



## TECNOLÓGICO NACIONAL DE MÉXICO CAMPUS QUERÉTARO

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN  
INGENIERÍA EN SISTEMAS COMPUTACIONALES

### PRÁCTICAS DE LABORATORIO DE HARDWARE

Título de la práctica	Circuitos lógicos con compuertas	Práctica No.	4
-----------------------	----------------------------------	--------------	---

Asignatura			Matemáticas discretas			Clave de la asignatura		AEF1041	
Unidad Temática			Unida 4- Álgebra booleana						
Objetivo de la práctica			Aplicar conocimientos de álgebra booleana para simplificar, analizar y diseñar un sistema automático de luces optimizado						
Atributo de egreso	Implementa aplicaciones computacionales	Diseña, desarrolla y aplica modelos computacionales	Diseña e implementa interfaces de hardware	Coordina y participa en equipos multidisciplinarios	Diseña, implementa y Administra Bases de Datos	Desarrolla y Administra Software	Evalúa tecnologías de hardware	Detecta áreas de oportunidad para crear proyectos	Diseña, configura y administra redes de computadoras

#### Fundamentación Teórica

**Álgebra de Boole:** Leyes base que permiten optimizar expresiones para el diseño y la construcción de circuitos lógicos.



















REGLAS DE ALGEBRA DE BOOLE	
PROPIEDAD	ALGEBRA BOOLE
CONMUTATIVA	$X + Y = Y + X$ $X \cdot Y = Y \cdot X$
DISTRIBUTIVA	$X \cdot (Y + Z) = X \cdot Y + X \cdot Z$ $X + (Y \cdot Z) = (X + Y) \cdot (X + Z)$
ASOCIATIVAS	$X + (Y + Z) = (X + Y) + Z$ $X(YZ) = (XY)Z$
DOBLE COMPLEMENTO	$X'' = X$
ABSORCIÓN	$X + (X \cdot Y) = X$ $X \cdot (X + Y) = X$
PROPIEDAD	ALGEBRA BOOLE
IDEMPOTENCIA	$X + X = X$ $X \cdot X = X$
ELEMENTO NEUTRO	$X + 0 = X$ $X \cdot 1 = X$
ACOTACIÓN	$X + 1 = 1$ $X \cdot 0 = 0$
DE MORGAN	$(X + Y)' = X' \cdot Y'$ $(X \cdot Y)' = X' + Y'$
INVERSO	$X + X' = 1$ $X \cdot X' = 0$

**Minitérminos:** Expresión lógica (producto) constituida de variables y operadores lógicos (AND, NOT)

**Maxitérminos:** Expresión lógica (suma) constituida variables y operadores lógicos (OR, NOT).

**Leyes del álgebra de Boole**

#### FUNCIONES LÓGICAS BÁSICAS

NOMBRE	AND - Y	OR - O	XOR O-exclusiva	NOT Inversor	NAND	NOR																																																																																	
SÍMBOLO																																																																																							
SÍMBOLO																																																																																							
TABLA DE VERDAD	<table><tr><th>a</th><th>b</th><th>z</th></tr><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></table>	a	b	z	0	0	0	0	1	0	1	0	0	1	1	1	<table><tr><th>a</th><th>b</th><th>z</th></tr><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></table>	a	b	z	0	0	0	0	1	1	1	0	1	1	1	1	<table><tr><th>a</th><th>b</th><th>z</th></tr><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></table>	a	b	z	0	0	0	0	1	1	1	0	1	1	1	0	<table><tr><th>a</th><th>z</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	a	z	0	1	1	0	<table><tr><th>a</th><th>b</th><th>z</th></tr><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></table>	a	b	z	0	0	1	0	1	1	1	0	1	1	1	0	<table><tr><th>a</th><th>b</th><th>z</th></tr><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></table>	a	b	z	0	0	1	0	1	0	1	0	0	1	1	0
a	b	z																																																																																					
0	0	0																																																																																					
0	1	0																																																																																					
1	0	0																																																																																					
1	1	1																																																																																					
a	b	z																																																																																					
0	0	0																																																																																					
0	1	1																																																																																					
1	0	1																																																																																					
1	1	1																																																																																					
a	b	z																																																																																					
0	0	0																																																																																					
0	1	1																																																																																					
1	0	1																																																																																					
1	1	0																																																																																					
a	z																																																																																						
0	1																																																																																						
1	0																																																																																						
a	b	z																																																																																					
0	0	1																																																																																					
0	1	1																																																																																					
1	0	1																																																																																					
1	1	0																																																																																					
a	b	z																																																																																					
0	0	1																																																																																					
0	1	0																																																																																					
1	0	0																																																																																					
1	1	0																																																																																					
EQUIVALENTE EN CONTACTOS																																																																																							
AXIOMA	$z = a \cdot b$	$z = a + b$	$z = a \cdot b + \bar{a} \cdot \bar{b}$	$z = \bar{a}$	$z = \bar{a} \cdot \bar{b}$	$z = \bar{a} + \bar{b}$																																																																																	

**Diagramas de circuitos lógicos**  
Consiste en compuertas lógicas y símbolos que representan directamente una expresión en aritmética booleana.

#### Referencias

Jiménez Murillo José A. (2010). Matemáticas para la computación. Ed. Alfaomega.  
Grimaldi, Ralph P. (1998). "Matemáticas discreta y combinatoria" 3ª. edición. Ed. Pearson Educación. México.

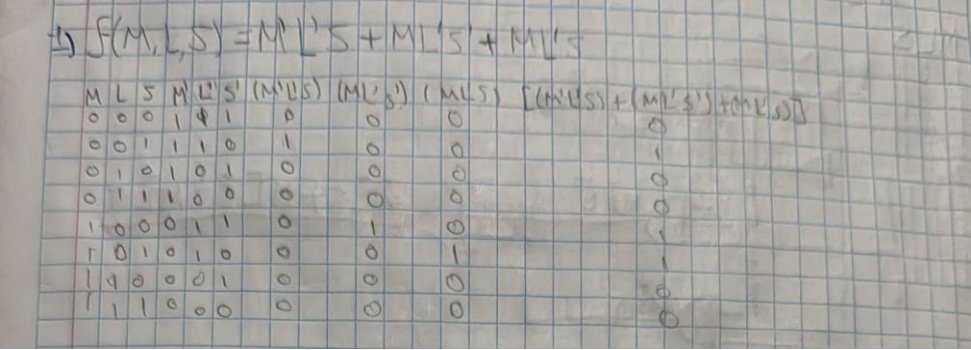




## TECNOLÓGICO NACIONAL DE MÉXICO CAMPUS QUERÉTARO

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN  
INGENIERÍA EN SISTEMAS COMPUTACIONALES

Requerimientos de Hardware y Software
Computadora, Procesador de palabras o cuaderno, hojas y bolígrafo
Descripción de la práctica
Aplicar herramientas de álgebra booleana y mapas de Karnaugh para analizar, simplificar y diseñar el funcionamiento de un sistema automático de luces mediante un circuito lógico, construyendo tablas de verdad como forma de comprobación.

Procedimiento (Instrucciones de desarrollo)
<p><b>Caso práctico:</b> Sistema de control de luces en una casa inteligente</p> <p><b>Contexto:</b> Se desea diseñar un sistema de luces automático en una casa que funcione según las siguientes condiciones:</p> <ol style="list-style-type: none"> <li>La luz debe encenderse si hay movimiento detectado (sensor de movimiento activo).</li> <li>La luz debe permanecer apagada durante el día (sensor de luz ambiental indica luz suficiente).</li> <li>La luz debe encenderse por la noche, aunque no haya movimiento, solo si el interruptor manual está activado.</li> </ol> <p><b>Implementación usando Álgebra Booleana</b></p> <p>Definimos las variables:</p> <ul style="list-style-type: none"> <li>• M: Movimiento detectado (1 = sí, 0 = no).</li> <li>• L: Luz ambiental suficiente (1 = sí, 0 = no).</li> <li>• S: Interruptor manual activado (1 = sí, 0 = no).</li> </ul> <p>La expresión booleana para el sistema sería: Salida de luz inicial sería:  <b><math>F(M,L,S) = M'L'S + ML'S' + ML'S</math></b></p> <p><b>Procedimiento por realizar:</b></p> <ol style="list-style-type: none"> <li>Elabora la tabla de verdad de la función inicial</li> </ol>  <ol style="list-style-type: none"> <li>Utiliza los métodos de Álgebra de Boole y Mapas de Karnaugh para simplificar la expresión booleana anterior y llegar a la siguiente expresión optimizada <b><math>M L' + L' S</math></b></li> </ol> <p>Donde:</p> <p><b><math>M L'</math></b>: La luz se enciende si hay movimiento y no hay suficiente luz ambiental.</p>





**TECNOLÓGICO NACIONAL DE MÉXICO  
CAMPUS QUERÉTARO**

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN  
INGENIERÍA EN SISTEMAS COMPUTACIONALES

L' S: La luz se enciende si no hay suficiente luz ambiental y el interruptor está activado

2)  $M'L'S + M'L'S' + M'L'S$   
 $M'L'S + M(L'S + L'S')$   
 $M'L'S + M(L'(S+S'))$   
 $M'L'S + M(L')$   
 $M'L'S + ML$   
 $L'(M'S + M)$   
 $L'(M + M')(M + S)$   
 $L'(1 + S)$   
 $L'(M + S)$   
 $ML + SL'$

3. Realiza la tabla de verdad de la expresión optimizada  $M L' + L' S$

3)

M	L	S	L'	(ML')	(SL')	[ML' + SL']
0	0	0	1	0	0	0
0	0	1	1	0	1	1
0	1	0	0	0	0	0
0	1	1	0	0	0	0
1	0	0	1	1	0	1
1	0	1	1	1	1	1
1	1	0	0	0	0	0
1	1	1	0	0	0	0

4. Compara las tablas e indica si son equivalentes en la salida de la función, es decir en la última columna de su tabla de verdad y responde las siguientes preguntas:

a) ¿Las tablas de verdad de ambas expresiones son equivalentes?

R= Si ya que al final en el ultima columna terminan de la misma manera está siendo una contingencia

b) Justifica tu respuesta mediante un análisis de un párrafo mínimo

R= Revisando ambas tablas se puede ver que en la ultima columna son iguales teniendo 0 en la fila 1,3,4,7 y 8, y teniendo 1 en las filas 2,5 y 7. Esto es posible ya que en si es la misma expresión, pero una más reducida que la otra, pero como es la misma expresión entonces debe de dar el mismo resultado haciendo así equivalentes

5. Diseña los circuitos lógicos de ambas expresiones

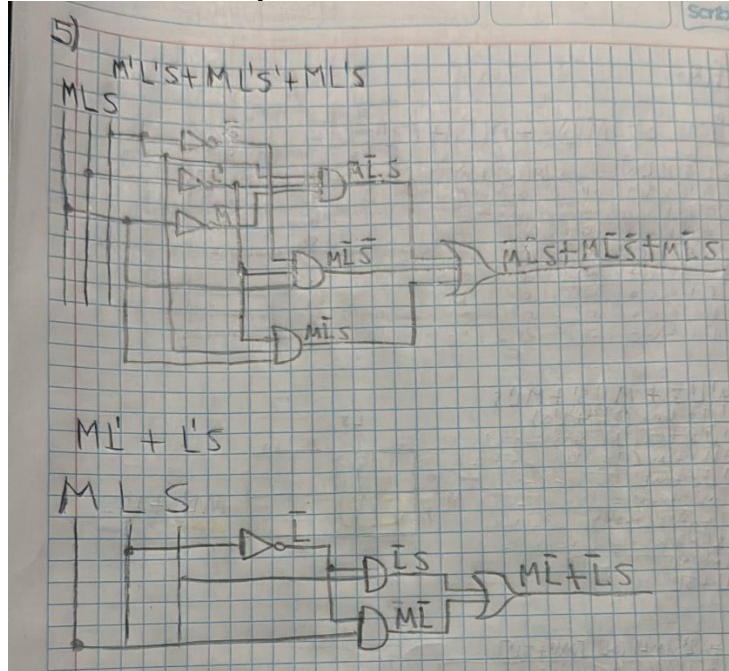
Usamos las siguientes compuertas lógicas: NOT para invertir el valor de las variables, AND para realizar la operación de multiplicación entre variables y OR para sumar los términos de productos



## TECNOLÓGICO NACIONAL DE MÉXICO CAMPUS QUERÉTARO

DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN  
INGENIERÍA EN SISTEMAS COMPUTACIONALES

combinar las salidas y obtener la salida final en el circuito.



6. Analiza la cantidad de recursos (compuertas lógicas) que se utilizan en la generación de cada circuito lógico e incorpora una conclusión de la actividad.

R= En el primer circuito se utilizaron 7 compuertas y para el segundo circuito fueron 4 compuertas. En conclusión, el reducir expresiones es muy importante por que ayuda en ahorrar dinero al momento de fabricar todos estos circuitos eso si siempre y cuando se pueda por que si no se puede y lo intentamos puede llegar a ser peor.

Evidencias a entregar

Reporte de la practica en archivo PDF

### Cuestionario

Responde las siguientes preguntas, justifica cada una de tus respuestas

1.- ¿Consideras que es importante simplificar las funciones proposicionales para optimizar recursos? Si por que como todo en lo físico siempre los recursos son finitos y no son infinitos entonces es bueno ahorrar para tener aun recursos disponibles además de que te ahorras dinero al momento de fabricación.

2.- ¿Cuál es la diferencia entre una compuerta OR y una AND?

R= OR se usa para cuando quieres decir que la proposición va a ser verdadera cuando cualquiera de las dos sea verdadero o ambas además su figura es como un triángulo con los vértices de la base un poco atrás para que parezca una flecha. El AND se usa para decir que ambas tienen que ser verdaderas para que sea todo verdadero y de figura es un círculo a la mitad.

3.- ¿Busca una herramienta en línea para diseñar circuitos lógicos y analiza si te facilita la generación y comprensión de los diagramas de circuitos lógicos con sus respectivas compuertas?

R= No busque una herramienta por que en libreta se me hace más cómodo y rápido por que al solo tener que dibujas es rápido y no tengo que estar prendiendo mi dispositivo, buscar en una pagina o iniciar el programa y ya ahí recién poder empezar.

Conclusiones







**TECNOLÓGICO NACIONAL DE MÉXICO  
CAMPUS QUERÉTARO**

**DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN  
INGENIERÍA EN SISTEMAS COMPUTACIONALES**

Es un tema muy interesante y que te pone a pensar el cómo reducir expresiones para el ahorro de recursos y también al tener que hacer menos procesos se hace mas eficiente el circuito resultante. Es muy importante tener las bases muy claras y saber cada una de las reglas para no cometer errores y que se cambie la expresión final. En resumen tema muy importante para hacer mas eficiente y reducir costos en los circuitos.

Instrumento de evaluación	Lista de cotejo	Guía de observación	Rúbrica
	X		

