

Argentina \$ 27.- // México \$ 54.-

TÉCNICO en ELECTRÓNICA

CONCEPTOS FUNDAMENTALES Y PRÁCTICA PROFESIONAL

Electrónica digital y compuertas lógicas

- ▶ Sistema binario
- ▶ Mapas de Karnaugh
- ▶ Compuertas lógicas
- ▶ Circuitos combinacionales

www.tecuserspremium.blogspot.com



MCELECTRONICA

USERS

TÉCNICO en ELECTRÓNICA

CONCEPTOS FUNDAMENTALES Y PRÁCTICA PROFESIONAL

Coordinación editorial

Paula Budris

Asesores técnicos

Federico Pacheco

Nuestros expertos

Diego Aranda
Esteban Aredez
Alejandro Fernández
Lucas Lucyk
Luis Francisco Macías
Mauricio Mendoza
Norberto Morel
David Pacheco
Federico Pacheco
Gerardo Pedraza
Mariano Rabioglio
Luciano Redolfi
Alfredo Rivamar
Federico Salguero



Técnico en electrónica es una publicación de Fox Andina en coedición con Dálaga S.A. Esta publicación no puede ser reproducida ni en todo ni en parte, por ningún medio actual o futuro sin el permiso previo y por escrito de Fox Andina S.A. Distribuidores en Argentina: Capital: Vaccaro Sánchez y Cia. S.C., Moreno 794 piso 9 (1091), Ciudad de Buenos Aires, Tel. 5411-4342-4031/4032; Interior: Distribuidora Interplazas S.A. (DISA) Pte. Luis Sáenz Peña 1832 (C1135ABN), Buenos Aires, Tel. 5411-4305-0114. Bolivia: Agencia Moderna, General Acha E-0132, Casilla de correo 462, Cochabamba, Tel. 5914-422-1414. Chile: META S.A., Williams Rebolledo 1717 - Ñuñoa - Santiago, Tel. 562-620-1700. Colombia: Distribuidoras Unidas S.A., Carrera 71 Nro. 21 - 73, Bogotá D.C., Tel. 571-486-8000. Ecuador: Disandes (Distribuidora de los Andes) Calle 7° y Av. Agustín Freire, Guayaquil, Tel. 59342-271651. México: Distribuidora Intermex, S.A. de C.V., Lucio Blanco #435, Col. San Juan Tlihuaca, México D.F. (02400), Tel. 5255 52 30 95 43. Perú: Distribuidora Bolivariana S.A., Av. República de Panamá 3635 piso 2 San Isidro, Lima, Tel. 511 4412948 anexo 21. Uruguay: Espert S.R.L., Paraguay 1924, Montevideo, Tel. 5982-924-0766. Venezuela: Distribuidora Continental Bloque de Armas, Edificio Bloque de Armas Piso 9no., Av. San Martín, cruce con final Av. La Paz, Caracas, Tel. 58212-406-4250.

Impreso en Sevagraf S.A. Impreso en Argentina.

Copyright © Fox Andina S.A. VI, MMXIII.

Anónimo

Técnico en electrónica / Anónimo ; coordinado por Paula Budris. - 1a ed. - Buenos Aires : Fox Andina; Dalaga, 2013.

576 p. ; 27x19 cm. - (Users; 23)

ISBN 978-987-1949-14-4

1. Informática. I. Budris, Paula, coord. II. Título.

CDD 005.3

En esta clase veremos...

NUESTRO PRIMER ACERCAMIENTO A LA ELECTRÓNICA DIGITAL Y LAS COMPUERTAS LÓGICAS; TAMBIÉN PRESENTAREMOS LAS TÉCNICAS QUE DEBEN CONOCERSE PARA TRABAJAR CON ESTE TIPO DE SEÑALES.



Aquí comenzaremos a introducirnos en un área de la electrónica llamada digital, que se diferencia de la analógica por no trabajar con señales de esta clase. La electrónica digital se basa en el uso de señales del tipo digital y sistemas del tipo binario, que codifican información que luego será interpretada y procesada.

En esta línea, veremos primero una explicación de los sistemas de numeración decimal, octal, hexadecimal y binario; luego distintos sistemas binarios, como Gray, BCD, y exceso 3, y la forma de realizar conversiones entre los diferentes sistemas. Además explicaremos las compuertas lógicas, sus tablas de verdad, simbología, características, y la combinación de compuertas para armar circuitos lógicos más complejos.

Finalmente, veremos los circuitos combinatoriales, entre los cuales se presentarán: el generador y detector de paridad, el multiplexor y demultiplexor, el codificador y decodificador, los conversores de código, y los comparadores. De ellos veremos los usos, diagramas, tablas de verdad y circuitos.

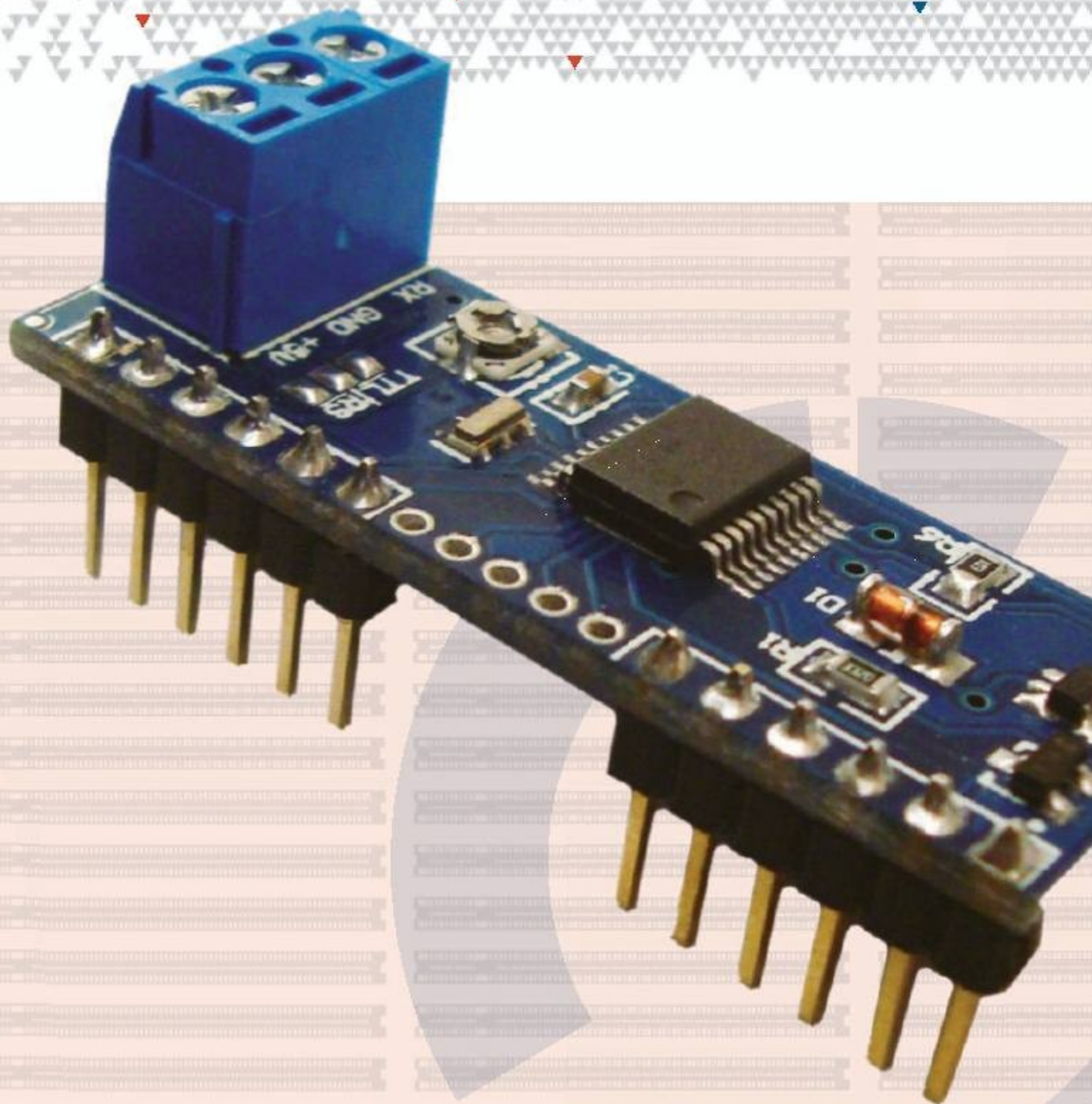
SUMARIO

- 02** **FUNDAMENTOS DE TÉCNICAS DIGITALES**
Sistemas de numeración y teorías relacionadas.
- 12** **COMPUERTAS LÓGICAS**
Características y familias lógicas.
- 20** **CIRCUITOS COMBINACIONALES**
Conoceremos diferentes circuitos, como el multiplexor y el codificador.



2

► Clase 11 //



FUNDAMENTOS DE TÉCNICAS DIGITALES

www.reduserspremium.blogspot.com.ar

LA ELECTRÓNICA FÍSICA CONSTA DE ELEMENTOS PREFABRICADOS QUE
PODEMOS ADQUIRIR E INSTALAR SEGÚN LAS ESPECIFICACIONES, PERO, PARA
SELECCIONARLOS, PRIMERO DEBEMOS ENTENDER SU FUNCIONAMIENTO.





Sistema binario

Antes de empezar a entender el funcionamiento de nuestros equipos digitales, necesitamos conocer su lenguaje de comunicación. Los seres humanos nos comunicamos en lenguajes predeterminados, como español, inglés portugués, francés, entre otros. Las máquinas utilizan como lenguaje el sistema binario.

Para entender el sistema binario, necesitamos introducir los sistemas de numeración que fueron aplicados previamente (incluso algunos sistemas aún los emplean) para concluir en el binario.

Uno de los primeros sistemas utilizados es el decimal que es el que manejamos a diario. Posee como base 10 dígitos diferentes que les corresponden al 0, 1, 2, 3, 4, 5, 6, 7, 8 y 9. Recibe el nombre de **sistema decimal** por herencia de los árabes e hindúes, quienes aplicaron este sistema a la numeración, basado en los dígitos de la mano principalmente para establecer una relación fácil de recordar. Este sistema es denominado posicional ya que el dígito representa el valor que se le asigna, dependiendo de su posición. El sistema decimal es un sistema en el que el valor de cada dígito depende de su lugar dentro del número.

Debemos tener en cuenta que, a partir de esta configuración, obtenemos los números enteros. Por

ejemplo, para obtener el número 695 es necesario efectuar las operaciones que vemos a continuación:

$$\begin{aligned}
 102.6 + 101.9 + 100.5 &= \\
 &= 600 + 90 + 5 = \\
 &= 695
 \end{aligned}$$

El sistema numérico utilizado como variante del decimal fue el sistema octal. Está basado en 8 dígitos (0, 1, 2, 3, 4, 5, 6 y 7). Posee dos caracteres menos que el decimal, lo que permite agruparlos en menores combinaciones.

Consideremos que el valor del sistema octal es el mismo que el del decimal y se utilizaba para evitar contar los dedos pulgares de la mano.

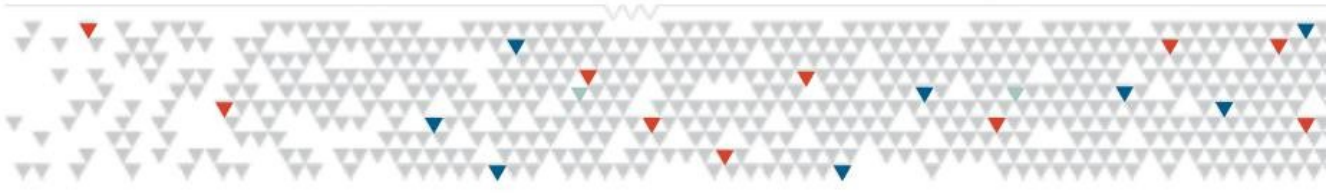
LAS CONVERSIONES DE SISTEMAS NUMÉRICOS POSIBILITARON LA AGRUPACIÓN DE INFORMACIÓN EN BYTES, LENGUAJE COMPUTACIONAL.



Billones			Millones			Unidades		
Millares de billón		Unidades de billón	Millares de millón		Unidades de millón	Millares		Unidades
Centenas	Decenas	Unidades	Centenas	Decenas	Unidades	Centenas	Decenas	Unidades
10^{17}	10^{16}	10^{15}	10^{14}	10^{13}	10^{12}	10^5	10^4	10^3
			10^{11}	10^{10}	10^9	10^6	10^2	10^1
								10^0

El sistema decimal es el más usado en el ámbito de la educación. Se trata del primer sistema numérico que aprendemos.





Convertir Decimal a octal

4

► Clase 11 //

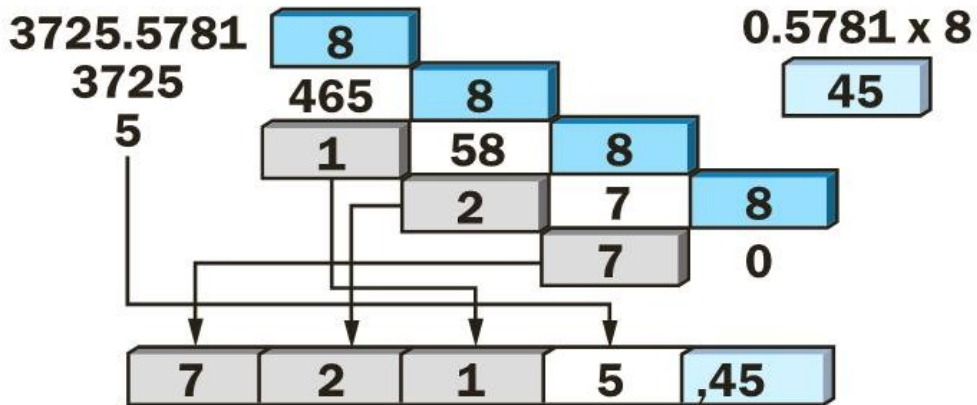


Diagrama de conversión decimal a octal mediante sustitución por restos.

Es utilizado en la informática para reemplazar al sistema **hexadecimal**, para utilizar números enteros en lugar de letras, como veremos más adelante. Se utiliza además para abreviar los números binarios que emplean caracteres de 6 bits. Cada 3 bits es convertido en un único dígito octal.

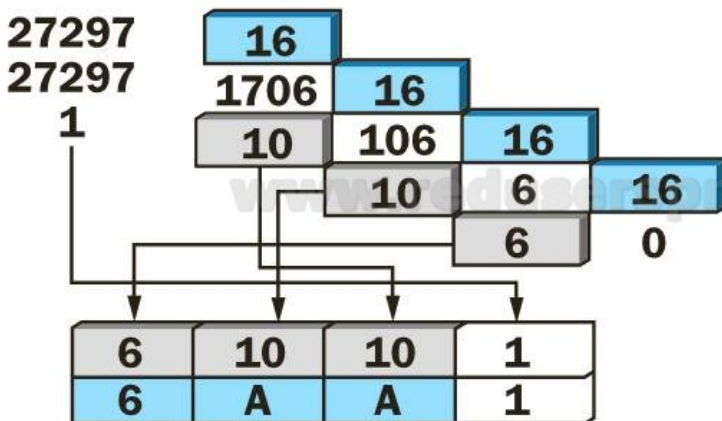
Como mencionamos antes, el sistema octal se utiliza para reemplazar, en determinadas, ocasiones al sistema hexadecimal. Este último sistema utiliza 16 símbolos sin repetir (de ahí el prefijo *hexa* representativo). Se usa principalmente en el mundo de la informática, ya que la unidad de trabajo es el byte, que representa al octeto de bits como unidad de memoria.

$7.83 + 2.82 + 1.81 + 5.80 + 4.8-1 + 5.8-2 =$
 $= 7.512 + 2.64 + 5.1 + 4.0,125 + 5.0,015625 =$
 $= 3584 + 128 + 5 + 0,5 + 0,078125 =$
 $= 3725,578125$
 $7215,45q = 3717,578125d$

(La letra *q* indica que es un número expresado en octal).

Posee por convención 16 caracteres (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F), y está basado en el sistema decimal de diez dígitos y seis letras. Las letras seleccionadas corresponden a las primeras seis del alfabeto latino, a las que se les asigna el correspondiente valor de: A = 10, B = 11, C = 12, D = 13, E = 14, F = 15. También es un sistema de numeración posicional, en el cual el valor de cada factor numérico se ve alterado dependiendo de la posición que este ocupe dentro de la cadena de números, multiplicado por la base del sistema (en este caso 16), elevado al exponente de la posición que ocupe. Ejemplificamos con 6AA116:

Convertir decimal a hexadecimal



$6.163 + A.162 + A.161 + 1.160 =$
 $= 6.4096 + 10.256 + 10.16 + 1.1 =$
 $= 24576 + 2560 + 160 + 1 =$
 $= 27297$
 $6AA116 = 27297$

Procedimiento para pasar del sistema decimal al hexadecimal. Los restos deben ser convertidos a letras hexadecimales si corresponde.

LAS CONVERSIONES DE SISTEMAS NUMÉRICOS POSIBILITARON LA AGRUPACIÓN DE INFORMACIÓN EN BYTES, LENGUAJE COMPUTACIONAL.



Para realizar operaciones con sistema hexadecimal, debemos tener en cuenta determinadas consideraciones. Recordemos que existe un límite, al igual que en el decimal, para realizar operaciones. Ejemplifiquemos una operación de suma.

Recordemos las reglas previamente comentadas y, al realizar operaciones numéricas correspondientes, no olvidemos que los dos únicos valores permitidos serán 1 y 0. Por lo que la suma se resumirá a: **Fórmula 2.**

$$7 + 7 = 14 = E$$

$$B + 1 = 11 = C$$

$$9 + 8 = 17$$

(17 no está en la escala hexadecimal, recordemos que el máximo es F = 15). Corregimos $17 - 16 = 1$; como superamos una vez el límite, tenemos 1 excedente. Entonces:

$$9 + 8 = 11 \text{ (En hexadecimal } 17 = 11).$$

$$7 + 7 = 14 = E$$

$$F + E = (15 + 14 = 29; 29 - 16 = 13; 1 13; 1D) 1D$$

Fórmula 1

$$1.24 + 0.23 + 0.22 + 1.21 + 1.20 =$$

$$= 16 + 0 + 0 + 2 + 1 =$$

$$= 19$$

En la actualidad, podemos realizar estas operaciones en calculadoras científicas reduciendo el engorroso trabajo de recordar todos los métodos, pero aun así es importante entender cómo funciona. Quizás el sistema numérico más utilizado en el mundo de la computación sea el **sistema binario**, ya que toda la codificación electrónica y de programación se encuentra basada en este sistema.

Con estas condiciones, se pueden realizar las operaciones aritméticas siguientes, respetando las reglas.

Existen sistemas binarios que difieren en algunas reglas en particular, que definen nuevos sistemas o códigos. Son utilizados, generalmente, en teorías digitales. El primero es conocido como **Código Gray.**

Fórmula 2

$$0 + 0 = 0$$

$$0 + 1 = 1$$

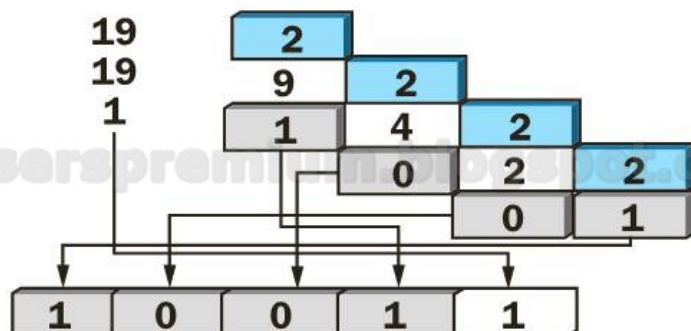
$$1 + 0 = 1$$

$$1 + 1 = 10 \text{ (2 no es un valor permitido, por lo que el primer valor volverá a 0 y se le asignará un segundo lugar, se le adiciona el 1).}$$

Para representar los números en el sistema binario, utilizamos solo dos caracteres: el 0 y el 1. En electrónica, los valores de 0 y 1 los utilizamos para establecer dos estados, encendido y apagado, dos niveles: ese es el principio de funcionamiento de la mayoría de los dispositivos actuales.

El sistema de representación está basado, al igual que los sistemas anteriores, en formato posicional de la base elevado al exponente que representa su posición en la cadena numérica. Por ejemplo, un número en binario 10011b lo convertimos mediante: **Fórmula 1.**

Convertir decimal a binario



Conversión de decimal a números binarios. Siempre se debe comprobar la conversión en un sentido como en el otro.





El sistema Gray (también conocido como **sistema binario reflejado**) es un código binario en el que dos valores sucesivos solo se diferencian en uno de sus dígitos. Se utilizó para prevenir señales erróneas en los switches electromecánicos. Actualmente, se utiliza en sistemas de televisión por cable y telecomunicaciones como sistema de corrección de errores. El sistema surge a partir del código binario convencional mediante un proceso reflejante que permite generar una secuencia de permutación cíclica.

El método conocido como **código Gray** consiste en tomar señales analógicas y convertirlas a grupos de códigos binarios, modificando solo un valor que garantice que cualquier transmisión variará solo en un bit. La ventaja de utilizar este método reside en que tiene un diseño eficiente, práctico y veloz.

Los sistemas de computación utilizan un estándar para representar números decimales en binario, conocido como **BCD** (*Binary-Coded Decimal*), en el que cada decimal es representado por un binario de 4 bits y no como un número binario puro. Con los decimales codificados, podemos realizar todas las operaciones matemáticas sin incurrir en las tediosas conversio-

Tablas comparativas de sistemas numéricos digitales

Decimal	Octal	Hexa-decimal	Binario
1	1	1	00001
2	2	2	00010
3	3	3	00011
4	4	4	00100
5	5	5	00101
6	6	6	00110
7	7	7	00111
8	10	8	01000
9	11	9	01001
10	12	A	01010
11	13	B	01011
12	14	C	01100
13	15	D	01101
14	16	E	01110
15	17	F	01111
16	20	10	10000
17	21	11	10001
18	22	12	10010
19	23	13	10011
20	24	14	10100

▲ **Tabla comparativa entre los cuatro sistemas estudiados. Se puede continuar su llenado utilizando los métodos ya descriptos.**

PARTE FRACCIONARIA

En todos los sistemas numéricos, podemos tratar con partes enteras y partes decimales. Estas partes decimales son tratadas de la misma manera que las enteras con la diferencia que los exponentes se reemplazan por números negativos relacionados con la posición que estos ocupan en la cadena de caracteres. De la coma a la izquierda, son considerados positivos; a la derecha, negativos. Otro método es la multiplicación de la parte fraccionaria por el representativo del sistema y reduciendo a su correspondiente factor.

nes que, por lo general, nos acarrearán demasiados errores en el momento de hacer operaciones complejas. Remarcamos que las operaciones en BCD entregan decimales en binarios de 4 bits por separado y no una cadena de números binarios, por lo que el sistema sirve para representar cifras, no al número en su totalidad

El código Exceso 3 es un derivado de BCD donde se le suma 3 (en binario) a cada dígito decimal y, luego, se lo convierte en un número binario de 4 bits.

Los distintos códigos surgen a partir de la necesidad de simplificar las operaciones y las posteriores codificaciones. La tec-

EL CÓDIGO BINARIO SE UTILIZÓ PARA REPRESENTAR DOS ESTADOS DE LA ELECTRÓNICA DE VÁLVULA ABIERTA (1) Y VÁLVULA CERRADA (0).



CALCULADORAS CIENTÍFICAS

nología se aceleró gracias a los métodos cada vez más eficaces para realizar operaciones aritméticas y, así, llevar a cabo operaciones digitales como las conocemos hoy.

Lógica binaria y álgebra de Boole

Las operaciones donde se implican números binarios comprenden ciertas reglas; como en la aritmética, llamamos álgebra a la suma, resta, multiplicación y división, pero la lógica binaria realiza estas operaciones según sus propias reglas. Los procesadores de los dispositivos electrónicos realizan estas operaciones denominadas **lógicas** con los datos contenidos en los programas que lo soliciten. A diferencia de las operaciones aritméticas en las que los resultados son valores numéricos, las operaciones lógicas nos entregan dos resultados que pueden ser leídos como **VERDADERO** o **FALSO**.

Operación unitaria (conocida como complemento) Se conoce como tal a todo elemento (a) de un conjunto A que le corresponda otro elemento distinto (b) del mismo conjunto A tal que se cumpla que el segundo sea el complemento del primero.
 $a = -b$ Ej: $8 = -8$



La calculadora científica puede ser de gran ayuda para efectuar operaciones complejas.

En todas las versiones de Windows, encontramos la calculadora científica como aplicativo base.

La calculadora base del sistema operativo Windows viene con una configuración básica, pero, mediante el menú Ver – Programador, obtendremos nuevas opciones para el cálculo en distintos sistemas numéricos: hexadecimal, decimal, octal y binario. La ventaja de esta calculadora es que podremos realizar operaciones aritméticas sin realizar ningún tipo de conversión previa.



	Binario	Decimal	Octal	Hexadecimal
Binario	_____	Fórmula Polinómica Pot. 2	Agrupar de a 3 bits	Agrupar de a 4 bits
Decimal	Entera/2 Fraccionaria/2	_____	Entera/8 Fraccionaria/8	Entera/16 Fraccionaria/16
Octal	C/dígito en binario 3 bits	Fórmula polinómica Pot. 8	_____	Pasar por binario
Hexadecimal	C/dígito en binario 4 bits	Fórmula polinómica Pot. 16	Pasar por binario	_____

Aquí vemos un cuadro de dos entradas que nos ofrece un pequeño ayuda memoria para recordar las operaciones por realizar.



Operación binaria (llamada suma y producto)

Debemos tener en cuenta que se define como tal operación al resultado de todo par ordenado (a, b) de elementos únicos de un conjunto A que le corresponda un único valor c como resultado de sumar los elementos del par ordenado, tal como vemos a continuación:

$$a + b = c \text{ Ej: } 3 + 4 = 7$$

Al producto del resultado c , como la multiplicación de los elementos del par ordenado.

$$a \cdot b = c \text{ Ej: } 3 \cdot 4 = 12$$

Operaciones nulas

Se define como operación nula al único resultado de multiplicar cualquier elemento del conjunto A por el número 0 que resulta siempre igual a 0.

$$a \cdot 0 = 0 \text{ Ej: } 3 \cdot 0 = 0$$

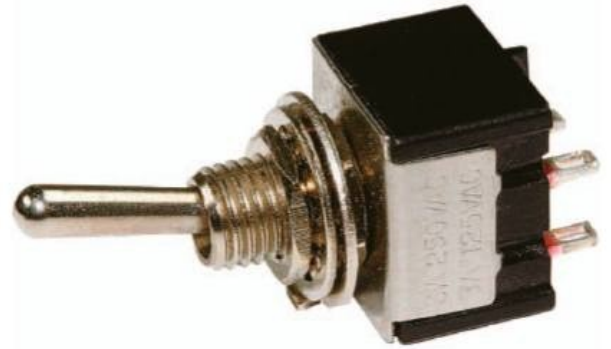
TABLA DE VERDAD

p	q	r	$\neg r$	$p \rightarrow q$	$((p \rightarrow q) \vee \neg r)$
V	V	V	F	V	V
V	V	F	V	V	V
V	F	V	F	F	F
V	F	F	V	F	V
F	V	V	F	V	V
F	V	F	V	V	V
F	F	V	F	V	V
F	F	F	V	V	V

Las tablas de verdad basadas en proposiciones son utilizadas para organizar los resultados de las operaciones algebraicas.

Las tablas de verdad son utilizadas para mostrar el valor de verdad de una proposición compuesta, es decir, determina el grado de verdad que posea cierta expresión para cada combinación de valores que se le pueda asignar a los componentes.

Para armar la tabla de verdad, se establece el sistema o definición, y las variables se asumen según el resultado.



▲ **Los interruptores de dos posiciones cumplen la función de verdadero y falso, o abierto y cerrado, que en binario será 0 y 1.**

Las operaciones booleanas responden a determinados **axiomas** que deben respetar. Para definir a una operación como perteneciente al álgebra de Boole, debe cumplir con:

▼ Ley asociativa de la suma. Si a, d y c pertenecen al conjunto A , entonces se debe cumplir que $(a + b) + c = a + (b + c)$.

▼ Ley asociativa del producto. Si a, d y c pertenecen al conjunto A , entonces se debe cumplir que $(a * b) * c = a * (b * c)$.

▼ Existe elemento nulo en la suma. Si a pertenece al conjunto A , entonces se debe cumplir que $a + 0 = a$.

▼ Existe elemento neutro en el producto. Si a pertenece al conjunto A , entonces se debe cumplir que $a * 1 = a$.

▼ Ley conmutativa de la suma. Si a y b pertenecen al conjunto A , entonces se debe cumplir que $a + b = b + a$.

▼ Ley conmutativa del producto. Si a y b pertenecen al conjunto A , entonces se debe cumplir que $a * b = b * a$.

▼ Ley distributiva de la suma respecto del producto. Si a, b y c pertenecen al conjunto A , entonces se debe cumplir que $a + (b * c) = (a + b) * (a + c)$.

▼ Ley distributiva del producto respecto a la suma. Si a, b y c pertenecen al conjunto A , entonces se debe cumplir que $a * (b + c) = (a * b) + (a * c)$.

▼ Existe elemento complemento para la suma. Si a pertenece al conjunto A , entonces existe un $\neg a$, que cumple $a + (\neg a) = 1$.

▼ Existe elemento complemento para el producto. Si a pertenece al conjunto A , entonces existe un $\neg a$, que cumple $a * (\neg a) = 0$.

Simbólicamente, las leyes se representan según su función, no se utilizan números, se utilizan símbolos representativos.

Propiedades	Intersección	Unión
Commutativa	$A \cap B = B \cap A$	$A \cup B = B \cup A$
Asociativa	$A \cap (B \cap C) = (A \cap B) \cap C$	$A \cup (B \cup C) = (A \cup B) \cup C$
Idempotente	$A \cap A = A$	$A \cup A = A$
Simplificación	$A \cap (B \cup A) = A$	$A \cup (B \cap A) = A$
Distributiva	$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
Elementos neutros	$A \cap E = A$	$A \cap \emptyset = A$
Absorción	$A \cap \emptyset = \emptyset$	$A \cup E = E$

A partir de los axiomas, podemos demostrar los teoremas fundamentales del álgebra:

▼ Ley de idempotencia para la suma implica que, para todo a perteneciente al conjunto A , se aplica $a + a = a$.

▼ Ley de idempotencia para el producto implica que, para todo a perteneciente al conjunto A , se aplica lo que ejemplificamos a continuación: $a * a = a$.

▼ Ley de absorción para la suma implica que, para todo a perteneciente al conjunto A , se aplica de la siguiente forma: $a + 1 = 1$.

▼ Ley de absorción para el producto implica que, para todo a perteneciente al conjunto A , se aplica $a * 0 = 0$.

▼ Ley de identidad para la suma implica que, para todo a perteneciente al conjunto A , se aplica $a + 0 = a$.

▼ Ley de identidad para el producto implica que, para todo a perteneciente al conjunto A , se aplica $a * 1 = a$.

▼ Ley de involución implica que, para todo a perteneciente al conjunto A , se aplica lo que mostramos a continuación: $-(-a) = a$.

▼ Ley del complemento implica que, para todo a perteneciente al conjunto A , se aplica que $-1 = 0$; $-0 = 1$

▼ Leyes de De Morgan implican que, para todo a y b perteneciente al conjunto A , se aplica que $-(a + b) = -a * -b$; $-(a * b) = -a + -b$

Mapas de Karnaugh

Cuando se realiza el cálculo de funciones booleanas, las operaciones pueden volverse muy tediosas y también complejas. Los mapas Karnaugh se empezaron a aplicar ante la necesidad de facilitar la resolución de estos inconvenientes y se convirtieron en una herramienta muy utilizada para la simplificación de circuitos lógicos.

El sistema utiliza técnicas que realiza el cerebro humano para detectar patrones y formas analíticas de reconocer, identificar y eliminar condiciones determinadas, minimiza expresiones del tipo

¿TE RESULTA ÚTIL?

Lo que estás leyendo es el fruto del trabajo de cientos de personas que ponen todo de sí para lograr un mejor producto. Utilizar versiones "pirata" desalienta la inversión y da lugar a publicaciones de menor calidad.

NO ATENTES CONTRA LA LECTURA. NO ATENTES CONTRA TI. COMPRA SÓLO PRODUCTOS ORIGINALES.

Nuestras publicaciones se comercializan en kioscos o puestos de vendedores; librerías; locales cerrados; supermercados e Internet (usershop.redusers.com). Si tienes alguna duda, comentario o quieres saber más, puedes contactarnos por medio de usershop@redusers.com



suma de productos y producto de sumas. Simplifica expresiones booleanas a mínimas cadenas, de modo que no exista otra con menor número de operaciones minimizando la cantidad de variables.

Previo a los mapas Karnaugh (de ahora en más los llamaremos **MK**) se utilizaban las tablas de verdad para obtener la función expresada por suma de productos. Con los MK, representamos la función deseada para obtener la respuesta directamente utilizando una **tabla de verdad bidimensional** (de dos entradas, verticales y horizontales).

Otorgamos jerarquía a los operadores siempre que cumplan que $a < b$ si se cumple que:

$$\begin{aligned} a + b &= b \\ a \cdot b &= a \\ -a + b &= 1 \\ a \cdot -b &= 0 \end{aligned}$$

Según el conjunto de números binarios, $a = 0$ y $b = 1$; entonces comprobamos que las afirmaciones anteriores son correctas.

Ley de Identidad	$A = A$ $\bar{A} = \bar{A}$
Ley Conmutativa	$A \cdot B = B \cdot A$ $A + B = B + A$
Ley Asociativa	$A \cdot (B \cdot C) = B \cdot A \cdot C$ $A + (B + C) = B + A + C$
Ley idempotencia	$A \cdot A = A$ $A + A = A$
Ley doble negativo/involución	$\bar{\bar{A}} = A$
Ley de Absorción	$A + 1 = 1$ $A \cdot 0 = 0$
Ley de Identidad	$A \cdot 1 = A$ $A + 0 = A$
Ley de Morgan	$\overline{AB} = \bar{A} + \bar{B}$ $\overline{A + B} = \bar{A} \cdot \bar{B}$
Ley Distributiva	$A \cdot (B + C) = (A \cdot B) + (A \cdot C)$ $A + (B \cdot C) = (A + B) \cdot (A + C)$

▲ Aquí vemos un cuadro con las leyes más importantes que debemos tener a mano para recordar el álgebra de Boole.

Para entender con claridad, una tabla de verdad de una función cualquiera de N variables posee 2N filas, para un MK debe poseer 2N cuadrados. Se ordenan según su peso y el código Gray para que solo una de las variables cambie entre las celdas adyacentes.

Cuando trabajamos con circuitos, una suma de productos se realiza con dos niveles de compuertas donde cada término representa una compuerta, y cada letra del producto es una entrada de la compuerta. Es más costoso adicionar una compuerta al circuito que incluir una única compuerta con múltiples entradas. Si podemos reducir un circuito mediante MK, tendremos dos circuitos equivalentes, de los cuales el reducido será más económico ya que poseerá menos compuertas o menor número de entradas.

La propiedad más destacable de los MK es la adyacencia de las celdas (dos celdas son adyacentes si no difieren en más de un bit) debido a que, si en las dos celdas adyacentes existe un valor 1, se puede realizar la operación "factor común" de estas celdas eliminando una variable.

Para entender con claridad, una tabla de verdad de una función cualquiera de N variables posee 2N filas, para un MK debe poseer 2N cuadrados. Se ordenan según su peso y el código Gray para que solo una de las variables cambie entre las celdas adyacentes.

CIRCUITOS DIGITALES

Compuertas Lógicas Digitales

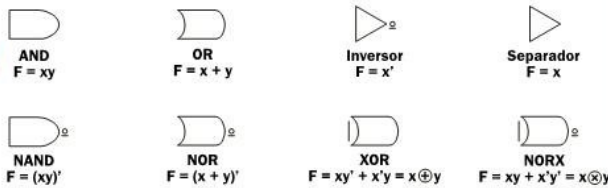


Tabla de equivalencia entre álgebra de Boole y circuitos digitales.

El equivalente en los circuitos digitales del álgebra de Boole son compuertas lógicas que responden en función a la programación. La equivalencia entre una forma y la otra se traduce en el diseño de los circuitos electrónicos que, posteriormente, son impresos en los circuitos integrados que casi siempre utilizaremos. Los símbolos representarán las compuertas que actúan según el diseño específico.

USO PRÁCTICO DEL DIAGRAMA PASO A PASO

Los MK son representaciones geométricas de los números binarios; un número binario de n bits se puede representar por un punto en el espacio N. Si representamos en un plano un número binario de un bit 0 y 1, y podemos unir ambos puntos mediante una línea que los conecte, a esta unión la llamaremos **cucho 1**. Si ahora reflejamos esta figura, obtendremos un cuadrilátero. Si al primer segmento superior le agregamos un 0 a la izquierda, y al segmento inferior (el reflejado) le agregamos un 1 a la izquierda, obtenemos un nuevo **cucho 2** con cuatro vértices, correspondiente a la combinación de dos variables ($2^2 = 4$). Es importante tener en cuenta que si recorremos los nuevos vértices, podremos observar cambios de una sola cifra.

El mapa de Karnaugh es uno de los métodos utilizados para la minimización de funciones algebraicas booleanas y es utilizado también para minimizar expresiones del tipo suma de productos o productos de sumas, obteniendo así otra suma de productos y otros productos de sumas. Como vimos se trata de una serie de cuadrados, donde cada cuadrado constituye una línea de la tabla de verdad. Dicha tabla se encarga de desplegar el valor de verdad de una proposición compuesta, para cada combinación de valores de verdad que se puedan establecer a sus componentes.

Operaciones de Álgebra de Boole

Operaciones	
Y	AND
$0 \cdot 0 = 0$	$0 \cdot 0 = 0$
$0 \cdot A = 0$	$0 \cdot 1 = 0$
$A \cdot 0 = 0$	$1 \cdot 0 = 0$
$A \cdot A = A$	$1 \cdot 1 = A$
O	OR
$A + 0 = A$	$0 + 0 = 0$
$A + 0 = 1$	$1 + 0 = 1$
$1 + A = A$	$0 + 1 = 1$
$1 + 1 = 1$	$1 + 1 = 1$
NO	NOT
$A'' = A$	$\overline{0} = 1$
$A' = \overline{A}$	$\overline{1} = 0$

Tabla comparativa entre las funciones del álgebra de Boole para un sistema cualquiera y un sistema binario. La variable A puede ser cualquier valor.

1

Tabla de 4 variables

A	B	C	D	F _n
0	0	0	0	1
0	0	0	1	
0	0	1	0	1
0	0	1	1	
0	1	0	0	1
0	1	0	1	
0	1	1	0	1
0	1	1	1	
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

16 cuadros, 8 posibles resultados

CD	00	01	11	10
00	1	1	0	1
01	0	0	0	1
11	0	0	0	0
10	1	1	0	1

Para comenzar armamos la tabla de verdad y posicionamos los valores de F_n en el MK.

2

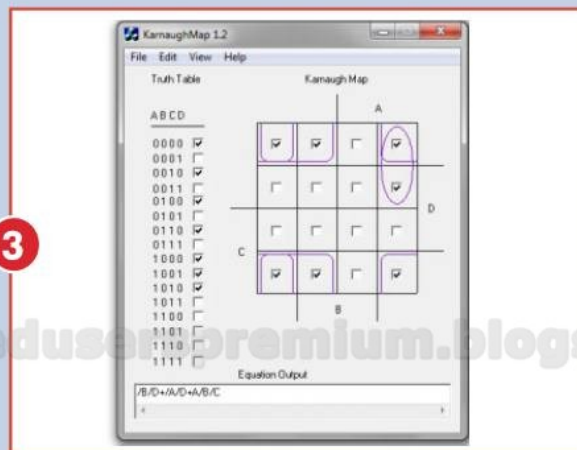
AB

	00	01	11	10
CD 00	1	1	0	1
01	0	0	0	1
11	0	0	0	0
10	1	1	0	1

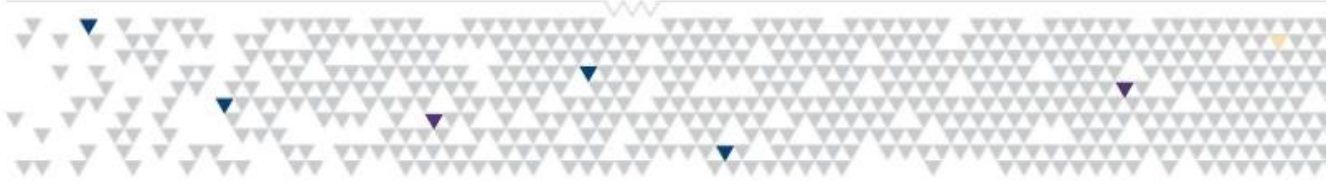
Podemos seleccionar como si se pudiese dar la vuelta y cerrar la figura

Luego agrupamos los 1 más próximos en grupos de dos en dos, solo vertical y horizontalmente con la menor cantidad de grupos posibles.

3



Leemos qué variables cambian dentro del rango seleccionado, hay tres rangos seleccionados, tres miniterminos. Comparemos con el programa.



12

► Clase 11 //



COMPUERTAS LÓGICAS

SUMAN, MULTIPLICAN, NIEGAN O AFIRMAN, INCLUYEN O EXCLUYEN SEGÚN SUS PROPIEDADES LÓGICAS. SE PUEDEN APLICAR A TECNOLOGÍA ELECTRÓNICA, ELÉCTRICA, MECÁNICA, HIDRÁULICA Y NEUMÁTICA.



www.reducerspremium.blogspot.com.ar



U

Una puerta lógica o compuerta lógica es, como bien dijimos, un dispositivo electrónico que representa la expresión física de un operador booleano en la lógica de conmutación. Cada puerta lógica consiste en una red de dispositivos interruptores que cumple las condiciones booleanas para el operador particular. Son, esencialmente, circuitos de conmutación integrados en un chip.

Es necesario que aclaremos, que las compuertas lógicas se comunican entre sí (incluidos los microprocesadores) usando el sistema binario, que consta de solo dos indicadores: 0 y 1.

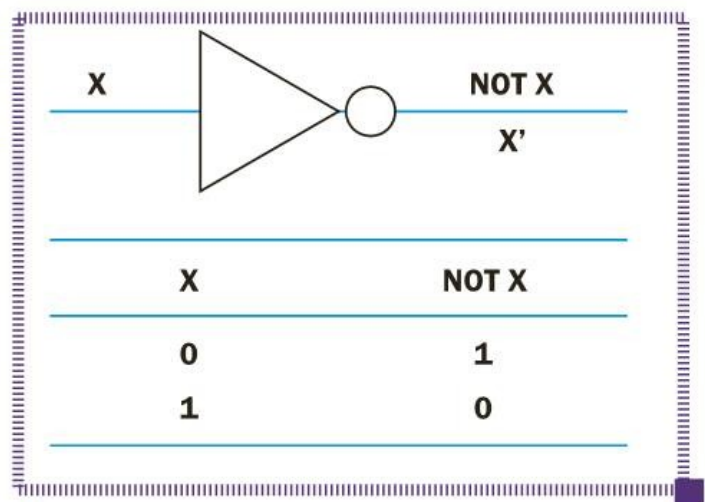
Así pues, encontramos gran variedad de compuertas estándares, cada una con un comportamiento perfectamente definido, y es posible combinarlas entre sí para obtener funciones nuevas. Desde el punto de vista práctico, podemos considerar cada compuerta como una caja negra, en la que introducimos valores digitales en sus entradas, y el valor del resultado aparece en la salida.

Cada compuerta tiene su tabla de verdad, que expresa en forma de lista el estado de su salida para cada combinación posible de estados en las entradas.

La compuerta **IF** (**SI** en español) realiza la función booleana de la igualdad. En los esquemas de un circuito electrónico se simboliza mediante un triángulo, cuya base corresponde a la entrada, y el vértice opuesto, a la salida. Su tabla de verdad es sencilla: la salida siempre toma el valor de la entrada. Esto significa que, si en la entrada hay un nivel

de tensión alto, también lo habrá en su salida; y, si la entrada se encuentra en un nivel bajo, su salida también será el mismo estado. Las compuertas IF se utilizan como amplificadores de corriente para permitir manejar dispositivos que tienen consumos de corriente elevados, desde otros que solo pueden entregar corrientes más débiles.

La compuerta **NOT** presenta, en su salida, un valor que es el opuesto del que está presente en su única entrada. Se simboliza en un esquema eléctrico en el mismo símbolo que la compuerta IF con un pequeño círculo agregado en su salida. Su función es la de negar, y comparte con la compuerta IF la característica de tener solo una entrada. Se utiliza cuando es necesario tener disponible un valor lógico opuesto a uno dado.



La puerta lógica NOT realiza la función booleana de inversión o negación de una variable lógica. Proporciona el estado inverso del que esté en su entrada.

CONJUNTO DE COMPUERTAS

Un conjunto de puertas lógicas completo es aquel con el que se puede implementar cualquier función lógica (puertas AND, OR y NOT; AND y NOT; OR y NOT; puertas NAND y puertas NOR). Un conjunto de puertas lógicas es completo si puede implementar todas las puertas de otro conjunto completo conocido. La equivalencia funcional entre compuertas lógicas es importante en el aspecto práctico, dado que un tipo de compuerta lógica puede tener mejores características operativas que otro, dependiendo de la tecnología utilizada.

CADA COMPUERTA CONSISTE EN UNA RED DE DISPOSITIVOS INTERRUPTORES QUE CUMPLE LAS CONDICIONES BOOLEANAS PARA EL OPERADOR.



La compuerta **AND**, llamada **Y** en español, con dos o más entradas, realiza la función booleana de multiplicación. Su salida será un 1 cuando todas sus entradas también estén en nivel alto. En cualquier otro caso, la salida será un 0. El resultado de multiplicar entre sí diferentes valores binarios solo dará como resultado 1 cuando todos ellos también sean 1.

La función booleana que realiza la compuerta OR (O) es la asociada a la suma, y matemáticamente la expresamos como **+**. Presenta un alto estado en su salida cuando, al menos una de sus entradas, también está en estado alto. En cualquier otro estado, la salida será 0. Tal como ocurre con las compuertas AND, el número de entradas puede ser mayor a dos.

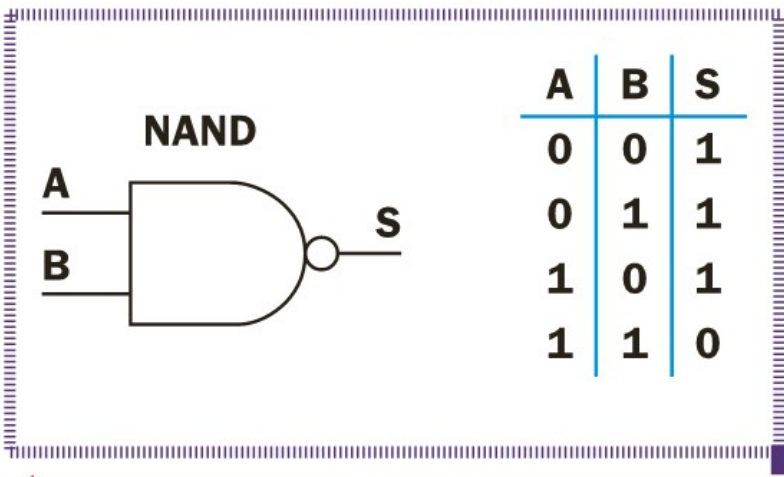
La compuerta **NAND (NO Y)** es la negación de la compuerta AND. Cualquier compuerta lógica se puede negar agregando una compuerta NOT que realice esta tarea. Debido a que es una situación muy común, se han fabricado compuertas que ya están negadas internamente, como es el caso de la NAND. Esto modifica su tabla de verdad, y la salida solo será un 0 cuando todas sus entradas estén en 1. El número de entradas debe ser de dos como mínimo, pero no nos será extraño encontrar NAND de tres o más entradas.

La compuerta **NOR (NO O)** es de forma similar a la NAND; una compuerta NOR es la negación de una compuerta OR, obtenida agregando una etapa NOT en su salida.

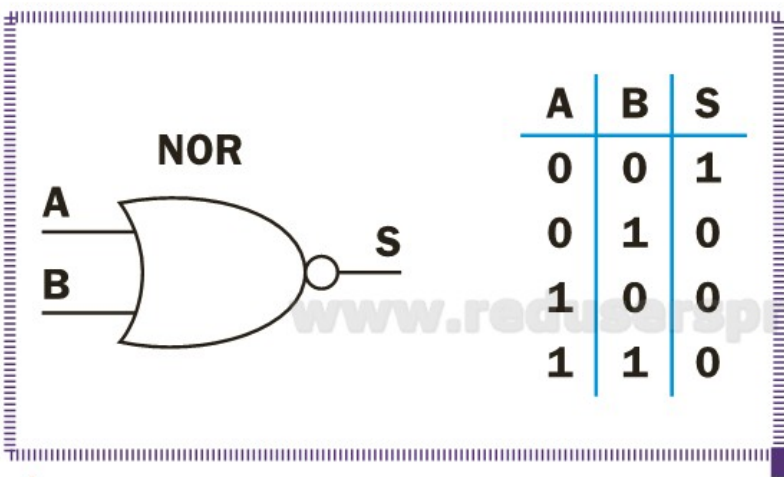
La salida de una compuerta NOR es 1 solamente cuando todas sus entradas son 0. Igual que en casos anteriores, la negación se expresa en los esquemas mediante un círculo en la salida. El número de entradas también puede ser mayor a dos.

En una compuerta **XOR (OR exclusiva)** realiza la función booleana de $A'B + AB$. Esta compuerta da como resultado 1 cuando los valores en las entradas son distintos (por ejemplo en una puerta de dos entradas). Si la puerta tuviese tres o más entradas, la XOR tomaría la función de suma de paridad: cuenta el número de unos a la entrada y, si resulta un número impar, pone un 1 a la salida para que el número de unos pase a ser par. Esto se debe a que la operación XOR es asociativa.

Finalmente, mencionaremos la compuerta **NXOR (No o Exclusivo)**, que no es más que una compuerta XOR con su salida negada, por lo que su salida estará en estado alto solo cuando sus



Podemos definir la puerta NO-Y, o NAND, como aquella que proporciona a su salida un 0 lógico únicamente cuando todas sus entradas están a 1.



La compuerta NOR proporciona, a su salida, un 1 lógico solo cuando todas sus entradas están a 0. Constituye un conjunto completo de operadores.

LA MAYORÍA DE LAS CENTRALES ELÉCTRICAS QUE EXISTEN HOY EN DÍA EN EL MUNDO TRANSFORMAN ENERGÍA MECÁNICA EN ELÉCTRICA.



La compuerta **AND**, llamada **Y** en español, con dos o más entradas, realiza la función booleana de multiplicación. Su salida será un 1 cuando todas sus entradas también estén en nivel alto. En cualquier otro caso, la salida será un 0. El resultado de multiplicar entre sí diferentes valores binarios solo dará como resultado 1 cuando todos ellos también sean 1.

La función booleana que realiza la compuerta OR (O) es la asociada a la suma, y matemáticamente la expresamos como **+**. Presenta un alto estado en su salida cuando, al menos una de sus entradas, también está en estado alto. En cualquier otro estado, la salida será 0. Tal como ocurre con las compuertas AND, el número de entradas puede ser mayor a dos.

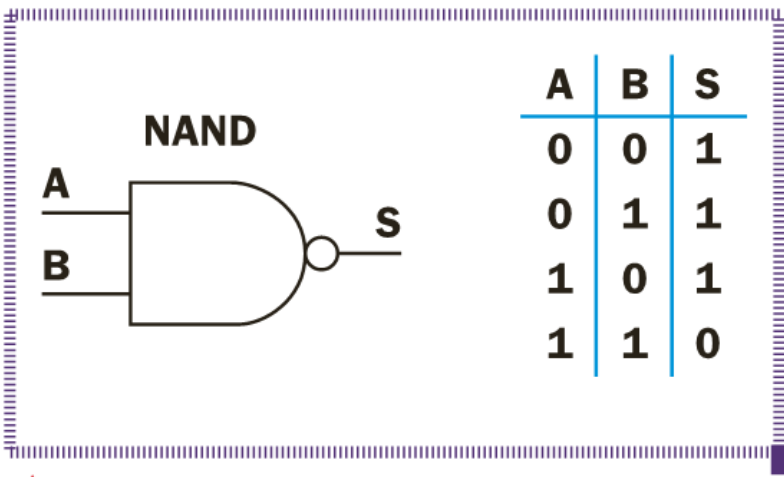
La compuerta **NAND (NO Y)** es la negación de la compuerta AND. Cualquier compuerta lógica se puede negar agregando una compuerta NOT que realice esta tarea. Debido a que es una situación muy común, se han fabricado compuertas que ya están negadas internamente, como es el caso de la NAND. Esto modifica su tabla de verdad, y la salida solo será un 0 cuando todas sus entradas estén en 1. El número de entradas debe ser de dos como mínimo, pero no nos será extraño encontrar NAND de tres o más entradas.

La compuerta **NOR (NO O)** es de forma similar a la NAND; una compuerta NOR es la negación de una compuerta OR, obtenida agregando una etapa NOT en su salida.

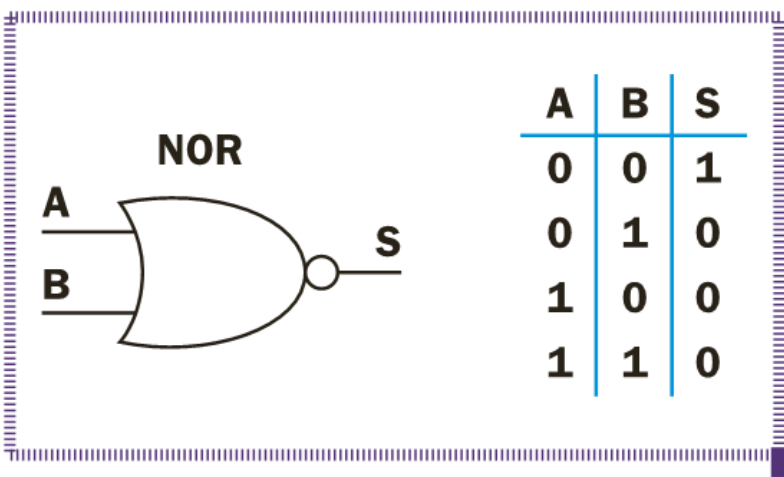
La salida de una compuerta NOR es 1 solamente cuando todas sus entradas son 0. Igual que en casos anteriores, la negación se expresa en los esquemas mediante un círculo en la salida. El número de entradas también puede ser mayor a dos.

En una compuerta **XOR (OR exclusiva)** realiza la función booleana de $A'B + AB$. Esta compuerta da como resultado 1 cuando los valores en las entradas son distintos (por ejemplo en una puerta de dos entradas). Si la puerta tuviese tres o más entradas, la XOR tomaría la función de suma de paridad: cuenta el número de unos a la entrada y, si resulta un número impar, pone un 1 a la salida para que el número de unos pase a ser par. Esto se debe a que la operación XOR es asociativa.

Finalmente, mencionaremos la compuerta **NXOR (No o Exclusivo)**, que no es más que una compuerta XOR con su salida negada, por lo que su salida estará en estado alto solo cuando sus



Podemos definir la puerta NO-Y, o NAND, como aquella que proporciona a su salida un 0 lógico únicamente cuando todas sus entradas están a 1.



La compuerta NOR proporciona, a su salida, un 1 lógico solo cuando todas sus entradas están a 0. Constituye un conjunto completo de operadores.

LA MAYORÍA DE LAS CENTRALES ELÉCTRICAS QUE EXISTEN HOY EN DÍA EN EL MUNDO TRANSFORMAN ENERGÍA MECÁNICA EN ELÉCTRICA.



MÁQUINA DE TURING

La máquina de Turing puede considerarse como un autómeta capaz de reconocer lenguajes formales. En ese sentido, puede reconocer los lenguajes recursivamente enumerables, de acuerdo a la jerarquía de Chomsky. Su potencia es, por tanto, superior a otros tipos de autómetas, como el autómeta finito o el autómeta con pila, o igual a otros modelos con la misma potencia computacional.

entradas son iguales y, en estado bajo, para las demás combinaciones posibles.

La **máquina de Turing** es un dispositivo de reconocimiento de lenguajes, que manipula símbolos de acuerdo con una tabla de reglas. A pesar de su simplicidad, puede ser adaptada para simular la lógica de cualquier algoritmo de computadora.

Dicha máquina tiene un control finito, una cabeza lectora y una cinta donde puede haber caracteres, y donde eventualmente viene la palabra de entrada. La cinta es de longitud infinita hacia la derecha, hacia donde se extiende en forma indefinida, y los espacios se llenan con el carácter blanco. La cinta no es infinita hacia la izquierda, por lo que hay un cuadro de la cinta que es el extremo izquierdo; la cabeza lectora es de lectura y escritura, por lo que la cinta puede ser modificada en curso de ejecución. Además, la cabeza se mueve bidireccionalmente (izquierda y derecha), por lo que puede pasar repetidas veces sobre un mismo segmento de la cinta.

Este modelo está conformado por un alfabeto de entrada y uno de salida, un símbolo especial llamado blanco, un conjunto de estados finitos y un conjunto de transiciones entre dichos estados. Su funcionamiento se basa en una función de transición, que recibe un estado inicial y una cadena de caracteres (la cinta, la cual es finita por la izquierda) pertenecientes al alfabeto de entrada. Luego va leyendo una celda de la cinta, borrando el símbolo, escribe el nuevo símbolo perteneciente al alfabeto de salida y, por último, avanza a la izquierda o a la derecha (solo una celda a la vez), repitiendo esto según se indique en la función de transición, para finalmente detenerse en un estado final o de aceptación, que representa la salida.

La máquina de Turing consta de un cabezal lector/escritor y de una cinta infinita en la que el cabezal lee el contenido, borra el contenido anterior y escribe un

LAS COMPUERTAS LÓGICAS SE COMUNICAN ENTRE SÍ USANDO EL SISTEMA BINARIO, QUE CONSTA DE SOLO DOS INDICADORES: 0 Y 1.



X	Y	X AND Y
0	0	0
0	1	0
1	0	0
1	1	1

Conocida por su nombre en inglés AND, la compuerta Y realiza la función booleana de producto lógico. Su símbolo es un punto, aunque se suele omitir.

nuevo valor. Las operaciones que se pueden realizar en esta máquina se limitan a: avanzar el cabezal lector/escritor hacia la derecha y avanzar el cabezal lector/escritor hacia la izquierda. El cómputo es determinado a partir de una tabla de estados de la forma: (estado, valor) (nuevo estado, nuevo valor, dirección).

Esta tabla toma como parámetros el estado actual de la máquina y el carác-

ter leído de la cinta, y da la dirección para mover el cabezal, el nuevo estado de la máquina y el valor que será escrito en la cinta.

Mediante este modelo teórico y el análisis de complejidad de algoritmos, fue posible la categorización de problemas computacionales de acuerdo con su comportamiento; así, aparece el conjunto de problemas denominados **P** y **NP**,



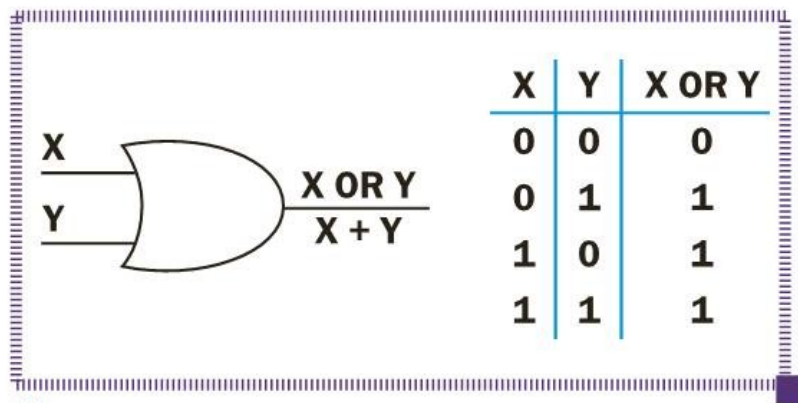


EN SU VERSION ORIGINAL,
LA MÁQUINA DE TURING
CONSISTE EN UNA CINTA
INFINITAMENTE LARGA CON
UNOS Y CEROS.



TURING

Turing, en 1948, definió su máquina como "... una ilimitada capacidad de memoria obtenida en la forma de una cinta infinita marcada con cuadrados, en cada uno de los cuales podría imprimirse un símbolo. En cualquier momento, hay un símbolo en la máquina, llamado el símbolo leído. La máquina puede alterar el símbolo leído, y su comportamiento está en parte determinado por ese símbolo, pero los símbolos en otros lugares de la cinta no afectan el comportamiento de la máquina. Sin embargo, la cinta se puede mover hacia adelante y hacia atrás a través de la máquina, lo que constituye una de sus operaciones elementales".



La compuerta OR puede ser definida como aquella que proporciona a su salida un 1 lógico si, al menos una de sus entradas, está a 1.

cuyas soluciones en tiempo polinómico son encontradas según el determinismo y no determinismo, respectivamente, de la máquina de Turing.

De hecho, se puede probar en forma matemática que, para cualquier programa de computadora, es posible crear una máquina de Turing equivalente.

Funcionamiento

Ahora nos preguntamos: ¿cómo funciona? Es un dispositivo que transforma un INPUT en un OUTPUT; tanto el INPUT como el OUTPUT constan de números en código binario (ceros y unos). En su versión original, la máquina de Turing consiste en una cinta infinitamente larga con unos y

ceros, que pasa a través de una caja. La caja es tan fina que solo el trozo de cinta que ocupa un bit (0 o 1) está en su interior. La máquina tiene una serie de estados internos finitos que también se pueden numerar en binario.

Para llevar a cabo algún algoritmo, la máquina se inicializa en algún estado interno arbitrario. A continuación, se pone en marcha, y la máquina lee el bit que se encuentra en ese momento en su interior; luego, ejecuta alguna operación con ese bit (lo cambia o no, dependiendo de su estado interno). Después, se mueve hacia la derecha o hacia la izquierda, y vuelve a procesar el siguiente bit de la misma manera.

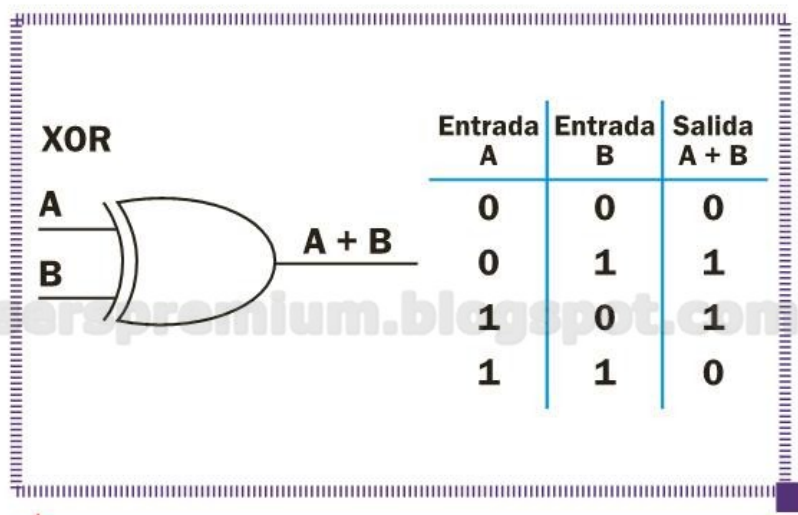


Imagen que refleja los elementos, partes y sectores de un generador de corriente alterna en las centrales eléctricas.

Una familia lógica es un conjunto de circuitos integrados compatibles entre sí, que se alimentan con el mismo voltaje y entienden los mismos niveles lógicos. Como consecuencia, las salidas de las puertas pueden conectarse directamente a las entradas de otras puertas de la misma familia sin ningún problema de adaptación ni de comunicación entre sí.

Una familia lógica va a estar determinada por el tipo de componente con que está construida, y la estructura básica con que se construyen las compuertas lógicas, así como sus valores.

Familias lógicas

Los circuitos integrados, como ya sabemos, se pueden clasificar en analógicos y digitales. Los circuitos integrados digitales funcionan con la base de la lógica booleana o álgebra de Boole, en la que cada operación es representada en electrónica digital por una compuerta lógica.

Algunos circuitos son considerados por su nivel de complejidad, es decir, por el número de puertas que estos contienen, y los fabricantes han desarrollado familias lógicas de dispositivos digitales que representan 0 y 1, por medio de parámetros eléctricos. En estos se definen propiedades de operación, como niveles de tensión y corriente, disipación de potencia, retardos de propagación, fan-out, entre otras.

Una familia lógica es el conjunto de circuitos integrados (CI's) los cuales pueden ser interconectados entre sí sin ningún tipo de Interfase o aditamento, es decir, una salida de un CI puede conectarse directamente a la entrada de otro CI de una misma familia. Se dice entonces que son compatibles.

Las familias pueden clasificarse en bipolares y MOS, podemos mencionar algunos ejemplos de familias bipolares: RTL, DTL, TTL, ECL, HTL, IIL. y de familias MOS: PMOS, NMOS, CMOS. Las tecnologías TTL y CMOS (metal oxido-semiconductor complementario) son los mas utilizadas en la fabricación de SSI (baja escala de integración) y MSI (media escala de integración).

AHUYENTADOR DE MOSQUITOS PASO A PASO

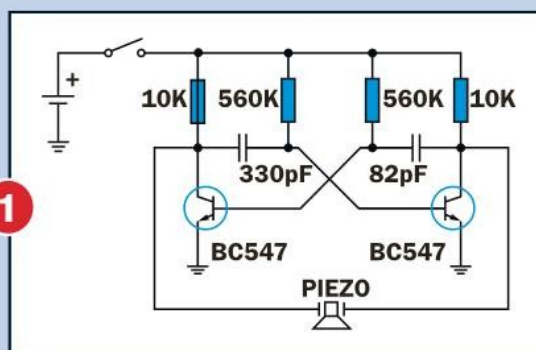
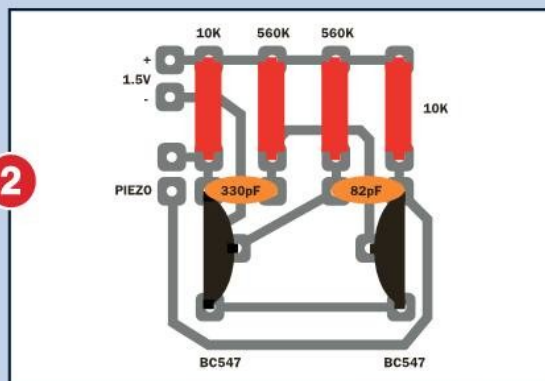
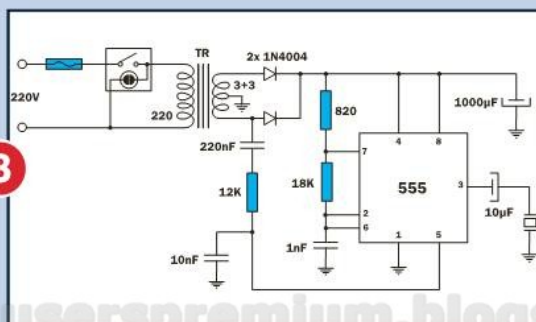


Diagrama circuital del ahuyentador. El corazón de este circuito consta de dos transistores NPN BC547, que se consiguen fácilmente en cualquier casa de electrónica.

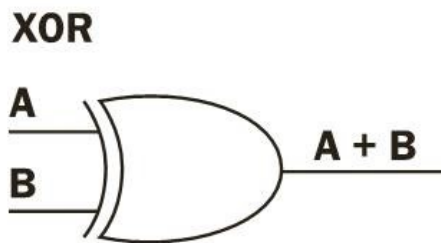


Otro vistazo, pero ahora cómo sería el diseño para PCB, de fácil implementación en un PCB virgen de 2,5 x 2,5 cm.



Circuito PLUS. Acá podemos ver otro diseño de un ahuyentador, en este caso implementado con un CI 555, en el que las resistencias regulan la oscilación del circuito.





Entrada A	Entrada B	Salida A + B
0	0	0
0	1	1
1	0	1
1	1	0

La compuerta XOR es aquella que proporciona un 1 lógico, solo si las dos entradas son iguales. Es verdadero si ambos componentes tienen el mismo valor lógico

Según el diseño y el montaje del sistema digital, el tipo de transistor u otros componentes electrónicos utilizados, se conocen dos tipos de familias lógicas. Por lo general, el tipo de componente usado en la construcción del circuito lleva el nombre de la tecnología utilizada, y cada familia está formada por subfamilias.

La primera familia que mencionaremos es la **bipolar**; esta emplea transistores bipolares (BJT), resistencias y diodos, es decir, dispositivos de unión. De esta familia, la más representativa es la **familia TTL**.

La familia TTL es caracterizada por su rapidez. Es saturante porque los transistores que la forman trabajan en corte-saturación (conmutación). Esa corriente de base elevada conlleva a que, cuando se desea pasar al corte, el exceso de carga acumulada en la base tarda en desalojarse, lo que contribuye a tiempos de conmutación mayor. Una de las mejoras introducidas en la familia TTL estándar es la utilización de un transistor de entrada multie-

misor, que favorece el paso de saturación al de corte, retirando la carga almacenada en la base del transistor durante la saturación. De esta familia, surgen variadas series, como la **74Sxxx**; esta trata de evitar que los transistores alcancen el estado de saturación, reduciendo así el exceso de carga en la zona de base. De esta forma, tarda menos en conmutar de saturación a corte, aumentando considerablemente la velocidad.

La serie **74LSxxx** es una mejora de la familia **74Sxxx**, la cual obtenía un gran aumento de velocidad respecto de la TTL estándar, a costa del aumento de las corrientes circulantes por las puertas.

LA FAMILIA TTL ES SATURANTE PORQUE LOS TRANSISTORES QUE LA FORMAN TRABAJAN EN CORTE-SATURACIÓN (CONMUTACIÓN).



Set de cuatro compuertas NAND (TTL), una de las primeras diseñadas en el mercado de la electrónica digital; fueron reemplazadas por las NAND CMOS.

La segunda familia por nombrar es la **MOSFET**. Estas familias emplean transistores MOSFET (transistores de efecto campo). Las más representativas son las familias NMOS, PMOS y CMOS. Puntualmente hablaremos de la familia CMOS.

Existen varias series en la familia CMOS de circuitos integrados digitales. La serie **4000** y la **14000** fueron las primeras. También existe la serie **74C** cuya característica principal es ser compatible terminal por terminal y función por función con los dispositivos TTL. Esto hace que sea posible reemplazar algunos circuitos TTL por un diseño equivalente CMOS.

La serie **74HC** está integrada por los CMOS de alta velocidad, que tienen un aumento de diez veces en la velocidad de conmutación. La serie **74HCT** es también de alta velocidad y compatible en lo que respecta a los voltajes con los dispositivos TTL.

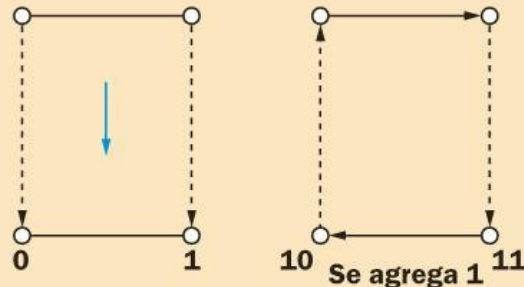
Los voltajes de alimentación en la familia CMOS tienen un rango muy amplio; estos valores van de 3 a 15V para los 4000 y los 74C. De 2 a 6V para los 74HC y 74HCT.

Por lo tanto, los márgenes de ruido se pueden determinar y llegan a ser de 1,5V. Esto es mejor que los TTL, ya que los CMOS pueden ser utilizados en medios con mucho más ruido. Los márgenes de ruido pueden hacerse todavía mayores si aumentamos el valor de VDD, ya que es un porcentaje de este.

En lo que a la disipación de potencia concierne, tenemos un consumo de potencia de solo 2,5 nW cuando VDD = 5V y, cuando VDD = 10V, la potencia consumida aumenta a solo 10 nW.

De esta forma podemos darnos cuenta que el aumento en el consumo no es considerable, teniendo en cuenta la potencia de consumo a la que nos estamos enfrentando, la cual aumenta desde 2,5 nW a tan solo 10 nW.

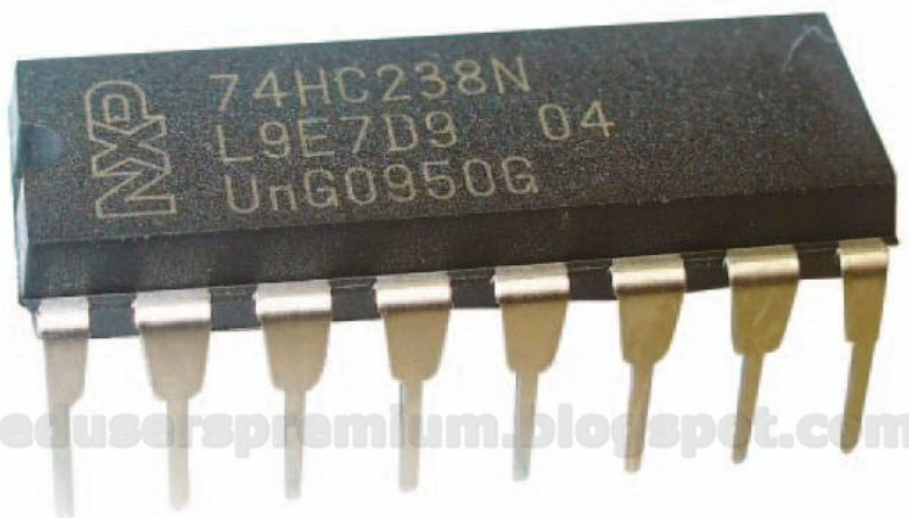
MÁRGENES DE RUIDO



PROYECTADO

Cuanto mayor sean los márgenes de ruido, los circuitos serán menos sensibles a las perturbaciones; se adecuarán para trabajar en entornos con mayor índice de ruido.

El ruido consiste en fluctuaciones no deseadas de la tensión real de una señal lógica. Puede ser debido a variación en la alimentación, perturbaciones electromagnéticas externas, etc. Si los niveles de tensión (verdadero/ falso) están muy próximos entre sí, sería relativamente fácil que una variación en la tensión provocada por ruido se pudiera pasar de un estado lógico a otro, o simplemente quedarse en la zona de transición.



▲ **Decodificador de tres Líneas (HIGH-CMOS) de rápida conmutación, muy utilizado en direccionamiento de memoria.**



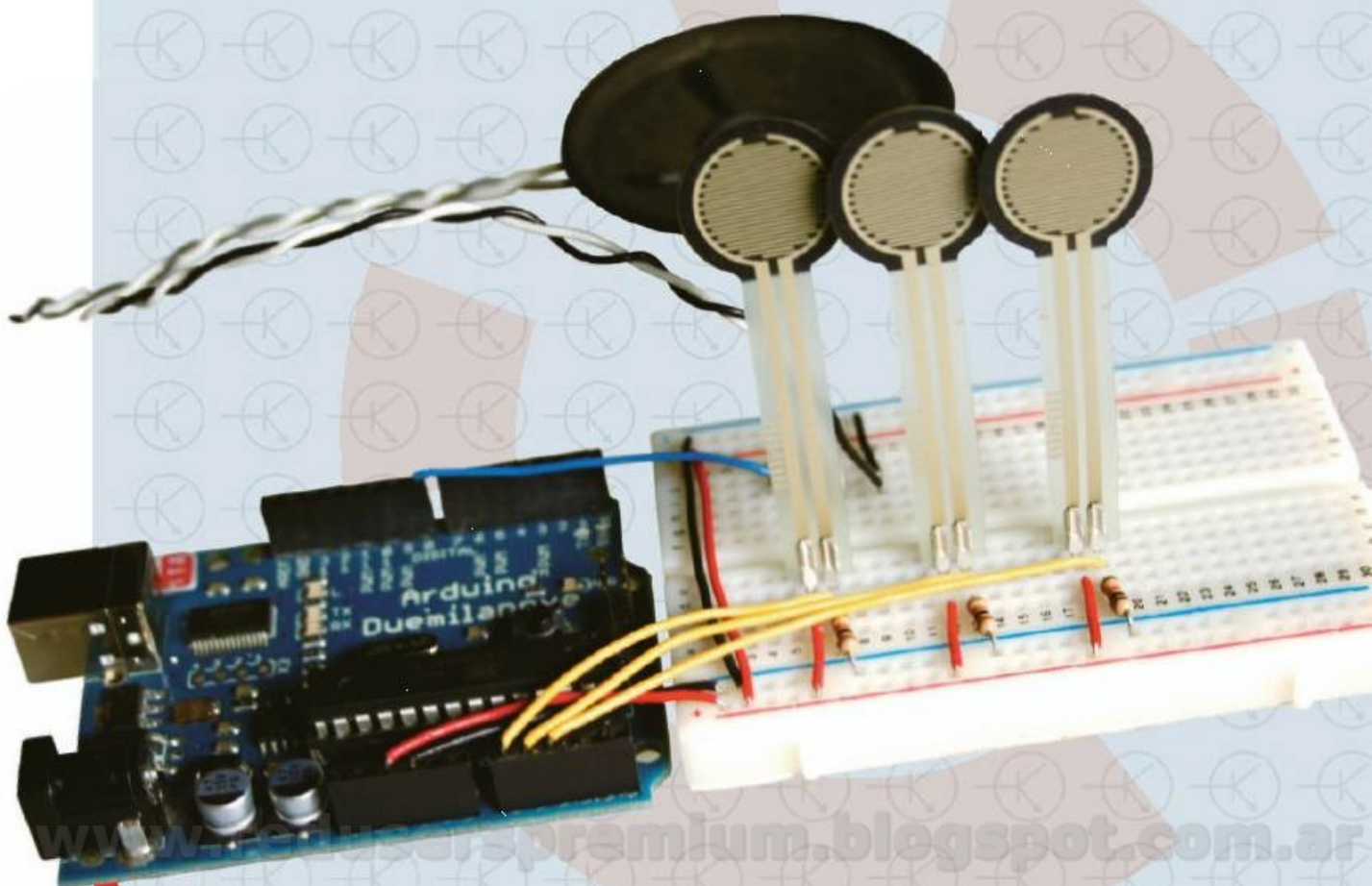
20

► Clase 11 //



CIRCUITOS COMBINACIONALES

LOS CIRCUITOS COMBINACIONALES SON CIRCUITOS LÓGICOS, EN LOS CUALES EL ESTADO DE SUS SALIDAS DEPENDE DE LA COMBINACIÓN DE LOS VALORES INSTANTÁNEOS DE SUS RESPECTIVAS ENTRADAS.



Generador y detector de paridad

X0	X1	X2	Fp	Fi
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	0

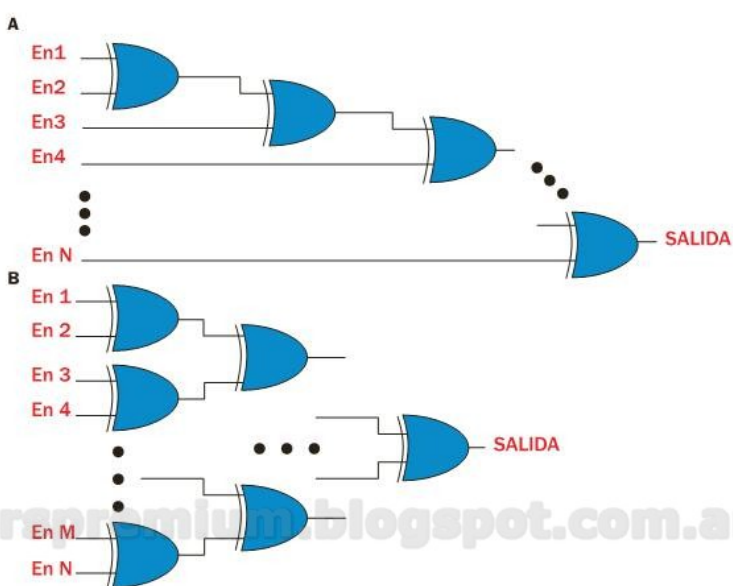
El circuito de la figura también es utilizado como receptor para verificar la paridad de un dato. Es necesario acoplar dos de estos circuitos para obtener al final un sistema de generación y verificación de paridad.

ESTOS CIRCUITOS NOS PERMITEN CHEQUEAR LA INFORMACIÓN ENVIADA Y DETECTAR ALGÚN ERROR PRODUCIDO EN ELLA.

X	Y	XOR
0	0	1
0	1	0
1	0	0

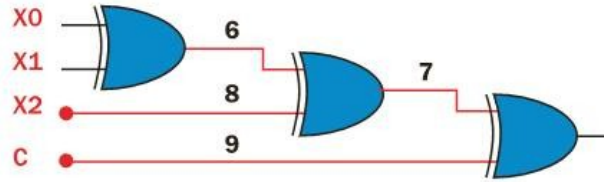
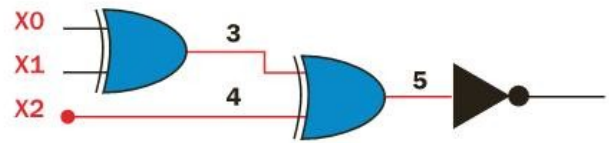
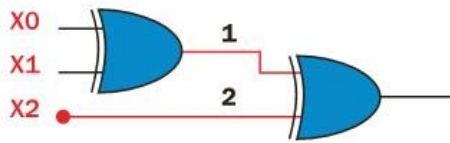
Como podemos observar, N compuertas XOR pueden conectarse en cascada y así formar un circuito de n entradas y una sola salida. De esta forma, podemos diferenciar los circuitos de paridad impar como aquellos cuya salida es 1 si un número impar de sus entradas es 1. Si la salida de cualquier circuito es invertida, se obtiene un circuito de paridad par, que se caracteriza porque su salida es 0 si un número par de sus entradas es 1.

Realizamos un ejemplo en el cual verificaremos tres bits mediante un circuito lógico. Realizamos la siguiente tabla de verdad.



En la imagen, vemos dos modelos de circuitos con paridad impar; el circuito B es mucho más rápido ya que posee una estructura en árbol.





C=0	par
C=1	impar

Multiplexor y demultiplexor

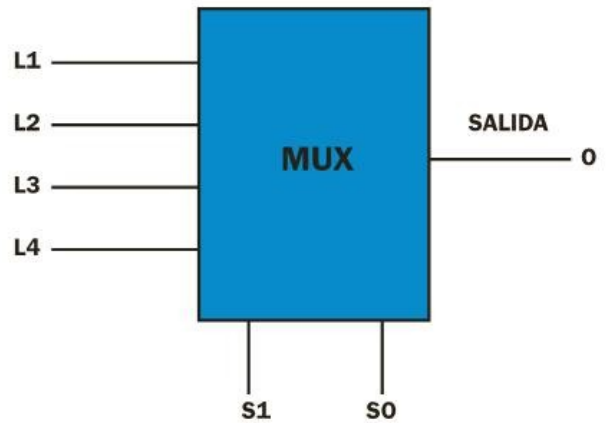
Los multiplexores son dispositivos que poseen varias señales de entrada y una sola señal de salida que toma el mismo valor que el de una entrada seleccionada. Se lo podría interpretar básicamente como una llave rotativa que selecciona una entrada.

Un multiplexor posee señales de habilitación o de selección, que son las encargadas de seleccionar el valor de una de las entradas que se transmitirá a la salida. Por ejemplo un multiplexor de ocho líneas de entradas necesita, como mínimo, tres señales de selección. O un multiplexor de cuatro líneas necesita dos señales de habilitación, y posee la siguiente tabla de verdad.

S1	S0	Y
0	0	L1
0	1	L2
1	0	L3
1	1	L4

Los decodificadores son dispositivos con una circuitería lógica en los cuales sus entradas presentan una combinación de ceros y unos, que identifican y seleccionan a una de las varias salidas que posee. Las salidas que son seleccionadas pasarán a un nivel activo, y todas las sali-

Sumamos la señal de control C que, en cero lógico, genera una función par y, cuando toma el valor uno lógico, genera la función impar.



Este es un esquema en bloque de un multiplexor de cuatro entradas y dos líneas de selección. Solo sirve para señales digitales.

das restantes que no fueron seleccionadas pasarán a un nivel inactivo. Las líneas de entradas también son llamadas **líneas de selección o de dirección**.

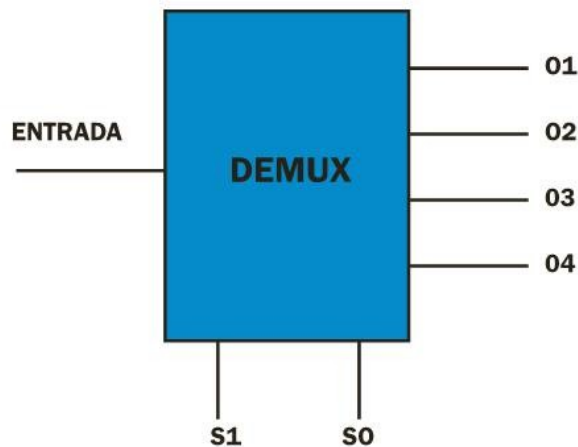
Un decodificador que posee ocho salidas necesitará tener, al menos, tres entradas de selección para que, a partir del estado de estas entradas, pueda identificarse una salida. Un decodificador con estas características se denomina **decodificador de 3 líneas a 8**.

MUX Y DEMUX ASOCIADOS

Por lo general, los demultiplexores se encuentran asociados a multiplexores. Este conjunto se presta muy bien para resolver el problema de tener, en un equipo digital, unidades que pueden transmitir información a varias unidades receptoras, lo que puede dar lugar a un gran número de conexiones. Se utiliza un MUX para seleccionar la señal proveniente de un trasmisor y colocarla en una línea, y un DEMUX para seleccionar un receptor para recibir dicha información

El tamaño de un multiplexor comercial no ve un límite en la cantidad de líneas de entrada que puede tener, ya que estos se expanden en una típica configuración tipo árbol, donde conectamos tres o más multiplexores en forma conjunta. Por ejemplo, si contamos con cinco multiplexores de cuatro líneas de entradas cada uno, mediante las conexiones correspondientes podemos tener un dispositivo de 16 líneas de entradas.

Codificador y decodificador



Las líneas de salidas están limitadas por la cantidad de líneas de habilitación que disponemos. El demultiplexor de la imagen es 4x2.

EN EL USO REAL DEL DEMULTIPLEXOR AL IGUAL QUE EN EL DEL MULTIPLEXOR, SE UTILIZAN LÍNEAS DE HABILITACIÓN QUE PERMITEN A LA LÍNEA DE SALIDA TRANSMITIR LA SEÑAL DE ENTRADA.

A2	A1	A0	00	01	02	03	04	05	06	07
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

En esta imagen observamos que cada una de las salidas toma el valor 1 para una única combinación de la entrada.

Como sabemos, los codificadores realizan la función contraria que la efectuada por los decodificadores. De esta forma, poseen varias entradas de las cuales solo una de ellas va a estar en el estado de activo.

En este sentido, debemos considerar que en los terminales de salida que correspon-

den al codificador, aparece la combinación binaria que se encarga de identificar a la única entrada activa.

Para este caso, podemos utilizar como ejemplo un codificador de 8 líneas a 3, el cual presenta la tabla de verdad que veremos en la siguiente página.

Si analizamos esta tabla de verdad tendremos en nuestras manos información relevante sobre el funcionamiento de un codificador de 8 líneas a 3.

Consideremos que, dependiendo del codificador con el que nos encontremos, la tabla de verdad será distinta.



I0	I1	I2	I3	I4	I5	I6	I7	A2	A1	A0
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

Tabla de verdad de un codificador de 8 líneas a 3.

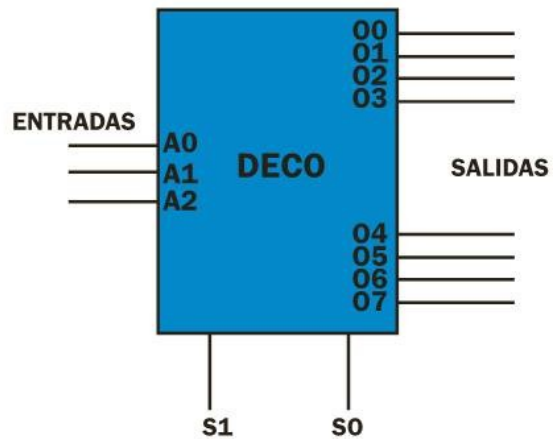
Podemos observar que esta tabla de verdad de 8 entradas independientes requiere 256 renglones, pero, teniendo en cuenta que solo una entrada puede estar activa, las posibilidades son solo ocho.

Un codificador que podemos encontrar en el mercado y que posee las características del descripto antes, es decir, 8 líneas de entrada a 3 de salida, es el 74x148, que posee todas sus entradas y sus salidas activas en bajo. Se deberían invertir los valores de la tabla de verdad expuesta con anterioridad.

Conversores de código

El convertor de códigos es un circuito combinacional lógico utilizado en los sistemas digitales. Como sabemos, las máquinas manejan el lenguaje de ceros y unos que es difícil de entender para las personas. Por esta razón, necesitamos los conversores de códigos para traducir el lenguaje de máquina al lenguaje de los humanos, y viceversa. Entre los códigos más utilizados y conocidos, podemos nombrar: BCD, octal, hexadecimal, decimal, etcétera.

Podemos desarrollar un convertor de códigos con el uso de dos dispositivos anteriormente vistos: el decodificador y el codificador. Citamos un ejemplo: podemos tener un código BCD recibido



Un decodificador de 3 líneas a 8. Este responde a la ecuación $p = 2q$. Donde p es líneas de salida, y q es líneas de selección.

por un decodificador que, en sus salidas, habilitará en total diez salidas (S0-S9) que representarán los números decimales; estas salidas se colocarán en las entradas de un codificador de siete segmentos. Hemos realizado, así, la conversión de

código BCD a uno de siete segmentos. Debemos tener en cuenta que la función básica de un convertor consiste en realizar la transformación de un código de n bits en otro de m bits, los cuales pueden ser independientes uno del otro.

DECODIFICADOR COMO DEMUX

Si observamos la tabla de verdad, podemos ver que un decodificador que suma una entrada de validación se comporta como un demultiplexor. Consideremos que, si esta entrada está activa, el dispositivo funciona, y, si no lo está, todas sus salidas estarán en cero, esto dependerá de si es activo en bajo o en alto. De hecho, no existen en el mercado circuitos integrados que realicen la función de demultiplexar con demultiplexores propiamente dichos, para esta función se utilizan decodificadores.

TÉCNICO en ELECTRÓNICA

CONCEPTOS FUNDAMENTALES Y PRÁCTICA PROFESIONAL

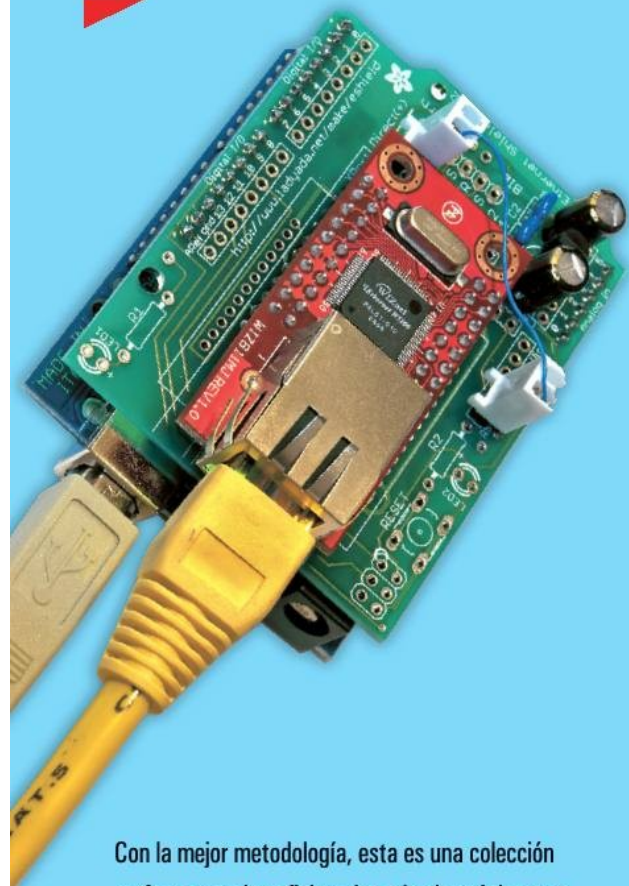


PROFESORES EN LÍNEA
profesor@redusers.com

SERVICIOS PARA LECTORES
usershop@redusers.com

SOBRE LA COLECCIÓN

CURSO VISUAL Y PRÁCTICO QUE BRINDA CONCEPTOS Y CONSEJOS NECESARIOS PARA CONVERTIRSE EN UN TÉCNICO EXPERTO EN ELECTRÓNICA. LA OBRA INCLUYE RECURSOS DIDÁCTICOS COMO INFOGRAFÍAS, GUÍAS VISUALES Y PROCEDIMIENTOS REALIZADOS PASO A PASO PARA MEJORAR EL APRENDIZAJE.



Con la mejor metodología, esta es una colección perfecta para los aficionados a la electrónica que deseen profesionalizarse y darle un marco teórico a su actividad, y para todos aquellos técnicos que quieran actualizar y profundizar sus conocimientos.

CONTENIDO DE LA OBRA

11/24

- 1 ▲ INTRODUCCIÓN A LAS REDES INFORMÁTICAS
- 2 ▲ PRINCIPIOS DE ELECTRÓNICA
- 3 ▲ EL LABORATORIO DE ELECTRÓNICA
- 4 ▲ CORRIENTE CONTINUA
- 5 ▲ CORRIENTE ALTERNA
- 6 ▲ DISPOSITIVOS ELECTRÓNICOS
- 7 ▲ CONSTRUCCIÓN DE CIRCUITOS
- 8 ▲ PROYECTOS: LUCES AUDIORÍTMICAS Y MICRÓFONO FM
- 9 ▲ DISEÑO DE CIRCUITOS IMPRESOS
- 10 ▲ SIMULACIÓN DE CIRCUITOS EN LA PC
- 11 **ELECTRÓNICA DIGITAL Y COMPUERTAS LÓGICAS**
- 12 ▼ TÉCNICAS DIGITALES APLICADAS
- 13 ▼ MICROPROCESADORES Y MICROCONTROLADORES
- 14 ▼ MICROCONTROLADORES PIC
- 15 ▼ PROYECTO: ANALIZADOR DE ESPECTRO CON PIC
- 16 ▼ CONECTIVIDAD POR CABLE
- 17 ▼ CONECTIVIDAD INALÁMBRICA
- 18 ▼ DISPLAYS
- 19 ▼ SENSORES Y TRANSDUCTORES
- 20 ▼ PROYECTO: MODIFICADOR DE VOZ
- 21 ▼ FUENTES DE ALIMENTACIÓN
- 22 ▼ PLATAFORMAS ABIERTAS
- 23 ▼ PLATAFORMA ARDUINO
- 24 ▼ PROYECTO: SISTEMA DE TELEMETRÍA CON ARDUINO



9 789871 949144



00011

PRÓXIMA ENTREGA



Técnicas digitales aplicadas

EN LA SIGUIENTE ENTREGA ANALIZAREMOS EL USO DE TÉCNICAS DIGITALES APLICADAS. CONOCEREMOS LAS MEMORIAS DE UN BIT, LOS CIRCUITOS SECUENCIALES, CONCEPTOS DE LÓGICA PROGRAMABLE Y APLICACIONES PARA ELECTRÓNICA DIGITAL.



www.redusersprofesional.blogspot.com.ar

