

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_1.** Realizar las siguientes operaciones en binario natural (verificar las respuestas en decimal):

	<b>Solución</b>		<b>Solución</b>
1. $1100110_2 + 1001011_2$	10110001	2. $1011_2 - 0101_2$	110
3. $11_2 + 11_2$	110	4. $1000_2 - 101_2$	11
5. $100_2 + 10_2$	110	6. $1100_2 - 1000_2$	100
7. $111_2 + 1_2$	1000	8. $1110001_2 \times 111_2$	1100010111
9. $110_2 + 100_2$	1010	10. $101010_2 \times 1001_2$	101111010
11. $1100_2 + 1000_2$	10100	12. $1011_2 \times 101_2$	110111
13. $1010_2 + 1011_2$	10101	14. $1100_2 \times 101_2$	111100
15. $1001_2 + 1011_2$	10100	16. $1011_2 \times 11_2$	100001
17. $1_2 + 1_2 + 1_2 + 1_2 + 1_2$	101	18. $1001111_2 \times 0110_2$	111011010
19. $1_2 + 1_2 + 1_2 + 1_2 + 1_2 + 1_2$	110	20. $1100_2 : 100_2$	11
21. $11_2 - 01_2$	10	22. $101100_2 : 100_2$	1011
23. $11_2 - 10_2$	01	24. $100100_2 : 11_2$	1100
25. $111_2 - 100_2$	011	26. $110000_2 : 110_2$	1000
27. $101_2 - 010_2$	11		

**U6\_2.** Realizar la operación  $0001000_2 - 111001_2$  en donde los operandos están en binario natural. Utilice 10 bits en las operaciones.

- Comprobar que en binario natural no se puede realizar la operación
- En complemento a 2

**Solución:**

- Como el resultado es negativo, no es representable en binario natural.
- 1111001111

**U6\_3.** Efectuar las siguientes restas en las que los operandos están en binario natural, dando los resultados binario y C-2 con 10 bits (verificar las respuestas en decimal):

- $10000000 - 110111$
- $00111111 - 00011100$
- $00011100 - 00111111$
- $100001 - 10000$

**Solución:**

- 0001001001 y 0001001001
- 0000100011 y 0000100011
- El resultado es negativo, no se puede representar binario natural, pero si en complemento a 2: 1111011101
- 0000010001 y 0000010001

**U6\_4.** Efectuar las siguientes restas, dando los resultados en C-2 con 10 bits

- $20_{10} - 30_{10}$
- $51_{10} - 64_{10}$
- $12_{10} - 19_{10}$

**Solución:**

- 1111110110
- 1111110011
- 1111111001

**U6\_5.** Representar los siguientes números en signo-magnitud y C-2 utilizando 7 bits:

- $+14_{10}$
- $-15_{10}$

**Solución:**

- 0001110 y 0001110
- 1001111 y 1110001

**U6\_6.** Dados los siguientes números binarios y sus representaciones, indicar el equivalente en decimal:

	<b>Solución</b>		<b>Solución</b>
1. $000011_2$ (C2)	$+3_{10}$	2. $1001001_2$ (SM)	$-9_{10}$
3. $1100101_2$ (C2)	$-27_{10}$		

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_7.** Realizar las siguientes operaciones teniendo en cuenta que todos los números son enteros en la representación en complemento a 2 con 12 bits. Dar el resultado tanto en la representación en complemento a 2 con 12 bits como su equivalente en decimal.

	<b>Solución</b>
1. 7D0 + 3E8	Desbordamiento en C-2 de 12 bits (Res: 3000 <sub>10</sub> )
2. A24 – 5DC	Desbordamiento en C-2 de 12 bits (Res: 1096 <sub>10</sub> )
3. 496 + 0EB	581 <sub>16</sub> / 1409 <sub>10</sub>
4. 0AF – 7D0	8DF <sub>16</sub> / -1825 <sub>10</sub>
5. 3E8 + 418	Desbordamiento en C-2 de 12 bits (Res: 2048 <sub>10</sub> )
6. 4B0 – 258	258 <sub>16</sub> / 600 <sub>10</sub>
7. B3C + 216	D52 <sub>16</sub> / -686 <sub>10</sub>
8. D44 + 2BC	000 / 0

**U6\_8.** Codificar en BCD:

	<b>Solución</b>		<b>Solución</b>
1. 235 <sub>10</sub>	0010 0011 0101	2. 956 <sub>10</sub>	1001 0101 0110
3. 35 <sub>10</sub>	0011 0101	4. 45785 <sub>10</sub>	0100 0101 0111 1000 0101
5. 98 <sub>10</sub>	1001 1000	6. 826 <sub>10</sub>	1000 0010 0110
7. 170 <sub>10</sub>	0001 0111 0000	8. 1807 <sub>10</sub>	0001 1000 0000 0111
9. 2469 <sub>10</sub>	0010 0100 0110 1001	10. 2905 <sub>10</sub>	0010 1001 0000 0101

**U6\_9.** Decodificar los siguientes números representados en BCD:

	<b>Solución</b>
1. 0001 0000 0000 0011 0100	10034
2. 1001 1100 0010	Imposible, 1100 no es BCD
3. 0001 0011 0101 0111	1357

**U6\_10.** Realizar las siguientes sumas en BCD:

	<b>Solución</b>
1. 10011000+00110101	0001 0011 0011
2. 1001+1001	0001 1000
3. 100110010011+010101101000	0001 0101 0110 0001
4. 00110111+01000110	1000 0011
5. 00100011+00010101	0011 1000
6. 010101100001+011100001000	0001 0010 0110 1001
7. 10000110+00010011	1001 1001
8. 01010001+01011000	0001 0000 1001
9. 01100111+01010011	0001 0010 0000
10. 10011000+10010111	0001 1001 0101

**U6\_11.** Decodificar el siguiente texto en ASCII:

59 6F 75 20 68 61 76 65 20 73 75 63 63 65 65 64 65 64 2E

**Solución:**

You have succeeded

**U6\_12.** Codificar el siguiente texto en ASCII (la comilla simple no pertenece al mensaje): 'Feliz Año'

**Solución:**

46 65 6C 69 7A 20 41 A4 6F

**U6\_13.** Completar los siguientes números añadiendo un bit de paridad siguiendo el criterio de paridad par:

<b>Solución</b>	<b>Paridad par</b>
1	1001001 <sub>2</sub> (a)
1	000010101 <sub>2</sub> (b)
0	10111101 <sub>2</sub> (c)

<b>Solución</b>	<b>Paridad impar</b>
0	0110001 <sub>2</sub> (a)
0	0111110 <sub>2</sub> (b)
1	10111000 <sub>2</sub> (c)

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_14.** Se pretende transmitir el texto "Hola" en código ASCII (básico con 7 bits) con VRC, LRC y paridad cruzada, utilizando paridad par. Construir los códigos hexadecimales (por columnas) que realmente se transmitirán (los datos se dispondrán en una matriz 7x4, añadiendo la fila de paridades como primera fila y la columna de paridades como última columna).

**Solución:**

0	0	0	1	1
1	1	1	1	0
0	1	1	1	1
0	0	0	0	0
1	1	1	0	1
0	1	1	0	0
0	1	0	0	1
0	1	0	1	0

.....48 6F 6C E1 AA

**U6\_15.** Un circuito lógico tiene 5 entradas y una salida. Cuatro de las entradas A, B, C, y D representan un dígito decimal en BCD y la quinta entrada, E, es de control. Cuando el control está en '0' lógico, la salida vale '0' si el número decimal es par y vale '1' si es impar. Cuando el control está a '1' lógico, la salida vale '0' cuando el número es múltiplo de 3 y vale '1' en el resto de los casos. Se pide diseñar el circuito. NOTA: Se considera que el cero es un múltiplo de 3.

**Solución:**

$$Z = D \bar{E} + \bar{A} \bar{C} D + BD + B \bar{C} E + A \bar{D} E + \bar{B} C \bar{D} E$$

**U6\_16.** Un circuito lógico acepta como entradas dos números enteros de 2 bits  $A = A_1A_0$  y  $B = B_1B_0$  y suministra una salida de 4 bits  $P = P_3P_2P_1P_0$  que es el producto numérico de A y B. Se pide diseñar y dibujar el circuito correspondiente.

**Solución:**

$$P_0 = A_0B_0; \quad P_1 = A_1B_0 (\bar{A}_0 + \bar{B}_1) + A_0B_1 (\bar{A}_1 + \bar{B}_0);$$

$$P_2 = A_1B_1 (\bar{A}_0 + \bar{B}_0); \quad P_3 = A_1A_0B_1B_0$$

**U6\_17.** a) Mostrar en una tabla con 4 entradas ( $X_1X_0$  e  $Y_1Y_0$ ) que representan a dos números binarios X e Y de 2 bits en complemento a dos, el resultado de la suma  $X+Y$ , también en complemento a 2. Utilice el menor número de bits de salida que necesite. b) Expresar en maxterms la función par el bit de signo del resultado. c) Expresar también esta misma función como producto de sumas en la forma más simplificada posible.

**Solución: a)**

$X_1$	$X_0$	$Y_1$	$Y_0$	$S_2$	$S_1$	$S_0$		$X_1$	$X_0$	$Y_1$	$Y_0$	$S_2$	$S_1$	$S_0$
0	0	0	0	0	0	0		1	0	0	0	1	1	0
0	0	0	1	0	0	1		1	0	0	1	1	1	1
0	0	1	0	1	1	0		1	0	1	0	1	0	0
0	0	1	1	1	1	1		1	0	1	1	1	0	1
0	1	0	0	0	0	1		1	1	0	0	1	1	1
0	1	0	1	0	1	0		1	1	0	1	0	0	0
0	1	1	0	1	1	1		1	1	1	0	1	0	1
0	1	1	1	0	0	0		1	1	1	1	1	1	0

b)  $S_2 = \Pi M (0,1,4,5,7,13) = (X_1 + X_0 + Y_1 + Y_0) (X_1 + X_0 + Y_1 + \bar{Y}_0) (X_1 + \bar{X}_0 + Y_1 + Y_0) (X_1 + \bar{X}_0 + Y_1 + \bar{Y}_0) (X_1 + \bar{X}_0 + \bar{Y}_1 + \bar{Y}_0) (\bar{X}_1 + \bar{X}_0 + Y_1 + \bar{Y}_0)$

c)  $S_2 = (X_1 + Y_1) (X_1 + \bar{X}_0 + \bar{Y}_0) (\bar{X}_0 + Y_1 + \bar{Y}_0)$

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_18.** Se pretende diseñar un circuito que multiplique dos números de 3 bits cada uno, codificados en complemento a 2.  $A(A_2A_1A_0) \times B(B_2B_1B_0) = R(R_5R_4R_3R_2R_1R_0)$  Se pide:

- a) Obtener la expresión de la función  $R_5$  como la mínima suma de productos. Dibujar el circuito resultante.
- b) Obtener la expresión de la función  $R_0$  como el mínimo producto de sumas. Dibujar el circuito resultante.

**Solución:**

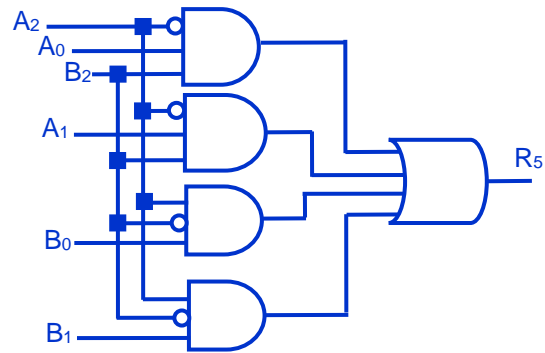
a) La función  $R_5$  es el bit de signo del producto, por lo que:

a1) El resultado será negativo,  $R_5 = 1$ , cuando los dos factores tengan distinto signo excepto cuando uno de los dos factores sea cero. En una tabla de Karnaugh de 64 celdas se representa:

		$B_2 = 0$				$B_2 = 1$				
		$B_1B_0$	00	01	11	10	00	01	11	10
$A_2 = 0$	$A_1A_0$	00								
	01					1	1	1	1	
	11					1	1	1	1	
	10					1	1	1	1	

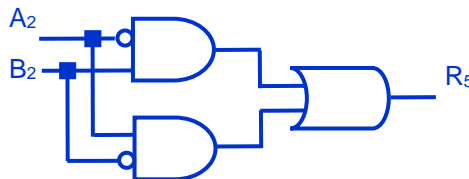
$A_2 = 1$	$A_1A_0$	00	1	1	1				
	01		1	1	1				
	11		1	1	1				
	10		1	1	1				



$$R_5 = \overline{A_2} A_0 B_2 + \overline{A_2} A_1 B_2 + A_2 \overline{B_2} B_0 + A_2 \overline{B_2} B_1$$

**NOTA:** Se considera válida una solución simplificada del problema en donde se señale de forma explícita, la exclusión de los valores  $A_2A_1A_0 = "000"$  y  $B_2B_1B_0 = "000"$  en la solución. En este caso la solución simplificada sería:

$$R_5 = \overline{A_2} B_2 + A_2 \overline{B_2}$$



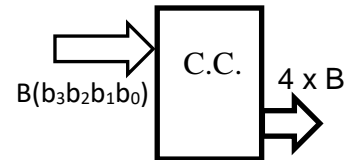
b) La función  $R_0$  es el bit de paridad del producto, por lo que valdrá 1 (será impar) cuando ambos factores sean impares.

$$R_0 = A_0 B_0$$



**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

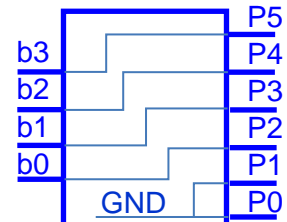
**U6\_19.** Diseñar un circuito combinacional cuya salida sea el valor de multiplicar por 4 un número en binario natural de 4 bits. Se quiere que la solución sea un diseño que utilice el menor número de puertas lógicas posibles y con el menor número de entradas.



**Solución:**

Como multiplicar por 4 es desplazar el número dos lugares a la izquierda, añadiendo dos '0' en los bits menos significativos, la solución más simple no necesita puerta lógica alguna.

$4xB = P(P_5P_4P_3P_2P_1P_0) \Rightarrow$



**U6\_20.** Sabiendo que X e Y son números en complemento a 2, se quiere diseñar un circuito que acepte como entrada un número X de 4 bits:  $X(X_3X_2X_1X_0)$ , y cuya salida  $Y(Y_4Y_3Y_2Y_1Y_0)$  satisfaga la siguiente ecuación:

- Y = +15                    si     $X > +4$
- Y = 2·X                    si     $+4 \geq X \geq -4$
- Y = -15                    si     $X < -4$

Se pide:

- a) Realizar la tabla de verdad
- b) Obtener la expresión de la función Y2 como la mínima suma de productos. Dibujar el circuito resultante.
- c) Obtener la expresión de la función Y1 como el mínimo producto de sumas. Dibujar el circuito resultante.

**Solución**

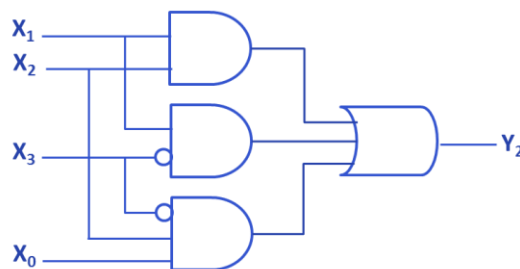
a)

$X_3$	$X_2$	$X_1$	$X_0$	$Y_4$	$Y_3$	$Y_2$	$Y_1$	$Y_0$
0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	0
0	0	1	0	0	0	1	0	0
0	0	1	1	0	0	1	1	0
0	1	0	0	0	1	0	0	0
0	1	0	1	0	1	1	1	1
0	1	1	0	0	1	1	1	1
0	1	1	1	0	1	1	1	1
1	0	0	0	1	0	0	0	1
1	0	0	1	1	0	0	0	1
1	0	1	0	1	0	0	0	1
1	0	1	1	1	0	0	0	1
1	1	0	0	1	1	0	0	0
1	1	0	1	1	1	0	1	0
1	1	1	0	1	1	1	0	0
1	1	1	1	1	1	1	1	0

b)

$X_1X_0$ \ $X_3X_2$	00	01	11	10
00	0	0	1	1
01	0	1	1	1
11	0	0	1	1
10	0	0	0	0

$Y_2 = \overline{X_3} X_2 X_0 + \overline{X_3} X_1 + X_2 X_1$

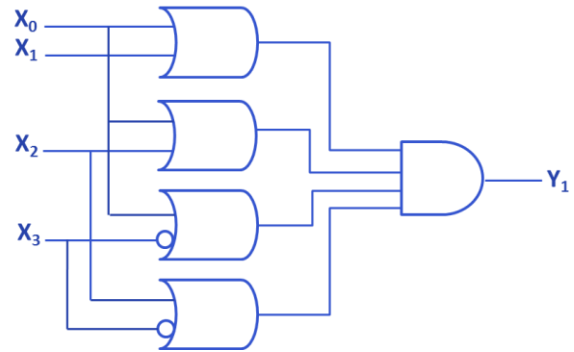


**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

b)

	$X_1X_0$	00	01	11	10
$X_3X_2$	00	0	1	1	0
	01	0	1	1	1
	11	0	1	1	0
	10	0	0	0	0

$$Y_1 = (X_1 + X_0) \cdot (X_2 + X_0) \cdot (X_2 + \overline{X_3}) \cdot (X_0 + \overline{X_3})$$



**U6\_21.** Se pretende diseñar un circuito cuyas dos entradas, A(A<sub>1</sub>A<sub>0</sub>) y B(B<sub>1</sub>B<sub>0</sub>), sean dos números en complemento a 2, y cuya salida S(S<sub>3</sub>S<sub>2</sub>S<sub>1</sub>S<sub>0</sub>) sea la suma de A y B, también en complemento a 2. Esto es, S=A+B. Se pide:

- Escribir la tabla de verdad.
- Obtener la expresión de la función S<sub>2</sub> como la mínima suma de productos.
- Obtener la expresión de la función S<sub>0</sub> como el mínimo producto de sumas.
- Obtener la expresión de la función S<sub>3</sub> como producto completo de Maxterms

**Solución: a)**

A <sub>1</sub>	A <sub>0</sub>	B <sub>1</sub>	B <sub>0</sub>	S <sub>3</sub>	S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	1	1
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	1	1	1	1
0	1	1	1	0	0	0	0
1	0	0	0	1	1	1	0
1	0	0	1	1	1	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	1	0	0	1	1	1	1
1	1	0	1	0	0	0	0
1	1	1	0	1	1	0	1
1	1	1	1	1	1	1	0

b)  $S_2 = A_1\overline{A_0} + B_1\overline{B_0} + A_1\overline{B_0} + A_1B_1 + \overline{A_0}B_1$

c)  $S_0 = (\overline{A_0} + \overline{B_0}) (A_0 + B_0)$

d)  $S_3 = \prod M(0,1,4,5,7,13) =$

$$= (A_1 + A_0 + B_1 + B_0) (A_1 + A_0 + B_1 + \overline{B_0}) (A_1 + \overline{A_0} + B_1 + B_0) (A_1 + \overline{A_0} + B_1 + \overline{B_0}) (A_1 + \overline{A_0} + \overline{B_1} + \overline{B_0})$$

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_22.** Un circuito lógico tiene 5 entradas y una salida. Cuatro de las entradas A, B, C, y D representan un dígito decimal en BCD y la quinta entrada, E, es de control. Cuando el control está en '0' lógico, la salida vale '0' si el número decimal es par y vale '1' si es impar. Cuando el control está a '1' lógico, la salida vale '0' cuando el número es múltiplo de 3 y vale '1' en el resto de los casos. Se pide diseñar el circuito. NOTA: Se considera que el cero es un múltiplo de 3.

**Solución:**

$$Z = D \overline{E} + \overline{A} \overline{C} D + BD + B \overline{C} E + A \overline{D} E + \overline{B} C \overline{D} E$$

**U6\_23.** Un circuito lógico acepta como entradas dos números enteros de 2 bits  $A = A_1A_0$  y  $B = B_1B_0$  y suministra una salida de 4 bits  $P = P_3P_2P_1P_0$  que es el producto numérico de A y B. Se pide diseñar y dibujar el circuito correspondiente.

**Solución:**

$$P_0 = A_0B_0; \quad P_1 = A_1B_0 (\overline{A_0} + \overline{B_1}) + A_0B_1 (\overline{A_1} + \overline{B_0});$$

$$P_2 = A_1B_1 (\overline{A_0} + \overline{B_0}); \quad P_3 = A_1A_0B_1B_0$$

**U6\_24.** a) Mostrar en una tabla con 4 entradas ( $X_1X_0$  e  $Y_1Y_0$ ) que representan a dos números binarios X e Y de 2 bits en complemento a dos, el resultado de la suma  $X+Y$ , también en complemento a 2. Utilice el menor número de bits de salida que necesite. b) Expresar en maxterms la función par el bit de signo del resultado. c) Expresar también esta misma función como producto de sumas en la forma más simplificada posible.

**Solución:**

a)

$X_1$	$X_0$	$Y_1$	$Y_0$	$S_2$	$S_1$	$S_0$		$X_1$	$X_0$	$Y_1$	$Y_0$	$S_2$	$S_1$	$S_0$
0	0	0	0	0	0	0		1	0	0	0	1	1	0
0	0	0	1	0	0	1		1	0	0	1	1	1	1
0	0	1	0	1	1	0		1	0	1	0	1	0	0
0	0	1	1	1	1	1		1	0	1	1	1	0	1
0	1	0	0	0	0	1		1	1	0	0	1	1	1
0	1	0	1	0	1	0		1	1	0	1	0	0	0
0	1	1	0	1	1	1		1	1	1	0	1	0	1
0	1	1	1	0	0	0		1	1	1	1	1	1	0

**b)**  $S_2 = \prod M (0,1,4,5,7,13) = (X_1 + X_0 + Y_1 + Y_0) (X_1 + X_0 + Y_1 + \overline{Y_0}) (X_1 + \overline{X_0} + Y_1 + Y_0) (X_1 + \overline{X_0} + Y_1 + \overline{Y_0}) (X_1 + \overline{X_0} + \overline{Y_1} + \overline{Y_0}) (\overline{X_1} + \overline{X_0} + Y_1 + \overline{Y_0})$

**c)**  $S_2 = (X_1 + Y_1) (X_1 + \overline{X_0} + \overline{Y_0}) (\overline{X_0} + Y_1 + \overline{Y_0})$

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_25.** Dado el número real negativo escrito en decimal:  $A = -5142,25$ , se pide representar  $A$  en los siguientes formatos:

a.1. En un formato signo-magnitud en coma fija para 16 bits, con 2 bits para la parte fraccionaria.

<b>A =</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>1</b>
------------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

a.2. En un formato signo-magnitud en coma fija para 32 bits, con 4 bits para la parte fraccionaria. **Es obligatorio dar el resultado en Hexadecimal.**

<b>A = 0x80014164</b>
-----------------------

a.3. En un formato en complemento a dos para 16 bits, con 2 bits para la parte fraccionaria.

<b>A =</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>
------------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

a.4. En un formato en complemento a dos para 32 bits, con 4 bits para la parte fraccionaria. **Es obligatorio dar el resultado en Hexadecimal.**

<b>A = 0xFFFEBE9C</b>
-----------------------

**U6\_26.** Dado el número real negativo, escrito en decimal:  $A = -2.040,125$ , se pide representar  $A$  en los siguientes formatos:

a.1. En complemento a dos para 16 bits, con 4 bits para la parte fraccionaria.

<b>A =</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0</b>
------------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

a.2. En complemento a dos para 32 bits, con 8 bits para la parte fraccionaria. Es obligatorio dar el resultado en Hexadecimal.

<b>A = 0xFFF807E0</b>
-----------------------



**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_27.** Se pretende diseñar un circuito cuyas dos entradas, A(A<sub>1</sub>A<sub>0</sub>) y B(B<sub>1</sub>B<sub>0</sub>), sean dos números en complemento a 2, y cuya salida S(S<sub>3</sub>S<sub>2</sub>S<sub>1</sub>S<sub>0</sub>) sea la suma de A y B, también en complemento a 2. Esto es, S=A+B. Se pide:

- a) Completar la tabla de verdad adjunta
- b) Obtener la expresión de la función S<sub>2</sub> como la mínima suma de productos.
- c) Obtener la expresión de la función S<sub>0</sub> como el mínimo producto de sumas.
- d) Obtener la expresión de la función S<sub>3</sub> como producto completo de Maxterms.

**Solución:**

a)

A <sub>1</sub>	A <sub>0</sub>	B <sub>1</sub>	B <sub>0</sub>	S <sub>3</sub>	S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	1	1
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	1	1	1	1
0	1	1	1	0	0	0	0
1	0	0	0	1	1	1	0
1	0	0	1	1	1	1	1
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	1	0	0	1	1	1	1
1	1	0	1	0	0	0	0
1	1	1	0	1	1	0	1
1	1	1	1	1	1	1	0

b)  $S_2 = A_1\overline{A_0} + B_1\overline{B_0} + A_1\overline{B_0} + A_1B_1 + \overline{A_0}B_1$

c)  $S_0 = (\overline{A_0} + \overline{B_0}) (A_0 + B_0)$

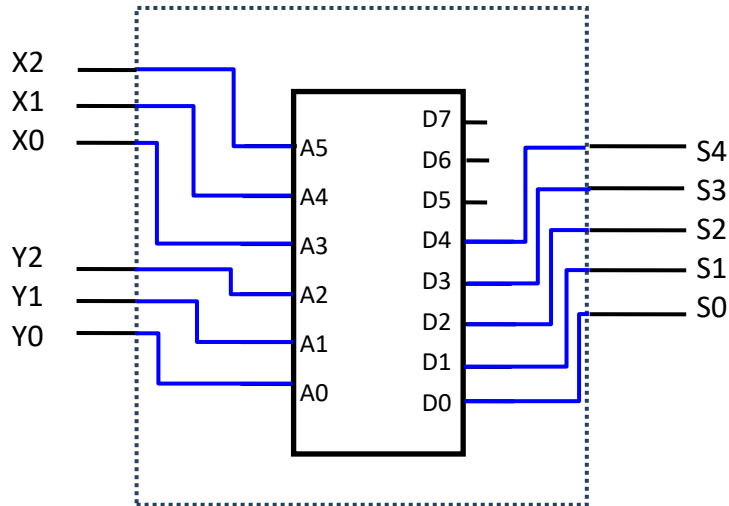
d)

$$S_3 = \prod M(0,1,4,5,7,13) = (A_1 + A_0 + B_1 + B_0) (A_1 + A_0 + B_1 + \overline{B_0}) (A_1 + \overline{A_0} + B_1 + B_0) (A_1 + \overline{A_0} + B_1 + \overline{B_0}) (\overline{A_1} + \overline{A_0} + B_1 + B_0)$$

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_28.** Se pretende diseñar un circuito multiplicador de dos números de 3 bits X e Y, codificados en signo-magnitud. Para simplificar su implementación, se ha utilizado una memoria de 64x8 (64 posiciones de 8 bits de datos cada una de ellas). Se pide:

- a. Indicar las conexiones de las variables de entrada (X2, X1, X0, Y2, Y1 e Y0) y las funciones de salida (S4, S3, S2, S1 y S0) con la memoria.



- b. Completar el contenido de las siguientes posiciones de memoria para que realice la función pedida. Justificar dichos contenidos.

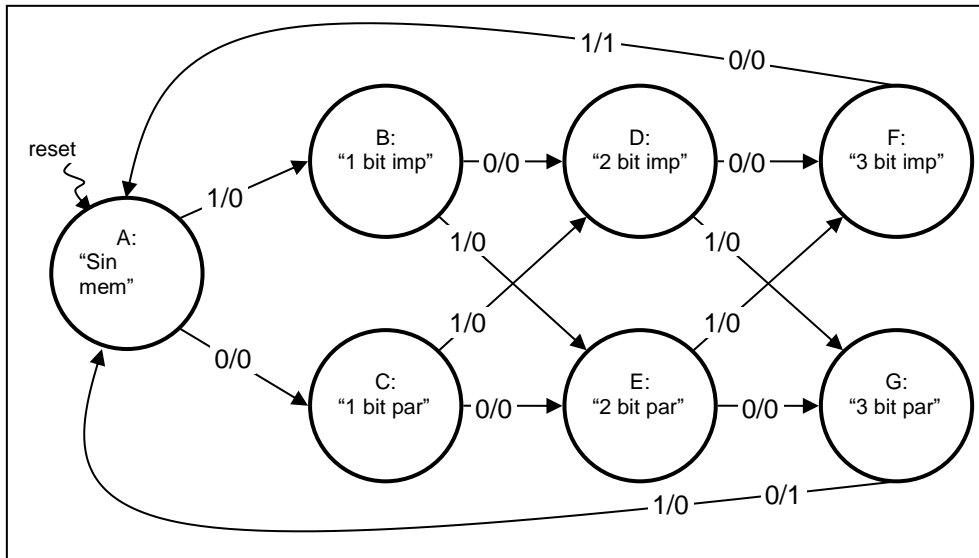
Dirección (Hexa)	X2	X1	X0	Y2	Y1	Y0	X*Y = S	Dato (Hexa.)
00	0	0	0	0	0	0	(+0)*(+0) = +0	00
01	0	0	0	0	0	1	(+0)*(+1) = +0	00
02	0	0	0	0	1	0	(+0)*(+2) = +0	00
03	0	0	0	0	1	1	(+0)*(+3) = +0	00
15	0	1	0	1	0	1	(+2)*(-1) = -2	12
1A	0	1	1	0	1	0	(+3)*(+2) = +6	06
24	1	0	0	1	0	0	(-0)*(-0) = +0	00
25	1	0	0	1	0	1	(-0)*(-1) = +0	00
27	1	0	0	1	1	1	(-0)*(-3) = +0	00
29	1	0	1	0	0	1	(-1)*(+1) = -1	11
2C	1	0	1	1	0	0	(-1)*(-0) = +0	00
2D	1	0	1	1	0	1	(-1)*(-1) = +1	01
32	1	1	0	0	1	0	(-2)*(+2) = -4	14
38	1	1	1	0	0	0	(-3)*(+0) = -0	00*
3B	1	1	1	0	1	1	(-3)*(+3) = -9	19
3F	1	1	1	1	1	1	(-3)*(-3) = +9	09

\* También vale 10, el signo-magnitud, el cero tiene dos codificaciones

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_29.** Se pide diseñar un circuito secuencial de MEALY que sirva como detector de paridad para un sistema que recibe en serie y sincronamente con la señal del reloj, paquetes de cuatro bits por una única línea de entrada. Al recibir el cuarto bit de cada bloque, la salida del circuito Z, tomará el valor 1 sólo si el número total de unos recibidos en el bloque es un número par y 0 en caso contrario. Tras la recepción del cuarto bit de cada bloque, el sistema debe estar listo para recibir y analizar sin ningún retardo los cuatro bits correspondientes del siguiente bloque. Realice el diagrama de estados mínimo, la tabla de transición y utilizando FF tipo-D, las ecuaciones de excitación (estado) y salida. Suponga que recibir 4 ceros es paridad par.

**Solución:**



Los siete posibles estados del sistema son:

- A: Estado en espera del primer bit. Asignación  $Q_2Q_1Q_0$  (000)
- B: Estado que ha recibido un bit y es un número impar de "1". Asignación (001)
- C: Estado que ha recibido un bit y es un número par de "1". Asignación (010)
- D: Estado que ha recibido dos bits y es un número impar de "1". Asignación (011)
- E: Estado que ha recibido dos bits y es un número par de "1". Asignación (100)
- F: Estado que ha recibido tres bit y es un número impar de "1". Asignación (101)
- G: Estado que ha recibido tres bit y es un número par de "1". Asignación (010)

Las ecuaciones de excitación y salida son:

$$D_0 = \overline{Q_2} Q_0 \overline{X} + \overline{Q_2} \overline{Q_0} X + \overline{Q_1} \overline{Q_0} X$$

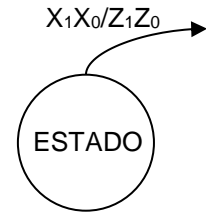
$$D_1 = \overline{Q_2} \overline{Q_1} X + \overline{Q_1} \overline{Q_0} \overline{X} + \overline{Q_2} Q_1 X$$

$$D_2 = \overline{Q_2} Q_1 \overline{X} + \overline{Q_2} Q_0 X + Q_2 \overline{Q_1} \overline{Q_0}$$

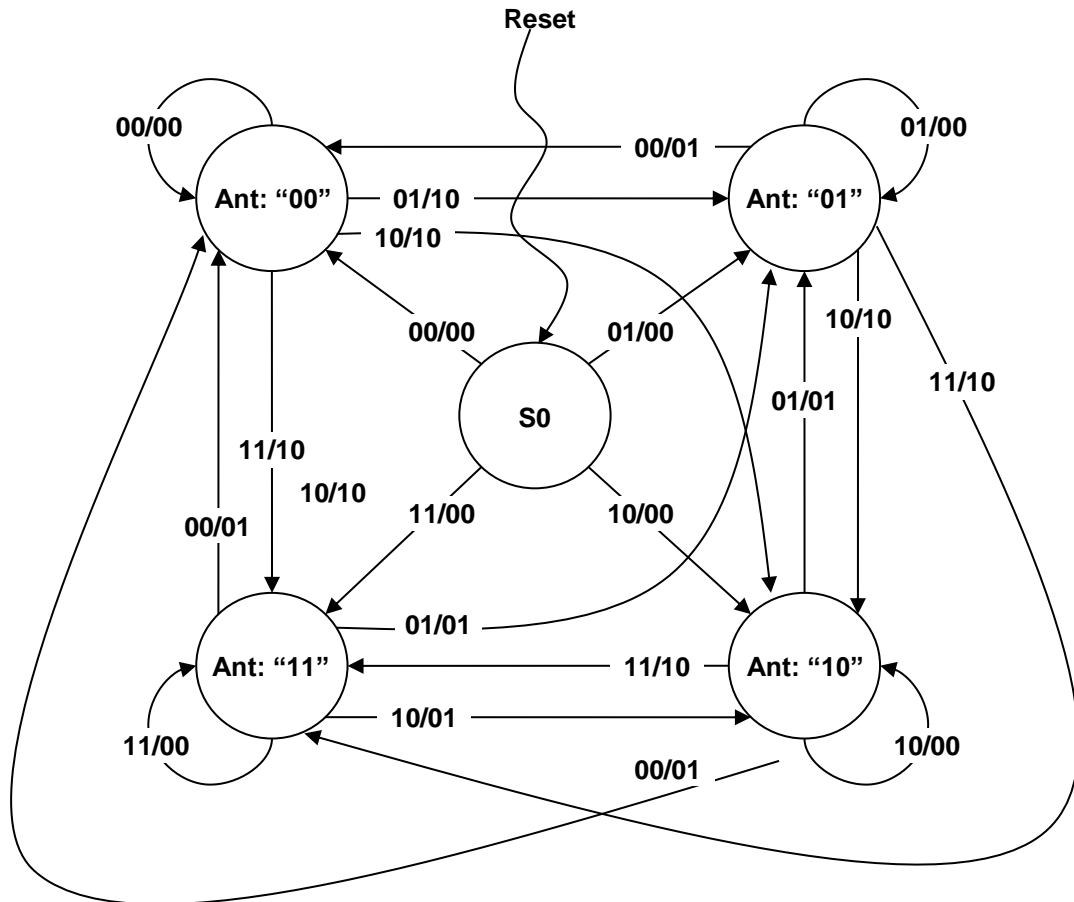
$$Z = Q_2 Q_1 \overline{X} + Q_2 Q_0 X$$

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_30.** Un circuito secuencial tiene dos entradas ( $X_1, X_0$ ) y dos salidas ( $Z_1, Z_0$ ). Las entradas representan un número binario de dos bits,  $N = X_1X_0$ . Si el valor actual de  $N$  es mayor que el valor anterior, entonces  $Z_1 = 1$ . Si el valor actual de  $N$  es menor que el valor anterior, entonces  $Z_0 = 1$ . Si el valor actual de  $N$  es igual que el valor anterior, las dos salidas permanecen a 0. Cuando se recibe el primer par de entradas, tras una señal asíncrona de de Reset, no hay ningún valor anterior de  $N$ , así que no es posible determinar si el valor de  $N$  actual es mayor o menor que el valor anterior, por lo que ambas salidas deben ser 0. Se pide diseñar una máquina de estado de Mealy para el circuito utilizando la nomenclatura indicada.



**Solución:**

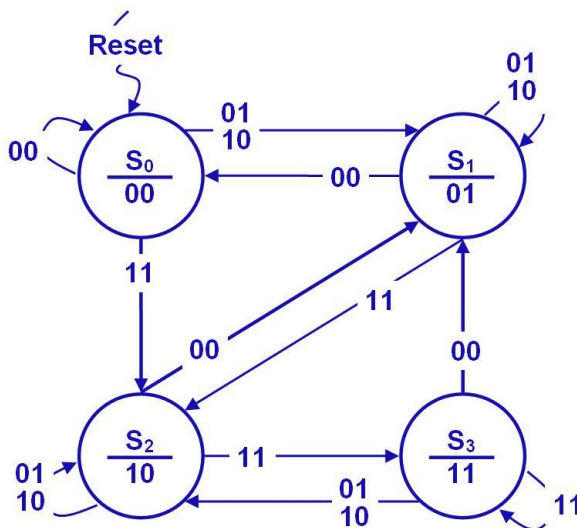


**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_31.** Se quiere diseñar un circuito secuencial que funcione como un sumador serie completo de n bits. Partiendo de una señal de Reset asíncrono que inicializa toda la lógica secuencial a "0", el circuito recibe en serie (síncronamente con el reloj) por sus dos entradas y empezando por los menos significativos los bits de los dos operandos  $A_i$  y  $B_i$ . El circuito debe suministrar por sus dos salidas tanto el bit de la suma  $S_i$  como el de acarreo  $C_{i+1}$ . Se supone que la suma continúa indefinidamente hasta que se recibe un nuevo Reset por la entrada asíncrona, momento en el que la introducción de nuevos operandos se entiende como el comienzo de una nueva suma. Se pide:

- Diseñe el diagrama de estados del circuito en su versión Moore, utilizando los estados necesarios de los diagramas adjuntos, añadiendo más si fuera necesario o dejando en blanco los que no se necesiten.
- Utilizando flip-flops tipo T, indique el número de biestables necesarios. Realice la tabla de asignación de estados, la tabla de transición entre estados, la función para la excitación de los biestables y la tabla de salida. Utilice las plantillas facilitadas eliminando, si las hubiere, las filas o columnas innecesarias o añadiendo las que considere necesarias.
- Expresé en su forma más simplificada como suma de productos las ecuaciones de salida.

**Solución a)**



Hay que considerar la suma completa bit a bit:  
 $S_i = C_i \oplus A_i \oplus B_i$  y  $C_{i+1} = A_i B_i + A_i C_i + B_i C_i$

**S<sub>0</sub>:** Estado inicial para el que el resultado de la suma es "0" ( $C_{i+1} = 0$  y  $S_i = 0$ )

**S<sub>1</sub>:** Estado para el que el resultado de la suma es "1" ( $C_{i+1} = 0$  y  $S_i = 1$ )

**S<sub>2</sub>:** Estado para el que el resultado de la suma es "2" ( $C_{i+1} = 1$  y  $S_i = 0$ )

**S<sub>3</sub>:** Estado para el que el resultado de la suma es "3" ( $C_{i+1} = 1$  y  $S_i = 1$ )

**b)**

Asignación de estados		
ESTADO	$Q_1$	$Q_0$
S0	0	0
S1	0	1
S2	1	0
S3	1	1

Transición de estados					
E. Actual		Entradas		E. Siguiete	
$Q_1^n$	$Q_0^n$	$A_i$	$B_i$	$Q_1^{n+1}$	$Q_0^{n+1}$
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	0	1
0	0	1	1	1	1
0	1	0	0	0	0
0	1	0	1	0	1
0	1	1	0	0	0
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	1	1
1	1	0	0	0	0
1	1	0	1	0	1
1	1	1	0	1	0
1	1	1	1	1	1

Entradas biestables	
$T_1$	$T_0$
0	0
0	1
0	1
0	1
1	0
0	1
0	0
0	0
1	1
1	1
0	0
0	0
0	1
1	0
0	1
0	1
0	0

Salidas	
$C_{i+1}$	$S_i$
0	0
0	0
0	0
0	0
0	0
0	1
0	1
0	1
0	1
1	0
1	0
1	0
1	0
1	1
1	1
1	1
1	1

c)  $S_i = Q_0$ ;  $C_{i+1} = Q_1$

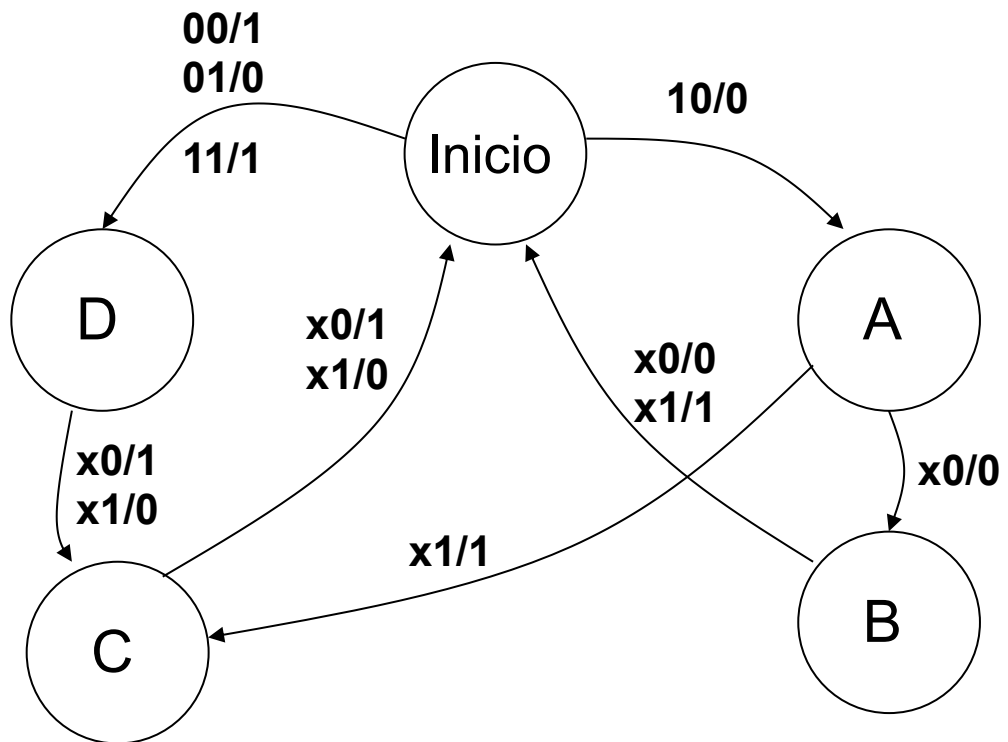
**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

**U6\_32.** Diseñe un circuito secuencial de Mealy que calcule, dependiendo del valor de una entrada "S" el complemento a 1 ( $S = 0$ ) o el complemento a 2 ( $S = 1$ ) de un número de 3 bits que se va introduciendo en serie por una entrada "X" comenzando por el bit menos significativo. El bit más significativo del complemento correspondiente aparecerá en la salida al tercer ciclo de reloj. Una vez que se ha empezado a hacer un tipo de complemento (a 1 o a 2) no se cambiará el cálculo aunque varíe la entrada "S".

Realice únicamente el diagrama de estados utilizando los estados necesarios del diagrama adjunto, añadiendo más si fuera necesario o dejando en blanco los que no se necesiten.

La nomenclatura a seguir es:

**Solución:**



**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

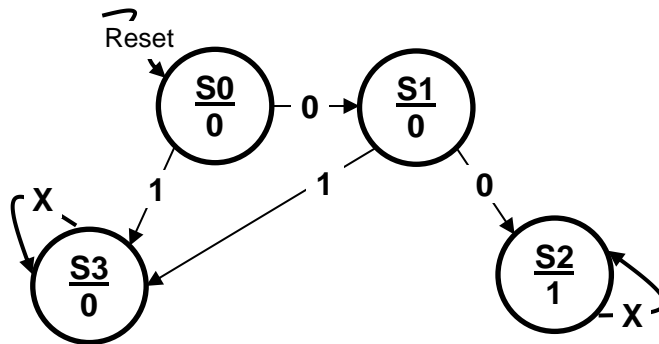
**U6\_33.** Se pide el diseño de una máquina de estados finita, FSM de Moore, que detecte si un número entero desconocido es múltiplo de cuatro. Cuando lo detecte, la FSM debe poner la única salida  $Z = '1'$ , en caso contrario  $Z = '0'$ . Después de una señal asíncrona de Reset, el número en binario de  $n$  bits, empezando por el bit de menos peso, llega al sistema secuencial bit a bit, con cada flanco ascendente del reloj. Una nueva señal de Reset señala el final de número actual y el comienzo del siguiente.

Se pide sólo el diagrama de estados de la FSM, en el diseño se penalizará el uso de más estados que los necesarios, así como las transiciones entre estados incorrectas o incompletas. Indicar **necesariamente** de forma breve, no más de una línea, un comentario que ayude a entender el significado de cada uno de los estados definidos.

**Nota:** El número cero es múltiplo de cuatro.

**Solución:**

Los múltiplos de cuatro son 0, 4, 8, 12, 16, 20...en binario todos terminan en "00". Esta será la secuencia a buscar por la FSM.



**Descripción de estados:**

**S0.** Estado inicial sin memoria, a la espera del bit menos significativo del número.

**S1.** El primer bit, el menos significativo es un '0', secuencia buscada.

**S2.** El segundo bit menos significativo es un '0', secuencia buscada y encontrada. Cualquier número que termine en "00", no importa el tamaño en bits es múltiplo de cuatro. Salida  $Z = '1'$

**S3.** El primer o el segundo bit menos significativo es un '1', secuencia errónea, cualquier número, no importa el tamaño en bits no es múltiplo de cuatro. Salida  $Z = '0'$ .

**U6\_34.** Dado el número decimal  $X = -9.547,25$  se pide su representación, en los siguientes formatos:

a1) Código binario con 20 bits, en signo magnitud, utilizando 4 bits para la parte fraccionaria.

<b>X =</b>	1	0	1	0	0	1	0	1	0	1	0	0	1	0	1	1	0	1	0	0
------------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

a2) Código binario con 20 bits, en complemento a dos, utilizando 4 bits para la parte fraccionaria.

<b>X =</b>	1	1	0	1	1	0	1	0	1	0	1	1	0	1	0	0	1	1	0	0
------------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

a3) Código hexadecimal con 32 bits en complemento a dos, utilizando 8 bits para la parte fraccionaria.

<b>X =</b>	<b>0XFFDAB4C0</b>
------------	-------------------

b) Se tienen dos números binarios ambos enteros y representados en hexadecimal,  $A = 0x830F$  codificado en complemento a dos y  $B = 0x7654$  codificado en signo magnitud. Se pide el resultado de la operación de resta  $R = A - B$ , en binario y en hexadecimal con un formato de 24 bits en complemento a dos.

**Nota:** En todo el ejercicio **NO** se permite el uso de calculadora. Para calificar el apartado b), **debe hacer necesariamente la operación en binario**.

**FUNDAMENTOS DE COMPUTADORES**  
**EJERCICIOS U6: Representación digital de los números**

A = 1000 0011 0000 1111

B = 0111 0110 0101 0100 → 1000 1001 1010 1011

$$\begin{array}{r} \phantom{1000\ 1001\ 1010\ 1011} \\ \phantom{1000\ 1001\ 1010\ 1011} \phantom{+} \\ \hline 1000\ 1001\ 1010\ 1100 \end{array}$$

A- 0000 0000 1000 0011 0000 1111

B- 1111 1111 1000 1001 1010 1100 +

1 1111 1111 0000 1100 1011 1011 → 0xFF0CBB