

# Proceso de desarrollo Taller N2 Fundamentos de Arquitectura

1<sup>st</sup> Gabriel Guzmán Rojas  
Ingeniería en Computadores  
Instituto Tecnológico de Costa Rica  
Cartago, Costa Rica  
gabguzman@estudiantec.cr

2<sup>nd</sup> Sebastián Hernández Bonilla  
Ingeniería en Computadores  
Instituto Tecnológico de Costa Rica  
Cartago, Costa Rica  
sebhernandez@estudiantec.cr

**Abstract**—El presente documento va a consistir en como se creo un circuito que permita decodificar las señales de 1 y 0, proveniente de switches. El enfoque de este taller es un avance sobre el proyecto del mismo curso, solo que los switches seran los dedos de la mano.

**Index Terms**—Circuito combinacional, Algebra Booleana, decodificador, BCD

original, diseñamos los circuitos combinacionales necesarios para implementar las funciones obtenidas.

## I. INTRODUCCIÓN

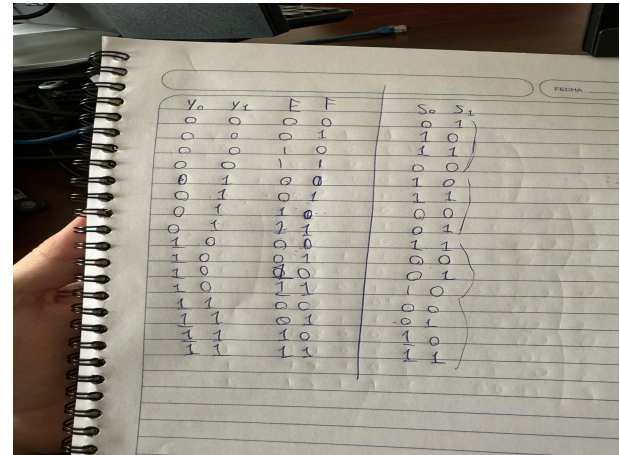
En la presente bitácora se describe el desarrollo de un decodificador con display de siete segmentos, utilizando principios de lógica combinacional. El proyecto tiene como objetivo implementar un sistema que sume el valor acumulado, representado por dos bits, con un valor de entrada adicional, utilizando un decodificador en lugar de un sumador convencional. Los resultados de la operación se mostrarán en un display de siete segmentos a través de un decodificador BCD.

Para la implementación se utilizaron los siguientes materiales: dos dip switch para la selección de valores binarios, un registro 74LS273 para almacenar el valor acumulado, un decodificador BCD 74LS248 para controlar el display de siete segmentos, compuertas lógicas AND (74LS08), NOT (74LS04), NAND (74LS00), XOR (74LS86), y OR (74LS32), así como un display de siete segmentos para la visualización del resultado.

La lógica combinacional desarrollada se encargará de gestionar la suma circular de los valores binarios, de manera que el resultado de la operación se mantenga dentro del rango de 0 a 3 (2 bits), haciendo uso eficiente de las compuertas lógicas y el decodificador.

## II. BITÁCORA

1) *Martes 27 de Agosto:* Iniciamos simulando un circuito de 4 bits, cuyo propósito era representar los dedos en un sistema específico. Posteriormente, construimos la tabla de verdad correspondiente, utilizando las entradas Y0, Y1, E y F. Las salidas obtenidas fueron s0 y s1. Con la tabla de verdad establecida, procedimos a desarrollar los mapas de Karnaugh para simplificar las expresiones lógicas. Después de validar que la tabla de verdad simplificada coincidiera con la



Y <sub>0</sub>	Y <sub>1</sub>	E	F	S <sub>0</sub>	S <sub>1</sub>
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	1	1
0	0	1	1	0	1
0	1	0	0	1	0
0	1	0	1	0	1
0	1	1	0	0	0
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	1	1	0
1	0	1	0	0	1
1	0	1	1	1	1
1	1	0	0	0	0
1	1	0	1	1	0
1	1	1	0	0	1
1	1	1	1	1	1

Figura I. Tabla de Verdad acumulador y sumador.

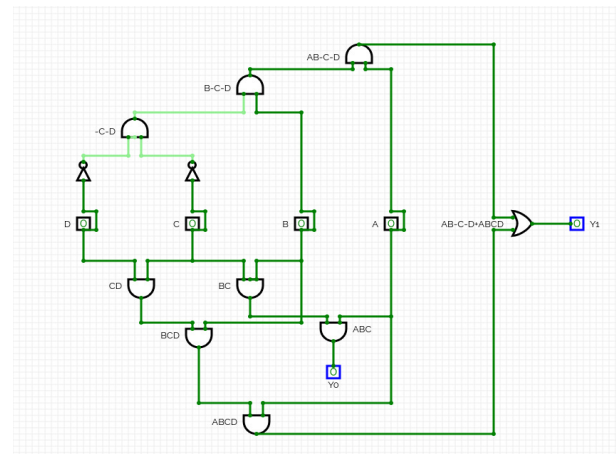


Figura II. Circuito combinacional de los 4 dedos.

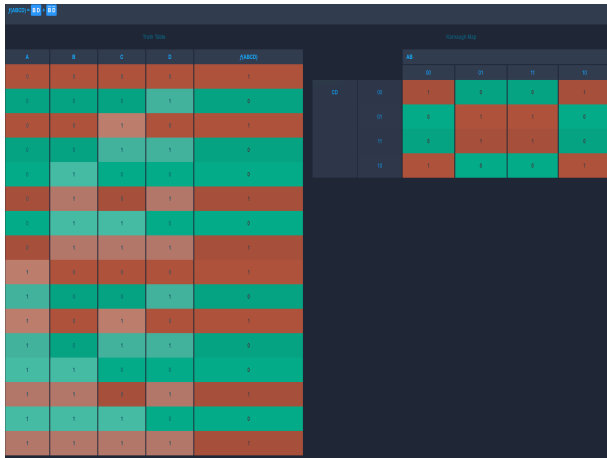


Figura III. Mapa de Karnaugh para S1

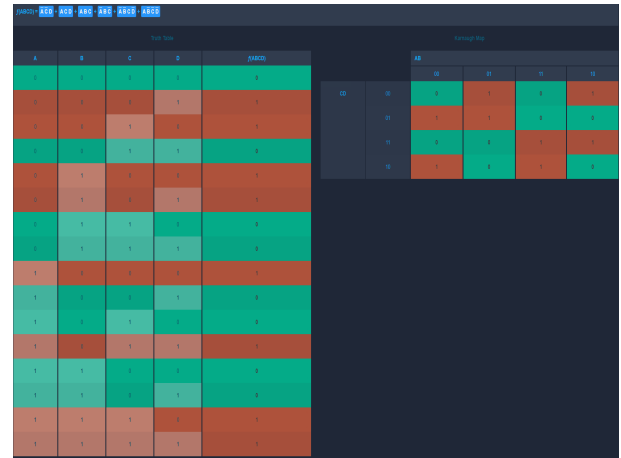


Figura VI. Mapa de Karnaugh para S0.

## Truth Table

B	D	Output
0	0	T
0	1	F
1	0	F
1	1	T

Figura IV. Tabla de verdad de S1.

## Logic Gate

View Non-minimized

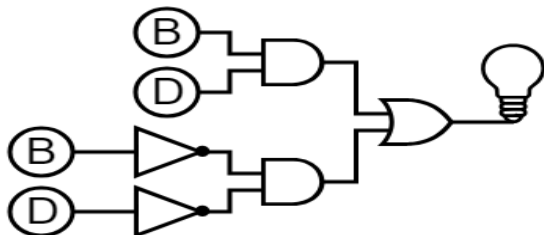


Figura V. Circuito combiancional de S1.

## Truth Table

A	B	C	D	Output
0	0	0	0	F
0	0	0	1	T
0	0	1	0	T
0	0	1	1	F
0	1	0	0	T
0	1	0	1	T
0	1	1	0	F
0	1	1	1	F
1	0	0	0	T
1	0	0	1	F
1	0	1	0	F
1	0	1	1	T
1	1	0	0	F
1	1	0	1	F
1	1	1	0	T
1	1	1	1	T

$$\overline{A}\overline{B}\overline{C}\overline{D} + \overline{A}\overline{B}\overline{C}D + \overline{A}\overline{B}C\overline{D} + \overline{A}\overline{B}CD + \overline{A}B\overline{C}\overline{D} + \overline{A}B\overline{C}D + \overline{A}BC\overline{D} + \overline{A}BCD$$

Figura VII. Tabla de Verdad de S0.

## Logic Gate

View Minimalized

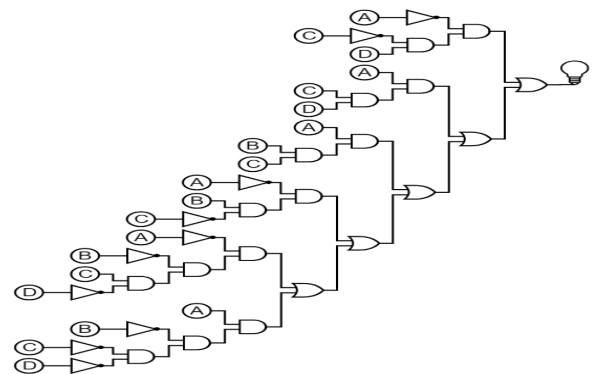


Figura VIII. Circuito Combinacional de S0.

2) *Jueves 29 de Agosto:* En esta sesión, comenzamos simulando ambos circuitos combinacionales conjuntamente: el circuito que representa los dedos y el sumador, los cuales generan como salidas s0 y s1. Antes de proceder a la implementación física, dibujamos el diagrama detallado del circuito, para visualizar cómo debía verse una vez ensamblado. Posteriormente, procedimos a armarlo de manera física, realizando pruebas exhaustivas en cada salida para asegurarnos de

que todo correspondiera correctamente con las simulaciones previas. El resultado fue exitoso, ya que el circuito funcionó perfectamente conforme a lo esperado.

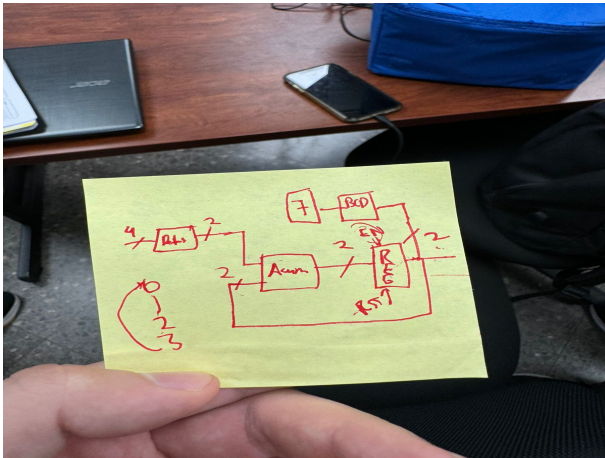


Figura IX. Diagrama Conceptual circuito integrado.

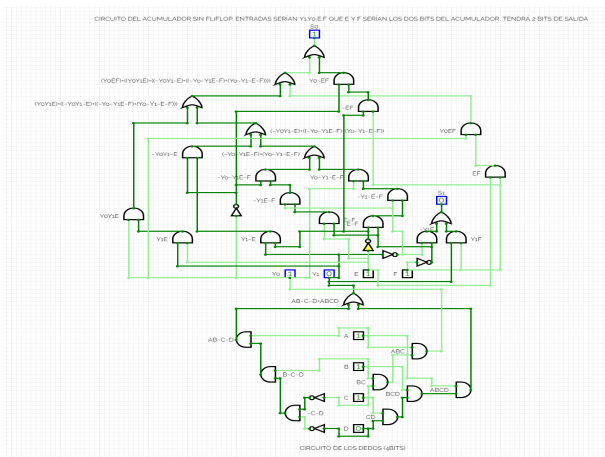


Figura X. Simulación de ambos circuitos combinatoriales juntos.



Figura XI. Armado de la salida de S0.

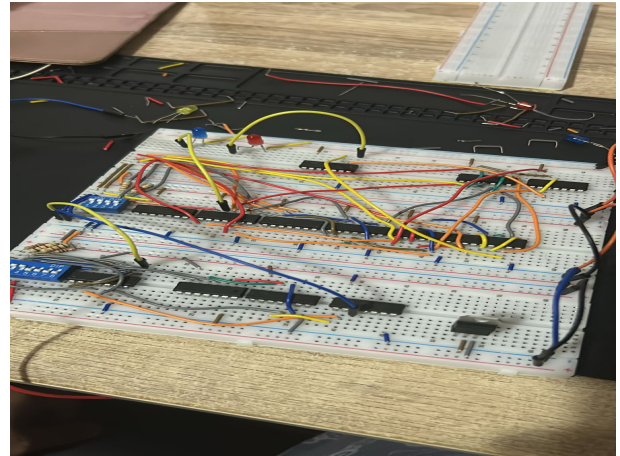


Figura XII. Ambos circuitos combinatoriales listos.

3) *Sábado 31 de Agosto:* Con la implementación de los circuitos combinatoriales finalizada, procedimos a la siguiente etapa del proyecto, que consistió en añadir el BCD, el registro, el botón y el display de 7 segmentos. Sin embargo, a pesar de nuestros esfuerzos, no obtuvimos los resultados esperados, ya que el circuito no funcionó como se había planificado.

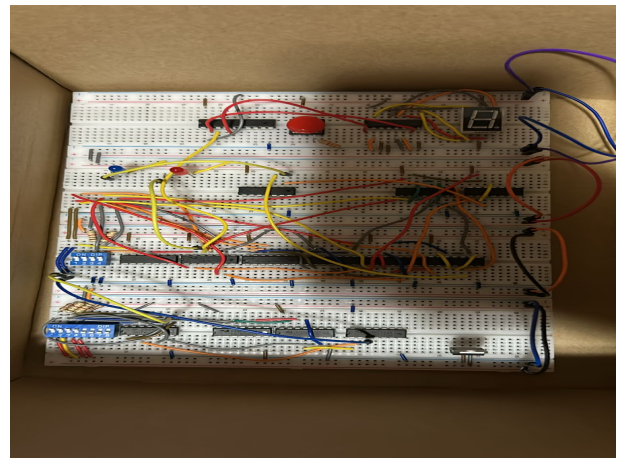


Figura XIII. Intento de integrar el registro, BCD y 7 segmentos.

4) *Martes 3 de Setiembre:* Este día se realizó el paso final del circuito, que correspondía al uso del registro, lo único que hasta el momento no había funcionado correctamente. El problema que enfrentábamos era que el registro no parecía operar como esperábamos, ya que el display de 7 segmentos no cambiaba de 0. Después de realizar varias pruebas, llegamos a la conclusión de que el clock y el master reset no estaban bien conectados.



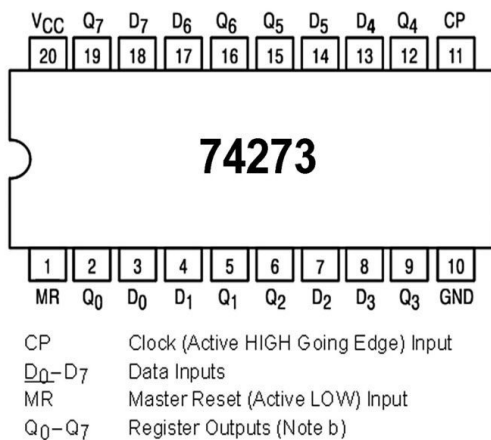


Figura XIV. Registro 74LS273

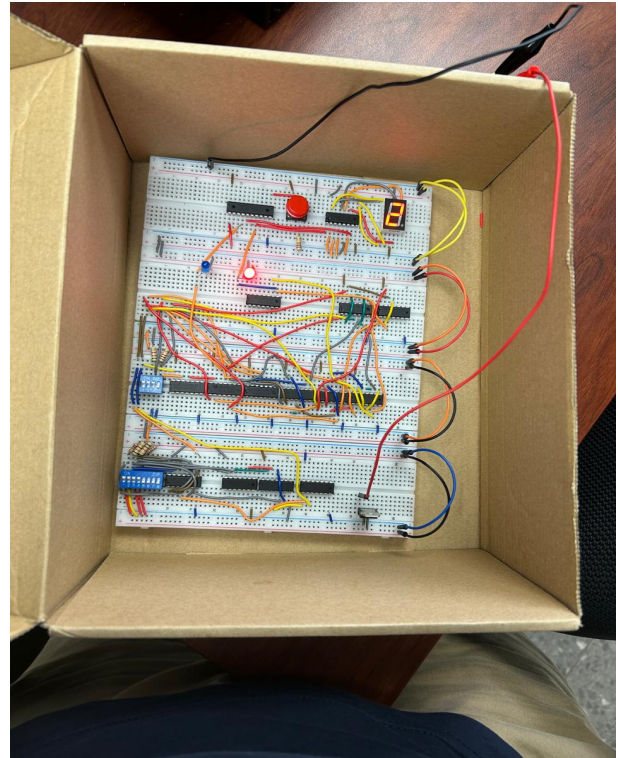


Figura XV. Circuito en funcionamiento

De igual manera en la imagen se ve el circuito final que es totalmente funcional. De forma extra a lo que se solicita, agregamos dos leds, para comprobar que las salidas funcionen  $S_0$  y  $S_1$ .

Como se observa en la figura XIV el registro que utilizamos fue el 74LS273, que cuenta con 8 salidas y 8 entradas, además de un clock y un master reset. El inconveniente surgía porque no estábamos utilizando correctamente ambos pines, principalmente porque no habíamos comprendido bien el esquemático del componente. Este indica que tanto el clock como el reset deben estar conectados, y además, que el reset funciona en Active Low. Inicialmente, habíamos entendido que eso significaba conectarlo a tierra, cuando en realidad debía estar conectado a corriente. Esto se debe a que el reset se activa al recibir un 0, y al estar siempre conectado a tierra, el registro se reiniciaba constantemente.

Una vez corregido esto, el circuito funcionó correctamente. El mismo principio se aplicó al clock, ya que configuramos el botón para que siempre envíe un 1, y cuando recibe un 0, cambia el número que el registro está almacenando.

5) *Sabado 7 de Setiembre:* Se realizó la implementación final del proyecto, que consistió en la integración de los LEDs y fotoresistores con el motor. Basándose en el video “How to Use a Photocell/Photoresistor”, se conectaron los LEDs en un arreglo 4x3 y los fotoresistores para detectar las sombras proyectadas, generando señales binarias que fueron procesadas por el circuito combinacional. El motor, controlado por el registro acumulador, se encendía cuando el acumulador estaba en los estados 2 o 1, y se apagaba en los estados 0 o 3, logrando así la integración entre los sensores, la lógica y el actuador.

6) *Lunes 9 de Setiembre:* Ese día se comenzó a armar el circuito de los fotoresistores y LEDs. El funcionamiento es sencillo: se utilizó un transistor BJT NPN, con el emisor conectado a tierra, la fotoresistencia ubicada entre la base y el emisor, mientras que el LED se conectó con su patilla negativa al colector y la positiva a la fuente de corriente, lo que permite controlar el encendido del LED en función de la luz detectada por la fotoresistencia.

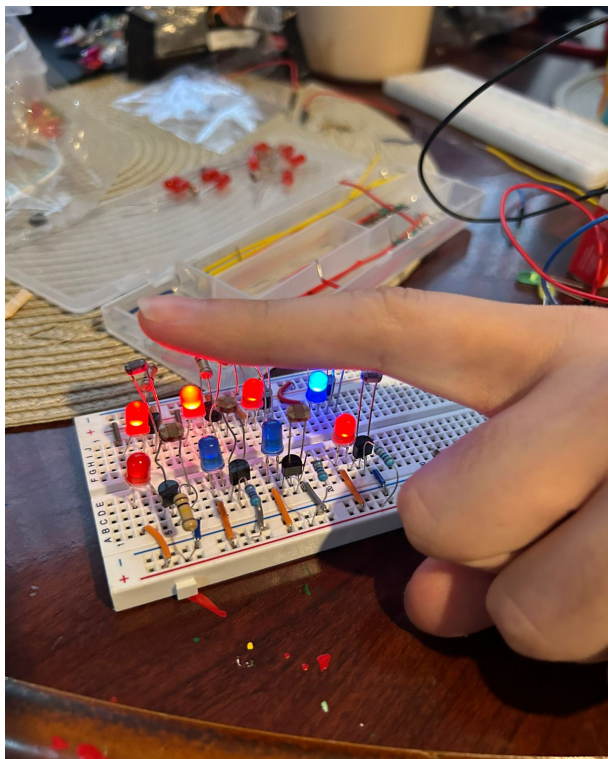


Figura XVI. Fotoresistores y LEDS.

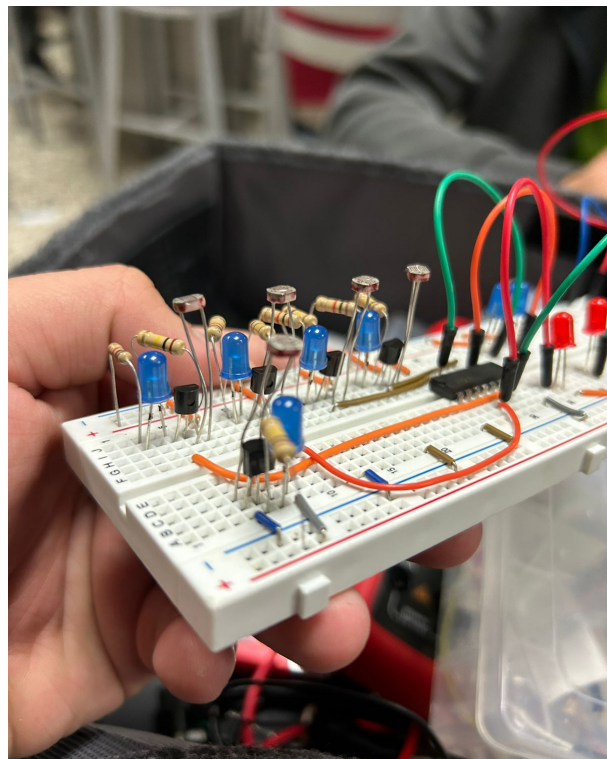


Figura XVII. Solución de problema con LEDS.

7) *Martes 10 de Setiembre:* Ese día se intentó solucionar un problema relacionado con la obtención de un 1 lógico y un 0, que impedía el funcionamiento correcto del circuito. Después de analizarlo detenidamente, se identificó que el problema se debía a la falta de una resistencia en el LED. Al agregar la resistencia adecuada, el circuito logró funcionar correctamente, permitiendo la distinción entre los niveles lógicos deseados.

8) *Miercoles 11 de Setiembre:* Ese día se realizó el montaje del circuito accionador, el cual se tuvo que implementar en una protoboard separada debido a que el motor requiere un voltaje considerablemente mayor al del resto del circuito. Tras esta etapa final de integración, el proyecto fue completado con éxito, asegurando el correcto funcionamiento de todas las partes, desde los sensores hasta el accionador, cumpliendo con los requisitos establecidos.

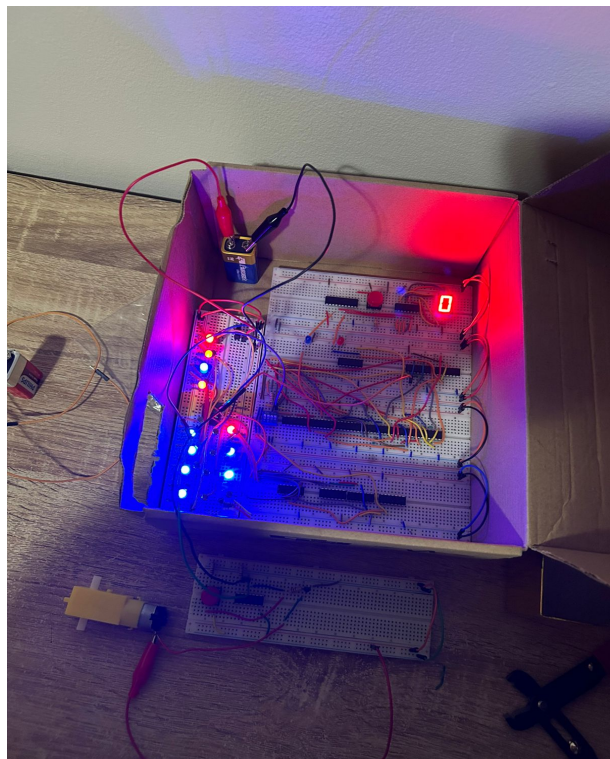


Figura XVIII. Circuito final.