

SIDAD NACIONAL DEL SANTA

Facultad de Ingeniería

EAP INGENIERIA DE SISTEMAS E INFORMATICA

**DISEÑO DE CIRCUITOS
COMBINATORIOS USANDO EL
CONVERTIDOR LOGICO DIGITAL PARA
APLICACIONES EN SISTEMAS
DIGITALES**

Ing. Carlos Guerra Cordero

INDICE

CAPITULO I :	INTRODUCCION
CAPITULO II:	REVISION BIBLIOGRAFICA
CAPITULO III:	MATERIALES Y METODOS
CAPITULO IV:	RESULTADOS Y DISCUSIÓN
CAPITULO V:	ANALISIS DE LOS RESULTADOS
CONCLUSIONES Y RECOMENDACIONES	
BIBLIOGRAFIA y REFERENCIAS	

CAPITULO I INTRODUCCION

Los circuitos lógicos para sistemas digitales pueden ser combinacionales o secuenciales. Un circuito combinacional consta de puertas lógicas cuyas salidas en cualquier momento están determinadas en forma directa por la combinación presente de las entradas sin tomar en cuenta las entradas previas. Y realiza una operación específica de procesamiento de información, en forma lógica por un conjunto de funciones booleanas.

Para el desarrollo del informe el alumno conocerá los símbolos, la tabla de verdad y la expresión booleana de cada compuerta lógica, para resolver problemas del mundo real en el campo de la electrónica digital. Además deberá conectar compuertas para construir lo que los ingenieros conocen como circuitos lógicos combinacionales

CAPITULO I

Problema

¿De que manera el diseño de circuitos combinatorios usando el convertidor lógico tendrá aplicaciones en sistemas digitales?

Hipótesis

Utilizando el convertidor lógico digital se podrá diseñar los circuitos lógicos combinacionales, para sistemas digitales.

Objetivos

1. Determinar las compuertas lógicas digitales, básicas y secundarias
2. Diseño de circuitos lógicos combinacionales.
3. Uso del convertidor lógico, para simular un circuito lógico combinacional



*Your complimentary
use period has ended.
Thank you for using
PDF Complete.*

[Click Here to upgrade to
Unlimited Pages and Expanded Features](#)

Importancia y justificación del estudio

El uso del convertidor lógico, es importante por que nos ayuda a ilustrar, analizar y reparar problemas de sistemas digitales en forma inmediata.

El bloque funcional básico de cualquier circuito lógico es la compuerta lógica. Todo estudiante o profesional que trabaja con electrónica digital, tienen que comprender el uso de las compuertas lógicas, y sus aplicaciones que lleven a desarrollar problemas



*Your complimentary
use period has ended.
Thank you for using
PDF Complete.*

[Click Here to upgrade to
Unlimited Pages and Expanded Features](#)

CAPITULO II

REVISION BIBLIOGRAFICA

SEÑALES ANALOGICAS

SEÑALES DIGITALES

ALGEBRA BOOLEANA

COMPUERTAS

CIRCUITO INTEGRADO

LED

CAPITULO III

MATERIALES Y METODOS

Estrategia del estudio.

- ” Se determinará las puertas lógicas digitales
- ” Se desarrollará un problema para aplicar el uso del software Workbench.
- ” Se usara el simulador Convertidor lógico digital, para la solución de un problema

Diseño y características de la Muestra

Puertas lógicas digitales

Circuitos Integrados digitales

Simulación de circuitos MultiSIM

Técnicas e Instrumentos de Recolección de Datos


La investigación será a base de lecturas y análisis de manuales de electrónica digital.


Uso del convertidor lógico digital para aplicación de problemas


CAPITULO IV

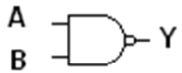
RESULTADOS Y DISCUSIÓN


Determinar las compuertas lógicas digitales


En español	A las entradas A y B se les aplica AND para generar la salida Y															
Como expresión booleana	$A * B = Y$ Símbolo de la operación AND															
Como símbolo lógico																
Como tabla de verdad	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	A	B	Y	0	0	0	0	1	0	1	0	0	1	1	1
A	B	Y														
0	0	0														
0	1	0														
1	0	0														
1	1	1														


En español	A las entradas A y B se les aplica la operación lógica OR para obtener Y como salida															
Como expresión booleana	$A + B = Y$ Símbolo de la operación OR															
Como símbolo lógico																
Como tabla de verdad	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	1
A	B	Y														
0	0	0														
0	1	1														
1	0	1														
1	1	1														

En español	A la entrada (A) se les aplica la operación lógica INVERSOR para obtener (A') como salida						
Como expresión booleana	$Y = A'$ Símbolo de la operación INVERSOR						
Como símbolo lógico							
Como tabla de verdad	<table border="1"> <thead> <tr> <th>A</th> <th>A'</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	A'	0	1	1	0
A	A'						
0	1						
1	0						

	Se aplica la operación lógica NAND a las entradas A y B para obtener la salida Y															
Como expresión booleana	$(A * B)' = Y$ Símbolo de la operación NAND															
Como símbolo lógico																
Como tabla de verdad	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	1	1	0	1	1	1	0
A	B	Y														
0	0	1														
0	1	1														
1	0	1														
1	1	0														

En español	La operación lógica NOR se aplica a las entradas A y B para obtener la salida Y															
Como expresión booleana	$(A + B)' = Y$ Símbolo de la operación NOR															
Como símbolo lógico																
Como tabla de verdad	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	0
A	B	Y														
0	0	1														
0	1	0														
1	0	0														
1	1	0														

En español	Se aplica la función lógica XOR a las entradas A y B para obtener la salida Y															
Como expresión booleana	$(A (+) B) = Y$ Símbolo de la operación XOR															
Como símbolo lógico																
Como tabla de verdad	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	0
A	B	Y														
0	0	0														
0	1	1														
1	0	1														
1	1	0														

En español	La operación lógica XNOR se aplica a las entradas A y B para obtener la salida Y															
Como expresión booleana	$(A (+) B)' = Y$ Símbolo de la operación XNOR															
Como símbolo lógico																
Como tabla de verdad	<table border="1"> <thead> <tr> <th>A</th> <th>B</th> <th>Y</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	1
A	B	Y														
0	0	1														
0	1	0														
1	0	0														
1	1	1														

BOOLEANAS

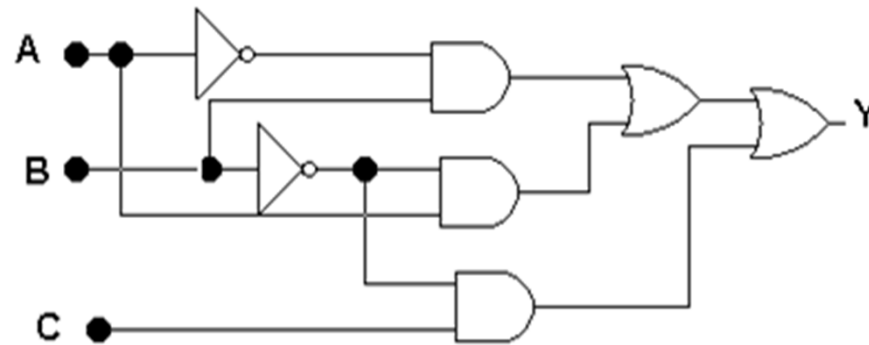
de circuitos a partir de sus expresiones

Supongamos que tenemos la expresión. Booleana $Y = A + B + C$ y se pide que se construya un circuito que realice esta función lógica.



Obtener el circuito lógico a partir de la expresión booleana

$$Y = AqB + A.Bq + BqC$$



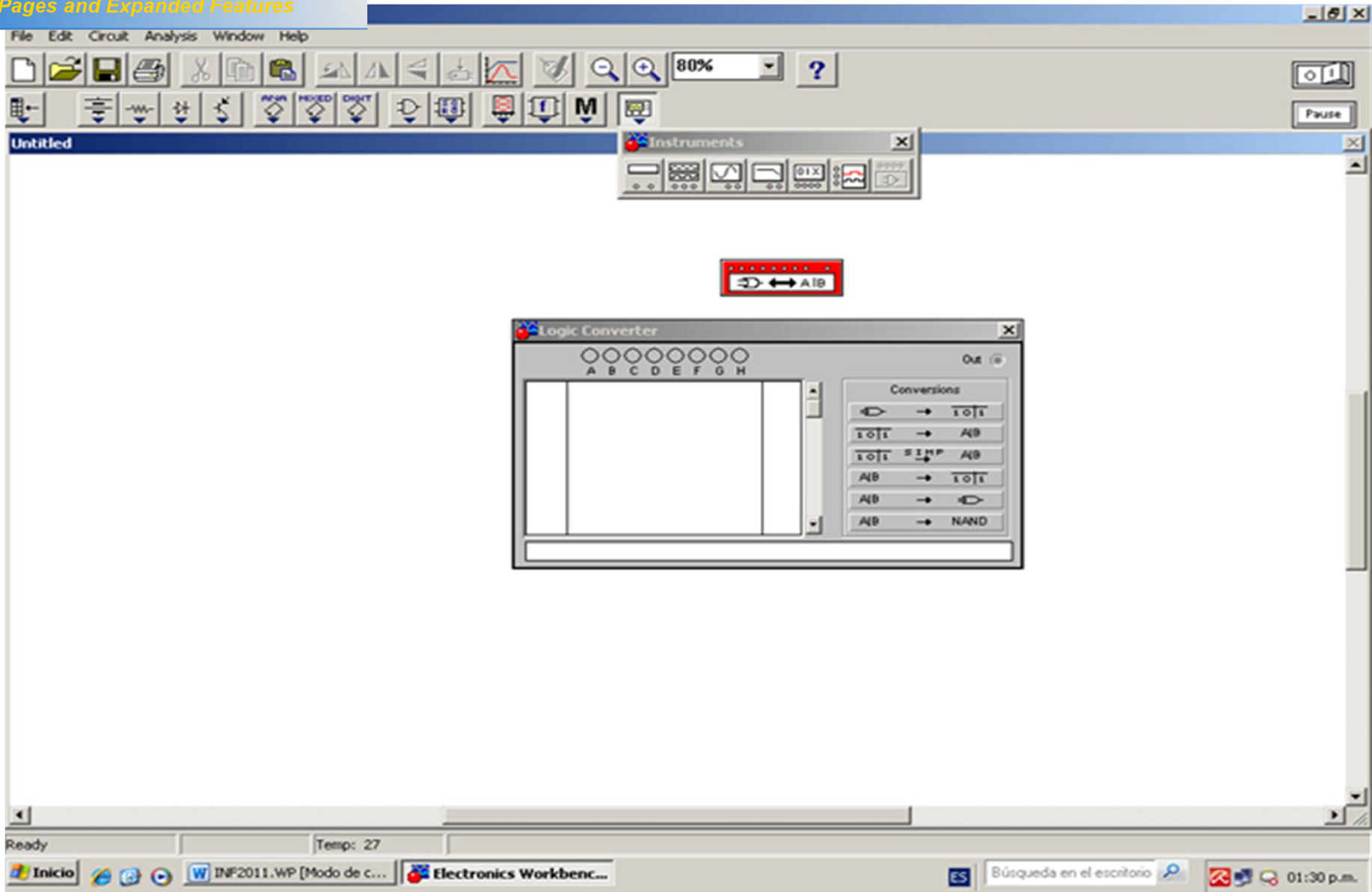
ad y expresiones booleanas

C	B	A	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Luego la expresión booleana será:

$$Y = C\bar{B}A + CB\bar{A}$$

en la simulación de circuitos



simulación:

a) Para introducir los términos de la expresión algebraica en el convertidor lógico los términos de la expresión algebraica. Para introducir el complemento de una variable se describe después de ella el símbolo ($\bar{}$) (comilla simple).

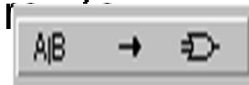
b) Para convertir una expresión booleana para una tabla de verdad, haga clic en el botón Expresión booleana para Tabla de verdad.



c) Para convertir una tabla de verdad de una expresión booleana simplificada, o para simplificar una expresión booleana existente, haga clic en el Simplificar



d) Para convertir una expresión booleana a un circuito, haga clic en el botón Expresión booleana para circuito.



e) Para convertir de un circuito lógico a tabla de verdad, haga clic en el botón circuito a tabla de verdad



Convertir una expresión booleana en una tabla de verdad

The screenshot shows a software window titled "Logic Converter". At the top, there are eight indicator lights labeled A through H, with A, B, and C being lit. Below the lights is a truth table with columns for A, B, C, and Out. The truth table contains the following data:

A	B	C	Out
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

At the bottom of the window, a text box contains the Boolean expression $A' B' C + ABC'$. To the right of the window, a "Conversions" panel contains several buttons. One button, labeled "A/B → 1 0 | 1", is highlighted with a callout that says "Paso 2: Presionar este boton". Another callout, "Paso 3: Despliegue de resultados", points to the truth table. A third callout, "Paso 1: Teclee la expresion booleana", points to the text box at the bottom.

una tabla de verdad en una expresión booleana simplificada

Paso 1: llenar la tabla de verdad

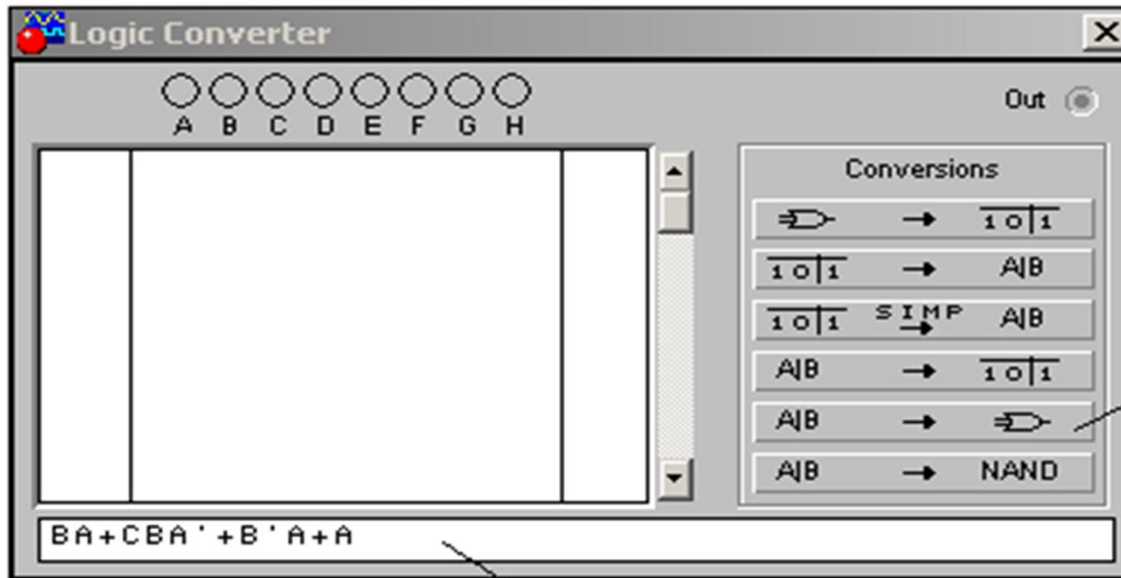
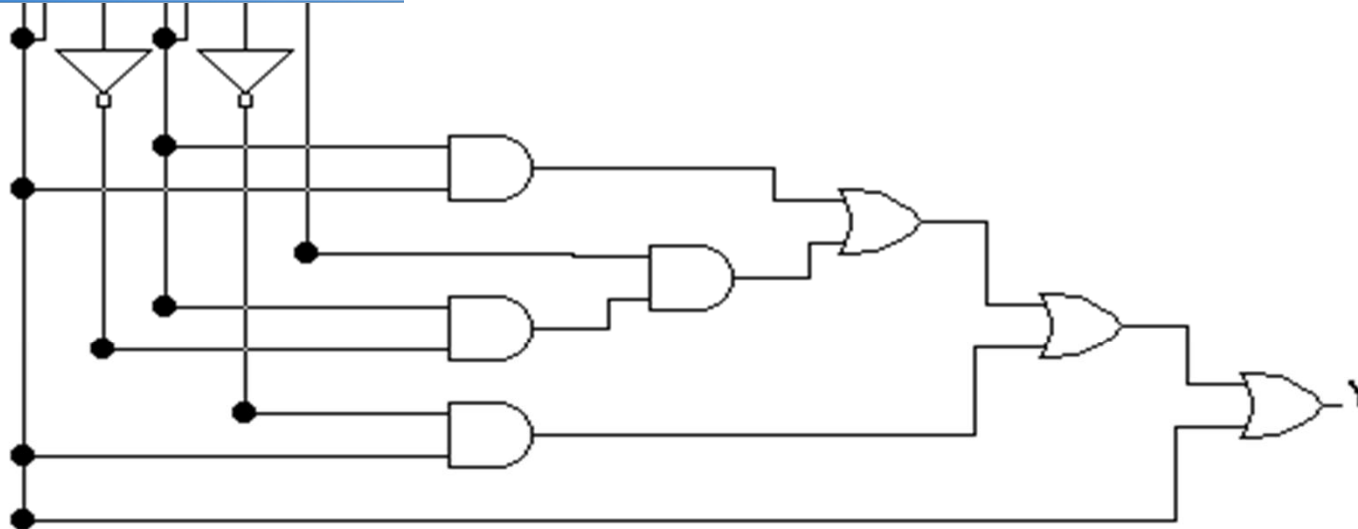
The screenshot shows a window titled "Logic Converter" with a blue title bar. At the top, there are eight indicator lights labeled A through H, with A, B, and C being lit. Below the lights is a truth table with columns for binary inputs (000 to 007) and a column for the output (0 to 1). The truth table is filled with 1s for rows 003, 004, 005, 006, and 007. To the right of the truth table is a "Conversions" panel with several buttons: \Rightarrow , \rightarrow , $\overline{101}$, $\overline{101} \rightarrow A|B$, $\overline{101} \xrightarrow{SIMP} A|B$, $A|B \rightarrow \overline{101}$, $A|B \rightarrow \Rightarrow$, and $A|B \rightarrow NAND$. At the bottom of the window, a text box displays the simplified Boolean expression "BC+A".

	A	B	C	D	E	F	G	H	Out
000	0	0	0						0
001	0	0	1						0
002	0	1	0						0
003	0	1	1						1
004	1	0	0						1
005	1	0	1						1
006	1	1	0						1
007	1	1	1						1

Paso 2: presionar este boton

Paso 3: resultado de expresion booleana

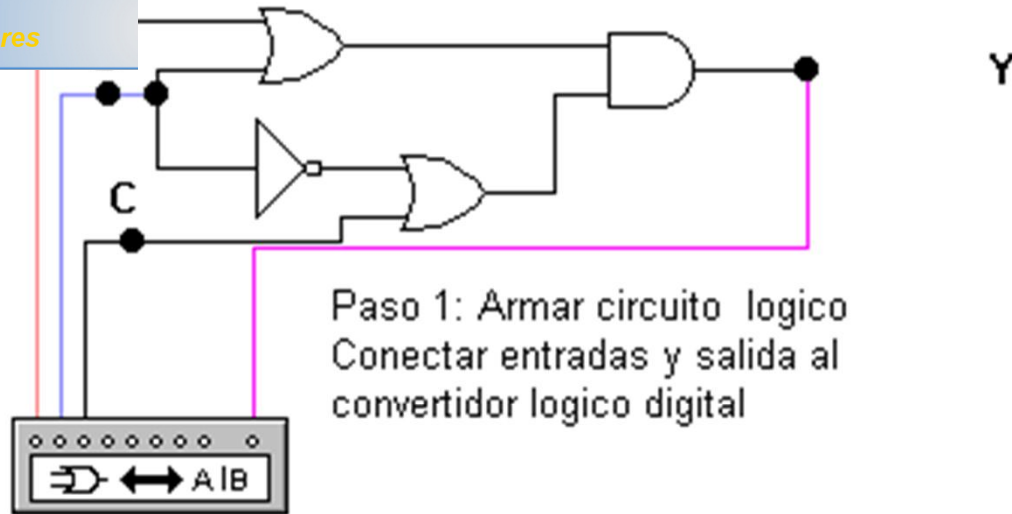
Convertir una expresión booleana a un circuito



Paso 2: Presionar este boton

Paso 1: Introducir Expresion booleana

Convertir un circuito lógico a tabla de verdad



Logic Converter

Out

	A	B	C	D	E	F	G	H
000	0	0	0					0
001	0	0	1					0
002	0	1	0					1
003	0	1	1					1
004	1	0	0					1
005	1	0	1					1
006	1	1	0					0
007	1	1	1					1

Conversions

- \Rightarrow \rightarrow 1 0 | 1
- 1 0 | 1 \rightarrow A|B
- 1 0 | 1 $\xrightarrow{\text{SIMP}}$ A|B
- A|B \rightarrow 1 0 | 1
- A|B \rightarrow \Rightarrow
- A|B \rightarrow NAND

Paso 2: activar boton

lice un circuito que indique el número de unos (1)
que hay en una palabra de 4 bits

A	B	C	D	X	Y	Z
0	0	0	0	0	0	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	0	0	1
0	1	0	1	0	1	0
0	1	1	0	0	1	0
0	1	1	1	0	1	1
1	0	0	0	0	0	1
1	0	0	1	0	1	0
1	0	1	0	0	1	0
1	0	1	1	0	1	1
1	1	0	0	0	1	0
1	1	0	1	0	1	1
1	1	1	0	0	1	1
1	1	1	1	1	0	0

de la variable X

Logic Converter

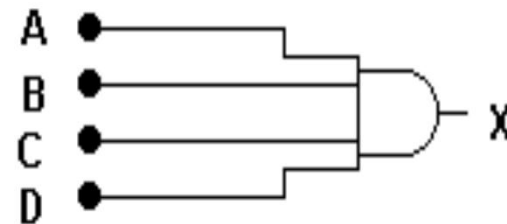
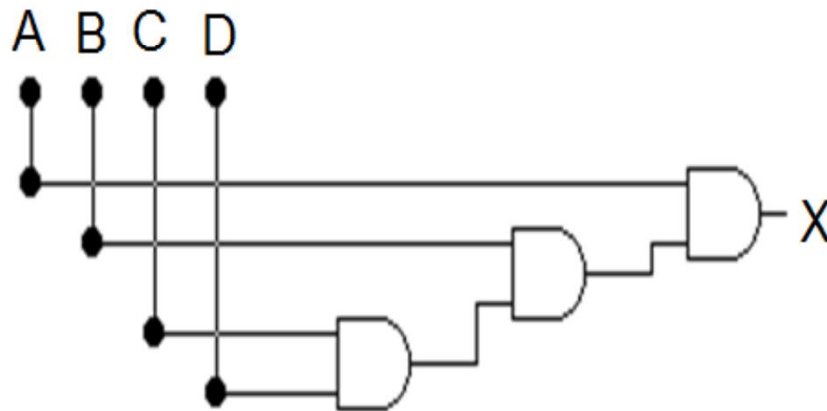
Out

	A	B	C	D	E	F	G	H
000	0	0	0	0				0
001	0	0	0	1				0
002	0	0	1	0				0
003	0	0	1	1				0
004	0	1	0	0				0
005	0	1	0	1				0
006	0	1	1	0				0
007	0	1	1	1				0
008	1	0	0	0				0
009	1	0	0	1				0
010	1	0	1	0				0
011	1	0	1	1				0
012	1	1	0	0				0
013	1	1	0	1				0
014	1	1	1	0				0
015	1	1	1	1				1

Conversions

- \Rightarrow \rightarrow $\overline{101}$
- $\overline{101} \rightarrow A\overline{B}$
- $\overline{101} \xrightarrow{\text{SIMP}} A\overline{B}$
- $A\overline{B} \rightarrow \overline{101}$
- $A\overline{B} \rightarrow \Rightarrow$
- $A\overline{B} \rightarrow \text{NAND}$

ABCD



de la variable Y

The screenshot shows a 'Logic Converter' window. At the top, there are eight indicator lights labeled A through H. Below them is a truth table with 16 rows (000 to 015) and four columns (A, B, C, D). The output column 'Out' shows the result of the function for each input combination. To the right of the truth table is a 'Conversions' panel with several buttons for converting between logic symbols and expressions. At the bottom of the window, a text box displays the simplified logic expression: $A'CD + AB'C + AB'D + ABC' + BC'D + BCD'$.

	A	B	C	D	Out
000	0	0	0	0	0
001	0	0	0	1	0
002	0	0	1	0	0
003	0	0	1	1	1
004	0	1	0	0	0
005	0	1	0	1	1
006	0	1	1	0	1
007	0	1	1	1	1
008	1	0	0	0	0
009	1	0	0	1	1
010	1	0	1	0	1
011	1	0	1	1	1
012	1	1	0	0	1
013	1	1	0	1	1
014	1	1	1	0	1
015	1	1	1	1	0

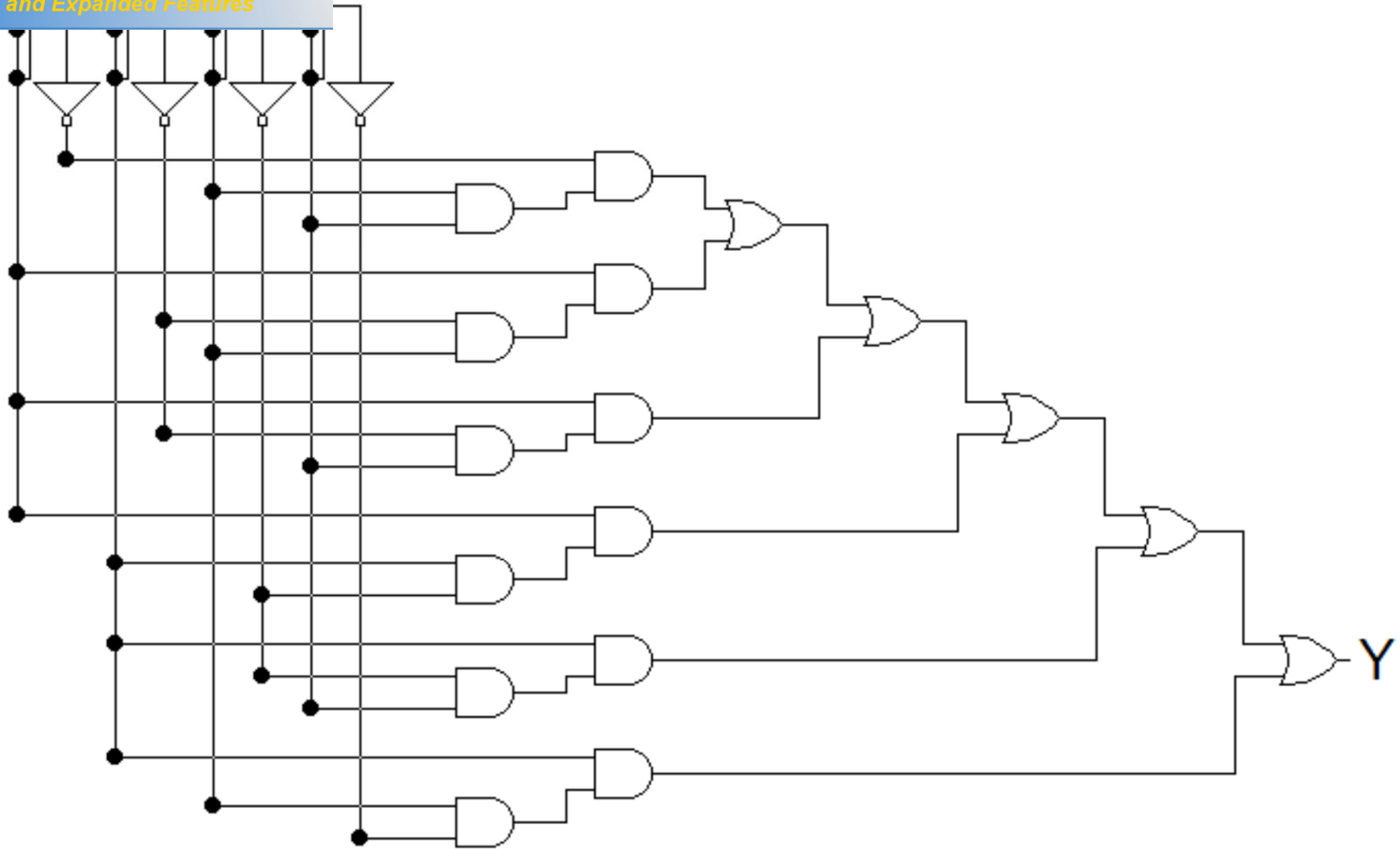
Conversions:

- \Rightarrow → $\overline{10|1}$
- $\overline{10|1}$ → $A|B$
- $\overline{10|1}$ $\xrightarrow{\text{SIMP}}$ $A|B$
- $A|B$ → $\overline{10|1}$
- $A|B$ → \Rightarrow
- $A|B$ → NAND

Logic Expression: $A'CD + AB'C + AB'D + ABC' + BC'D + BCD'$

La función es:

$$Y = A'CD + AB'C + AB'D + ABC' + BC'D + BCD'$$



de la variable Z

The screenshot shows a 'Logic Converter' window with a truth table for variables A, B, C, D, E, F, G, and H. The output column shows the function Z. Below the table is a Boolean expression: $A'B'C'D + A'B'CD' + A'BC'D' + A'BCD + AB'C'D' + AB'CD + ABC'D + ABCD$.

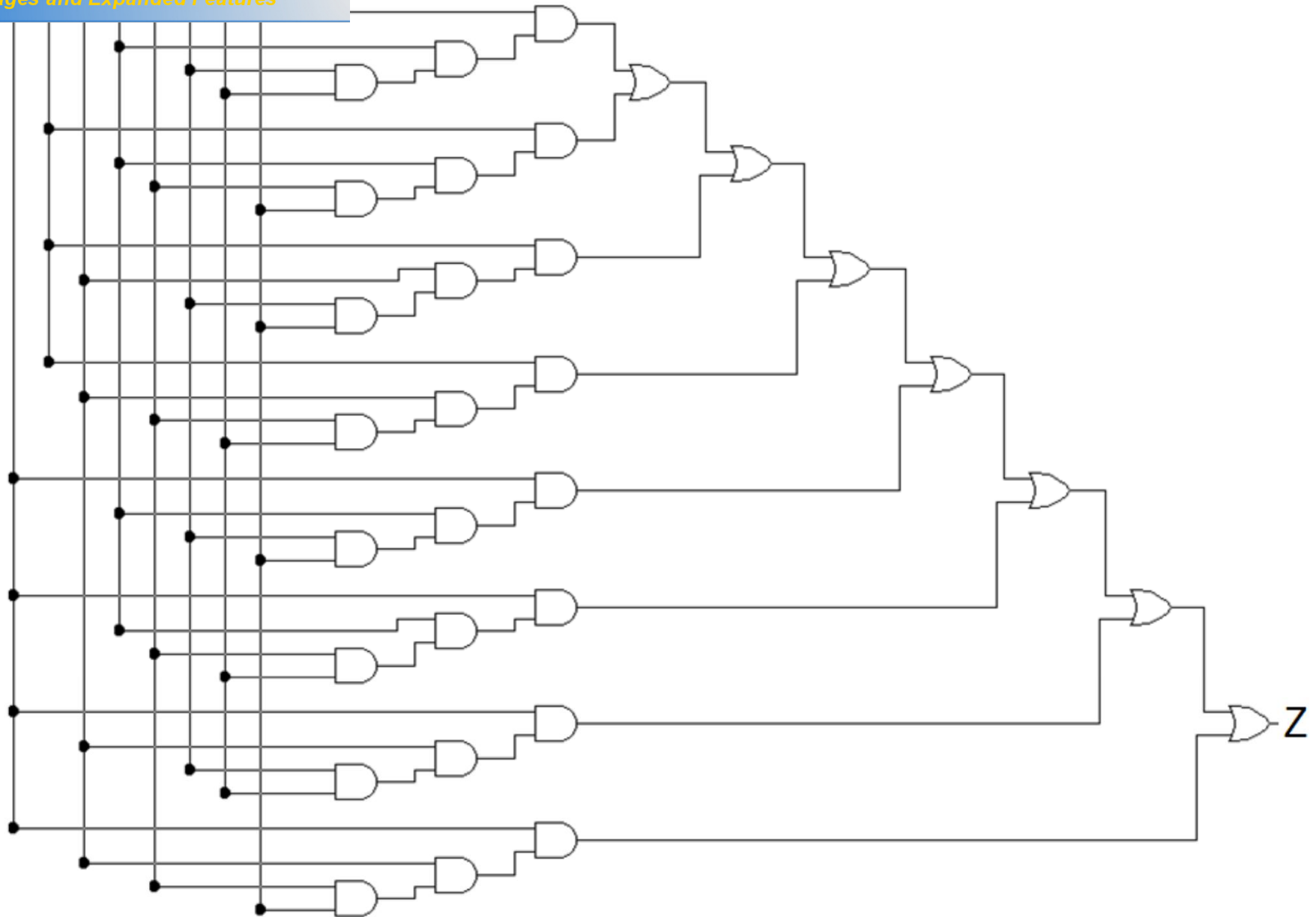
	A	B	C	D	Out
000	0	0	0	0	0
001	0	0	0	1	1
002	0	0	1	0	1
003	0	0	1	1	0
004	0	1	0	0	1
005	0	1	0	1	0
006	0	1	1	0	0
007	0	1	1	1	1
008	1	0	0	0	1
009	1	0	0	1	0
010	1	0	1	0	0
011	1	0	1	1	1
012	1	1	0	0	0
013	1	1	0	1	1
014	1	1	1	0	1
015	1	1	1	1	0

Conversions:

- \Rightarrow $\rightarrow \overline{101}$
- $\overline{101} \rightarrow A/B$
- $\overline{101} \xrightarrow{SIMP} A/B$
- $A/B \rightarrow \overline{101}$
- $A/B \rightarrow \Rightarrow$
- $A/B \rightarrow NAND$

La función es:

$$Z = A'B'C'D + A'B'CD' + A'BC'D' + A'BCD + AB'C'D' + AB'CD + ABC'D + ABCD$$



Conclusiones Recomendaciones

1. Determinamos las compuertas lógicas digitales, básicas y secundarias
2. Construimos el circuito a partir de sus expresiones booleanas y obtenemos su tabla de verdad
3. Para la conversión de simulación usando el convertidor lógico, podemos realizarlo en tres formas:
 - a) De la expresión booleana a tabla de verdad, y circuito lógico
 - b) De tabla de verdad a expresión booleana, y circuito lógico
 - c) De circuito lógico a tabla de verdad, y expresión booleana
4. Podemos diseñar cualquier circuito lógico. Para la resolución de problemas utilizando el convertidor lógico digital.
5. El convertidor lógico, nos permite simular un circuito lógico combinacional, para después implementarlo, obtener resultados reales
6. Es necesario la aplicación de este software para la resolución de problemas prácticos
7. Los estudiantes con este sistema les ayuda al aprendizaje de su curso de Electrónica Digital, Sistemas Digitales.

BIBLIOGRAFÍAS

Autor: Antonio García Guerra. Ed: Centro de estudios

Ramón Areces

- Electrónica Digital Principios y Aplicaciones Roger Tokheim 7Ed.
- Electrónica Digital Bignell . Donovan 3Ed.
- Análisis y Diseño de circuitos lógicos digitales Nelson Nagle 3Ed
- Sistemas Digitales Principios y aplicaciones Ronald Tocci 6Ed.
- Electrónica Digital, Lógica Digital Integrada Teoría y Problemas y Simulación 2º Ed. Acha, Santiago
- Electrónica Digital en la Práctica, Reina Acedo, Rafael García, Vásquez Martínez



*Your complimentary
use period has ended.
Thank you for using
PDF Complete.*

[Click Here to upgrade to
Unlimited Pages and Expanded Features](#)

REFERENCIAS

www.abcdatos.com/tutoriales/tutorial/z6410.html

<http://www.electronicworkbench.com>

<http://www.weblibrosgratis.com/2010/04/problemaseselectronicadigital>

www.forosdeelectrónica.com/tutoriales/historia.htm

www.es.wikipedia.org/wiki/electronica#aplicaciones_de_la_electronica

www.ecured.cu/index.php/electronica_digital

www.dte.uvigo.es/logica_programable/documentos/emulador/folleto.pdf



*Your complimentary
use period has ended.
Thank you for using
PDF Complete.*

[Click Here to upgrade to
Unlimited Pages and Expanded Features](#)

GRACIAS

cguerra@uns.edu.pe