

Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ciencias y Sistemas

Organización Computacional

Ing. Otto René Escobar Leiva

Auxiliar Javier Gutiérrez



## PRÁCTICA #2

### LogicCalc

Nombre	Carné	Participación
Bismarck Estuardo Romero Lemus	201708880	34%
Josué Nabí Hurtarte Pinto	202202481	33%
Naomi Rashel Yos Cujcuj	202001814	33%

## Introducción

Los cálculos aritméticos, comparativos y lógicos nos permiten la construcción de una Unidad Aritmética Lógica Básica. Esta se realiza por medio de componentes electrónicos, de manera física y simulada con el fin de conocer e implementar las compuertas lógicas, además del uso de multiplexores, demultiplexores, comparadores y decodificadores.

En este proyecto se creó una LogicCalc la cual cuenta con una unidad comparativa, circuitos combinacionales lógicos y aritméticos que tienen como entrada dos números de 4 bits cada uno y mostrando su resultado en displays de 7 segmentos.

El ingreso de un número en código binario nos obliga a utilizar un decodificador para obtener su resultado BCD, dónde será capaz de realizar su respectivo cálculo. Además, de presentar varias opciones, es necesario aplicar un multiplexor o demultiplexor, según el caso, para hacer funcionar el requerimiento

## Descripción del problema

Como estudiantes de ingeniería del curso de organización computacional se nos dejó el proyecto de realizar una calculadora llamada "LogicCalc". Esta calculadora busca proporcionar soluciones óptimas mediante el uso de lógica combinacional, permitiendo llevar a cabo operaciones aritméticas, lógicas y comparativas. Entre las funciones incluidas se encuentran la suma, resta, multiplicación, potencia, comparación entre dos números, así como las operaciones lógicas AND, OR y NOT y XOR. Para lograr esto, nos enfocamos en trabajar con especificaciones particulares destinadas a una Unidad Aritmética Lógica Básica.

# Objetivos

## Objetivo General

1. Construir una Unidad Aritmética Lógica Básica (ALU).

## Objetivos Específicos

1. Aprender el funcionamiento de Multiplexores, Demultiplexores, Comparadores y Decodificadores.
2. Construir un diseño óptimo, logrando utilizar la menor cantidad de dispositivos.
3. Aprender el funcionamiento de Operaciones Lógicas, Aritméticas y Comparativas con números binarios

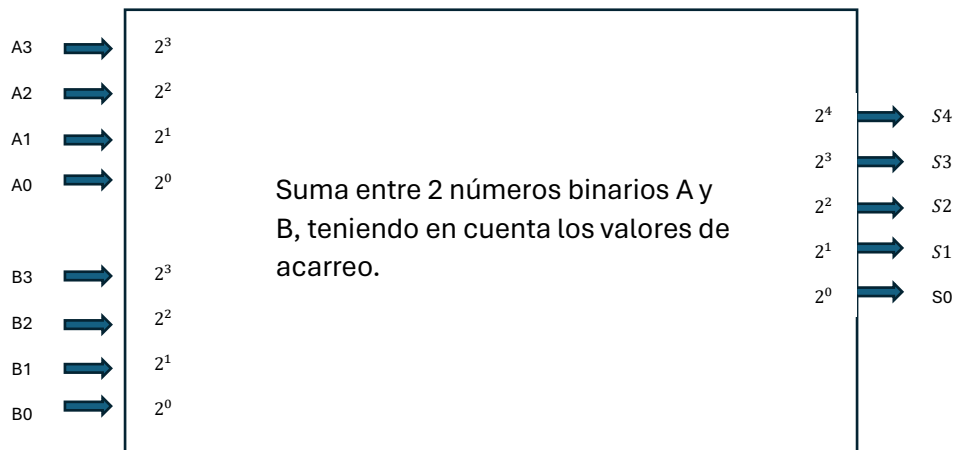
# Funciones booleanas

**Suma [0,0,0]:** Suma entre 2 números binarios A y B (Entradas), teniendo en cuenta los valores de acarreo respectivamente. Para las entradas se manejarán solo números positivos.

Entradas									Salidas						
A <sub>3</sub>	A <sub>2</sub>	A <sub>1</sub>	A <sub>0</sub>		B <sub>3</sub>	B <sub>2</sub>	B <sub>1</sub>	B <sub>0</sub>		C <sub>out</sub>	S <sub>4</sub>	S <sub>3</sub>	S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>
0	0	0	0		0	0	0	0		0	0	0	0	0	0
0	0	0	0		0	0	0	1		0	0	0	0	0	1
0	0	0	0		0	0	1	0		0	0	0	1	0	0
0	0	0	0		0	0	1	1		0	0	0	1	1	1
0	0	0	0		0	1	0	0		0	0	1	0	0	0
0	0	0	0		0	1	0	1		0	0	1	0	0	1
0	0	0	0		0	1	1	0		0	0	1	1	0	0
0	0	0	0		0	1	1	1		0	0	1	1	1	1
0	0	0	0		1	0	0	0		0	1	0	0	0	0
0	0	0	0		1	0	0	1		0	1	0	0	0	1
0	0	0	0		1	0	1	0		0	1	0	1	0	0
0	0	0	0		1	0	1	1		0	1	0	1	1	0
0	0	0	0		1	1	0	0		0	1	1	0	0	0
0	0	0	0		1	1	0	1		0	1	1	0	0	1
0	0	0	0		1	1	1	0		0	1	1	1	0	0
0	0	0	0		1	1	1	1		0	1	1	1	1	1
0	0	0	1		0	0	0	1		0	0	0	1	0	0
0	0	1	0		0	0	1	0		0	0	1	0	0	0
0	1	0	0		0	1	0	0		0	1	0	0	0	0
0	1	0	1		0	1	0	1		0	1	0	1	0	0
0	1	1	0		0	1	1	0		0	1	1	0	0	0
0	1	1	1		0	1	1	1		0	1	1	1	0	0
1	0	0	0		1	0	0	0		1	0	0	0	0	0
1	0	0	1		1	0	0	1		1	0	0	1	0	0
1	0	1	0		1	0	1	0		1	0	1	0	0	0
1	0	1	1		1	0	1	1		1	0	1	1	0	0
1	1	0	0		1	1	0	0		1	1	0	0	0	0
1	1	0	1		1	1	0	1		1	1	0	1	0	0
1	1	1	0		1	1	1	0		1	1	1	0	0	0
1	1	1	1		1	1	1	1		1	1	1	1	0	0
0	1	1	0		1	0	1	0		1	0	0	0	0	0

Como se puede observar por tabla de verdad no es practico ya que sería muy largo, como estamos trabajando con 4 variables independientes usaremos un algoritmo de base 10.

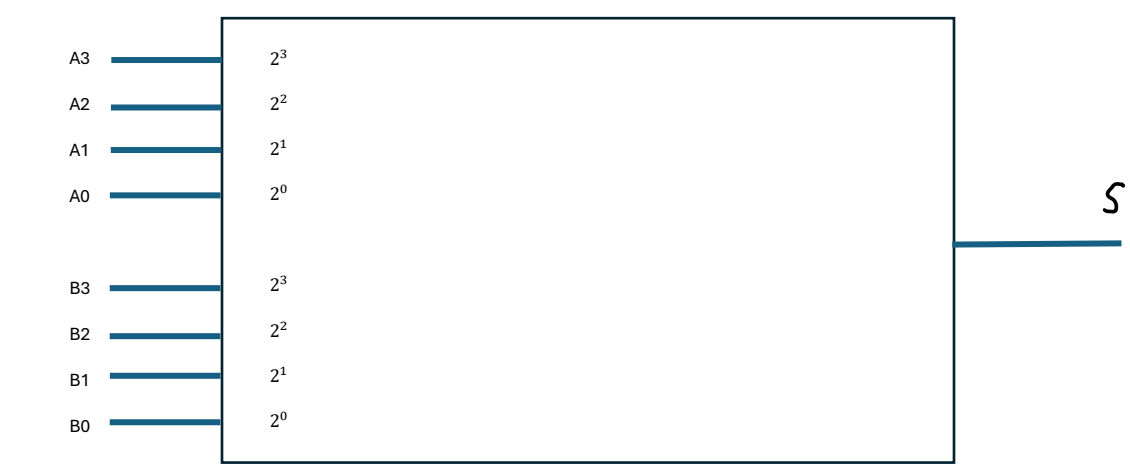
## Black Box



### Algoritmo deducido de base decimal

	$C_{in3}$	$C_{in2}$	$C_{in1}$	
	$B_3$	$B_2$	$B_1$	$B_0$
+	$A_3$	$A_2$	$A_1$	$A_0$
	$C_{in3} + A_3 + B_3$	$C_{in2} + A_2 + B_2$	$C_{in1} + A_1 + B_1$	$A_0 + B_0$
$S_4$	$S_3$	$S_2$	$S_1$	$S_0$

**Multiplicación [0,1,0]:** Para una multiplicación de dos entradas (A y B) de cuatro bits dan una salida “S”. El máximo resultado a mostrar es 99 (11 x 9)



Se utilizó el algoritmo inducido de base 10 en lugar de tablas de verdad por la cantidad de variables con la cual trabajar.

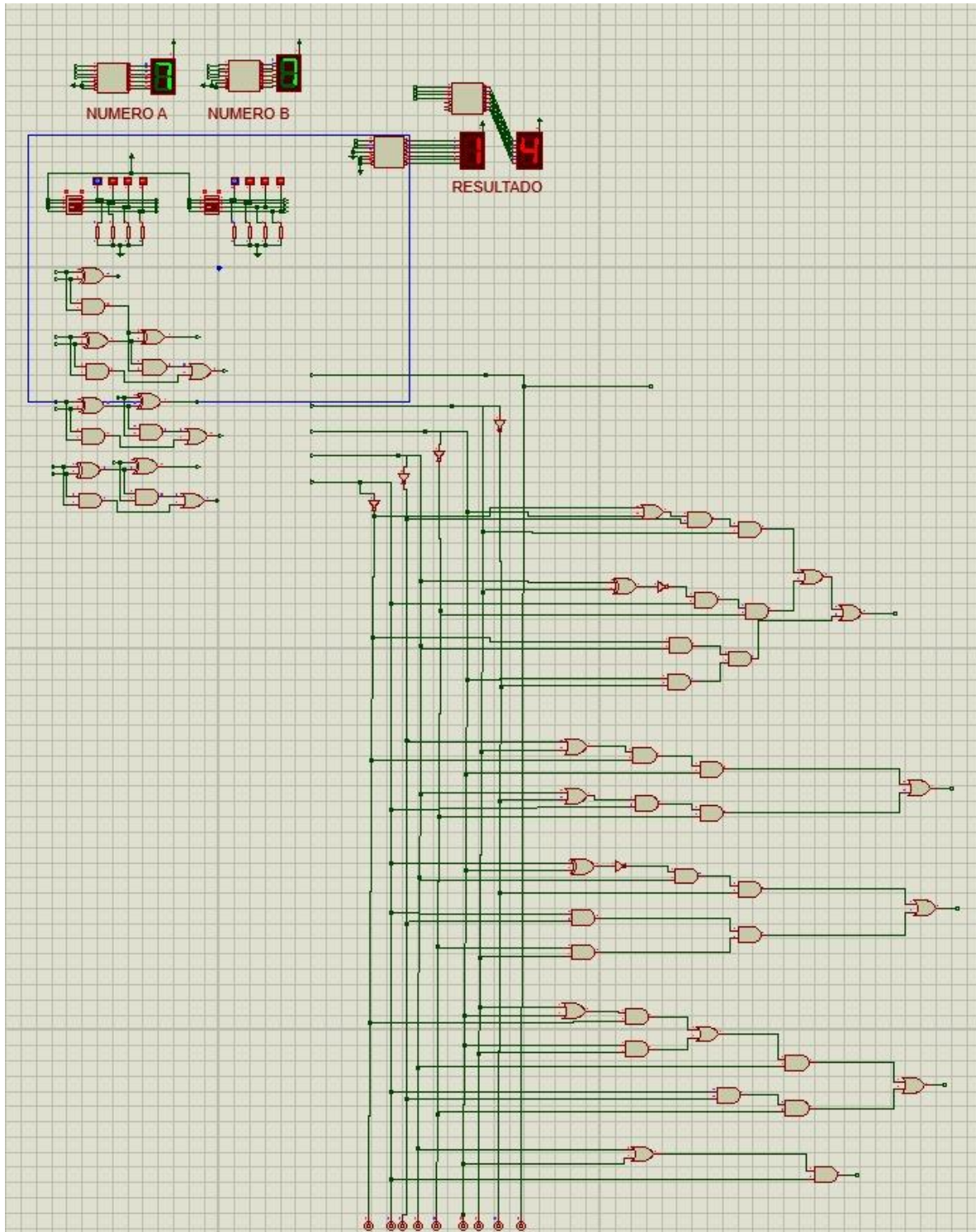
				A3	A2	A1	A0
				B3	B2	B1	B0
				B0A3	B0A2	B0A1	B0A0
			B1A3	B1A2	B1A1	B1A0	
		B2A3	B2A2	B2A1	B2A0		
	B3A3	B3A2	B3A1	B3A0			
COUT	B3A3	B2A3+B3A2	B1A3+B2A2+B3A1	B0A3+B1A2+B2A1+B3A0	B0A2+B1A1+B2A0	B0A1+B1A0	B0A0
M7	M6	M5	M4	M3	M2	M1	M0

**Potencia [0,1,1]:** Procederá a elevar al Cuadrado o al Cubo el Número binario A (Entrada), teniendo en cuenta los valores de acarreo respectivamente. Para definir a qué potencia se elevará el número, se tomará como base el valor del número B, si este es 2, el número A se elevará al cuadrado (Números del 0-9); si este el valor de B es 3, el número A se elevará al cubo (Números del 0-4).

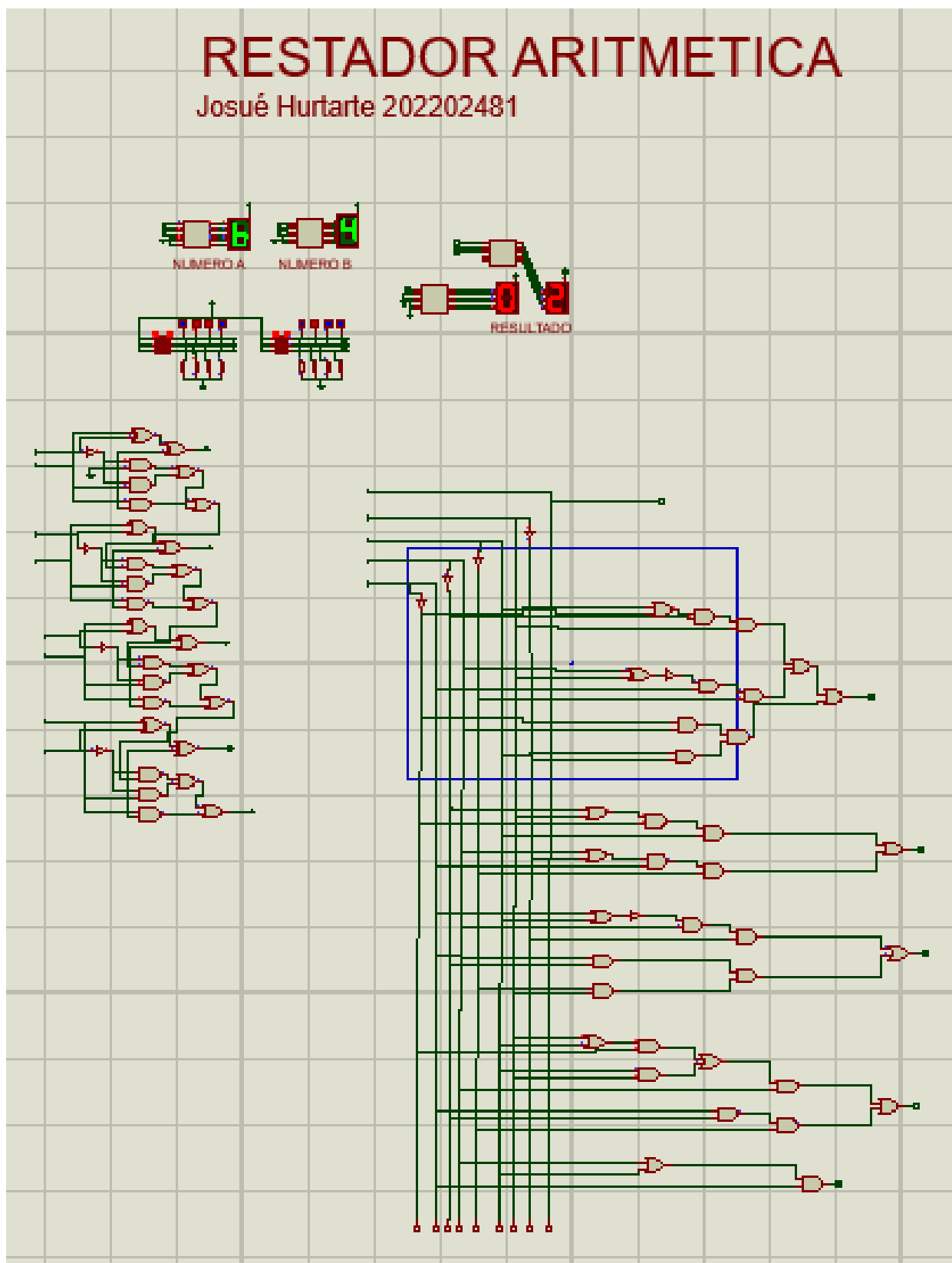
				A3	A2	A1	A0
A^B				A3	A2	A1	A0
				A0A3	A0A2	A0A1	A0A0
			A1A3	A1A2	A1A1	A1A0	
		A2A3	A2A2	A2A1	A2A0		
	A3A3	A3A2	A3A1	A3A0			
COUT	A3A3	A2A3+A3A2	A1A3+A2A2+A3A1	A0A3+A1A2+A2A1+A3A0	A0A2+A1A1+A2A0	A0A1+A1A0	A0A0
E7	E6	E5	E4	E3	E2	E1	E0

## Diagramas del diseño del circuito

SUMA [0,0,0]

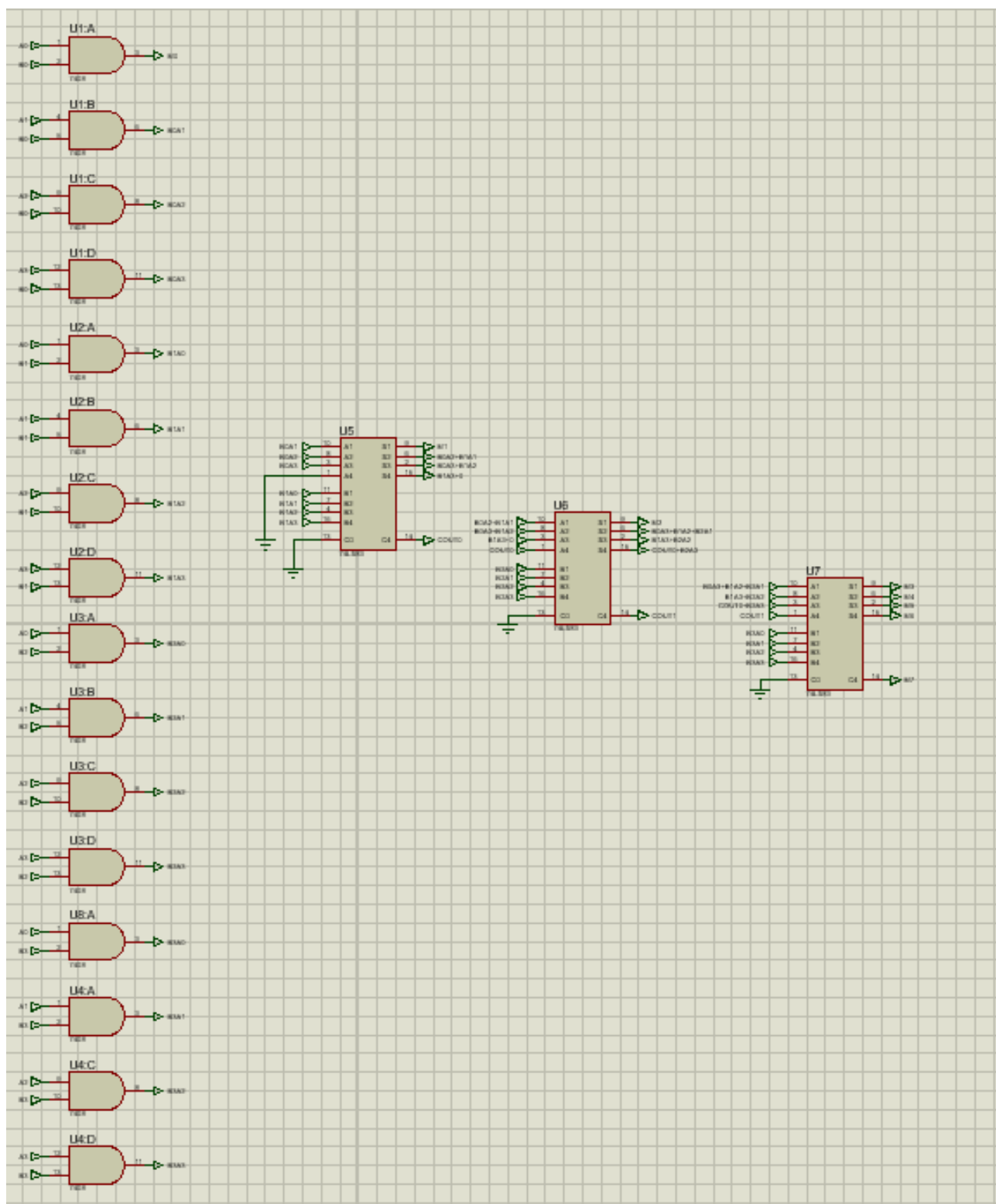


RESTA [0,0,1]

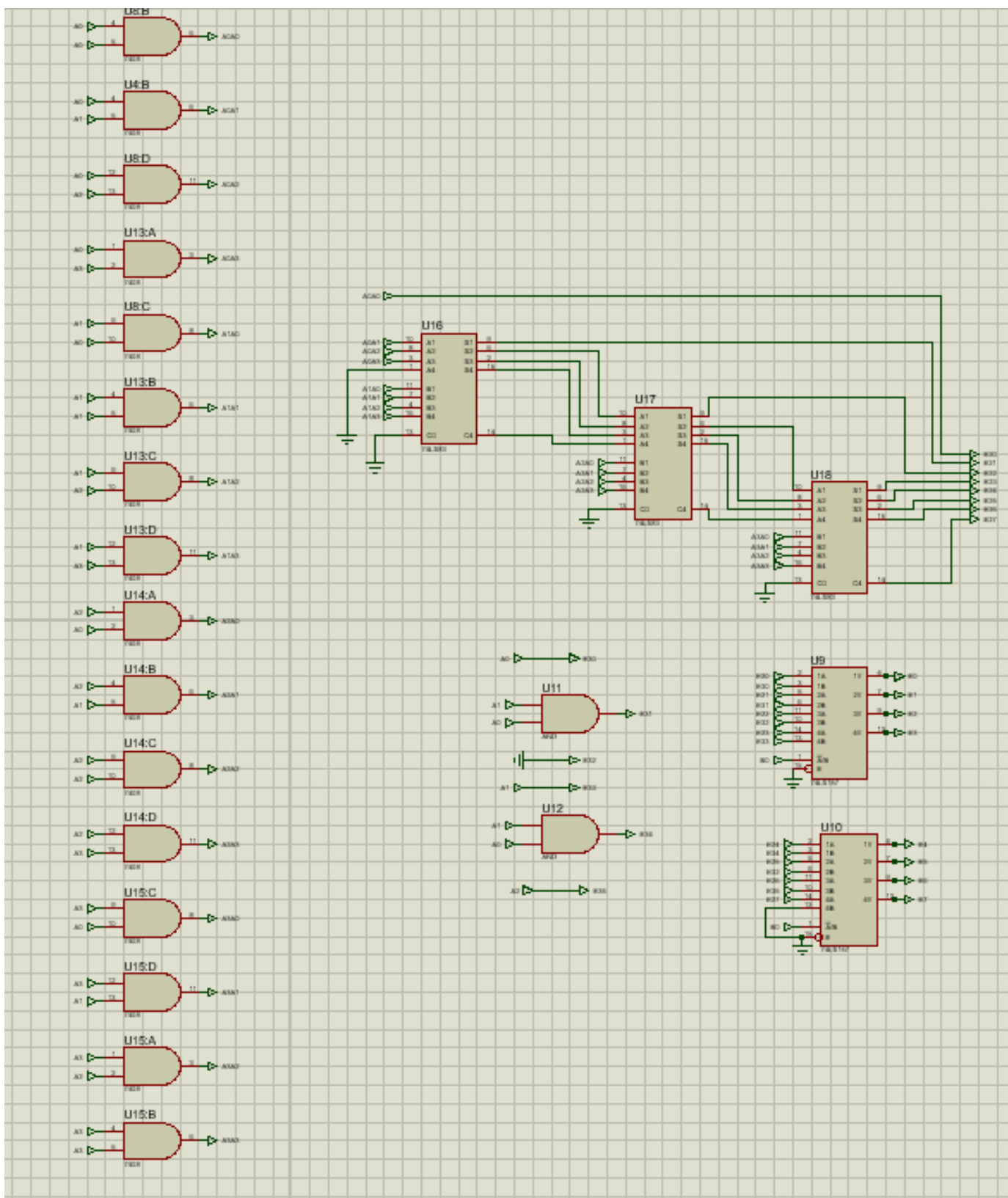




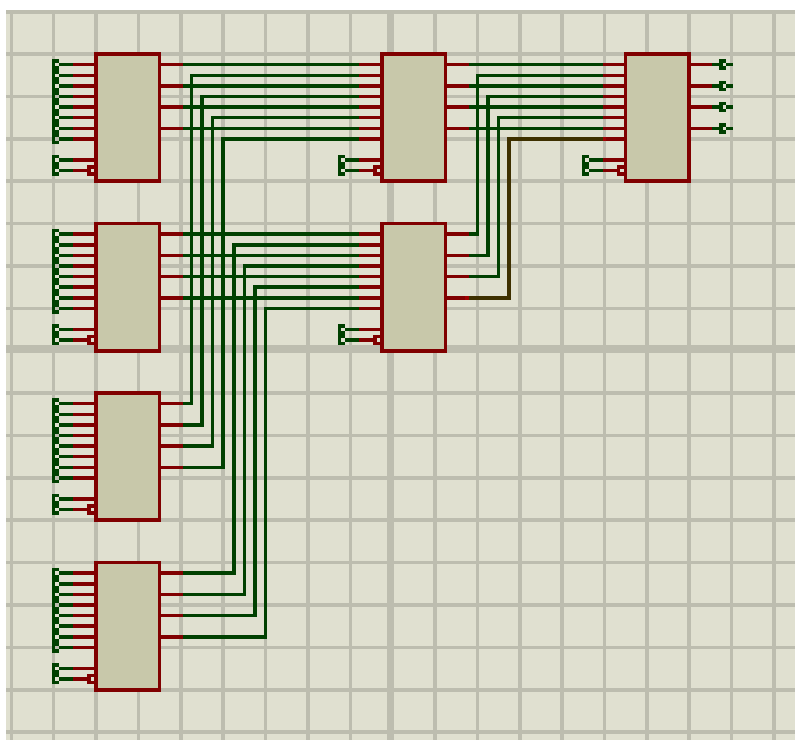
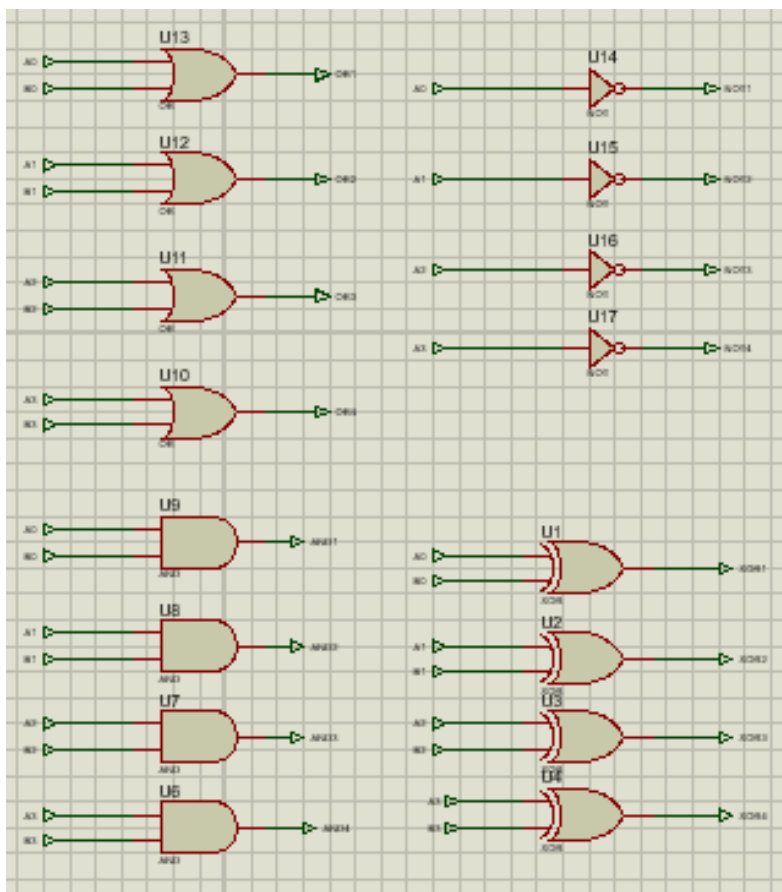
Multiplicación [0,1,0]:



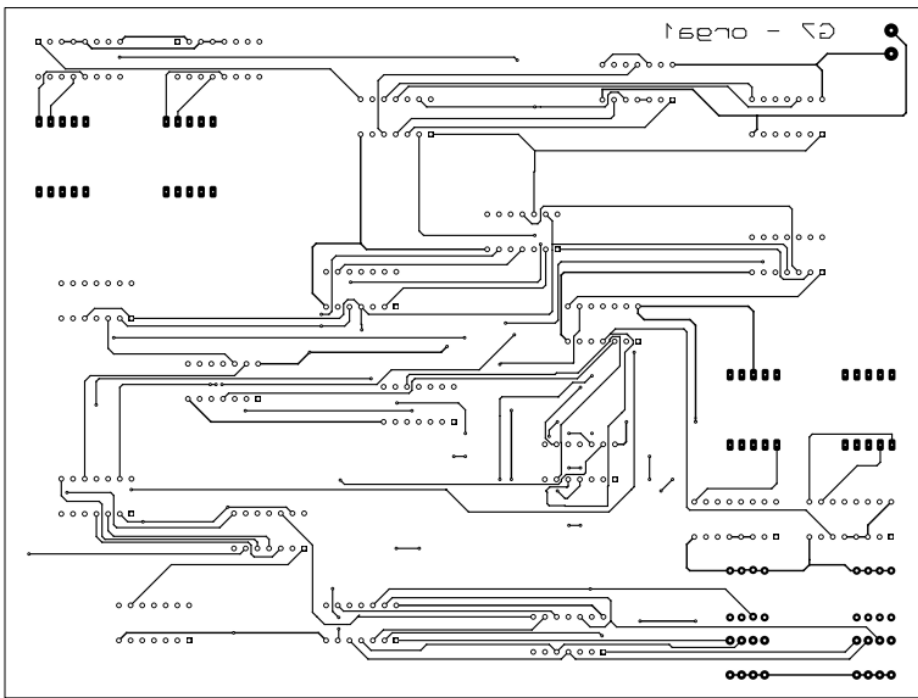
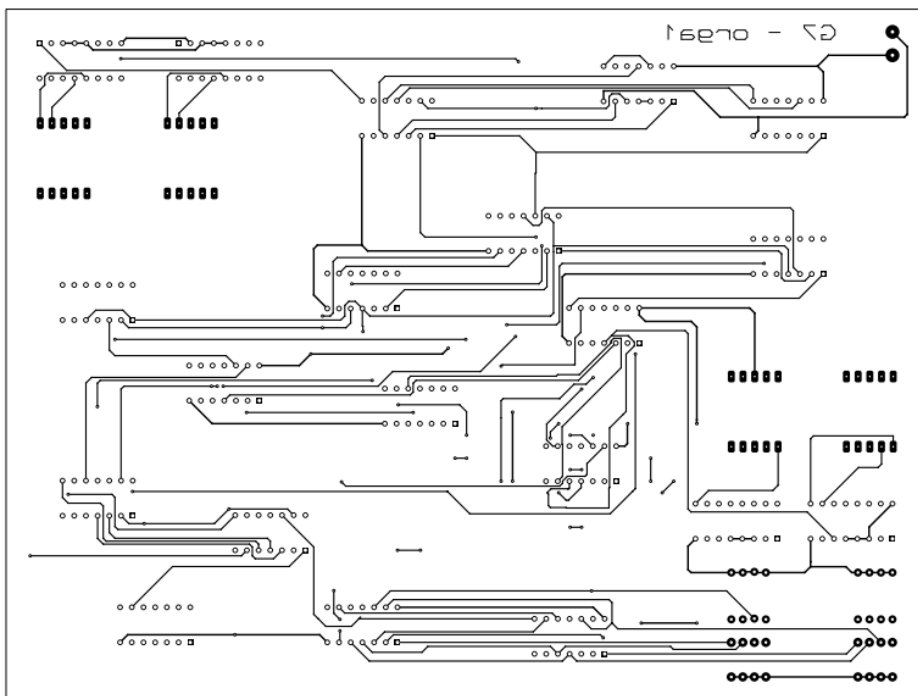
Potencia  $[0,1,1]$ :



## Bloque lógico



## Placa Sumadora



## Equipo utilizado

El equipo utilizado en la siguiente practica fue el siguiente:

EQUIPO UTILIZADO
Protoboard
Dip Switch de 4 entradas
Cable para protoboard
Compuertas OR
Compuertas AND
Compuertas NOT
Compuertas XOR
Resistencias de diferente kilo ohmios
batería de 9v
multímetro
Placa fenólica
Cloruro Férrico
Leds de diferentes colores
Brocas para PCB
Barreno
Papel termotransferible
Marcador permanente negro
Cautín
Multiplexor
Transistor npn 2n2222
Demultiplexores
Comparador
Multiplexores
Decoder

## Presupuesto

### Gastos

	Nombre	Cantidad	Precio	Precio total
1	Transistor npn 2n2222	8	1.25	10
2	Led amarillo de 3mm	5	1	5
3	Led verde de 3mm	5	1	5
4	dip switch de 8 posiciones	6	5	30
5	Compuerta not sn74ls04n	8	8	64
6	Compuerta and sn74ls08n	15	6	90
7	Compuerta or sn74ls32	8	6	48
8	Compuerta XOR 7432	7	7	49
9	Demultiplexores 74138	4	10	40
10	Multiplexores 74157	4	12	48
11	Decoder 7447	4	15	60
12	Comparador 7485	4	15	60
13	Cargador de 5v	1	0	0
14	Resistencias de 1k ohm	15	0.75	11.25
15	Placa de cobre	3	12	36
16	Brocas 1/32	2	3	6
17	Brocas 1/16	2	3	6
18	Ácido férrico 120ml	2	9	18
	TOTAL			586.25

### Aporte de cada integrante

INTEGRANTES	APOORTE (Q)
Bismarck Estuardo Romero Lemus	195
Naomi Rashel Yos Cujcuj	196.25
Josué Nabí Hurtarte Pinto	195
<b>TOTAL</b>	<b>586.25</b>

## Conclusiones

- La elaboración de unidad aritmética lógica (ALU) nos incentivó al aprendizaje de distintas operaciones con números binarios como su parte aritmética, lógica y comparativa con el propósito de aprender gracias a la lógica combinatorial que se pueden realizar este tipo de operaciones e implementarlo de manera física y ver su funcionamiento.
- Se le dio una solución al problema mediante electrónica digital y aprendizaje de diferentes usos para la lógica secuencial.
- Se logro implementar una ALU capaz de realizar operaciones como la suma, resta, multiplicación y potencia sobre números de 4 bits. Los resultados fueron los esperados, demostrando la efectividad de nuestra ALU.