14/3/2022

Arquitectura de computadoras

Ing. Gabriela Reynosa

Álgebra de Boole y Compuertas lógicas

Ing. Gabriela Reynosa

Retroalimentación

- El complemento de un producto de variables es igual a la suma de los complementos de las variables.
$$\overline{XY} = \bar{X} + \bar{Y}$$

- El complemento de una suma de variables es igual al producto de los complementos de las variables.
$$\overline{X + Y} = \bar{X}\,\bar{Y}$$

Simplificación Lógica

1. $A + 0 = A$
2. $A + 1 = 1$
3. $A \cdot 0 = 0$
4. $A \cdot 1 = A$
5. $A + A = A$
6. $A + \overline{A} = 1$
7. $A \cdot A = A$
8. $A \cdot \overline{A} = 0$
9. $\overline{\overline{A}} = A$
10. $A + AB = A$
11. $A + \overline{A}B = A + B$
12. $(A + B)(A + C) = A + BC$

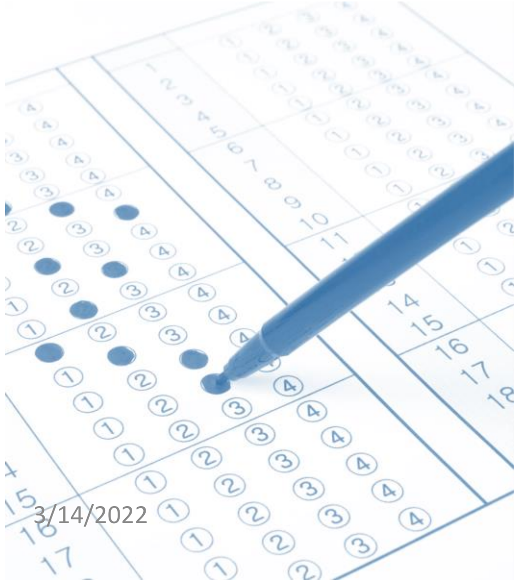
Ejemplo

Aplicar los teoremas de DeMorgan a las expresiones $\overline{WXYZ}$ y $\overline{W+X+Y+Z}$.

Respuestas

$$\overline{WXYZ} = \overline{W}+\overline{X}+\overline{Y}+\overline{Z}$$

$$\overline{W+X+Y+Z} = \overline{W}\,\overline{X}\,\overline{Y}\,\overline{Z}$$



Ejemplo:
Reducir la siguiente expresión

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

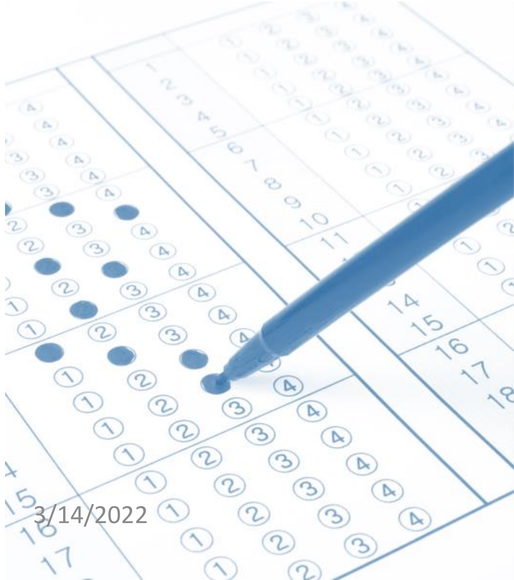
1. $A + 0 = A$	7. $A \cdot A = A$
2. $A + 1 = 1$	8. $A \cdot \overline{A} = 0$
3. $A \cdot 0 = 0$	9. $\overline{\overline{A}} = A$
4. $A \cdot 1 = A$	10. $A + AB = A$
5. $A + A = A$	11. $A + \overline{A}B = A + B$
6. $A + \overline{A} = 1$	12. $(A + B)(A + C) = A + BC$

Paso 1. Aplicar la ley distributiva al segundo y tercer término del siguiente modo:

$$AB + AB + AC + BB + BC$$

Paso 2. Aplicar la regla 7 ($BB = B$) al cuarto término.

$$AB + AB + AC + B + BC$$



Paso 3. Aplicar la regla 5 ($AB + AB = AB$) a los dos primeros términos.

$$AB + AC + B + BC$$

Paso 4. Aplicar la regla 10 ($B + BC = B$) a los dos últimos términos.

$$AB + AC + B$$

Paso 5. Aplicar la regla 10 ($AB + B = B$) al primero y tercer término.

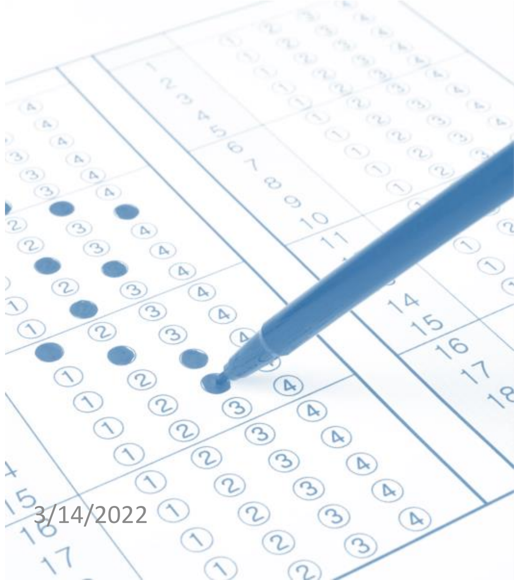
$$B + AC$$

Ejercicio:
Reducir la siguiente expresión

$$[A\overline{B}(C+BD)+\overline{A}\overline{B}]C$$

1. $A+0=A$	7. $A \cdot A = A$
2. $A+1=1$	8. $A\cdot\overline{A}=0$
3. $A\cdot 0=0$	9. $\overline{\overline{A}}=A$
4. $A\cdot 1=A$	10. $A+AB=A$
5. $A+A=A$	11. $A+\overline{A}B=A+B$
6. $A+\overline{A}=1$	12. $(A+B)(A+C)=A+BC$

Respuesta: $\bar{B}C$



Resolver los siguientes ejercicios

Ejercicios	Respuesta
(a) $AB + B(CD + EF)$	$AB + BCD + BEF$
(b) $(A + B)(B + C + D)$	$AB + AC + AD + BB + BC + BD$
(c) $\overline{(\overline{A + B}) + C}$	$(A + B)\overline{C} = A\overline{C} + B\overline{C}$

Actividad de compuertas

Ir al enlace

Sistemas de Numeración

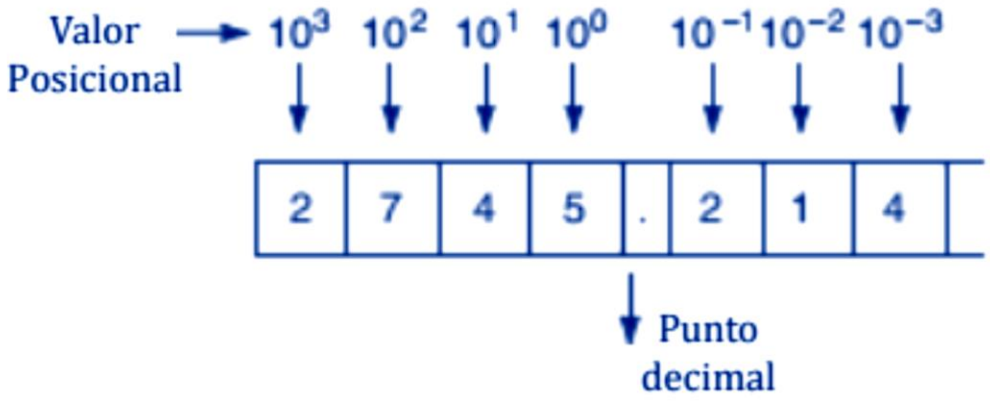
Ing. Gabriela Reynosa

Sistemas Numéricos

- Los sistemas numéricos se usan en tecnología digital para representar cantidades, valores y magnitudes. Los más utilizados son el decimal, el binario y el hexadecimal.
- Cada sistema utiliza **símbolos** y son éstos los que se utilizan para representar las cantidades.
- Son sistemas de valor posicional (el valor de un dígito depende de su posición).
- **Cualquier cantidad se expresa por la suma de los productos del valor de cada símbolo y su valor posicional (peso).**

Sistema Decimal

- El Sistema Decimal está compuesto por 10 dígitos (0, 1, 2, 3, 4, 5, 6, 7, 8, 9).
- Es un sistema de base 10.
- Una serie de N dígitos puede expresar $10N$ cantidades y la cantidad mayor que puede expresar el sistema es de 10^N -1.

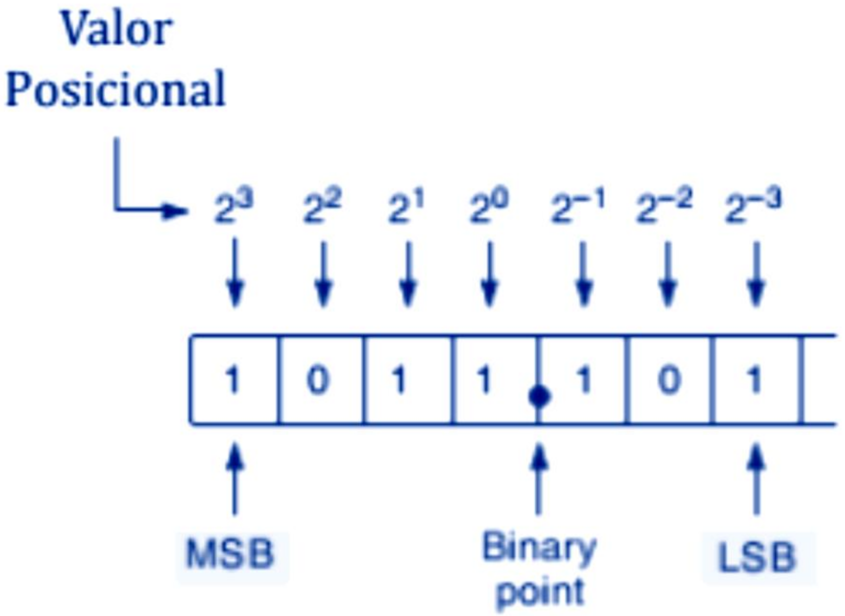


- Por ejemplo:

2,745.214

Sistema Binario

- El Sistema Binario está compuesto por 2 símbolos (0, 1). Se llaman **bits.**
- Es un sistema de base 2.
- Una serie de N dígitos puede expresar 2^N cantidades y la cantidad mayor que puede expresar el sistema es de 2^N -1.
- A cada bit le corresponde un valor de voltaje.
- Cero y cinco volts respectivamente.



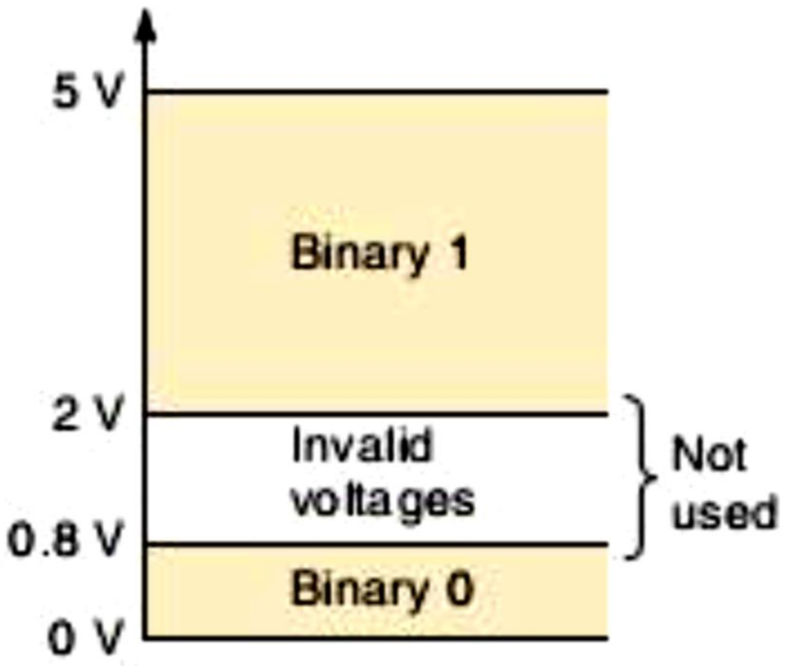
Sistema Binario

- Todos los sistemas numéricos tienen **alternancia**.
- El conteo binario tiene alternancia entre unos y ceros.
- Tiene alternancia a cada paso para 2^0, cada dos pasos para 2^1, cada cuatro pasos para 2^2.
- Sin embargo, la **desventaja** es que deben representarse sus cantidades por medio de otros sistemas más prácticos y comunes.

Weights

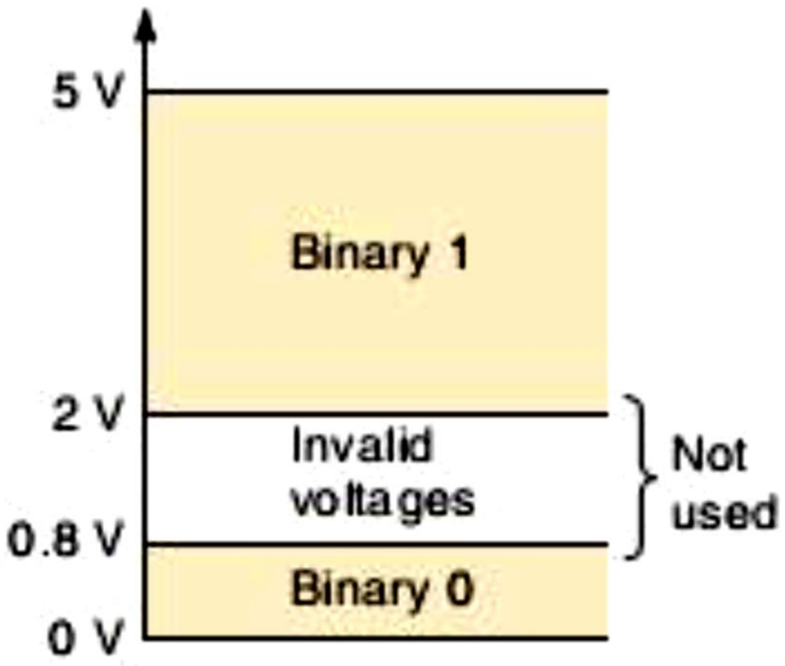
$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$	Decimal equivalent
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

LSB



Niveles de Voltaje

- Lógico 0 y Lógico 1 son los estados binarios. Son representados por intervalos de voltaje (y no por valores nominales, debido a las variaciones de voltaje).
- En los Sistemas Digitales, el valor exacto no es importante. En los sistemas analógicos sí.



Niveles de Voltaje

- En circuitos integrados de la familia estándar TTL, las compuertas de los CI reconocen como lógico 1 los valores de voltaje entre 2 y 5V, mientras que reconocen como lógico 0 los valores entre 0 y 0.8V.
- En la zona de “Voltajes No Válidos” el fabricante no garantiza ningún estado.

Sistema Hexadecimal

- El Sistema Hexadecimal de numeración tiene 16 símbolos:
- Los dígitos del 0 al 9 y letras de la A a la F.
- Todos los sistemas de numeración tienen sus contrapartes en otros sistemas y pueden convertirse fácilmente.

DEC	BINARIO	HEX
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Sistema Hexadecimal

- Es un sistema posicional de base 16.
- Una serie de N dígitos puede expresar 16^N cantidades y la cantidad mayor que puede expresar el sistema es de 16^N -1.
- Se utiliza principalmente como método abreviado de representación de números binarios.

DEC	BINARIO	HEX
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Conversiones

Conversión Binario - Decimal

- Asignados los pesos, se multiplica cada bit por su respectiva potencia de 2 y se suman los resultados

101011_2

2^5	2^4	2^3	2^2	2^1	2^0
1	0	1	0	1	1

$32 + 0 + 8 + 0 + 2 + 1$

$101011_2 = 43_{10}$

Conversión Decimal - Binario

- **Método 1.** Descomponer el número por la suma de potencias de 2, multiplicadas por uno o por cero.

$$25_{10}$$

$$16 + 8 + 1$$

$$2^4 + 2^3 + 2^0$$

$$(1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (0 \times 2^1) + (1 \times 2^0)$$

$$25_{10} = 11001_2$$

Conversión Decimal - Binario

- **Método 2.** Dividir el número por 2, recuperando los residuos (1 ó 0) provenientes de las divisiones sucesivas.

$$37_{10}\ a\ binario:$$

$37 \div 2 = 18 + residuo$
1

$18 \div 2 = 9 + residuo\ 0$

$9 \div 2 = 4 + residuo\ 1$

$4 \div 2 = 2 + residuo\ 0$

$2 \div 2 = 1 + residuo\ 0$

$1 \div 2 = 0 + residuo\ 1$

100101

Ejercicios

- **Convertir:**

$53_{10}\ a\ binario$ 1 1 0 1 0 1

$107_{10}\ a\ binario$ 1 1 0 1 0 1 1

Conteo Hexadecimal

- Se incrementa el valor de cada dígito desde 0 hasta F. Una vez un dígito llega a la F, el siguiente paso restablece la cuenta a 0 y se incrementa el dígito a la izquierda en 1.

26, 27, 28, 29, 2A, 2B, 2C, 2D, 2F, 30, 31, 32

278, 279, 27A, 27B, 27C, 27D, 27E, 27F, 280

Conversión Hexadecimal - Decimal

- Se multiplica el valor de cada hex (nombre de los “dígitos”) por su respectiva potencia de 16 y se suman sus resultados.

$$3CA_{16} = ?_{10}$$

$$3CA_{16} = 970_{10}$$

Conversión Decimal - Hexadecimal

- Nuevamente se utiliza el método de las divisiones repetidas, pero esta vez, el divisor es 16.

$$687_{10} = ?_{16}$$

$$687_{10} = 2AF_{16}$$

Conversión Hexadecimal – Binario

- Se puede realizar la conversión del Sistema Hexadecimal al Sistema binario convirtiendo cada símbolo hexadecimal por su valor equivalente representado en 4 bits.

$$
\begin{array}{lccc}
9F2_{16} = & 9 & F & 2 \\
 & \downarrow & \downarrow & \downarrow \\
 & = 1\ 0\ 0\ 1 & 1\ 1\ 1\ 1 & 0\ 0\ 1\ 0
\end{array}
$$

$$= 100111110010_2$$

$$ACDC_{16} = ?_2$$

$$ACDC_{16} = \color{red}{1010110011011100_2}$$

Conversión Binario - Hexadecimal

- Se agrupan los bits de 4 en 4 a partir del LSB, y se cambia por su respectivo símbolo en hexadecimal.
- Si el binario tiene una cantidad de dígitos que no es divisible por 4 (es decir, que tiene grupos que no tienen 4 bits), se agregan ceros a su **izquierda** para completar.

$$1110100110_2 = \underbrace{0\,0\,1\,1}_{3}\;\underbrace{1\,0\,1\,0}_{A}\;\underbrace{0\,1\,1\,0}_{6}$$
$$= 3A6_{16}$$

$$1001101011_2 = ?_{16}$$

$$1001101011_2 = 26B_{16}$$

Utilidad del Sistema Hexadecimal

El sistema Hexadecimal puede ser usado para abreviar cadenas binarias.

En operaciones computacionales, las cadenas de 64 bits no son extrañas.

A pesar de que todos los circuitos digitales trabajan en binario, se utiliza la representación hexadecimal por conveniencia.

Ejercicios

- **Convertir:**

$2AF47_{16} = ?_2$ 0010 1010 1111 0100 0111

$4756_{10} = ?_{16}$ **1294 Revisar procedimiento en jamboard pág. 10**

8	7	4	(decimal)
↓	↓	↓	
1000	0111	0100	(BCD)

9	4	3	(decimal)
↓	↓	↓	
1001	0100	0011	(BCD)

Código BCD

- El Decimal Codificado en Binario (Binary Coded Decimal) es un tipo de codificación que representa a un número decimal mediante una cadena binaria.
- Se forma al convertir **cada dígito** del número original en decimal a número **binario de 4 bits.**

Código BCD

- No es un sistema numérico (es un código).
- No es lo mismo que el número binario directo (el que ocupa todo el número decimal)
- Requiere más bits que el binario directo para representar los números decimales
- Se utilizan los números de 4 bits del 0000 al 1001

Resumen de Sistemas y Códigos

Decimal	Binary	Hexadecimal	BCD	GRAY
0	0	0	0000	0000
1	1	1	0001	0001
2	10	2	0010	0011
3	11	3	0011	0010
4	100	4	0100	0110
5	101	5	0101	0111
6	110	6	0110	0101
7	111	7	0111	0100
8	1000	8	1000	1100
9	1001	9	1001	1101
10	1010	A	0001 0000	1111
11	1011	B	0001 0001	1110
12	1100	C	0001 0010	1010
13	1101	D	0001 0011	1011
14	1110	E	0001 0100	1001
15	1111	F	0001 0101	1000

Otros Conceptos

Byte: Agrupación de 8 bits

Nibble: Agrupación de 4 bits

Word: Agrupación de 16 bits que representan cierta unidad de información. El tamaño de la palabra depende del tamaño del bus de datos del sistema que usa esa información.

Código ASCII

El código ASCII (Código Estándar Estadounidense para el Intercambio de Información) es el código alfanumérico más utilizado.

Con él se puede transferir información entre computadoras y dispositivos externos (impresoras, por ejemplo).

Utiliza 7 bits para codificar letras, símbolos, números y otros.

Los números del 0 al 31 (del HEX0 al HEX1F están reservados para caracteres de control, mientras que del decimal 32 (HEX20) en adelante representan caracteres imprimibles.

Fin de la presentación!