

Universidad San Carlos de Guatemala

Facultad de Ingeniería

Organización Computacional, Sección A

Ing. Juan Carlos Maeda Juárez

Auxiliar: Javier Alejandro Gutiérrez de León

Vacaciones Del Primer Semestre 2024



Práctica 2

LogicCalc

No	Nombre	Carné	CUI
1.	Josseline Griselda Montecinos Hernández	202201534	3010008160101
2.	Evelyn Maricely Balcarcel Rivera	202200110	3149328281501
3.	Pablo Andrés Rodríguez Lima	202201947	3014214630101
4.	Carlos Manuel Lima Y Lima	202201524	3009269950101

Guatemala, 16 de junio del 2024.

INTRODUCCIÓN

La unidad aritmética es un componente esencial en muchos sistemas digitales, encargada de realizar operaciones básicas como suma, resta, multiplicación y exponenciación en números binarios. En este proyecto, nos enfocaremos en diseñar y construir una unidad aritmética utilizando compuertas lógicas y sumadores en un protoboard.

A través de este proceso, exploraremos conceptos fundamentales de aritmética binaria, o mejor llamada booleana, comprensión de compuertas lógicas, manejo de condiciones especiales en las operaciones. Además, se abordará la necesidad de multiplexar operaciones y visualizar los resultados. Este proyecto tiene como finalidad aplicar habilidades prácticas en diseño de circuitos digitales, así como para comprender la importancia de la precisión y la eficiencia en las operaciones aritméticas en sistemas electrónicos.

Por otro lado, en el campo de la electrónica digital, los bloques Integración a Escala Media, son esenciales para el desarrollo de circuitos complejos. Los bloques aritméticos MSI son particularmente importantes porque permiten realizar operaciones aritméticas básicas, como adiciones, sustracciones y multiplicaciones, de forma eficiente y confiable. Estos bloques digitales combinacionales MSI están diseñados para llevar a cabo cálculos matemáticos utilizando compuertas lógicas, registros y otros componentes digitales. Están preparados para recibir dos o más operandos y producir una salida que refleja el resultado de la operación.

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

DESCRIPCIÓN DEL PROBLEMA

Para satisfacer los requisitos establecidos por Intel Corporation para el desarrollo del prototipo de calculadora "LogicCalc", es necesario diseñar un circuito combinacional que pueda llevar a cabo operaciones aritméticas, lógicas y comparativas entre dos números binarios de 4 bits. Esto implica la creación de una Unidad Aritmética Lógica Básica (ALU) que pueda realizar una variedad de operaciones, como suma, resta, multiplicación, división, AND, OR, XOR, entre otras.

El diseño de esta ALU debe permitir la flexibilidad para seleccionar la operación deseada por el usuario, lo cual será controlado por un controlador. Este controlador será responsable de interpretar las entradas del usuario y activar los componentes necesarios dentro de la ALU para realizar la operación especificada. Se debe diseñar una tabla de control que asigne un código único a cada operación posible, y el controlador utilizará esta tabla para determinar qué operación realizar en función de las entradas proporcionadas por el usuario.

En resumen, se necesita un diseño cuidadoso y preciso que integre componentes de hardware y lógica de control para satisfacer los requisitos de Intel Corporation y desarrollar con éxito el prototipo de la calculadora "LogicCalc". Además, es esencial considerar la eficiencia en términos de uso de recursos y velocidad de procesamiento para garantizar un rendimiento óptimo del circuito. Se deben implementar técnicas de diseño que minimicen la complejidad del circuito y maximicen su capacidad de realizar múltiples operaciones de manera simultánea o secuencial. La verificación y validación del diseño serán aspectos críticos para asegurar la precisión y confiabilidad de la calculadora "LogicCalc".

FUNCIONES BOOLEANAS Y MAPAS DE KARNAUGH

Suma

In			Out	
A	B	Ci	Cout	Suma
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Función Booleana Suma = $A'B'Ci + A'BCi' + AB'Ci' + ABC$

Suma = $A \text{ XOR } B \text{ XOR } C$ Cout

= $A'BCi + AB'Ci + ABC' + ABCi$ Cout

= $AB + Ci(A \text{ XOR } B)$

Resta

In			Out	
A	B	Ci	Cout	Resta
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Función Booleana Resta = $A'Ci + A'B' + BCi$

= $A \text{ XOR } (B \text{ XOR } Ci)$ Cout

= $A'B + A'Ci + Bci$

✚ Multiplicación

ENTRADAS A					A3	A2	A1	A0
ENTRADAS B				*	B3	B2	B1	B0
					B0A3	B0A2	B0A1	B0A0
				B1A3	B1A2	B1A1	B1A0	
		B2A3		B2A2	B2A1	B2A0		
	B3A3	B3A2		B3A1	B3A0			
SALIDAS	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0

✚ Potencia

A3, A2, A1, A0 son las entradas para la simulación del circuito.

PD	ENTRADAS				A ²	SALIDAS							
	A3	A2	A1	A0		R7	R6	R5	R4	R3	R2	R1	R0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	0	0	0	0	0	0	1
2	0	0	1	0	4	0	0	0	0	0	1	0	0
3	0	0	1	1	9	0	0	0	0	1	0	0	1
4	0	1	0	0	16	0	0	0	1	0	0	0	0
5	0	1	0	1	25	0	0	0	1	1	0	0	1
6	0	1	1	0	36	0	0	1	0	0	1	0	0
7	0	1	1	1	49	0	0	1	1	0	0	0	1
8	1	0	0	0	64	0	1	0	0	0	0	0	0
9	1	0	0	1	81	0	1	0	1	0	0	0	1

✚ Mapas de Karnaugh

Función 1 R_2

$A_1 A_0$	00	01	11	10
$A_3 A_2$	00	0	0	0
01	0	0	0	1
11	x	x	x	x
10	0	0	x	x

$R_2 = A_1 A_0' A_4$

Función 2 R_3

$A_1 A_0$	00	01	11	10
$A_3 A_2$	00	0	1	0
01	0	1	0	0
11	x	x	x	x
10	0	0	x	x

$\bar{A}_3 A_2 \bar{A}_1 A_0 + \bar{A}_3 \bar{A}_2 A_1 A_0$

$R_3 = A_0 \bar{A}_3 (A_2 \text{ XOR } A_1) A_4$

Función 3 R_4

$A_1 A_0$ $A_3 A_2$	00	01	11	10
00	0	0	0	0
01	1	1	1	0
11	x	x	x	x
10	0	1	x	x

$$R_4: A_2 \bar{A}_1 + A_3 A_0 + A_2 A_0$$

Función 4 R_5

$A_1 A_0$ $A_3 A_2$	00	01	11	10
00	0	0	0	0
01	0	0	1	1
11	x	x	x	x
10	0	0	x	x

$$R_5: A_2 A_1 A_0$$

PD	ENTRADAS				A^3	SALIDAS							
	A3	A2	A1	A0		R7	R6	R5	R4	R3	R2	R1	R0
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	0	0	0	0	0	0	1
2	0	0	1	0	9	0	0	0	0	1	0	0	0
3	0	0	1	1	27	0	0	0	1	1	0	1	1
4	0	1	0	0	64	0	1	0	0	0	0	0	0

Salidas Cuadrado:

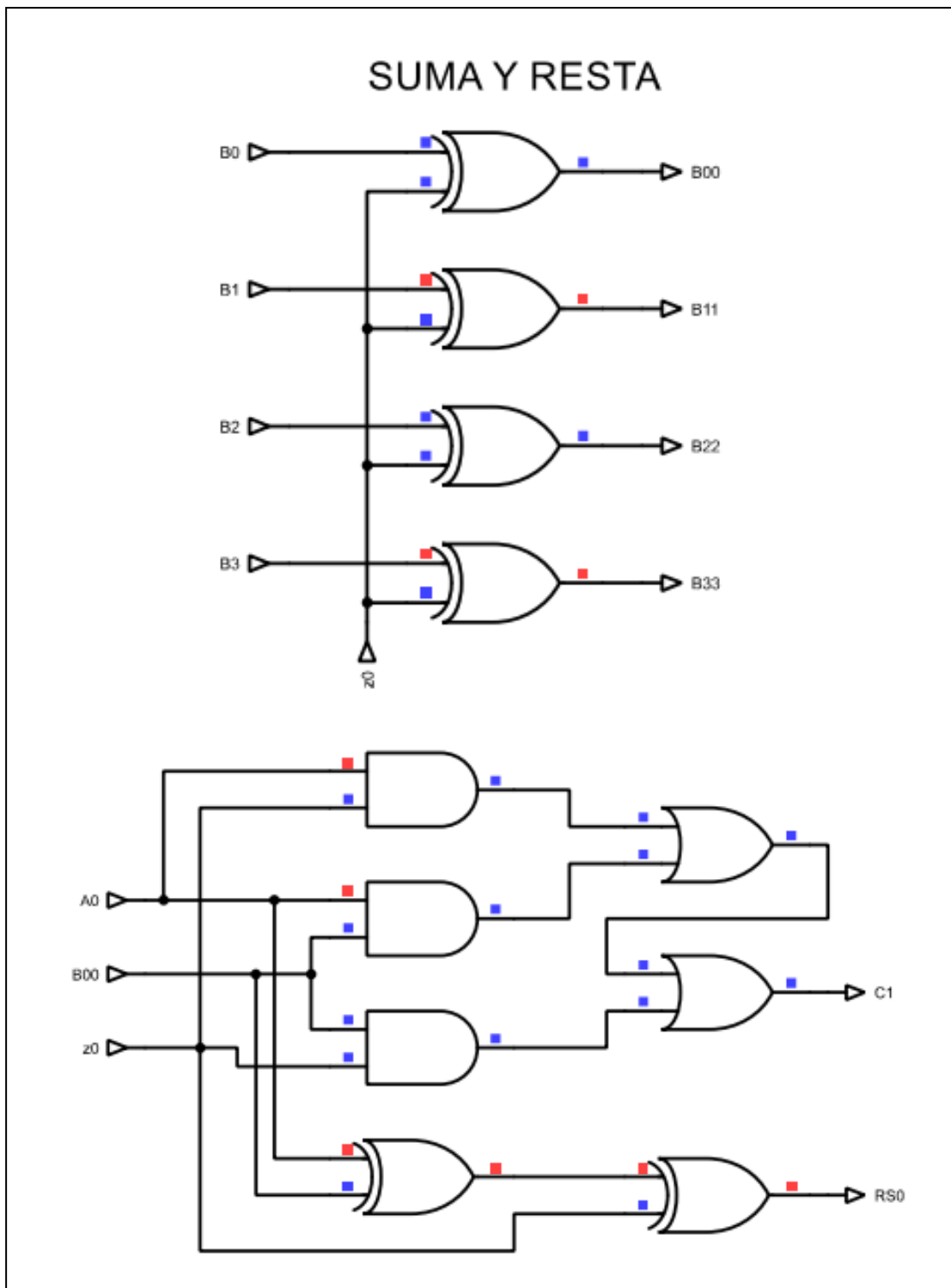
F0	A0
F1, F7	Tierra
F2	$A1 A_0' A_4$
F3	$A_0 A_3' (A_2 \text{ xor } A_1) A_4$
F4	$A_2 A_1' + A_2 A_0 + A_3 A_0$
F5	$A_2 A_1 A_4$
F6	A3

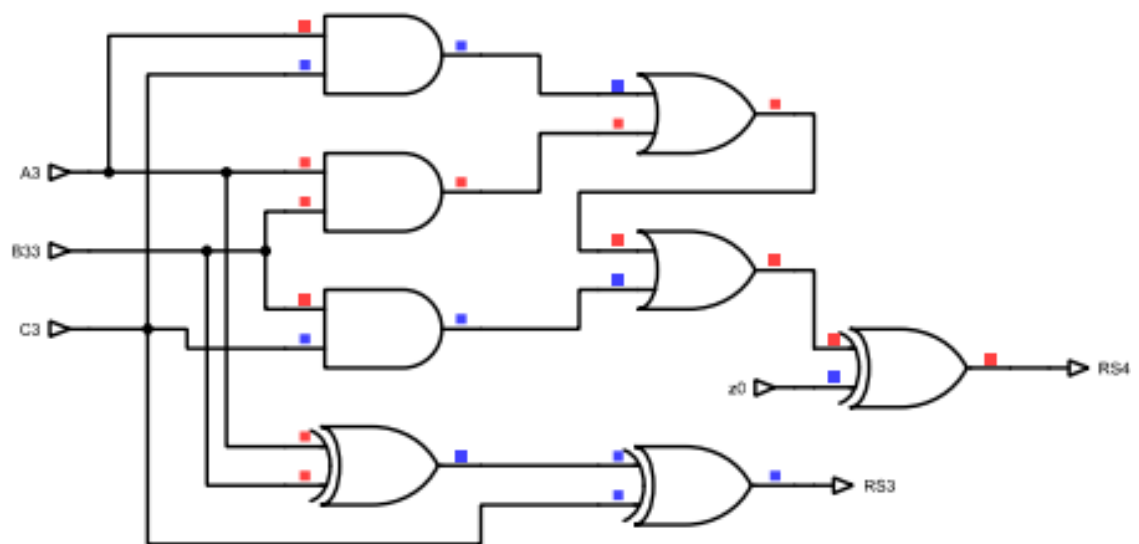
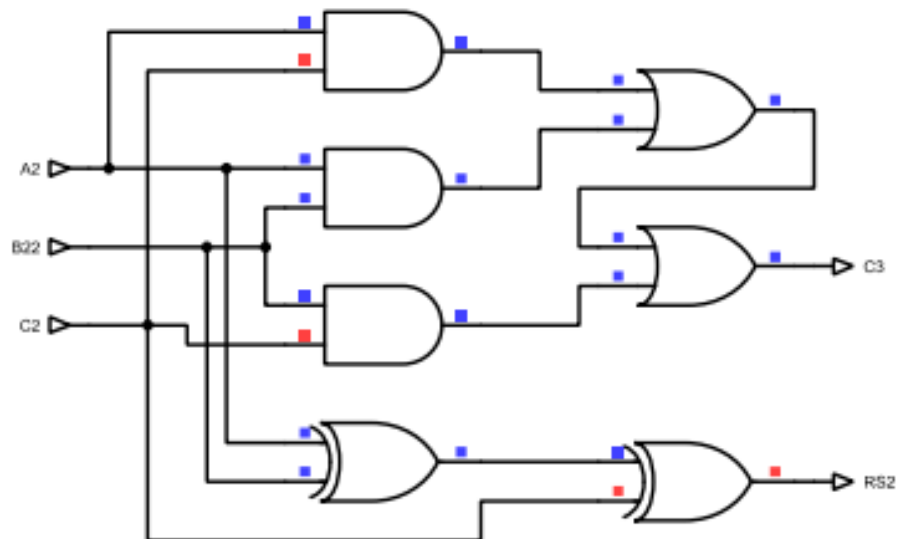
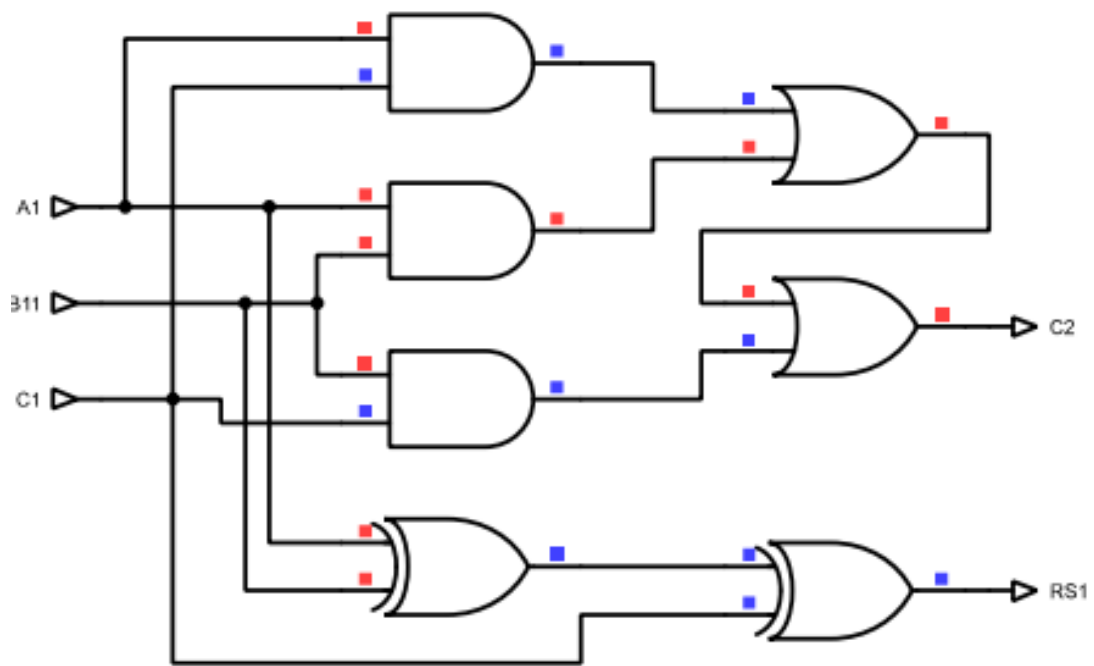
Salidas Cubo:

F0	A0
F1	$A1 A_0$
F2, F7	Tierra
F3	A1
F4	$A1 A_0$
F5	Tierra
F6	A2

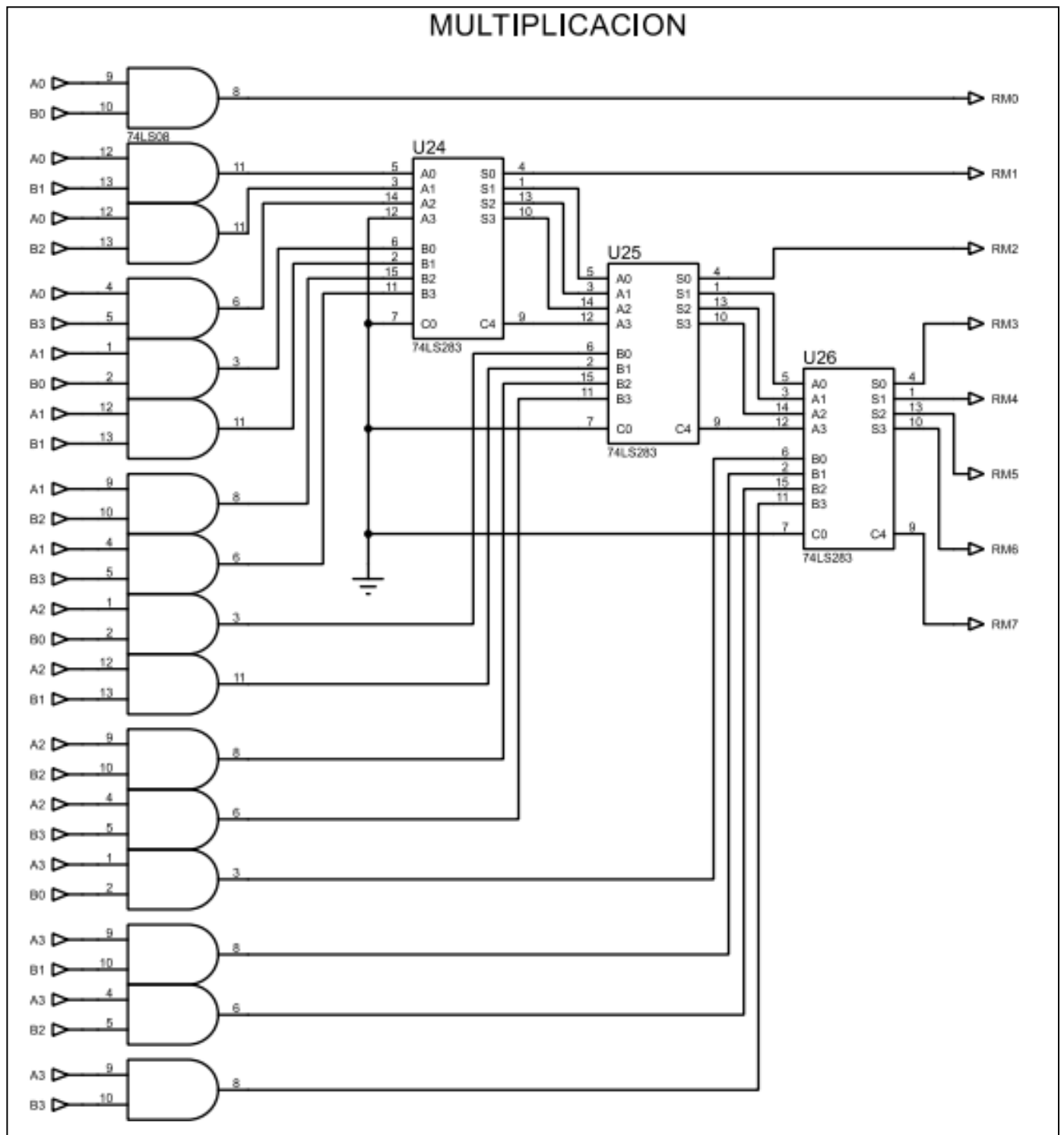
DIAGRAMAS DE LOS DISEÑOS DESARROLLADOS

✚ Suma-Resta

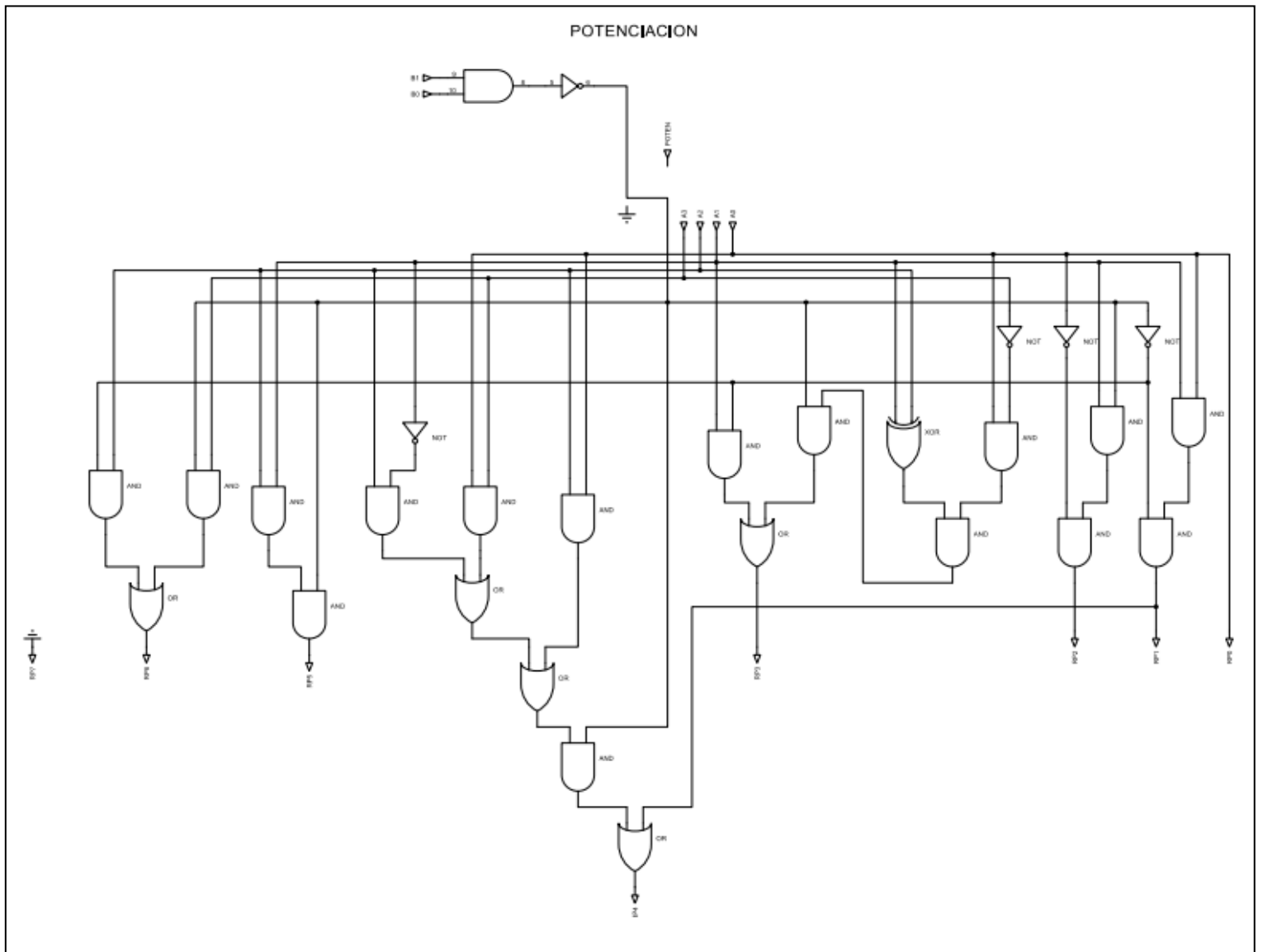




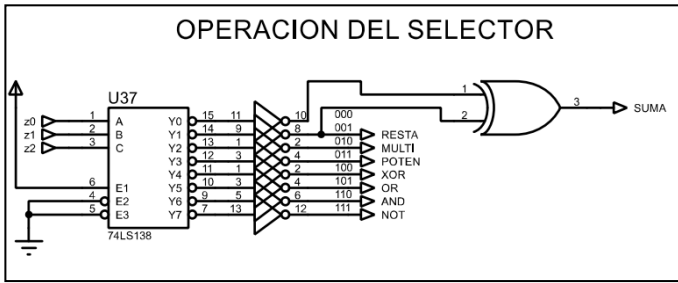
Multiplicación



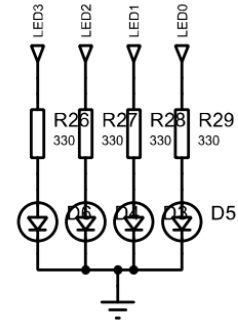
Potencia



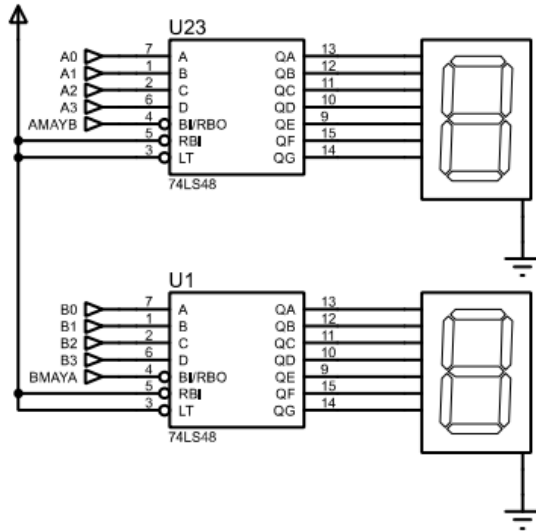
Selector



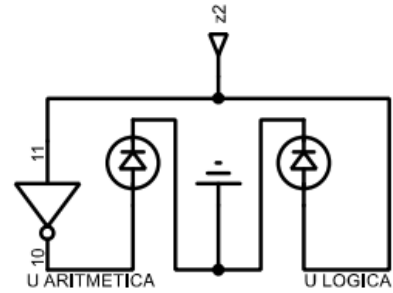
RESULTADO UNIDAD LOGICA



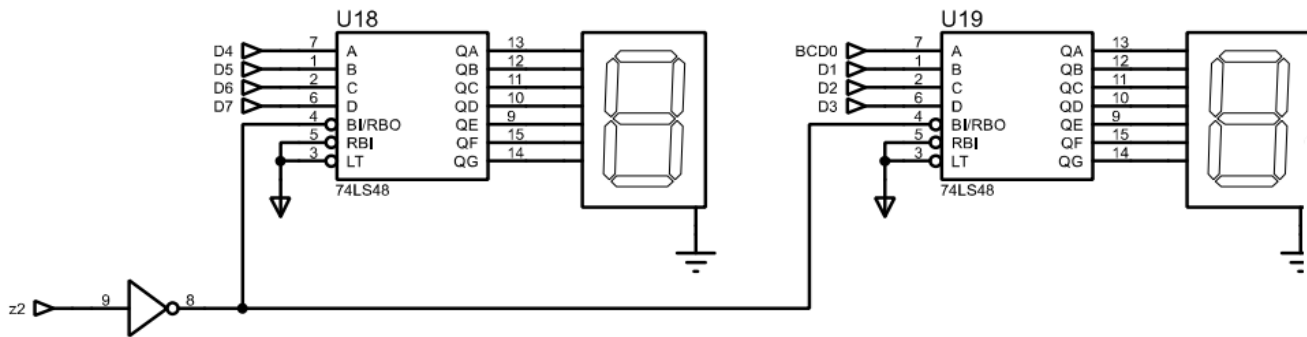
RESULTADO UNIDAD COMPARATIVA



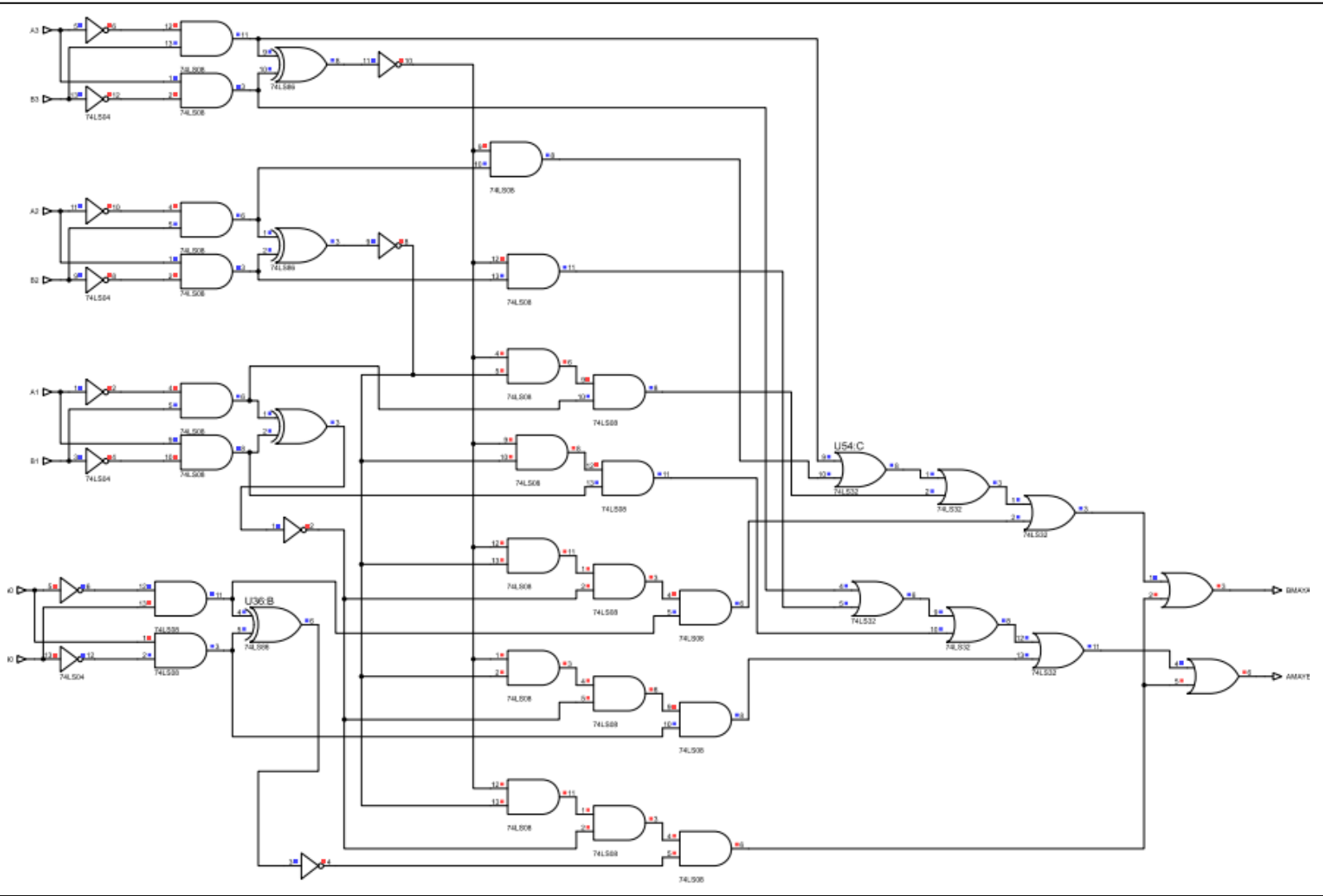
LEDS INDICADORES DE UNIDAD



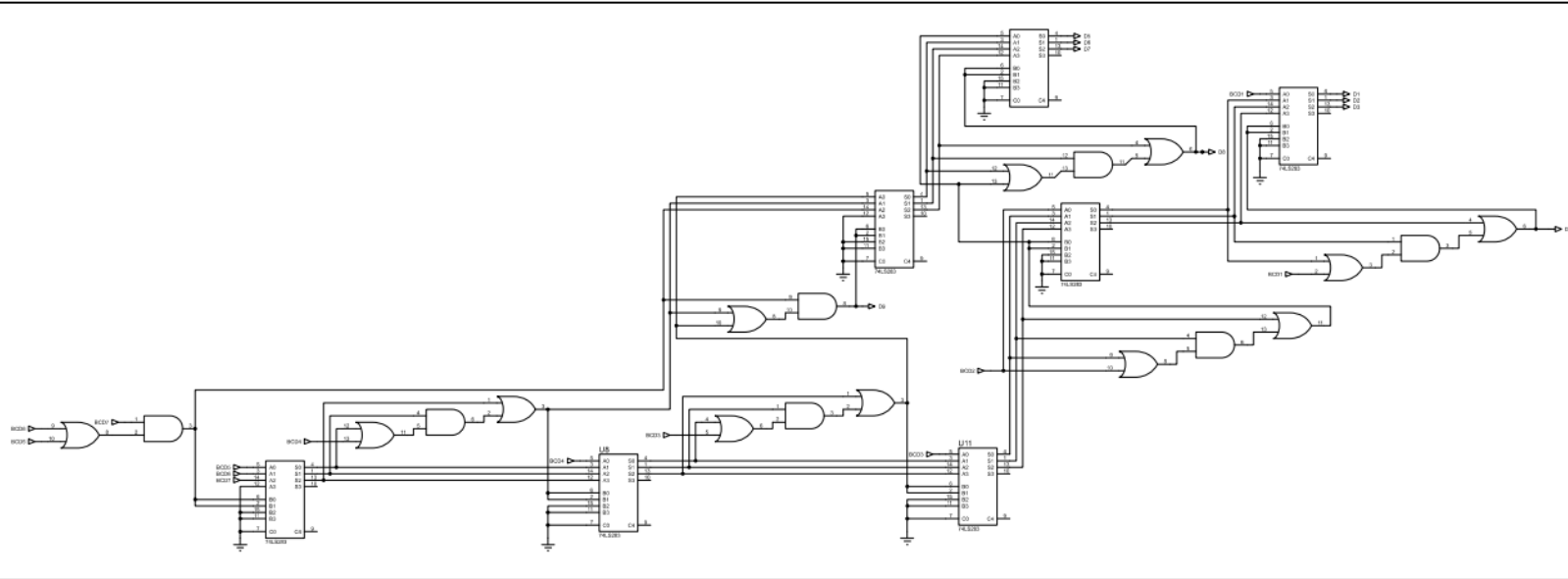
RESULTADO UNIDAD ARITMETICA



Unidad Comparativa

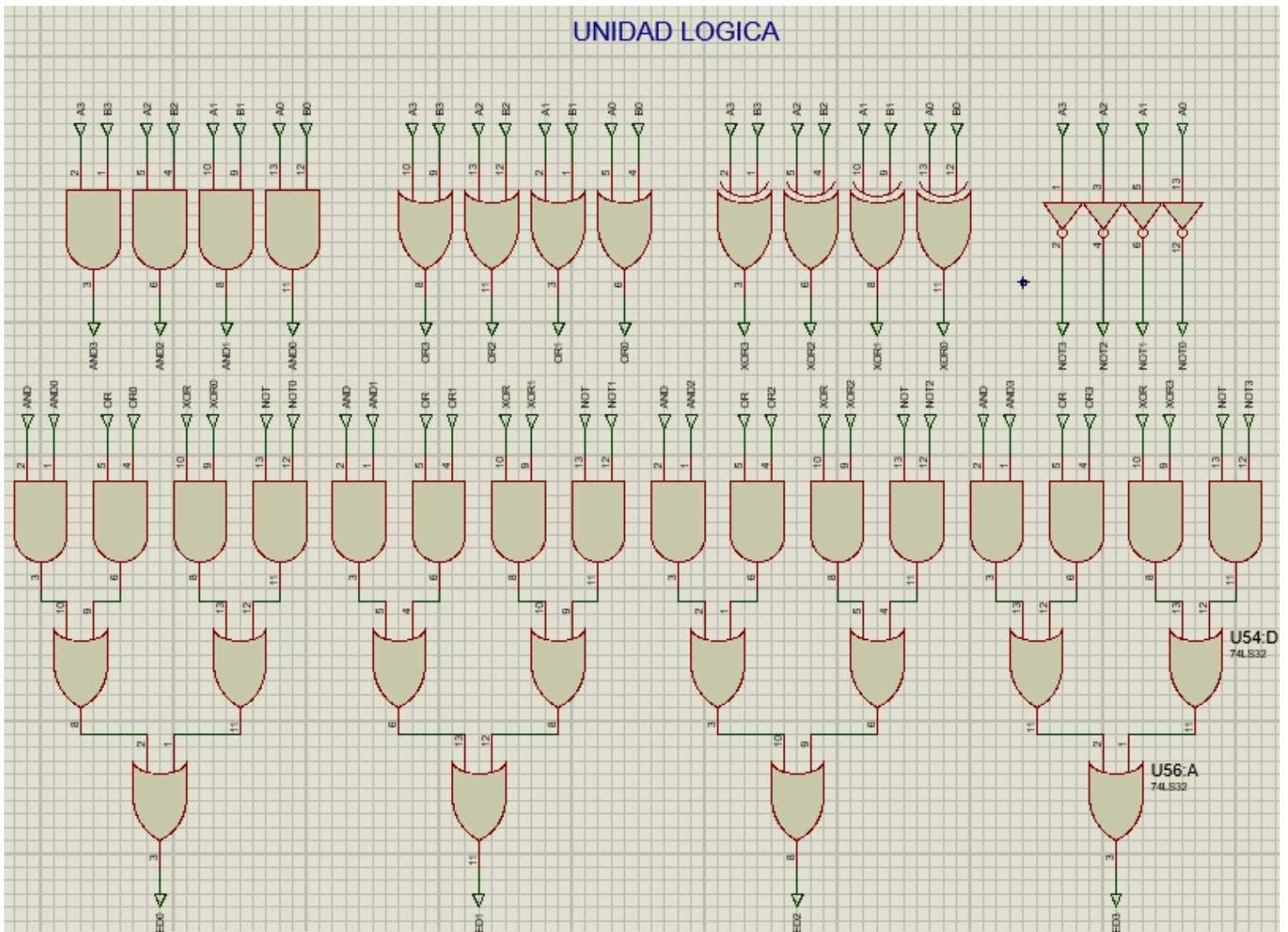


BCD A Decimal



Unidad Lógica

UNIDAD LOGICA



EQUIPO UTILIZADO

Equipo
Protoboard
Dip switch
Cable para protoboard
Display 7 Segmentos
Compuertas OR
Compuertas AND
Compuertas XOR
Compuertas NOT
Comparadores
Resistencias
Multiplexores
Led's
Decoder
Microcontrolador (Arduino)
Demultiplexores
Sumadores 74LS283
Multímetro

PRESUPUESTO

- Gastos

Componente	Cantidad	Precio Unitario	Total
Alambre para protoboard (Rojo, morado, amarillo, azul, verde, naranja, blanco)	49	Q 2.5	Q 122.5
Switch Dip de 4 posiciones	3	Q 3	Q 9
Resistencia 330 Ohm a 1 W	32	Q 0.9	Q 28.8
Display 7 Segmentos Cátodo Común	4	Q 6	Q 24
Componte 74LS04 (NOT)	6	Q 5	Q 30
Componte 74LS08 (AND)	17	Q 5	Q 85
Componte 74LS32 (OR)	11	Q 5.5	Q 60.5
Componte 74LS86 (XOR)	7	Q 7	Q 49
Componte 74LS48 (Decoder)	4	Q 15	Q 60
Componte 74LS283 (SUMADOR)	13	Q 10	Q 130
Componte 74LS138 (DEMUX)	1	Q10	Q 10
Led (Rojo y Verde)	10	Q 1	Q 10

Total: Q 618.00

- Aportación Económica

Nombre	Carné	Aportación Económica
Josseline Montecinos	202201534	Q. 154.50
Pablo Rodríguez	202201947	Q. 154.50
Evelyn Balcarcel	202200110	Q. 154.50
Carlos Lima	202201524	Q. 154.50
	Total:	Q 618.00

APORTACIÓN DE TRABAJO DE CADA INTEGRANTE

Unidad Lógica y comparativa:

- Carlos Manuel Lima Y Lima

Multiplicación y conversor BCD a decimal

- Evelyn Maricely Balcarcel Rivera

Suma-resta, potencia, "mux y demux"

- Griselda Montecinos Hernández

Simulación, Arduino y el selector

- Pablo Andrés Rodríguez Lima

CONCLUSIONES

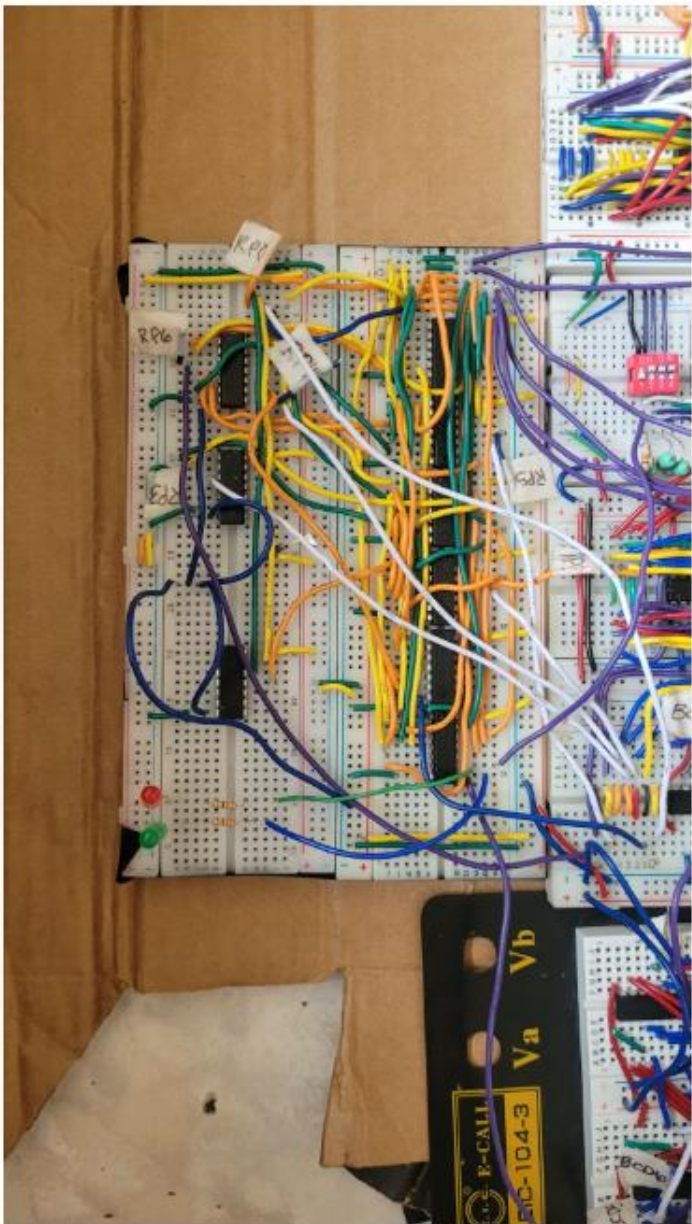
El desarrollo del prototipo de la calculadora "LogicCalc" para Intel Corporation implica un enfoque meticuloso y multidisciplinario que abarca tanto aspectos de hardware como de software. La creación de una Unidad Aritmética Lógica Básica (ALU) capaz de realizar una amplia gama de operaciones aritméticas, lógicas y comparativas entre números binarios de 4 bits requiere un diseño complejo y eficiente.

El controlador encargado de seleccionar y ejecutar las operaciones debe ser capaz de interpretar las entradas del usuario y activar los componentes adecuados dentro de la ALU para realizar la operación deseada. La eficiencia en el uso de recursos y la velocidad de procesamiento son aspectos críticos para considerar para garantizar un rendimiento óptimo del circuito. Es esencial emplear técnicas de diseño que minimicen la complejidad del circuito y maximicen su capacidad para realizar múltiples operaciones de manera simultánea o secuencial.

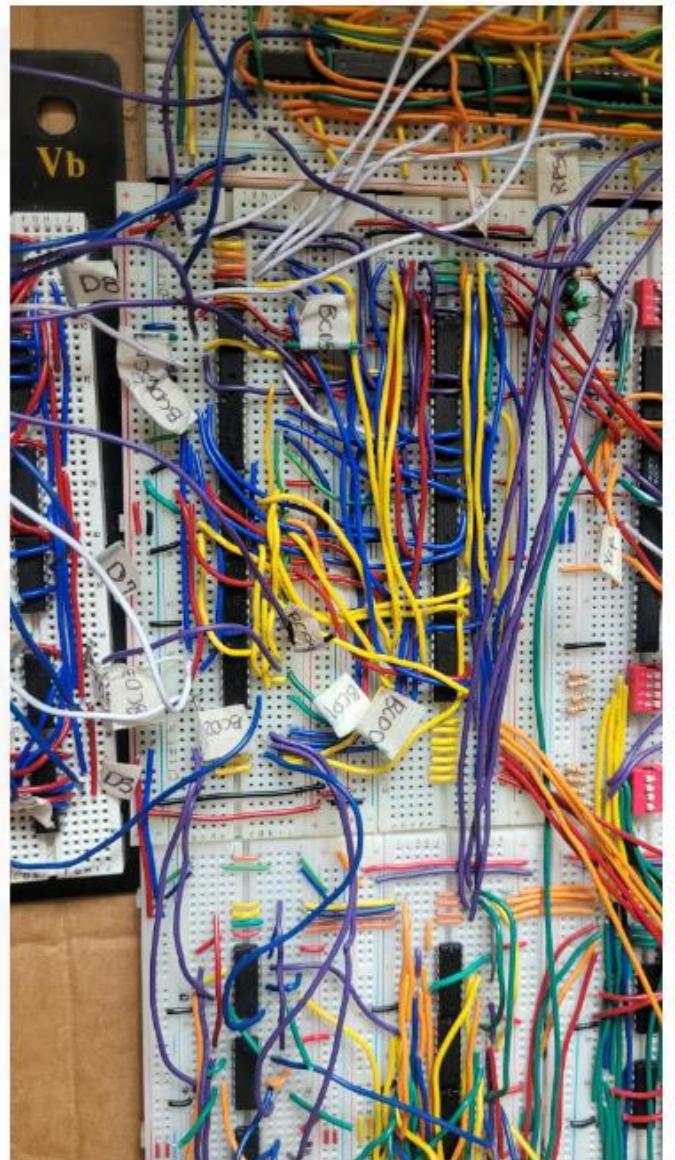
El análisis detallado de las operaciones aritméticas y lógicas requeridas, así como de las interacciones entre los componentes, es fundamental para el éxito del proyecto. La verificación y validación exhaustivas del diseño son procesos críticos para garantizar la precisión y confiabilidad de la calculadora "LogicCalc"

ANEXOS

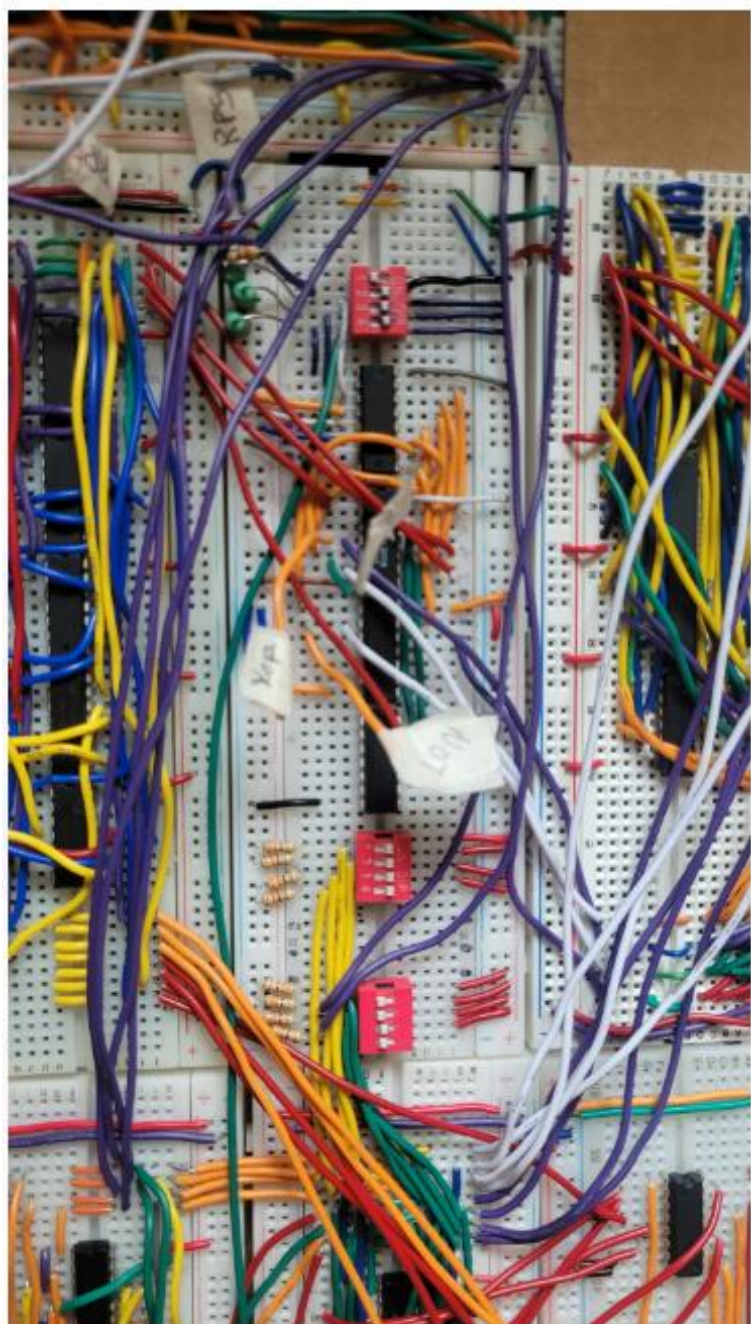
Parte Física



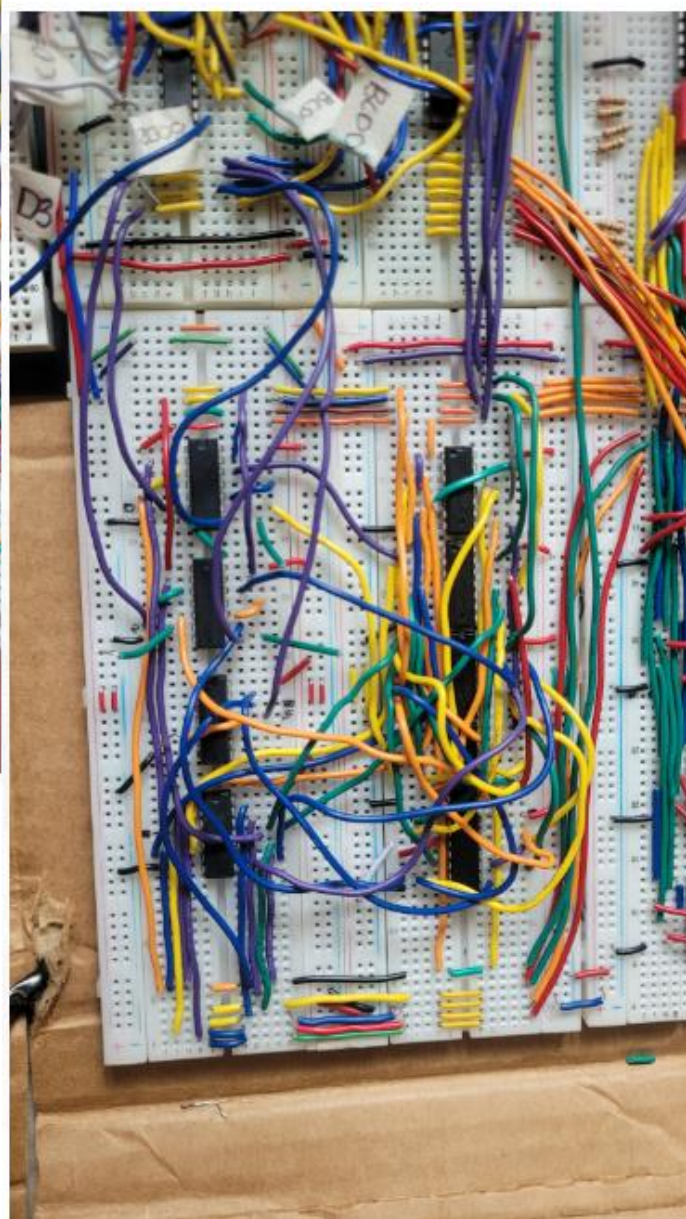
Potencia



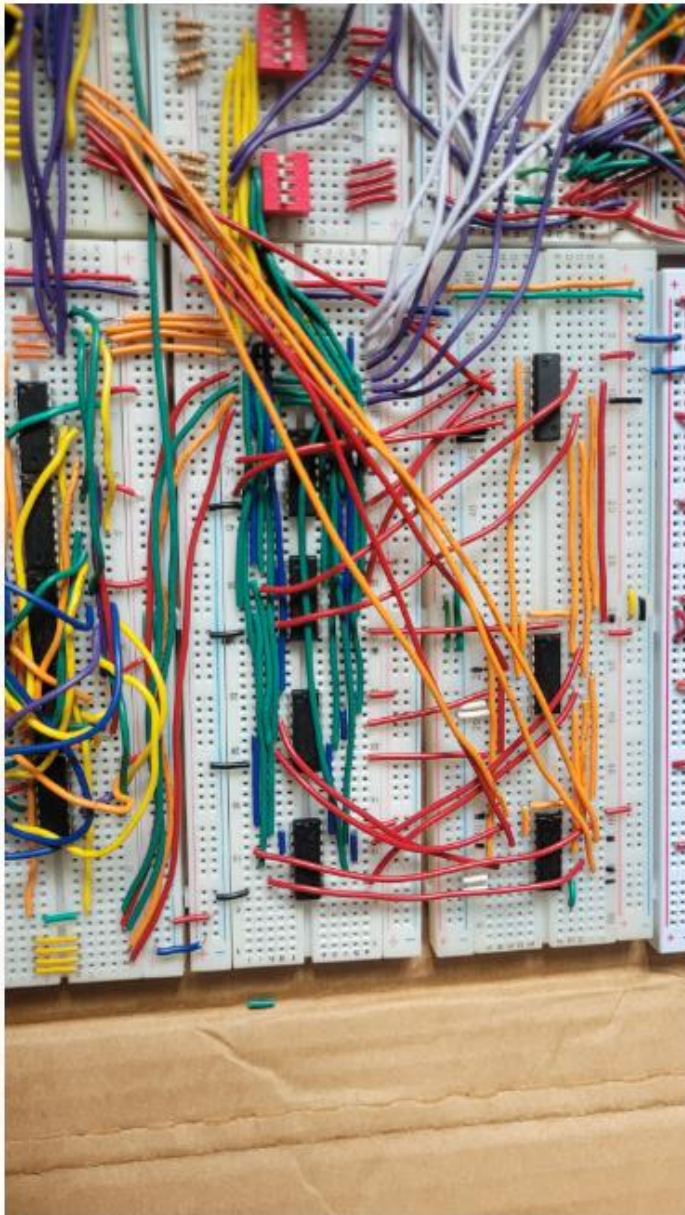
Multiplexor de Salidas Aritmeticas



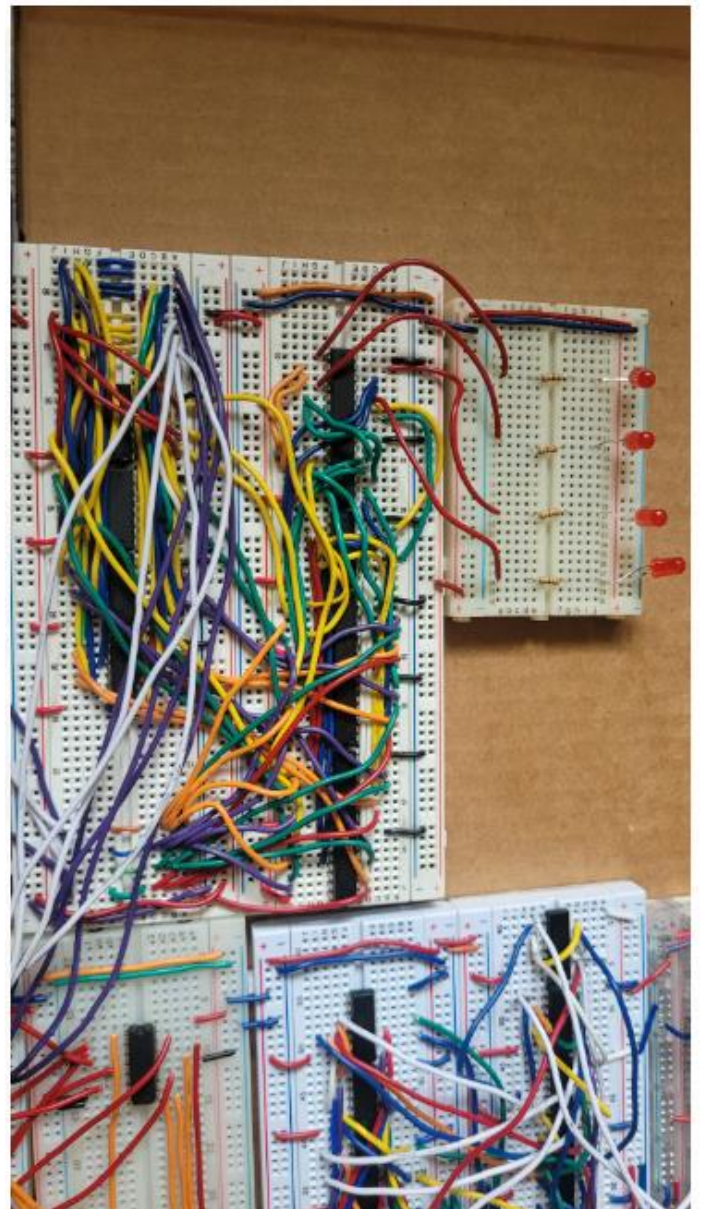
Selector



Suma - Resta



Multiplicacion



Unidad Logica

Unidad Comparativa

