

SIMULACIÓN Y RESOLUCIÓN DE PROBLEMAS IMPLEMENTANDO CIRCUITOS COMBINACIONALES

Bryan V. Alvarado ^{1*}, Evelin E. Hidalgo ¹, Jorge A. Cruz ¹

Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, Sangolguí, Ecuador

bvalvarado@espe.edu.ec

Resumen - Un circuito combinacional es un circuito electrónico, en el que el valor de sus salidas en un determinado instante, dependen del valor de las entradas en ese mismo instante. Es decir, es un circuito que carece de memoria. Trabajan con números, y con la tecnología con la que están realizados, estos números están representados en binario.

Índice de Términos -Binario-Circito-Combinacional-Electrónico.

I. INTRODUCCIÓN

Un circuito comparador compara dos entradas binarias (A y B de n bits) para indicar la relación de igualdad o desigualdad entre ellas por medio de "tres banderas lógicas" que corresponden a las relaciones A igual B, A mayor que B y A menor que B. El mismo que va a ser diseñado y resuelto con los prerrequisitos ya establecidos.

Un circuito votador es conocido por el uso de redundancia en aplicaciones electrónicas con altos requisitos de seguridad ante fallos. El mismo que va a ser diseñado para aviones de seguridad modernos como norma de seguridad

Finalmente se va a diseñar un circuito que tenga como entradas la representación binaria de la hora actual menos ocho y que proporcione a la salida el número de turno que está trabajando (si procede) o "0" si es hora de descanso para un horario laboral de una factoría es de 8 horas diarias

II. ESTADO DEL ARTE

En 2009, Zhaohui Gan, Gang Shi y Tao Shang de la Facultad de Ciencias de la Información e Ingeniería, Universidad de Ciencia y Tecnología de Wuhan realizaron un estudio implementando circuitos lógicos combinacionales en el diseño automático de la programación genética (GP). GP usando una estructura de árbol o una estructura lineal en lugar de cadenas para representar un circuito lógico combinacional. En el método GP, los nodos hoja y los nodos padres representan señales de entrada y puertas lógicas en circuitos, respectivamente. Los circuitos también evolucionan con algoritmos de selección clonal basado en gráficos (Zhaohui Gan, 2009, p.1) [1].

Alidoust Zahra y Basiri Mohammad, del Departamento de ingeniería informática Departamento de Ingeniería Universidad de Isfahan y de la Universidad Shahrekord, Irán en 2017 centraron su estudio en mejorar el algoritmo genético a través de la agrupación para el diseño de circuitos lógicos combinacionales, los diseños evolutivos son buenas alternativas para el diseño de circuitos lógicos pero tienen un inconveniente común, a saber, la alta aleatoriedad de su método cruzado. Para superar este inconveniente el método propuesto, se adopta el algoritmo k-means para optimizar el algoritmo genético con el fin de aumentar la eficiencia y reducir el costo de producción. Los criterios de optimización de los elementos del circuito, como el recuento de puertas de los transistores y el

consumo de energía (Zahra Alidousti, 2017, p.1) [2].

Mukherjee, B. y Dandapat, Dept, Jadavpur University Kolkata, en el Simposio internacional 2010 presentaron un estudio sobre el diseño de sistemas electrónicos mediante el método combinacional cíclico para VLSI de baja potencia. El objetivo principal del circuito cíclico es introducir retroalimentación estructural y evitar la retroalimentación lógica para obtener una salida primaria combinatoria. El estudio del circuito cíclico incluye análisis funcional, es decir, para determinar qué valores aparecerán, el análisis de tiempo que determina cuándo aparecerán estos valores y el área. El objetivo del estudio es diseñar y verificar diferentes circuitos combinacionales en un método cíclico mediante la optimización del área, la potencia y el retraso. (L Mukherjee, B. y Dandapat, 2010, p.1) [3].

Youjun Xu, Dantong Ouyang¹, Yuxin Ye, y Jialiang He, Facultad de Informática y Tecnología Jilin University, Changchun, China, en su trabajo de investigación basado en la solución de problemas SAT con álgebra booleana expusieron que en las últimas décadas, se proponen muchos métodos de SAT, basado en la resolución, por medio de la regla de extensión, en el estudio de la regla de extensión, encontramos que el problema SAT se puede resolver con algoritmos de configuración de golpe. Si podemos encontrar un conjunto de aciertos de un conjunto de cláusulas, y si no hay ningún par de literales complementarios en el conjunto de aciertos, el conjunto de cláusulas es satisfactoria. El algoritmo BHS basado en el álgebra booleana propuesto por Jiang es un algoritmo de conjunto de golpes eficiente. (Xu, Y., Ouyang, D., Ye, 2010, p.1) [4].

Ruanqianqian Huang y Franklyn Turbak del Wellesley College, Department of Computer Science en 2019 implementaron un diseño para la conversión bidireccional entre bloques y texto para App Inventor mediante fragmentos de código visual que evitan errores semánticos sintácticos y estáticos y reducen la carga cognitiva para este fin diseñaron un sistema de modo dual para MIT App Inventor que admite representaciones textuales para bloques, espacios de trabajo, pantallas y proyectos completos que permiten la conversión bidireccional entre bloques isomorfos y representaciones de texto, permitiendo que individuos de varios niveles de experiencia en programación se relacionen con la interfaz (R. Huang, F. Turbak, 2019) [5].

Para el Producto de Unidad presente, se utilizó una estructura de árbol o una estructura lineal en lugar de cadenas para representar un circuito lógico combinacional (Zhaohui Gan, 2009, p.1), para la optimización de los elementos del circuito, como el recuento de puertas de los transistores y el consumo de energía se utilizó parte de los postulados empleados en algoritmo genético con circuitos combinacionales (Zahra Alidousti, 2017, p.1) el entorno de programación para el aplicativo se hizo sobre la base de MIT App Inventor, el sistema tiene la capacidad de proyectar un número decimal de salida en la pantalla del dispositivo android simulando la proyección de un display de 7 segmentos (R. Huang, F. Turbak, 2019).

III. MARCO TEÓRICO

Circuito Combinacional

Un circuito combinacional es un circuito electrónico, en el que el valor de sus salidas en un determinado instante, dependen del valor de las entradas en ese mismo instante. Es decir, es un circuito que carece de memoria. Trabajan con números, y con la tecnología con la que están realizados, estos números están representados en binario[6]. El siguiente circuito:

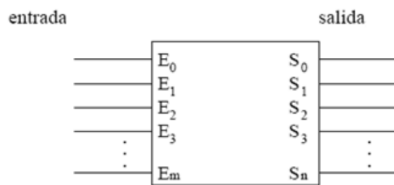


Fig 1.Circuito Combinacional

En un determinado instante van a depender de las entradas en ese preciso instante. Estos circuitos se caracterizan porque no almacenan información. Las salidas están relacionadas con las entradas a través de una función booleana. Cada bit de salida de un circuito combinacional, se obtiene mediante una función booleana aplicado a las variables de entrada. Así, si un circuito tiene n salidas, necesitaremos n funciones booleanas para caracterizarlo. Así, un circuito combinacional que tiene 3 entradas: A, B y C, y dos salidas F, G, que son dos funciones booleanas que dependen de las variables de entrada: $F(A,B,C)$ y $G(A,B,C)$, siendo por ejemplo, el valor de las funciones:

$$F = A + C'D$$

$$G = AB' + D$$

Se obtiene un circuito combinacional como este:

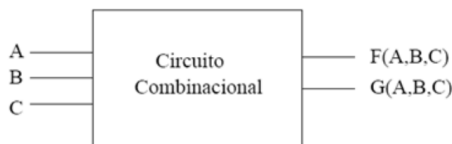


Fig 2.Circuito Combinacional

Circuito comparador

Un circuito comparador compara dos entradas binarias (A y B de n bits) para indicar la relación de igualdad o desigualdad entre ellas por medio de "tres banderas lógicas" que corresponden a las relaciones A igual B, A mayor que B y A menor que B. Cada una de estas banderas se activará solo cuando la relación a la que corresponde sea verdadera, es decir, su salida será 1 y las otras dos producirán una salida igual a cero[7].

Circuito votador Es conocido el uso de redundancia en aplicaciones electrónicas con altos requisitos de seguridad ante fallos. Las técnicas de redundancia implican utilizar una pluralidad de circuitos independientes para obtener un solo resultado. Cada uno de los circuitos genera una señal de salida, por lo que es necesario un sistema votador ("voter.^{en} inglés), que indica cual es el valor de salida más repetido. Dicho valor más repetido pasa a considerarse el resultado de salida del sistema de redundancia[7].

App Inventor

App Inventor es un entorno de desarrollo de software creado por Google para la elaboración de aplicaciones destinadas al sistema operativo de Android. El lenguaje es gratuito y se puede acceder fácilmente de la web. Las aplicaciones creadas con App Inventor están limitadas por su simplicidad, aunque permiten cubrir un gran número de necesidades básicas en un dispositivo móvil. Con App Inventor, se espera un incremento importante en el número de aplicaciones para Android debido a dos grandes factores: la simplicidad de uso, que

facilitará la aparición de un gran número de nuevas aplicaciones; y Google Play, el centro de distribución de aplicaciones para Android donde cualquier usuario puede distribuir sus creaciones libremente[6].

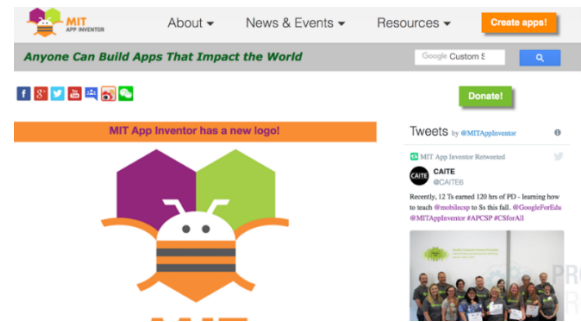


Fig 3.App Inventor

Display De 7 Segmentos

El display de 7 segmentos es un dispositivo electrónico que se utiliza para representar visualmente números y algunos caracteres. Este display es muy popular debido a su gran efectividad y simplicidad al momento de utilizarlo.

Partes de un display de 7 segmentos

Se le conoce como 7 segmentos por que cuenta con siete diodos led principales y uno extra para representar un punto. También cuenta con una carcasa para cubrirlos y 10 terminales: 2 son de alimentación (2 de Vcd o 2 de Gnd), 1 es para visualizar un punto y 7 son para representar cada uno de los números según la combinación que se le ponga, estos están representados por una letra del abecedario desde la "A" hasta la letra "G"[6].

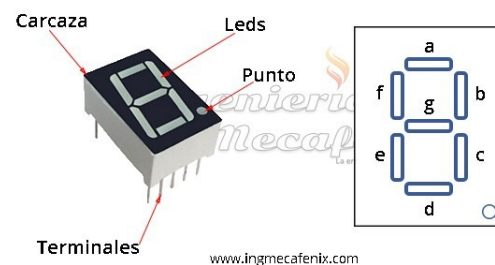


Fig 4.Display 7 segmentos

Funcionamiento de un display

Para poder representar los números o caracteres con este dispositivo solo basta con saber la configuración de cada una de sus leds y combinarlos.

IV. PROPUESTA

ENUNCIADO 1:

Las normas de seguridad de los modernos aviones exigen que, para señales de vital importancia para la seguridad del aparato, los circuitos deben estar triplicados para que el fallo de uno de ellos no produzca una catástrofe. En caso de que los tres circuitos no produzcan la misma salida, ésta se escogerá mediante votación. Diseñe el circuito "votador" que ha de utilizarse para obtener como resultado el valor mayoritario de las tres entradas. El proceso de votación consiste en tomar el valor mayoritario de las entradas. De esta forma, la salida, f, del circuito tendrá la siguiente codificación : Análisis

– f = 0 si hay más ceros que unos en las entradas

– f = 1 si hay más unos que ceros en las entradas

El circuito votador tiene tres señales de entrada : a, b y c, que son las salidas de los circuitos triplicados .

Tabla de verdad:

a	b	c	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Simplificación mediante el mapa de Karnaugh

ab/c	00	01	11	10
0	0	0	1	0
1	0	1	1	1

	\bar{c}	c
$\bar{a}\bar{b}$		
$\bar{a}b$		1
$a\bar{b}$	1	1
ab		1

$$f = A + B + C = ab + bc + ac$$

$$f = a \cdot b + c \cdot (a + b)$$

$$f = ab + ac + bc$$

ENUNCIADO 2:

El horario laboral de una factoría es de 8 horas diarias, divididas en tres turnos: de 8 a 11 (primer turno), de 11 a 13 (segundo turno), de 13 a 16 (descanso) y de 16 a 19 (tercer turno). Se pretende diseñar un circuito que tenga como entradas la representación binaria de la hora actual menos ocho y que proporcione a la salida el número de turno que está trabajando (si procede) o "0" si es hora de descanso. Diseñe un módulo adicional que permita visualizar en un display de 7 segmentos el número del turno que corresponda.

Análisis

Para implementar este circuito consideramos que cada turno se desarrolla en un lapso de tiempo específico considerando la hora laboral como: 8 a 9, 9 a 10, 10 a 11, 18 a 19 quedando un total de 11 horas posibles. En un primer análisis la HORA ACTUAL son números decimales del 8 al 18 que al momento de cumplir la condición impuesta por el enunciado (HORA ACTUAL - 8), queda un intervalo del 0 al 10, finalmente las ENTRADAS que se van a utilizar para el desarrollo del problema es la representación en binario del último intervalo (0 al 10). Ejemplo: Hora actual: 8
Hora actual - 8: 8 - 8 = 0

Representación binaria: 0 0 0 0

Hora actual: 9 Hora actual - 8: 9 - 8 = 1

Representación binaria: 0 0 0 1

Así sucesivamente hasta completar el intervalo, para la representación binaria del 10 se necesita una entrada de 4 bits

TURNOS	HORAS	Hora actual - 8	BINARIOS ENTRADAS				Número de turno		SALIDAS DISPLAY						
			A3	A2	A1	A0	salida 2	salida 1	a	b	c	d	e	f	g
1er. TURNO	8	0	0	0	0	0	0	1	0	1	1	0	0	0	0
	9	1	0	0	0	1	0	1	0	1	1	0	0	0	0
	10	2	0	0	1	0	0	1	0	1	1	0	0	0	0
	11	3	0	0	1	1	1	0	1	1	0	1	1	0	1
2do. TURNO	12	4	0	1	0	0	1	0	1	1	0	1	1	0	1
	13	5	0	1	0	1	0	0	0	1	1	1	1	1	0
	14	6	0	1	1	0	0	0	0	1	1	1	1	1	0
	15	7	0	1	1	1	0	0	0	1	1	1	1	1	0
DESCANSO	16	8	1	0	0	0	1	1	1	1	1	1	0	0	1
	17	9	1	0	0	1	1	1	1	1	1	1	0	0	1
	18	10	1	0	1	0	1	1	1	1	1	1	0	0	1

Al desarrollar la tabla de verdad la combinación de la salida 2 y salida 1 arrojan en binario lo siguiente ("1" encendido y "0" apagado):

Primer turno: 01

Segundo turno: 10

Tercer turno: 11

Descanso: 00

Descanso: 00

• PARA SALIDA 2

min-Términos (3, 4, 8, 9, 10)

$$\text{salida 2} = F(a_0, a_1, a_2, a_3)$$

salida 2 =

$$\begin{aligned} &= \bar{a}_3 \cdot \bar{a}_2 \cdot a_1 \cdot a_0 + \bar{a}_3 \cdot a_2 \cdot \bar{a}_1 \cdot \bar{a}_0 + a_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot \bar{a}_0 + a_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot a_0 + a_3 \cdot \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 \\ &= \bar{a}_3 \cdot \bar{a}_2 \cdot a_1 \cdot a_0 + \bar{a}_3 \cdot a_2 \cdot \bar{a}_1 \cdot \bar{a}_0 + a_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot (\bar{a}_0 + a_0) + a_3 \cdot \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 \\ &= \bar{a}_3 \cdot \bar{a}_2 \cdot a_1 \cdot a_0 + \bar{a}_3 \cdot a_2 \cdot \bar{a}_1 \cdot \bar{a}_0 + a_3 \cdot \bar{a}_2 \cdot \bar{a}_1 + a_3 \cdot \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 \\ &= a_3 \cdot (\bar{a}_2 \cdot \bar{a}_1 + \bar{a}_2 \cdot a_1 \cdot \bar{a}_0) + \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot a_0 + \bar{a}_3 \cdot a_2 \cdot \bar{a}_1 \cdot \bar{a}_0 \\ &= a_3 \cdot [\bar{a}_2 \cdot (\bar{a}_1 + a_1 \cdot \bar{a}_0)] + \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot a_0 + \bar{a}_3 \cdot a_2 \cdot \bar{a}_1 \cdot \bar{a}_0 \\ &= a_3 \cdot [\bar{a}_2 \cdot (\bar{a}_1 + a_1)] + \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot a_0 + \bar{a}_3 \cdot a_2 \cdot \bar{a}_1 \cdot \bar{a}_0 \\ &= a_3 \cdot (\bar{a}_2 \cdot \bar{a}_1 + \bar{a}_2 \cdot a_1 \cdot \bar{a}_0) + \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot a_0 + \bar{a}_3 \cdot a_2 \cdot \bar{a}_1 \cdot \bar{a}_0 \\ &= \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 + \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_0 + \bar{a}_3 \cdot \bar{a}_2 \cdot a_1 \cdot a_0 + \bar{a}_3 \cdot a_2 \cdot \bar{a}_1 \cdot \bar{a}_0 \end{aligned}$$

• PARA SALIDA 1

min-Términos (0, 1, 2, 8, 9, 10)

$$\text{salida 1} = F(a_0, a_1, a_2, a_3)$$

salida 1 =

$$\begin{aligned} &= \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot \bar{a}_0 + \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot a_0 + \bar{a}_3 \cdot \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 + a_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot \bar{a}_0 + a_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot a_0 + a_3 \cdot \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 \\ &= \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot (\bar{a}_0 + a_0) + \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 \cdot (\bar{a}_3 + a_3) + a_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \cdot (\bar{a}_0 + a_0) \\ &= \bar{a}_3 \cdot \bar{a}_2 \cdot \bar{a}_1 + \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 + a_3 \cdot \bar{a}_2 \cdot \bar{a}_1 \\ &= \bar{a}_2 \cdot \bar{a}_1 \cdot (\bar{a}_3 + a_3) + \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 \\ &= \bar{a}_2 \cdot \bar{a}_1 + \bar{a}_2 \cdot a_1 \cdot \bar{a}_0 \\ &= \bar{a}_2 \cdot (\bar{a}_1 + a_1 \cdot \bar{a}_0) \\ &= \bar{a}_2 \cdot ((\bar{a}_1 + a_1) \cdot (\bar{a}_1 + \bar{a}_0)) \\ &= \bar{a}_2 \cdot (\bar{a}_1 + \bar{a}_0) \end{aligned}$$

ENUNCIADO 3:

Se pretende diseñar un circuito comparador de 2 números de 2 bits, A = (a1, a0) y B = (b1, b0). Dicho circuito deberá tener tres salidas M, l, m, de tal forma que: Diseñe exclusivamente con puertas NOR.

Análisis:

$$M = 1 \text{ si } A \leq B$$

$$L = 1 \text{ si } A = B$$

$$m = 1 \text{ si } A \geq B$$

Entradas				Salidas		
A		B		A>B	A=B	A<B
a1	a0	b1	b0	M	L	m
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

Diseño en puertas NOR

- $M = a0 \overline{b1} \overline{b0} + a1 a0 \overline{b0} + a1 \overline{b1}$

$$M = \overline{a0 \overline{b1} \overline{b0}} + \overline{a1 a0 \overline{b0}} + \overline{a1 \overline{b1}}$$

$$M = \overline{a0 + b1 + b0} + \overline{a1 + a0 + b0} + \overline{a1 + b1}$$

- $L = \overline{a1 a0 \overline{b1} \overline{b0}} + \overline{a1 a0 \overline{b1} b0} + \overline{a1 a0 b1 \overline{b0}} + \overline{a1 a0 b1 b0}$

$$L = \overline{a1 a0 \overline{b1} \overline{b0}} + \overline{a1 a0 \overline{b1} b0} + \overline{a1 a0 b1 \overline{b0}} + \overline{a1 a0 b1 b0}$$

$$L = a1 + a0 + b1 + b0 + a1 + a0 + b1 + b0 + \overline{a1 + a0 + b1 + b0} + \overline{a1 + a0 + b1 + b0}$$

- $m = \overline{a1 b1} + \overline{a1 a0 b0} + \overline{a0 b1 b0}$

$$m = \overline{a1 b1} + \overline{a1 a0 b0} + \overline{a0 b1 b0}$$

$$m = \overline{a1 + b1} + \overline{a1 + a0 + b0} + \overline{a0 + b1 + b0}$$

V. RESULTADOS

Funciones resultantes aplicando min términos.

- $M = 0100 + 1000 + 1001 + 1100 + 1101 + 1110$

$$M = \overline{a1} a0 \overline{b1} \overline{b0} + a1 \overline{a0} \overline{b1} \overline{b0} + a1 a0 \overline{b1} \overline{b0} + a1 a0 \overline{b1} b0 + a1 a0 b1 \overline{b0} + a1 a0 b1 b0$$

- $L = 0000 + 0101 + 1010 + 1111$

$$L = \overline{a1} \overline{a0} \overline{b1} \overline{b0} + \overline{a1} a0 \overline{b1} b0 + a1 \overline{a0} \overline{b1} \overline{b0} + a1 a0 b1 b0$$

- $m = 0001 + 0010 + 0011 + 0110 + 0111 + 1011$

$$m = \overline{a1} a0 \overline{b1} b0 + \overline{a1} \overline{a0} b1 \overline{b0} + \overline{a1} a0 b1 \overline{b0} + \overline{a1} a0 b1 b0 + a1 \overline{a0} \overline{b1} \overline{b0} + a1 \overline{a0} b1 \overline{b0}$$

Simplificación de términos mediante el Mapas de Karnaugh

$$M = \overline{a1} a0 \overline{b1} \overline{b0} + a1 \overline{a0} \overline{b1} \overline{b0} + a1 a0 \overline{b1} \overline{b0} + a1 a0 \overline{b1} b0 + a1 a0 b1 \overline{b0} + a1 a0 b1 b0$$

$\begin{matrix} b1 \ b0 \\ a1 \ a0 \end{matrix}$	00	01	11	10
00				
01	1			
11	1	1		1
10	1	1		

$$M = a0 \overline{b1} \overline{b0} + a1 a0 \overline{b0} + a1 \overline{b1}$$

$$L = \overline{a1} \overline{a0} \overline{b1} \overline{b0} + \overline{a1} a0 \overline{b1} b0 + a1 \overline{a0} \overline{b1} \overline{b0} + a1 a0 b1 b0$$

$\begin{matrix} b1 \ b0 \\ a1 \ a0 \end{matrix}$	00	01	11	10
00	1			
01		1		
11			1	
10				1

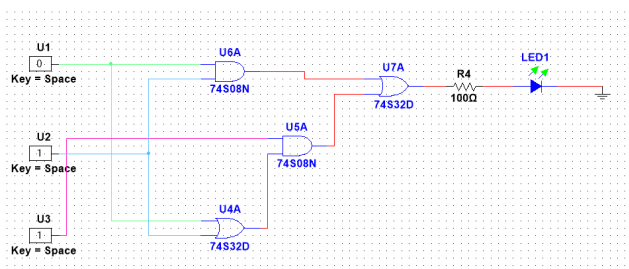
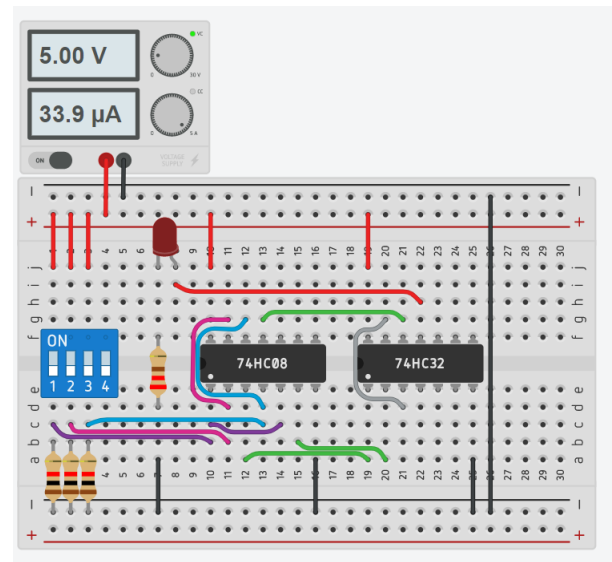
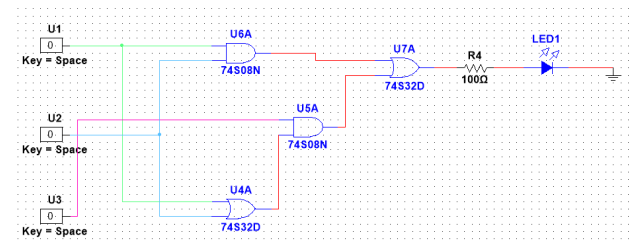
$$L = \overline{a1} \overline{a0} \overline{b1} \overline{b0} + \overline{a1} a0 \overline{b1} b0 + a1 \overline{a0} \overline{b1} \overline{b0} + a1 a0 b1 b0$$

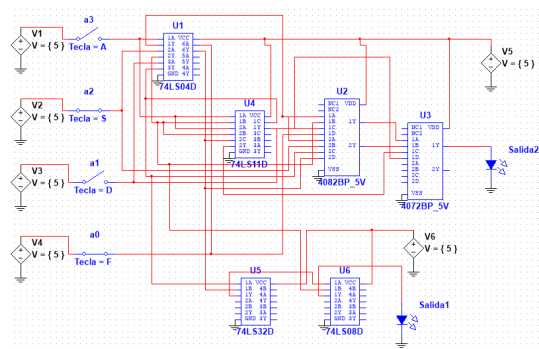
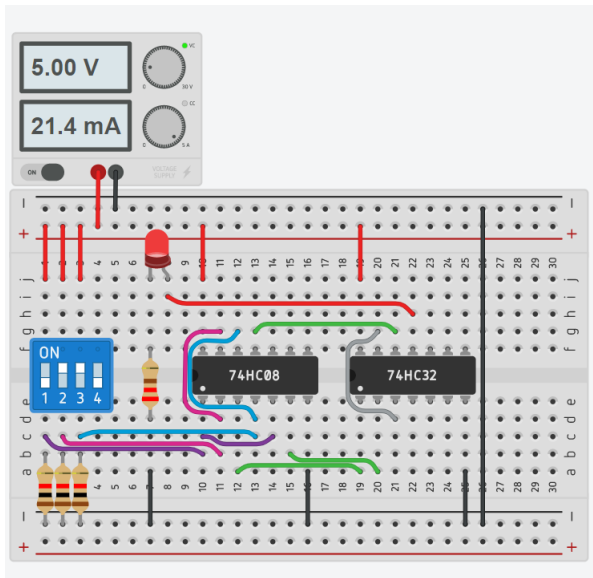
$$m = \overline{a1} \overline{a0} \overline{b1} b0 + \overline{a1} \overline{a0} b1 \overline{b0} + \overline{a1} \overline{a0} b1 b0 + \overline{a1} a0 \overline{b1} \overline{b0} + \overline{a1} a0 b1 b0 + a1 \overline{a0} \overline{b1} \overline{b0}$$

$\begin{matrix} b1 \ b0 \\ a1 \ a0 \end{matrix}$	00	01	11	10
00		1	1	1
01			1	1
11				
10				1

$$m = \overline{a1} b1 + \overline{a1} \overline{a0} b0 + \overline{a0} b1 b0$$

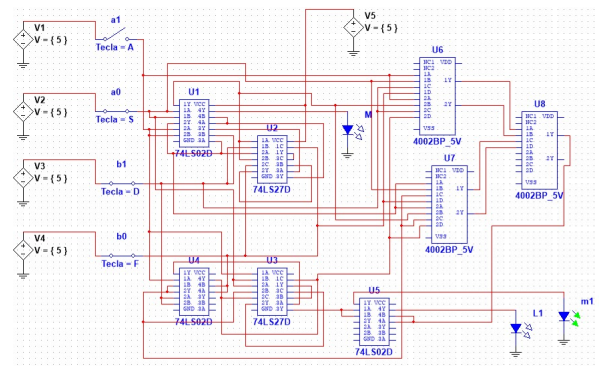
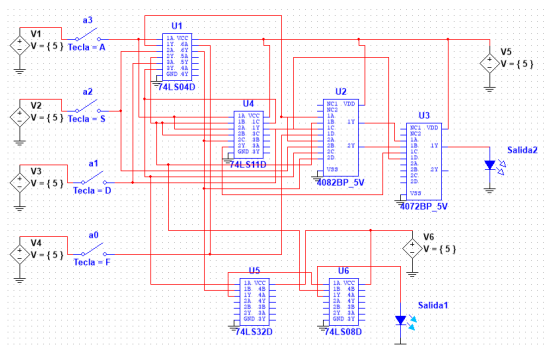
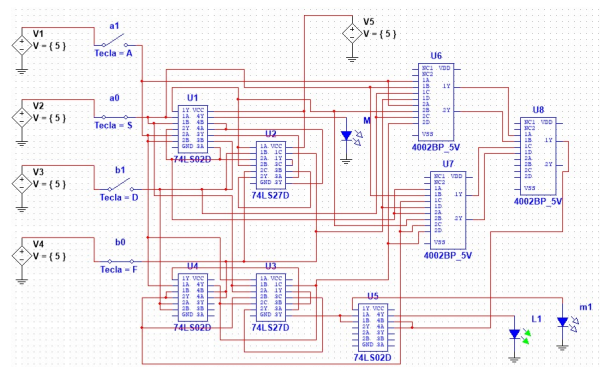
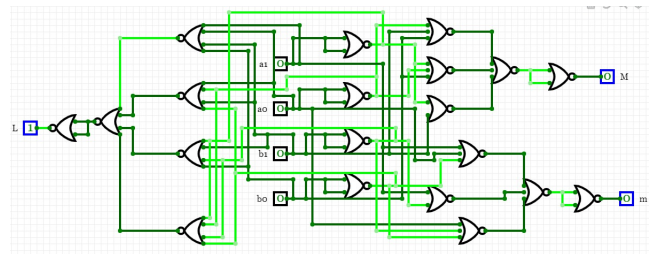
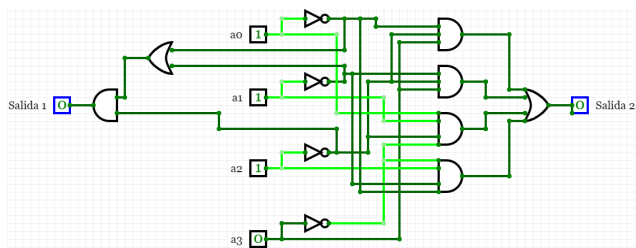
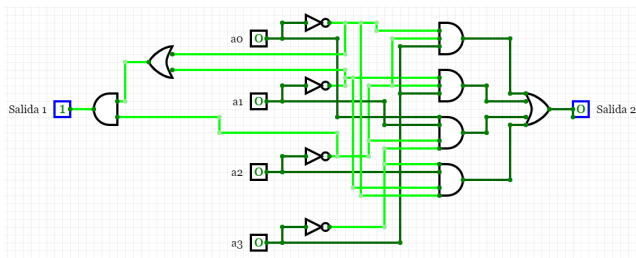
Enunciado 1:

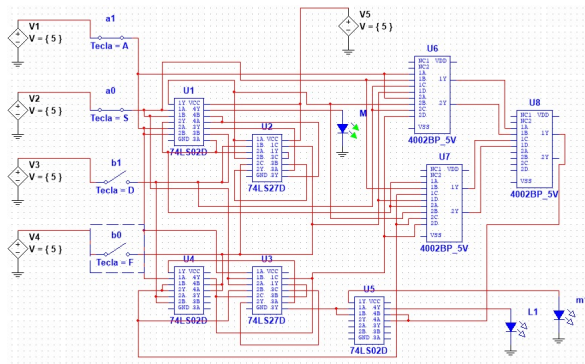




Enunciado 3:

Enunciado 2:





VI. CONCLUSIONES

En conclusión:

- El circuito diseñado utilizado como norma de seguridad de aviones modernos tiene tres señales de entrada que son las salidas triplicadas así si un circuito llegara a fallar quedarían dos en buen estado, evitándose una catástrofe.
- Dado un problema que conlleva realizar una o varias acciones se lo puede resolver fragmentándolo en una o varias funciones lógicas implementadas en un circuito combinacional, estas funciones por lo general son expresiones extensas según el problema o el número de entradas o salidas que se necesite, a estas expresiones se las pueden simplificar en una expresión más trabajada por medio de aplicar los teoremas del Álgebra Booleana o los Mapas de Karnaugh, siendo este último el más óptimo para simplificar expresiones extensas, basado en la resolución del ejercicio número 2 se puede llegar a una función simplificada solo aplicando los teoremas del Álgebra Booleana.
- Se comprendió con claridad el funcionamiento de las compuertas lógicas y su implementación en los diferentes integrados. También se analizó las diferencias al simplificar los términos de forma manual y mediante los mapas de Karnaugh, observando que al usar los mapas de Karnaugh, es mucho más efectivo y rápido de implementar. En general sirvió para comprender el funcionamiento básicos de los aparatos electrónicos, como también para manejar de mejor manera el Álgebra de Boole.

REFERENCIAS

- [1] Gan, Z., Shi, G. y Shang, T. (2009). La síntesis automática de circuitos lógicos combinacionales con algoritmo de selección clonal basado en gráficos. 2009 Conferencia Internacional sobre Inteligencia Artificial e Inteligencia Computacional. doi: 10.1109 / aici.2009.267
- [2] Alidousti, Z. y Basiri, ME (2017). CGACLC: Mejora del algoritmo genético a través de la agrupación para el diseño de circuitos lógicos combinacionales. 2017 Tercera Conferencia Internacional sobre Reconocimiento de Patrones y Análisis de Imágenes (IPRIA). doi: 10.1109 / pria.2017.7983062
- [3] Mukherjee, B. y Dandapat, AK (2010). Diseño de circuitos combinacionales mediante el método combinacional cíclico para VLSI de baja potencia. Simposio internacional 2010 sobre diseño de sistemas electrónicos. doi: 10.1109 / ised.2010.29
- [4] Xu, Y., Ouyang, D., Ye, Y. y He, J. (2010). Solución de problemas SAT con álgebra booleana. Quinta Conferencia Internacional 2010 sobre la frontera de la informática y la tecnología. doi: 10.1109 / fcst.2010.1
- [5] Huang, R. y Turbak, F. (2019). Un diseño para la conversión bidireccional entre bloques y texto para App Inventor. 2019 IEEE Blocks and Beyond Workshop (BB). doi: 10.1109 / bb48857.2019.8941197
- [6] Gonzales,Erick,(2018). Display 7 Segmentos ánodo y cátodo común.Recuperado de: <https://hetpro-store.com/TUTORIALES/display-7-segmentos-anodo-catodo-comun/>
- [7] Posada.Fernando,(2019). Creando aplicaciones para móviles Android con MIT App Inventor 2. DOI (web) 104438/2695-4176_OTE2019847 - 19 - 121 - 5.[https : //intef.es/wp - content/uploads/2019/03/MIT - App - Inventor - 2.pdf](https://intef.es/wp-content/uploads/2019/03/MIT-App-Inventor-2.pdf)