

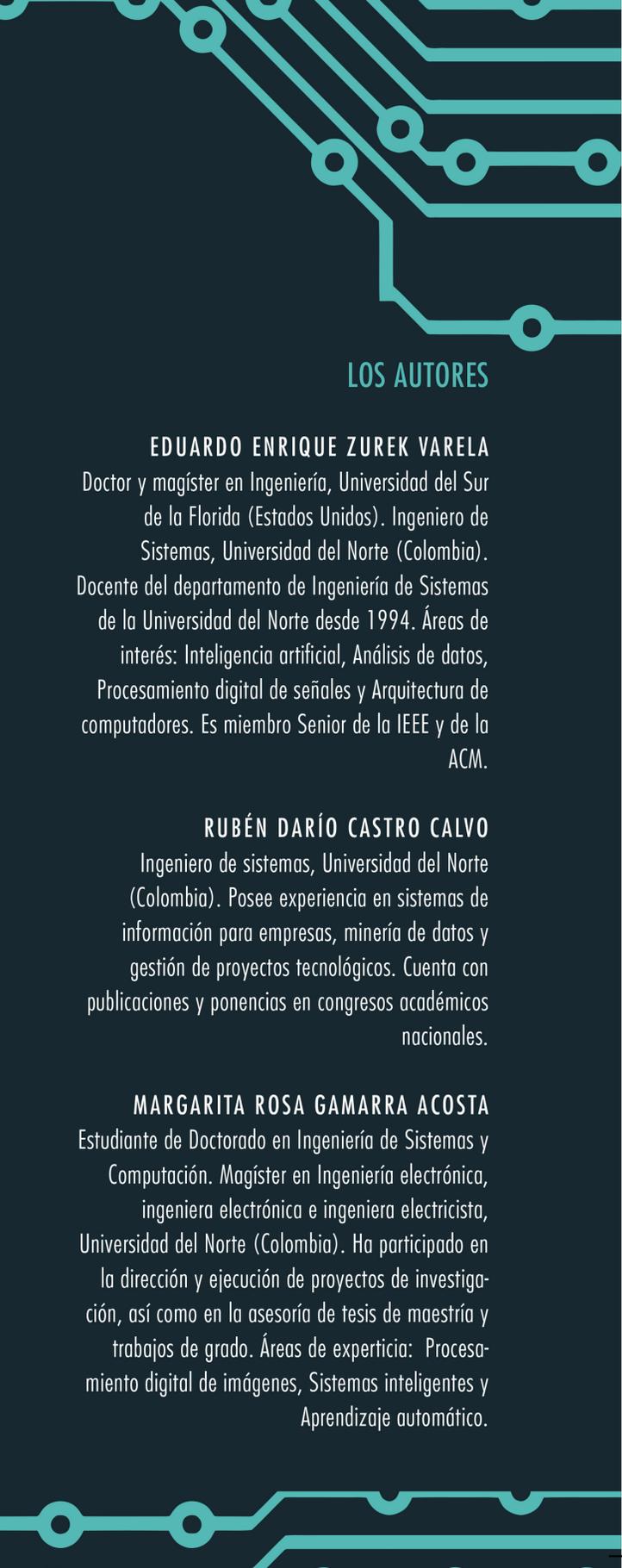


CONMUTACIÓN DISEÑO DIGITAL

EDUARDO ENRIQUE ZUREK VARELA, RUBÉN DARÍO CASTRO CALVO
Y MARGARITA ROSA GAMARRA ACOSTA

 UNIVERSIDAD
DEL NORTE

Editorial



LOS AUTORES

EDUARDO ENRIQUE ZUREK VARELA

Doctor y magíster en Ingeniería, Universidad del Sur de la Florida (Estados Unidos). Ingeniero de Sistemas, Universidad del Norte (Colombia). Docente del departamento de Ingeniería de Sistemas de la Universidad del Norte desde 1994. Áreas de interés: Inteligencia artificial, Análisis de datos, Procesamiento digital de señales y Arquitectura de computadores. Es miembro Senior de la IEEE y de la ACM.

RUBÉN DARÍO CASTRO CALVO

Ingeniero de sistemas, Universidad del Norte (Colombia). Posee experiencia en sistemas de información para empresas, minería de datos y gestión de proyectos tecnológicos. Cuenta con publicaciones y ponencias en congresos académicos nacionales.

MARGARITA ROSA GAMARRA ACOSTA

Estudiante de Doctorado en Ingeniería de Sistemas y Computación. Magíster en Ingeniería electrónica, ingeniera electrónica e ingeniera electricista, Universidad del Norte (Colombia). Ha participado en la dirección y ejecución de proyectos de investigación, así como en la asesoría de tesis de maestría y trabajos de grado. Áreas de experticia: Procesamiento digital de imágenes, Sistemas inteligentes y Aprendizaje automático.

CONMUTACIÓN
DISEÑO DIGITAL

CONMUTACIÓN DISEÑO DIGITAL

EDUARDO ENRIQUE ZUREK VARELA
RUBÉN DARÍO CASTRO CALVO
MARGARITA ROSA GAMARRA ACOSTA

Área metropolitana
de Barranquilla (COLOMBIA), 2018

 **UNIVERSIDAD
DEL NORTE**
Editorial

Zurek Varela, Eduardo Enrique.

Comutación diseño digital / Eduardo Enrique Zurek Varela, Margarita Rosa Gamarra Acosta. Barranquilla, Colombia : Editorial Universidad del Norte, 2017.

102 p. : il. ; 24 cm.

Incluye referencias bibliográficas

ISBN 978-958-741-966-5 (impreso)

ISBN 978-958-741-967-2 (PDF)

1. Electrónica digital. 2. Diseño lógico. 3. Circuitos de conmutación. 4. Álgebra booleana. I. Gamarra Acosta, Margarita Rosa. II. Tít.

(CO-BrUNB) (621.395 Z96 ed. 23)



Vigilada Mineducación

www.uninorte.edu.co

Km 5, vía a Puerto Colombia, A.A. 1569

Área metropolitana de Barranquilla (Colombia)

© Universidad del Norte, 2018
Eduardo Enrique Zurek Varela
Rubén Darío Castro Calvo
Margarita Rosa Gamarra Acosta

Coordinación editorial
Zoila Sotomayor O.

Diseño y diagramación
Álvaro Carrillo Barraza

Diseño de portada
Joaquín Camargo Valle

Corrección de textos
Henry Stein

Impreso y hecho en Colombia
Imageprinting (Bogotá)
Printed and made in Colombia

© Reservados todos los derechos. Queda prohibida la reproducción total o parcial de esta obra, por cualquier medio reprográfico, fónico o informático así como su transmisión por cualquier medio mecánico o electrónico, fotocopias, microfilm, *offset*, mimeográfico u otros sin autorización previa y escrita de los titulares del copyright. La violación de dichos derechos constituye un delito contra la propiedad intelectual.



CONTENIDO

PREFACIO	1
Capítulo 1	
INTRODUCCIÓN	3
1. SISTEMAS DE NUMERACIÓN Y CAMBIOS DE BASE	3
2. CODIFICACIÓN	6
3. NÚMERO DECIMAL CODIFICADO EN BINARIO (BCD)	7
4. CÓDIGOS DE GRAY	7
5. SUMAS Y RESTAS EN BASE DIEZ Y EN BASE DOS	9
6. EJERCICIOS DEL CAPÍTULO	12
Capítulo 2	
COMPUERTAS LÓGICAS BÁSICAS Y ÁLGEBRA BOOLEANA	15
1. COMPUERTAS BÁSICAS	15
2. ÁLGEBRA BOOLEANA	16
3. REGLAS DEL ÁLGEBRA BOOLEANA	17
4. EL DUAL DE UNA ECUACIÓN BOOLEANA	21
5. MAPAS DE KARNAUGH	22
6. IGUALDAD ENTRE SUMA DE PRODUCTOS Y PRODUCTO DE SUMAS	28
7. MAPA DE KARNAUGH DE CINCO VARIABLES	29
8. MAPAS DE KARNAUGH CON INDIFERENCIA	40
9. EJEMPLOS RESUELTOS	41
10. EJERCICIOS DEL CAPÍTULO	47

Capítulo 3

CIRCUITOS LÓGICOS COMBINACIONALES	51
1. DISPLAY DE SIETE SEGMENTOS	51
2. COMPARADOR DE DOS NÚMEROS	54
3. SEMISUMADOR Y SUMADOR COMPLETO	54
4. DECODIFICADORES	59
5. CODIFICADORES	60
6. CODIFICADOR CON PRIORIDAD	61
7. MULTIPLEXORES	62
8. DEMULTIPLEXORES	64
9. EJERCICIOS DEL CAPÍTULO	65

Capítulo 4

CIRCUITOS LÓGICOS SECUENCIALES	67
1. PRELIMINARES	67
2. LATCHES Y FLIP FLOPS	70
3. APLICACIONES BÁSICAS DE LOS LATCHES Y LOS FLIP FLOPS	79
4. TABLAS DE EXCITACIÓN	82
5. EJERCICIOS DEL CAPÍTULO	91

Apéndice 1

ANÁLISIS COMBINACIONAL	93
-----------------------------------------	-----------

Apéndice 2

TRANSISTORES	101
REFERENCIA	104



PREFACIO

El diseño digital es un área que se encarga del diseño de circuitos electrónicos digitales. Estos son utilizados en el diseño y construcción de sistemas como codificadores digitales, transmisión y almacenamiento de datos, grabación digital y muchas otras aplicaciones que se fundamentan en hardware digital.

Este texto presenta algunas herramientas básicas para el diseño de circuitos digitales y proporciona conceptos fundamentales necesarios en el diseño de sistemas digitales. Puede ser un texto guía para un curso introductorio de diseño digital en pregrados de ingeniería; de hecho, es el resultado de un curso introductorio de diseño digital desarrollado en el programa de pregrado de Ingeniería de Sistemas de la Universidad del Norte.

En este curso se tratarán de manera general los siguientes temas:

- Compuertas lógicas básicas
- Sumas y restas
- Codificación
- Álgebra booleana
- Mapas de Karnaugh
- Circuitos lógicos combinacionales
- Latches
- Flip Flops (SR, JK, D, T)

Una herramienta que será de gran utilidad para este curso es el software libre denominado Logisim, diseñado por Carl Burch, disponible en el siguiente enlace:

<http://www.cburch.com/logisim/download.html>

Ahí encontraremos una ventana como la siguiente y hacemos clic donde muestra la flecha:

Logisim
a graphical tool for designing and simulating logic circuits

Getting Logisim

Logisim should run on any platform supporting Java, version 5 or later.

1. Logisim requires Java 5 or later. If you do not already have it on your computer, Java is available from java.sun.com.
2. Download Logisim from Logisim's SourceForge.net page. You will three choices of which release to download.
 - A `-jar` file - runs on any platform, though not necessarily.
 - A MacOS `-tar.gz` file
 - A Windows `-exe` fileIf you use MacOS or Windows, I would recommend using the release specific to your platform.
3. To execute the program:
 - *With the generic `-jar` file:* On Windows and MacOS systems, you will likely be able to start Logisim by double-clicking the JAR file. If that doesn't work, or if you use Linux or Solaris, you can type `java -jar logisim-xx.jar` at the command line.
 - *With the MacOS X version:* Once the downloaded `-tar.gz` version is uncompressed (this will likely happen automatically), just double-click the Logisim icon to start. You may want to place the icon into the Applications folder.
 - *With the Windows version:* Just double-click the Logisim icon. You may want to create a shortcut on the desktop and/or in the Start menu to make starting Logisim easier.

If you find Logisim useful, please send me a comment!

Copyright and authorship

Copyright (c) 2005, Carl Burch.

Logisim is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

Logisim is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

El enlace redirecciona a otra página donde es necesario presionar el botón verde que se muestra en la siguiente imagen:

Home / Browse / Home & Education / Education / Logisim

Logisim
Brought to you by: carlburch

★★★★★ 83 Reviews

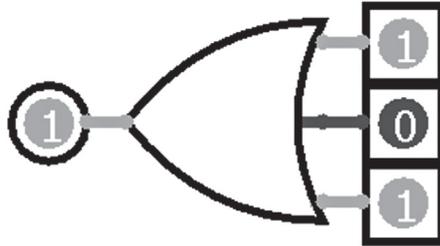
Last Update: 2013-04-29

Downloads: 5,654 This Week

Download Get Updates Share This

Windows Mac Linux

Nota: Si cuenta con un sistema operativo diferente de Microsoft Windows y no puede instalar el software, revise el enlace <http://sourceforge.net/projects/circuit/files/2.7.x/2.7.1/>



Capítulo 1

INTRODUCCIÓN

1. SISTEMAS DE NUMERACIÓN Y CAMBIOS DE BASE

El lenguaje que utilizan los computadores internamente se conoce como *sistema numérico binario*, en el cual todos los números y las operaciones se describen únicamente mediante las cifras uno (1) y cero (0), en representaciones de pulsos eléctricos que también se conocen como *pulsos de encendido y apagado*. Para entender mejor el concepto veamos el siguiente ejemplo:

Dado el arreglo



la representación en pulsos eléctricos sería la siguiente:



Fuente: elaboración propia.

Figura 1.1. Representación del arreglo dado en pulsos eléctricos

Si bien el lenguaje utilizado por los dispositivos electrónicos es el sistema binario, es importante mencionar que este no es el único que existe; lo más generalizado hoy en día en cuanto a representaciones numéricas estándar es el uso del *sistema numérico decimal*, representado por los dígitos del 0 al 9, pero hay otros sistemas de numeración tradicionales en las ciencias de la computación; los más importantes son los que se describen brevemente a continuación. Como veremos más adelante, es posible expresar un valor en diferentes sistemas de numeración (en diferentes bases).

Sistema de numeración binario (en base 2): Como ya se dijo, es utilizado para representar eventos en los que solo pueden existir dos estados finitos o infinitos numerables, es decir, discretos.

Sistema de numeración decimal (en base 10): Es un sistema utilizado en casi todo el mundo; los valores se expresan utilizando los dígitos del 0 al 9 y las potencias del número 10. Por ejemplo, 123 se puede expresar como $1*10^2 + 2*10^1 + 3*10^0$.

Sistema de numeración octal (en base 8): Es un sistema de numeración útil en ciencias de la computación debido a su flexibilidad al momento de trabajar con palabras de datos o bits (un byte) porque permite compactar información binaria de manera más fácil. Utiliza los dígitos del 0 al 7.

Sistema de numeración hexadecimal (en base 16): Al igual que el sistema en base 8, el sistema de numeración hexadecimal permite compactar aun más la información binaria con el requerimiento adicional de usar letras aparte de los dígitos del 0 al 9 del sistema en base 10; bajo esa idea, el sistema en base 16 utiliza la letra A para representar el 10 en sistema decimal, la B para el 11, la C para el 12, la D para el 13, la E para el 14 y la F para el 15. En ocasiones, los cambios de base de sistemas de numeración en base mayor a 10 pueden utilizar mayores recursos de un equipo.

Para realizar cambios de base en términos generales basta con realizar multiplicaciones sucesivas si se quiere ir a una base superior o divisiones sucesivas si lo que se pretende es convertir una cantidad en una base a otra base menor. Por ejemplo, para convertir 123 en base 10 a base 2 se realiza el proceso de las divisiones sucesivas, como podemos ver en la figura 1.2, y el resultado será