

$$\begin{array}{r} 10011010 \\ + 00000001 \\ \hline 10011011 \end{array}$$

Por lo que:  $[N]_2 = 10011011$

**Ejemplo 1.35.** Realice las siguientes operaciones tanto en complemento a dos como en complemento a uno.

a)  $-A - B$ ;  $A = 100110$ ;  $B = 1001$

*Complemento a 2*

$$[A]_2 = (1,011010)$$

$$[B]_2 = (1,110111)$$

$$\begin{array}{r} 1,011010 \\ + 1,110111 \\ \hline 11,010001 \end{array}$$

Por lo tanto:  $-A - B = 1,010001$

*Complemento disminuido:*

$$[\overline{A}]_2 = (1,011001); [\overline{B}]_2 = (1,110110)$$

$$\begin{array}{r} 1,011001 \\ + 1,110110 \\ \hline 11,001111 \end{array}$$

Por lo tanto:  $-A - B = (1,001111)$

La relación entre  $\overline{[N]}_2$  y  $[N]_2$  se realiza por el ajuste de dos unidades al resultado de la suma en complemento a uno, ya que cada sumando requiere de su ajuste a uno, y que en la operación existen dos de ellos.

$$\begin{array}{r} 1,001111 \\ + 0,000010 \\ \hline 1,010001 \end{array}$$

b)  $-A - B$ ;  $A = 110111$ ;  $B = 1101$

*Complemento a 2*

$$[A]_2 = (1,1001001); [B]_2 = (1,1110011)$$

$$\begin{array}{r} 1,1001001 \\ + 1,1110011 \\ \hline 11,0111100 \end{array}$$

Por lo tanto:  $-A - B = (1,0111100)$

*Complemento a uno*

$$[\overline{A}]_2 = (1,1001000); [\overline{B}]_2 = (1,1110010)$$

$$\begin{array}{r} 1,1001000 \\ +1,1110010 \\ \hline 11,0111010 \end{array}$$

Por lo tanto,  $-A -B$  en complemento a uno:  $(1,0111010)$

Verifique la relación entre los complementos.

**Ejemplo 1.36.** Realice las siguientes operaciones

$$\begin{array}{r} \text{EAF2D} \\ - \text{F35DE} \\ \hline \end{array} \qquad \begin{array}{r} - 9\text{AE13} \\ - \text{A35DE} \\ \hline \end{array}$$

- empleando el complemento a 15
- exprese en base 10 los resultados del inciso a.

$$\text{a) } \text{EAF2D} = (0, \text{EAF2D}); \quad \text{F35DE} = (0, \text{F35DE})$$

$$\text{Obteniendo el complemento a 15: } [\overline{0, \text{F35DE}}]_{16} = (\text{F}, 0\text{CA21})$$

$$\begin{array}{r} 0, \text{EAF2D} \\ + \text{F}, 0\text{CA21} \\ \hline \text{F}, \text{F794E} \end{array}$$

El resultado es negativo, y se encuentra en complemento a 15. Pasaremos a complemento 16

$$\begin{array}{r} \text{F}, \text{F794E} \\ + \quad \quad \quad 1 \\ \hline \text{F}, \text{F794F} \end{array}$$

El resultado sigue siendo negativo, pero en complemento 16, por lo tanto, para calcular la magnitud en base 16, se obtendrá el complemento a 16 de este último resultado.

$$\begin{array}{r} [\text{F}, \text{F794F}]_{16} = \quad 086\text{B0} \\ + \quad \quad \quad 1 \\ \hline \quad \quad \quad 86\text{B1} \end{array}$$

Expresado en base 10:

$$(\text{N})_{10} = 8 * (16)^3 + 8 * (16)^2 + 6 * (16)^1 + 1 * (16)^0 = 3448_{10}$$

$$\text{F}, \text{F794F}_{16} = - 3448_{10}$$

$$\text{b) } 9\text{AE13} = (0, 09\text{AE13}); \quad \text{A35DE} = (0, 0\text{A35DE})$$

Obteniendo el complemento a 15 para ambas expresiones:

$$[\overline{0,09AE13}]_{16} = (F, F651EC); [\overline{0,0A35DE}]_{16} = (F, F5CA21)$$

$$\begin{array}{r} F, F651EC \\ + F, F5CA21 \\ \hline F, EC1C0D \end{array}$$

El resultado es negativo, y se encuentra en complemento a 15. Pasaremos a complemento 16

$$\begin{array}{r} F, EC1C0D \\ + \quad \quad \quad 2 \\ \hline F, EC1C0F \end{array}$$

El resultado sigue siendo negativo, pero en complemento 16. Para la magnitud en base 16, se obtendrá el complemento a 16 del resultado final.

$$[F, EC1C0F]_{16} = 0, 13E3F1$$

En base 10:

$$(N)_{10} = 1 * (16)^5 + 3 * (16)^4 + 14 * (16)^3 + 3 * (16)^2 + 15 * (16)^1 + 1 * (16)^0 = 1303537_{10}$$

$$F, EC1C0D_{16} = -1303537_{10}$$

**Ejemplo 1.37** Realice la operación  $-A-B-C$ , siendo  $A= 100110$ ;  $B= 101$ ;  $C= 110110$

- empleando el complemento a uno
- calcule en base 10 el resultado obtenido

$$A= (0, 0100110); B= (0, 0000101); C= (0, 0110110)$$

$$[\overline{A}]_2 = (1,1011001); [\overline{B}]_2 = (1,1111010); [\overline{C}]_2 = (1,1001001)$$

$$\begin{array}{r} (-A) + (-B) = \quad 1, 1011001 \\ \quad \quad \quad 1, 1111010 \\ \hline \quad \quad \quad 1, 1010011 \end{array}$$

$$\begin{array}{r} (-A) + (-B) + (-C) = 1, 1010011 \\ \quad \quad \quad 1, 1001001 \\ \hline \quad \quad \quad 1, 0011100 \end{array} \quad \leftarrow \text{Negativo}$$

Para obtener el complemento a 2, se tiene que agregar tres unidades, uno por cada número A,B y C ya que todos se encuentran en complemento a uno.

$$\begin{array}{r} 1, 0011100 \\ \quad \quad \quad 11 \\ \hline 1, 0011111 \end{array}$$

Este último resultado sigue siendo negativo, pero en complemento a dos. Para determinar su equivalente en base 10 primero se obtiene el complemento a dos de él.



$$[1, 0011111] = 0,1100001$$

Ahora se obtiene su equivalente en base 10 :  $(N)_{10} = 1 * (2)^6 + 1 * (2)^5 + 1 * (2)^0 = 97_{10}$

Resultado:  $1,0011111 = -97_{10}$

## 1.7 Códigos

Un código es un sistema elaborado y diseñado para transmitir, manipular y aplicar información, de preferencia pública o privada, ocasionalmente para ambos al mismo tiempo. Emplea símbolos para representar la información que se desea construir, además de que obedece a una semántica específica y tiene correspondencia unívoca, esto es, que un solo código representa a un solo significado.

Definitivamente dicho sistema de codificación debe de guardar una estructura y por consiguiente debe de comunicar información útil para el usuario.

Ejemplo de un código son los colores de la luz de un semáforo. En este caso la aplicación del código es pública; los símbolos que emplea son los colores: verde, amarillo y rojo; su semántica corresponde a *sigas, precaución y alto*, respectivamente; la estructura que guarda es única, tanto en colores como en secuencia: *verde – amarillo – rojo – verde* y así sucesivamente; en cuanto a la utilidad de este código es más que evidente, considerando la universalidad, aceptabilidad y actos de cultura que ha generado en el comportamiento cívico alrededor del mundo.

En los sistemas digitales existen códigos binarios para aplicaciones específicos. Tal es el caso de los códigos BCD, Gray, ASCII, detección de errores, entre otros. Algunos se emplean para manipulación o representación numérica (BCD y Gray), otros para representación alfanumérica (ASCII), y existen los de aplicación varia, como los empleados para detectar errores en el envío y recepción de información entre dos dispositivos. En este capítulo se revisarán únicamente los códigos BCD, Gray y ASCII, los demás quedan fuera del alcance de este libro y se recomienda al lector referirse a la bibliografía correspondiente para la lectura del resto de ellos.

### 1.7.1 Código BCD

Se emplea para operaciones numéricas. Es una alternativa para representar al sistema numérico decimal utilizando sólo cuatro dígitos. La relación entre este código y su equivalente en binario se muestra en la tabla 1.6

Decimal	Binario	BCD
0	0000	0000
1	0001	0001
2	0010	0010
3	0011	0011
4	0100	0100
5	0101	0101
6	0110	0110
7	0111	0111
8	1000	1000
9	1001	1001

Tabla 1.6 Relación entre los números decimal, binario y BCD

**Ejemplo 1.38.** Equivalencias decimal – binario – BCD de diferentes números.

Decimal	Binario	BCD
10	1010	0001 0000
15	1111	0001 0101
34	100010	0011 0100
99	1100011	1001 1001
235	11101011	0010 0011 0101

**Ejemplo 1.39.** Realizar las siguientes operaciones numéricas, empleando código BCD.

$$\text{A) } \begin{array}{r} 3 \\ + 1 \\ \hline 4 \end{array} = \begin{array}{r} 0011 \\ + 0001 \\ \hline 0100 \end{array}$$

$$\text{B) } \begin{array}{r} 31 \\ + 15 \\ \hline 46 \end{array} = \begin{array}{r} 0011\ 0001 \\ + 0001\ 0101 \\ \hline 0100\ 0110 \end{array}$$

$$\text{C) } \begin{array}{r} 5 \\ + 6 \\ \hline 11 \end{array} ; \begin{array}{r} 0101 \\ + 0110 \\ \hline 1011 \end{array}$$

Excede el límite del código BCD, por lo tanto es necesario agregar al resultado el número 6 expresado en BCD

$$\begin{array}{r} 0101 \\ + 0110 \\ \hline 1011 \\ + 0110 \\ \hline 0001\ 0001 \end{array} \longrightarrow \text{Resultado} = 11_{10}$$

$$\begin{array}{r} 9 \\ + 8 \\ \hline 17 \end{array} ; = \begin{array}{r} 1001 \\ + 1000 \\ \hline 1\ 0001 \end{array}$$

↑  
carry

El número final queda excedido con un Carry, por lo tanto se deberá de realizar la operación de suma con el número 6 en BCD

$$\begin{array}{r}
 1001 \\
 + 1000 \\
 \hline
 1\ 0001 \\
 + 0110 \\
 \hline
 0001\ 0111
 \end{array}$$

$\downarrow \quad \downarrow$   
 1      7

Con lo que el resultado es  $17_{10}$

**Ejemplo 1.40** Realizar la suma en BCD de los números  $67 + 53$

$6 = 0110$ ;  $7 = 0111$ ;  $5 = 0101$ ;  $3 = 0011$ ;  $67 = 0110\ 0111$ ;  $53 = 0101\ 0011$ , entonces:

$$67 + 53 = 0110\ 0111 + 0101\ 0011 = 1011\ 1010.$$

Obsérvese que la representación  $1010$  no tiene significado en BCD. Para ajustar esta incongruencia, se deberá de sumar el número seis en BCD a esa expresión.

$$\begin{array}{r}
 1011\ 1010 \\
 + \quad 0110 \\
 \hline
 1100\ 0000
 \end{array}$$

Así mismo, deberá de ajustarse la expresión  $1100$  agregándole el número seis.

$$\begin{array}{r}
 1100\ 0000 \\
 + 0110 \\
 \hline
 1\ 0010\ 0000
 \end{array}$$

$$67 + 53 = 0001\ 0010\ 0000_{\text{BCD}}$$

### 1.7.2 Código Gray.

Entre sus características podemos mencionar que es un código en donde no es relevante el “peso”, según la posición que guarda el bit. Así mismo, presenta un solo cambio en bit entre un número y su subsecuente. Por último, esta herramienta se puede diseñar con un número no determinado de bits, esto es, se puede contar con un código Gray de dos, tres, cuatro o tantos dígitos como el usuario considere apropiado.

Para mostrar las ventajas de éste código con respecto a otro que modifique más de un bit a la vez se empleará el siguiente ejemplo. Suponga que en un sistema de control de posición de un robot manipulador, el cual emplea servomotores, se aplica un código de detección de posición montado en el eje de la flecha del motor, de acuerdo a la figura 1.2. Para la detección y medición de la posición se emplearán tres sensores ópticos de reflexión. En este caso el color oscuro representa placa

metálica no perforada, mientras que el color blanco representa placa perforada. Así cuando la placa gire el receptor del sensor se activará o no por efecto del espejo colocado justo en línea recta tras la placa. De esta manera, se puede identificar una secuencia establecida de acuerdo al código binario con tres bits, esto es, se identifican las posiciones 000, 001, 010, 011, 100, 101, 110 y 111. Como se observará, la precisión de dicho instrumento no es muy apropiada ya que se tiene una relación de  $360^\circ/8$ . Sin embargo ése no es el principal problema en la aplicación.

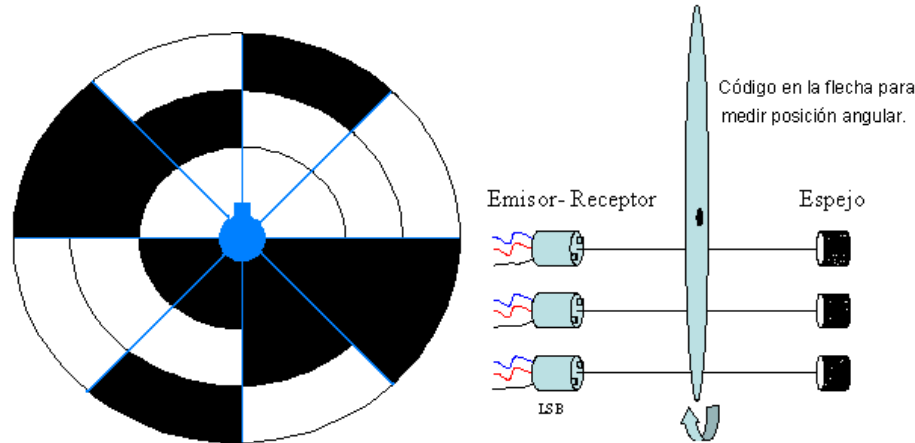


Figura 1.2 Código de posición montado en la flecha de un motor.

Para encontrar otro más grave, considérese lo siguiente. Suponga que el disco gira en cualquier sentido de modo que lo podamos detener entre la frontera de dos posiciones, digamos entre 101 y 110. La cuestión es, si se encuentra en la frontera, ¿cuál será la señal que captarán los receptores? En este caso ambos valores podrían ser aceptables, generándose por lo tanto un problema de medición que repercutiría de manera inmediata en la estabilidad del eslabón del robot debido a una mala decisión tomada por el controlador, como consecuencia de una errónea medición. El problema se hace mayor si existen más de dos bits cambiando al mismo tiempo, por ejemplo, entre 000 y 111. Los dos problemas anteriores se pueden disminuir en gran medida agregando más bits a la placa y, sobre todo, modificando la estrategia de codificación. En este caso en el código Gray modifica de manera sustancial la posibilidad de error. Observe la figura 1.3 del siguiente ejemplo.

**Ejemplo 1.41.** Implementar el código Gray de 3 bits para medir la posición angular de la flecha de un motor eléctrico.

Decimal	Binario	Gray
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

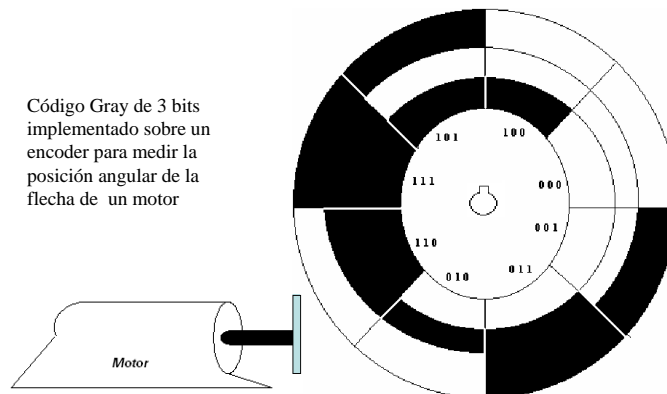


Figura 1.3 Código Gray de tres bits sobre la flecha de un motor.

### 1.7.3 Código Gray con cuatro dígitos.

Como su nombre lo dice, emplea cuatro dígitos para construir sus 16 combinaciones, por lo tanto si se aplicará a la flecha del motor nos ofrecería mayor precisión en la medida angular, esto es, hasta  $360^\circ/16$ . En la tabla 1.7 se observan las 16 combinaciones correspondientes.

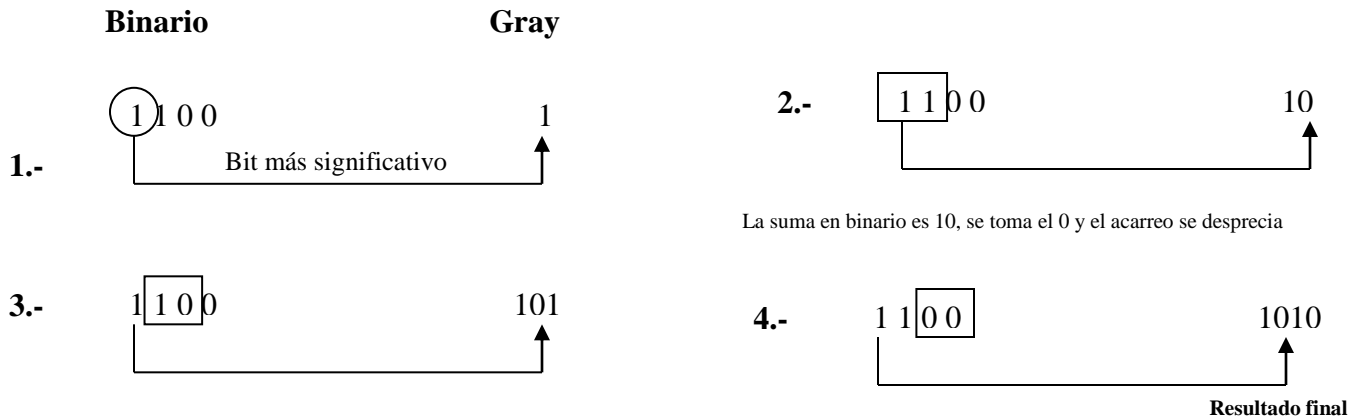
Decimal	Binario	Gray	Decimal	Binario	Gray
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

Tabla 1.7 Código Gray de cuatro dígitos

#### Conversión Binario - Gray.

- 1.- El bit más significativo del número Gray será el mismo tal cual el número en Binario.
- 2.- Posteriormente deberá de hacerse la suma en binario de los bits contiguos, de dos en dos, del número binario. Lo que resulte será el valor del siguiente bit en el código Gray.

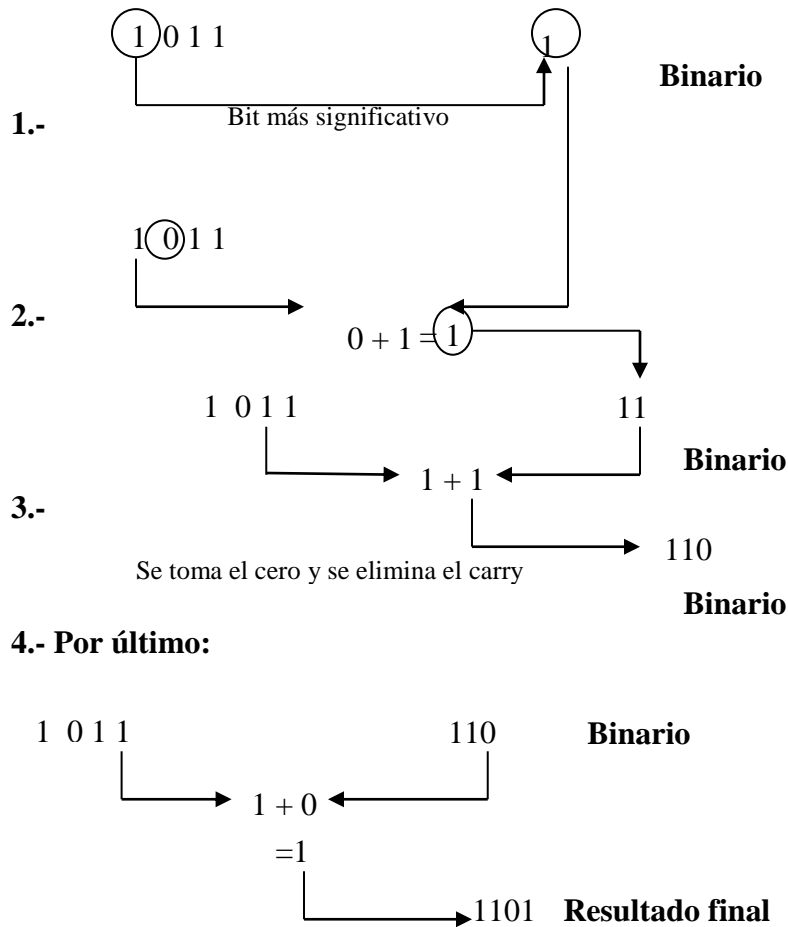
**Ejemplo 1.42.** Convertir el número binario 1100 en Gray



#### Conversión Gray - Binario

- 1.- Para convertir de Gray a binario se sigue manteniendo el valor del bit más significativo.
- 2.- Sumar cada bit generado en código Gray con el bit adjunto del número en binario. Así se genera el siguiente bit en binario, sin considerar los acarreos.

**Ejemplo 1.43.** Convertir el siguiente número en Gray a su equivalente en binario.



### 1.5.3 Código ASCII.

Transmitir información requiere del uso de códigos no sólo numéricos: números, letras, símbolos de puntuación, caracteres matemáticos y comandos no desplegables necesitan ser representados por un código específico. Por ejemplo, suponga que desea escribir las letras del teclado de una computadora, 27 incluyendo la ñ. Sin embargo, considerando que se emplearán tanto minúsculas como mayúsculas, la cantidad de caracteres crece a 54. Por lo tanto será necesario seleccionar un código que al menos cubra esa cantidad, que en este caso sería uno con al menos seis dígitos o bits. El más común, más no necesariamente el único, es el código ASCII (*American Standar Code for Information Interchange*). Consiste de 7 bits y representa a 128 caracteres, los cuales pueden ser revisados en la tabla 1.8. Cada vez que el usuario oprime una tecla de una computadora, un conjunto de 7 bits de información son transmitidos internamente al procesador, el cual interpreta y opera de acuerdo al carácter que representa. Por lo tanto, una palabra quedará representada por toda una cadena de bits, así, el texto “Hoy examen” transmitirá de izquierda a derecha el siguiente conjunto de información.

0100010 1001000 1101111 1111001 0100000 1100101 1111000 1100001 1101101 1100101 1101110 0100010

Caracter	ASCII 7 bits	Caracter	ASCII 7 bits	Caracter	ASCII 7 bits
NUL	0000000	'	0101100	X	101 1000
SOH	0000001	-	0101101	Y	101 1001
STX	0000010	.	0101110	Z	101 1010
ETX	0000011	/	0101111	[	1011011
EOT	0000100	0	011 0000	\	1011100
ENQ	0000101	1	011 0001	]	1011101
ACK	0000110	2	011 0010	^	1011110
BEL	0000111	3	011 0011	_	1011111
BS	0001000	4	011 0100	`	1100000
HT	0001001	5	011 0101	a	1100001
LF	0001010	6	011 0110	b	1100010
VT	0001011	7	011 0111	c	1100011
FF	0001100	8	011 1000	d	1100100
CR	0001101	9	011 1001	e	1100101
SO	0001110	:	0111010	f	1100110
SI	0001111	;	0111011	g	1100111
DEL	0010000	<	0111100	h	1101000
DC1	0010001	=	0111101	i	1101001
DC2	0010010	>	0111110	j	1101010
DC3	0010011	?	0111111	k	1101011
DC4	0010100	@	1000000	l	1101100
NAK	0010101	A	100 0001	m	1101101
SYN	0010110	B	100 0010	n	1101110
ETB	0010111	C	100 0011	o	1101111
CAN	0011000	D	100 0100	p	1110000
EM	0011001	E	100 0101	q	1110001
SUB	0011010	F	100 0110	r	1110010
ESC	0011011	G	100 0111	s	1110011
FS	0011100	H	100 1000	t	1110100
GS	0011101	I	100 1001	u	1110101
RS	0011110	J	100 1010	v	1110110
US	0011111	K	100 1011	w	1110111
"Espacio"	0100000	L	100 1100	x	1111000
!	0100001	M	100 1101	y	1111001
"	0100010	N	100 1110	z	1111010
#	0100011	O	100 1111	{	1111011
\$	0100100	P	101 0000		1111100
%	0100101	Q	101 0001	}	1111101
&	0100110	R	101 0010	~	1111110
,	0100111	S	101 0011	Del	1111111
(	0101000	T	101 0100		
)	0101001	U	101 0101		
*	0101010	V	101 0110		
+	0101011	W	101 0111		

Tabla 1.8 Código ASCII

## Ejercicios

1.- Liste en orden ascendente los primeros veinte dígitos de los sistemas numéricos de base tres, cinco y trece.

2.- Realizar las siguientes conversiones empleando los métodos de multiplicación y/o división.

a) Convertir  $(0.2543)_{10}$  a su equivalente en Octal

b) Convertir  $(0.5321)_{10}$  a su equivalente en binario

3.- El equivalente en base 8 del número  $234_{10}$  es:

a)  $352_{16}$

b)  $351_8$

c)  $352_8$

d)  $461_8$

4.- El equivalente en base 16 del número  $234_{10}$  es:

a)  $EE_{16}$

b)  $EB_{16}$

c)  $EA_{16}$

d)  $461_8$

5.- El equivalente a base once del número  $18.6_9$  es igual:

a)  $16.37_{11}$

b)  $61.37_{11}$

c)  $16.73_{11}$

d)  $AF25_{11}$

6.- El equivalente a base ocho del número  $0.1285_{10}$  es igual a:

a)  $0.1016_8$

b)  $0.6101_8$

c)  $1.016_8$

d)  $1.016_8$

7.- La conversión correcta a base seis del número  $(3BA.25)_{14}$  es:

a)  $2523.01_6$

b)  $2523.10_6$

c)  $3252.10_6$

d)  $3252.22_6$

8.- Liste en código Gray a cinco bits, los números decimales siguientes:

Decimal	Gray
2	
3	
5	
9	
12	
15	
18	
21	
26	

9.- El resultado en Código BCD de la operación numérica  $25_{10} - 15_{10}$ , es:

a) 0000 0001

b) 1 0000 0001

c) 0001 0000

d) 74H00

10.- El resultado en Código BCD de la operación numérica  $0110\ 0111 + 0101\ 0011$  es:

a) 1011 1010

b) 0010 0000

c) 01100110

d) 000100100000

11.- Realice la operación  $11_{10} + 9_{10}$  en código BCD y muestre el resultado en código GRAY.

12.- Realice las siguientes operaciones y exprese el resultado en binario.

a)  $(14.5)_8 + (2A.2)_{16}$

b)  $(17.5)_8 + (1A.2)_{16}$

c)  $(12.7)_8 + (1E.2)_{16}$

d)  $(17.6)_8 + (1C.9)_{16}$



13.- Realice las siguientes operaciones y exprese el resultado en base tres.

- a)  $(5.4)_9 + (1.2)_4$
- b)  $(4.5)_7 + (2.1)_4$
- c)  $(3.8)_9 + (2.2)_5$
- d)  $(5.4)_9 + (1.2)_3$
- e)  $(7.4)_8 + (2.2)_3$
- f)  $(18.4)_9 + (12.2)_3$

14.- Del ejercicio anterior, exprese el resultado en las bases cuatro, cinco y seis.

15.- La suma de  $4163_8 + 7520_8$  es igual a:

- a)  $14703_8$
- b)  $14708_8$
- c)  $13703_8$
- d)  $13713_8$

16.- La suma de:  $2A58_{16} + 71D0_{16}$  es igual a:

- a)  $9C28_{16}$
- b)  $9D28_{16}$
- c)  $9CG8_{16}$
- d)  $9C38_{16}$

17.- La resta de:  $6204_8 - 5173_8$  es igual a:

- a)  $1011_8$
- b)  $1012_8$
- c)  $1923_8$
- d)  $11011_8$

18.- La resta de:  $9F1B_{16} - 4A36_{16}$  es igual a :

- a)  $9C28_{16}$
- b)  $9D28_{16}$
- c)  $55E5_{16}$
- d)  $54E5_{16}$

19.- Empleando el complemento a dos, realice las operaciones siguientes.

- a)  $12_{10} - 5_{10}$
- b)  $5_{10} - 12_{10}$
- c)  $-9_{10} - 5_{10}$
- d)  $-19_{10} - 15_{10}$

20.- Empleando el complemento a dos realice las siguientes operaciones.

- a)  $-62_{10} - 40_{10}$
- b)  $-40_{10} - 20_{10}$
- c)  $-30_{10} - 35_{10}$

21.- El complemento a uno de  $01100101_2$  es:

- a)  $10011010_2$
- b)  $10011011_2$
- c)  $10011011_2$
- d)  $11011010_2$

22.- El complemento a uno de  $11010100_2$  es:

- a)  $00101011_2$
- b)  $10011011_2$
- c)  $10011011_2$
- d)  $11011010_2$

23.- Realice las operaciones siguientes y exprese el resultado en base tres, cuatro, cinco y seis.

- a)  $-(5.4)_9 - (1.2)_4$
- b)  $-(5.4)_8 - (2.3)_5$

24.-Empleando el complemento correspondiente, realizar las siguientes operaciones y expresar el resultado en base diez.

- a)  $(132)_4 - (231)_4$ . Complemento a 3 y 4.
- b)  $(245)_6 - (315)_6$ . Complemento a 5 y 6.
- c)  $(131)_7 - (230)_7$ . Complemento a 6 y 7.
- d)  $(245)_8 - (314)_8$ . Complemento a 7 y 8.

25.- La operación numérica  $1001_2 - 1111_2$  es igual a:

- a)  $11000_{c1}$
- b)  $2110_{c2}$
- c)  $11101_{c2}$
- d)  $11001_{c1}$

26.- La operación numérica  $-1001_2 - 0011_2$  es igual a:

- a)  $11000_{c2}$
- b)  $2110_{c1}$
- c)  $10010_{c1}$
- d)  $10010_{c2}$

27.- Seleccione el resultado correcto de la operación  $101_2 - 1100_2$ .

- a)  $7_{10}$
- b)  $351_8$
- c)  $-7_{10}$
- d)  $-6_{10}$

28.- Empleando complemento a 16, la opción apropiada para la operación de resta  $(FACB)_{16} - (ABCD)_{16}$ , es:

- a)  $1,4EFF$
- b)  $0,4EFE$
- c)  $1,4EFE$
- d)  $0, FFFF$

29.- Seleccione el resultado correcto de la operación  $-25_{10} - 46_{10}$ , empleando el complemento a 2.

- a)  $(1,1101010)_{2c}$
- b)  $-(1,0010101)_{2c}$
- c)  $(0,0010101)_{2c}$
- d)  $(1,0111001)_{2c}$

30.- Empleando complemento a 15, la opción apropiada para la operación de resta  $(FACB)_{16} - (ABCD)_{16}$ , es:

- a)  $4EFD$
- b)  $4EFE$
- c)  $3EFD$
- d)  $3EFF$

31.-Empleando el complemento a dos, realice la operación correspondiente y el resultado obtenido mostrarlo en base 10. **Nota:** agregar el bit de signo a los números A y B.

- a)  $-A - B =$ ;  $A = 100110$ , sea  $B = 1001$ .
- b)  $-A - B =$ ;  $A = 110110$ , sea  $B = 1011$

32.-Empleando el complemento a uno, realice la operación y el resultado obtenido expresarlo en base 10. **Nota:** agregar el bit de signo a los números A y B

- a)  $-A - B$ ;  $A = 110010$ , sea  $B = 1110$ .
- b)  $-A - B - C$ ;  $A = 100110$ ;  $B = 101$ ;  $C = 110110$

33.- Realice la operación siguiente empleando complemento a uno y exprese el resultado en base diez:  $-(5.4)_{10} - (2.3)_{10}$

34.- Realice las siguientes operaciones empleando el complemento a quince y el resultado obtenido expresarlo en base 10

- a)  $\begin{array}{r} EAF2D \\ -F35DE \\ \hline \end{array}$
- b)  $\begin{array}{r} -9AE13 \\ -A35DE \\ \hline \end{array}$

## Bibliografía

Floyd, T., *Digital Fundamentals*, 8<sup>th</sup> edition, Prentice Hall, New Jersey, 2003.

Mano, M., *Digital Design, Third Edition*, Prentice Hall, Englewood Cliffs, NJ, 2002

Nelson, P., Nagle H., Carroll D., and Irwin J., *Digital Logic Analysis and Design*, Prentice Hall, Englewood Cliffs, NJ, 1995.

Roth, C., *Fundamentals of Logic Design*, 5<sup>th</sup> edition, Thomson Brooks/Cole, Belmont CA, 2004.

Tocci J., Widmer S., *Sistemas Digitales, principios y aplicaciones*, Octava edición, Naucalpan de Juárez, México, 2003.

Wakerly F., *Diseño Digital y Prácticas*, Prentice Hall, Naucalpan de Juárez, México, 1992.