Sistema Digital. Paso de mundo analógico a digital.

Tipos de Sistemas Digitales.

Representación de la información.

Sistemas de Numeración. Cambios de Base.

Sistema Binario, hexadecimal y octal.

Funciones Lógicas.

Álgebra de Boole. Propiedades.

Tabla de verdad. Simplificación algebráica.

Funciones básicas: AND, OR, NOT, NAND Y NOT.

Análisis y diseño de circuitos combinacionales.

- **<u>CONCEPTO DE SISTEMA</u>:**

Cualquier método capaz de transmitir o de procesar información.

En los sistemas, la información se representa por medio de cantidades físicas (señales).

Ej. Cantidades físicas: voltajes, intensidades, cantidad de luz, volumen de agua, cantidad de arena en un reloj...

- **<u>TIPOS DE SISTEMAS</u>:**

<u>ANALÓGICOS</u>: Las cantidades físicas son analógicas, no están limitadas a un valor concreto (continuidad física de cualquier señal real).

<u>DIGITALES</u>: Las cantidades físicas que representan la información sólo pueden tomar una serie de valores discretos. Ej.: Sistema binario: sólo pueden tomar dos valores, “0” o “1”.

- CARACTERÍSTICAS DE LOS SITEMAS DIGITALES:

•SINCRONOS/ASINCRONOS:

dependen o no de una señal de reloj.

•COMBINACIONALES/SECUENCIALES:

no dependen o si dependen de la “historia” previa del sistema.

•TRANSMISIÓN SERIE/PARALELO:

forma de transmitir o recibir, secuencialmente o toda la información a la vez.

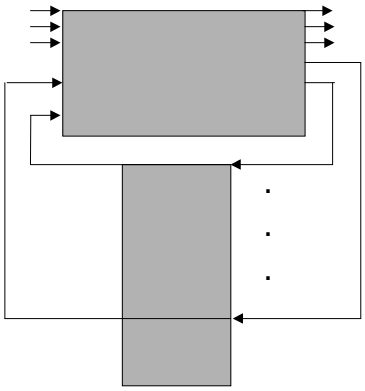
Ejemplo: Sensor de presión.

Circuito Combinacional

Para cada t, la salida sólo depende del valor de las variables de entrada para ese mismo t.

Circuito Secuencial: combinacional + registro.

Posibilidad de almacenar información sobre la historia pasada de las entradas a dicho registro.



REPRESENTACIÓN DE LA INFORMACIÓN

Sistemas de numeración:

- **Base.**
- **Resolución.**
- **Rango de representación.**

Sistema binario.

Sistema hexadecimal.

Sistema Octal.

Resumen Métodos de conversión para las bases más comunes

Binario a:	**Octal**	**Sustitución: grupos de tres**
	Hexadecimal	**Sustitución: grupos de cuatro.**
	Decimal	**Suma polinomio**
Octal a:	**Binario**	**Sustitución: por registro de 3 b**
	Hexadecimal	**Sustitución: por binario de 4 b**
	Decimal	**Suma**
Hexadecimal a:	**Decimal**	**Sustitución**
	Octal	**Sustitución**
	Binario	**Suma**

ALGEBRA DE BOOLE

- **Punto de partida: Sistema binario, [0,1], se representa por dos voltages.**
- **Herramientas matemáticas para manipular las señales binarias con el fin de DISEÑAR CIRCUITOS DIGITALES.**
- **Def.: Algebra de Boole.**

Un conjunto B dotado con dos operaciones algebráicas + y - es un álgebra de Boole si y sólo si se verifican los postulados:

1- las operaciones + y - son conmutativas: $a+b = b+a$; $a \bullet b = b \bullet a$, $\forall$ a y b $\in$ B.

2- $\exists$ 0 y 1 $\in$ B tal que: $a+0 = 0+a = a$; $a \bullet 1 = 1 \bullet a = 1$ $\forall$ a $\in$ B.

3- cada operación es distributiva respecto de la otra:

$$a+(b \bullet c) = (a+b) \bullet (a+c); \; a \bullet (b+c) = a \bullet b + a \bullet c, \; \forall \text{ a y b} \in B.$$

4- $\forall$ a $\in$ B $\exists$ su complemetario, ca, $\in$ B tal que: $a + ca = 1$; $a \bullet ca = 0$. (ca = $\bar{a}$)

PROPIEDADES DEL ÁLGEBRA DE BOOLE

<u>Teorema 1</u> (IDEMPOTENCIA): Para todo elemento del álgebra de Boole se cumple:

$$b = b + b; \; b = b \bullet b$$

<u>Teorema 2</u>: Para todo elemento del álgebra de Boole se cumple:

$$1 = b + 1; \; 0 = b \bullet 0$$

<u>Teorema 3</u> (LEY DE ABSORCIÓN):Para todo elemento del álgebra de Boole se cumple:

$$a + ab = a; \; a(a+b) = a$$

<u>Teorema 4</u>: En un álgebra de Boole, las operaciones suma y producto son asociativas:

$$a + (b + c) = (a + b) + c = a + b + c; \; a(bc) = (ab)c = abc$$

PROPIEDADES DEL ÁLGEBRA DE BOOLE

<u>Teorema 5</u>: Para todo elemento del álgebra de Boole existe complementario, el cual es único.

<u>Teorema 6</u> (TEOREMA DE MORGAN): Para todo elemento del álgebra de Boole se cumple:

$$\overline{a+b+c+\ldots f} = \overline{a}\cdot\overline{b}\cdot\overline{c}\cdot\ldots\cdot\overline{f}$$

$$\overline{a\cdot b\cdot c\cdot\ldots\ldots\cdot f} = \overline{a}+\overline{b}+\overline{c}+\ldots+\overline{f}$$

<u>Teorema 7</u> (TEOREMA DE SHANNON): El complementario de una función se obtiene reemplazando cada variable por su complementaria y, al mismo tiempo, intercambiando las operaciones + y •.

$$\overline{f(x_1,x_2,\ldots,x_n,+,\cdot)} = f(\overline{x_1},\overline{x_2},\ldots,\overline{x_n},\cdot,+)$$

PROPIEDADES DEL ÁLGEBRA DE BOOLE

<u>Teorema 8</u> (TEOREMA DE EXPANSIÓN):

$$f(x_1,x_2,\ldots,x_n) = x_1\cdot f(1,x_2,\ldots,x_n) + \overline{x_1}\cdot f(0,x_2,\ldots,x_n)$$

f se ha expandido o desarrollado respecto a x_1.

DEFINICIONES:

VARIABLE BOOLEANA, Xi; representa cualquier elemento del álgebra de Boole.

FORMA BOOLEANA: Una expresión es una forma booleana si y sólo si se ha obtenido de alguna de las siguientes expresiones:

1- Una variable, Xi es una forma booleana.

2- 0 y 1 $\in$ álgebra de Boole es una forma booleana.

3- Si f_1 y f_2 son formas booleanas, entonces: f_1+f_2 y f_1f_2 también.

Una forma booleana puede utilizarse para describir una función de conmutación: FUNCIÓN LÓGICA, F(Xi).

F(Xi) → completamente especificada: exixte un único valor de salida para cada combinación de entrada.

→ Incompletamente especificada: si a una o más combinaciones de entrada se le pueden asignar los valores 0 o 1 indistintamente.

DEFINICIONES:

TÉRMINO CANÓNICO: de una función lógica e todo producto (productos canónicos) o suma (sumas canónicas) en el cual aparecen todas las variables o sus complementos de esa función.

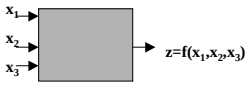
FORMA CANÓNICA: es toda función expresada como productos de sumas canónicas o como sumas de productos canónicos.

TABLA DE VERDAD: de una función lógica es la representación de la misma en la que se especifica el valor de la salida (0 o 1) para cada combinación de entrada. A partir de ella se deduce la forma canónica de dicha función como:

a) productos canónicos o,

b) sumas canónicas.

Ejemplo: circuito de conmutación combinacional y función lógica que establece.



$z=f(x_1,x_2,x_3)=\prod(0,3)=\sum(1,2,4,5,6,7)$

Tabla de verdad:

	x_1 x_2 x_3	$z=f(x_1,x_2,x_3)$
0	000	0
1	001	1
2	010	1
3	011	0
4	100	1
5	101	1
6	110	1
7	111	1

REALIZACIÓN DE FUNCIONES LÓGICAS MEDIANTE FUNCIONES BÁSICAS* EN EL ÁLGEBRA DE BOOLE

AND: producto lógico = 1 si y sólo si todas la entradas son 1.

$$a \cdot b$$

OR: suma lógica = 1 si al menos una entrada es 1.

$$a + b$$

NOT: inversora = función de complementación.

$$\overline{a}$$

***Simbología según norma IEEE Std. 91-1973.**

REALIZACIÓN DE FUNCIONES LÓGICAS MEDIANTE FUNCIONES BÁSICAS* EN EL ÁLGEBRA DE BOOLE

NAND = 1 si cualquiera de las entradas es 0.

$$\overline{a \cdot b} = \overline{a} + \overline{b}$$

NOR = si y sólo si todas las entradas son 0.

$$\overline{a + b} = \overline{a} \cdot \overline{b}$$

***Simbología según norma IEEE Std. 91-1973.**

REALIZACIÓN DE FUNCIONES LÓGICAS MEDIANTE FUNCIONES BÁSICAS* EN EL ÁLGEBRA DE BOOLE

OR-EXCLISIVE=EOR =1 si y sólo si el número de 1 en la entrada es impar.

$$a \oplus b$$

NOR-EXCLUSIVE = ENOR = 1 si y sólo si el número de 1 en la entrada es par.

$$\overline{a \oplus b}$$

{OR,NOT}, {AND,NOT}, {NAND}, {NOR}son CONJUNTOS COMPLETOS.

***Simbología según norma IEEE Std. 91-1973.**

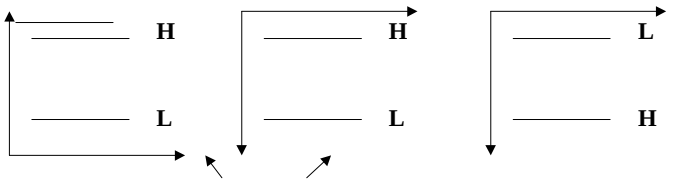
REPRESENTACIÓN DE VARIABLES LÓGICAS

Puertas lógicas son dispositivos electrónicos. No entienden de números sino de tensiones. Según relación número-tensión, hay dos tipos de lógica:

a) lógica positiva

b) lógica negativa

El dispositivo no altera su modo de funcionamiento por el tipo de lógica.



a) de tensiones positivas y negativas **b) lógica negativa**

ANÁLISIS DE CIRCUITOS COMBINACIONALES

Obtener a partir de un determinado circuito, las funciones lógicas mínimas que representan el comportamiento del mismo:

1- Determinar todas las entradas.

2- Salidas en cada etapa.

3- Último nivel: Función lógica.

4- Minimizar o simplificar si es posible.

5- Especificar función de salida.

DISEÑO DE CIRCUITOS COMBINACIONALES

Obtener el circuito lógico mínimo que funciones con las especificaciones iniciales:

1- Especificaciones del problema a diseñar. Simular problema señalando entradas y salidas.

2- Construir tabla de verdad de la función lógica a implementar.

3- Expresión canónica.

4- Simplificar por alguno de los métodos conocidos.

Si la implementación se va a realizar empleando un determinado tipo de puertas lógicas (NAN, NOR..), y/o un número de entradas prefijado, realizar las transformaciones adecuadas en f para que la expresión final simplificada resulte en la forma deseada.

5- Realizar diseño, según las opciones especificadas.