

Universidad Nacional de México Autónoma

Facultad de Estudios Superiores Aragón

Materia:
Laboratorio de Diseño Lógico

Núm de práctica:
2 Reducción de circuitos

Alumnos:

Soto Franco Diego Rodrigo

Grupo:
8460



REDUCCIÓN DE CIRCUITOS

OBJETIVO

El objetivo de la práctica es reducir y simplificar el diseño de un circuito lógico que controle una lámpara ahorradora, utilizando álgebra de Boole y Mapas de Karnaugh.

MATERIAL Y EQUIPO

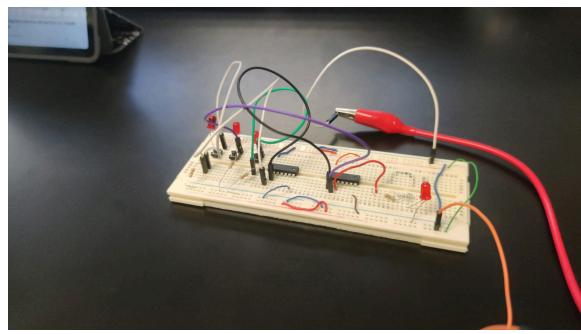
- Fuente de poder.
- Caimanes.
- Protoboard.
- 2 Jumper macho-macho.
- Alambre.
- Circuitos integrados 7432, 7408, 7404
- 4 diodos emisores de luz (led),
- 4 Resistencias 330 ohms a 1/2 watt
- 3 Push bottom

DESARROLLO

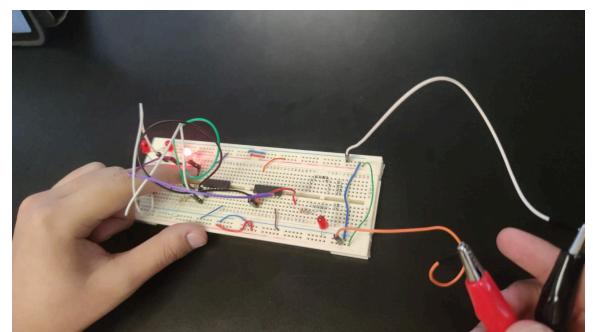
1) Arma el circuito de la figura 2.2.

Empezamos a armar nuestro circuito de la figura 2.2 alambrando el circuito ya que se parecía un poco a los circuitos de la práctica 1, analizamos la tabla de verdad para compararla con los resultados en el circuito.

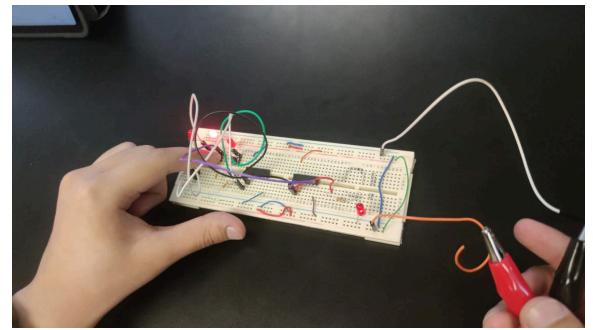
Todas en 0



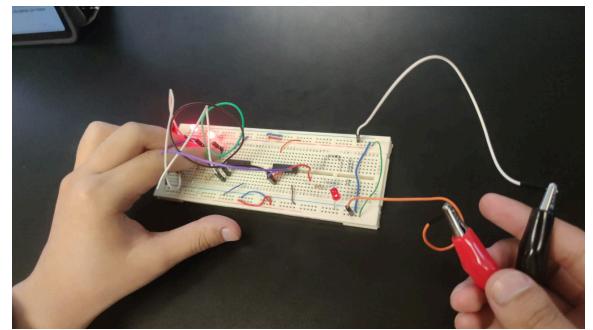
001



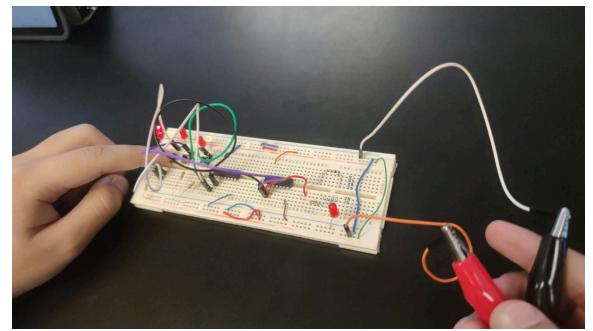
010



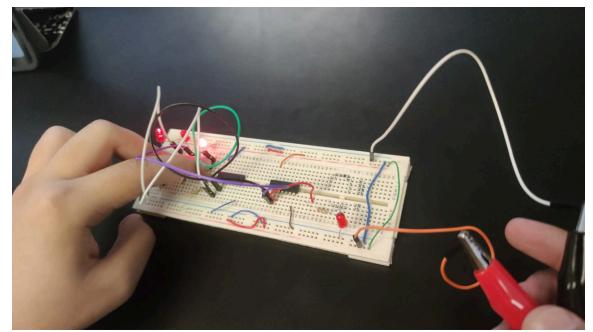
011



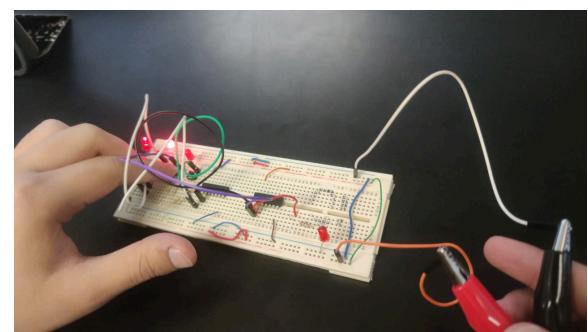
100



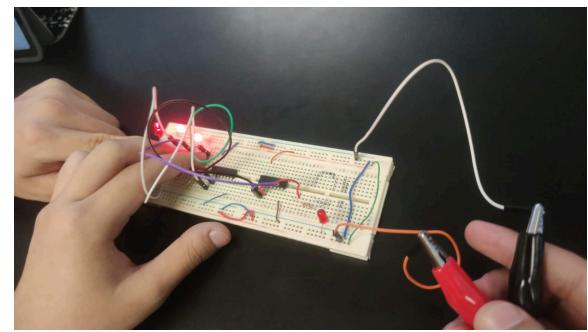
101



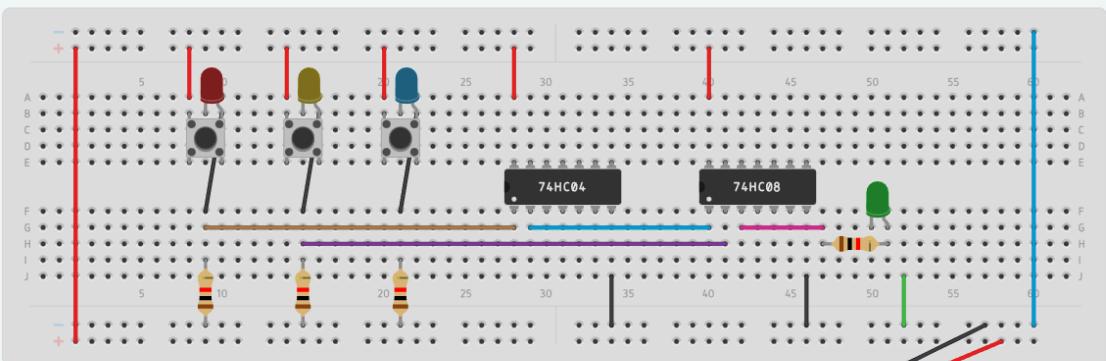
110



111



Simulacion



El circuito está diseñado para ilustrar la operación de compuertas lógicas y cómo las combinaciones de entradas (pulsadores) pueden influir en el resultado de una operación lógica (encender o apagar un LED).

2) Leer y entender el siguiente enunciado, realizar su tabla de verdad, simplificar con ayuda de álgebra de Boole, simular con compuertas lógicas.

- El circuito debe controlar una lámpara ahorradora con las siguientes condiciones:
 - **Encendido Manual:** La lámpara puede encenderse manualmente mediante un interruptor (contacto).
 - **Encendido Automático:** La lámpara se enciende automáticamente si:
 1. **Sensor de Movimiento** está activado.
 2. **Contador de Hora** indica que es hora de encenderla.

Variables:

- **M:** Estado del encendido manual (1: encendido, 0: apagado).
- **S:** Estado del sensor de movimiento (1: detecta movimiento, 0: no detecta movimiento).
- **C:** Estado del contador de hora (1: es la hora de encender la lámpara, 0: no es la hora).

Salida:

- **L:** Estado de la lámpara (1: encendida, 0: apagada).

2. Tabla de Verdad

M	S	C	S * C	$L = M + (S * C)$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	1

1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

3. Simplificación con Álgebra de Boole

La función booleana que representa la salida es: $L=M+(S \cdot C)L = M + (S \cdot C)L=M+(S \cdot C)$

Esto significa que la lámpara se encenderá ($L = 1$) si:

- Se enciende manualmente ($M = 1$), independientemente de los valores de S y C .
- O si tanto el sensor de movimiento ($S = 1$) como el contador de hora ($C = 1$) están activados.

4. Simulación con Compuertas Lógicas

Diseño del Circuito:

- Usar una **compuerta AND** para combinar las entradas S y C .
- Usar una **compuerta OR** para combinar la salida de la compuerta AND con M .

El circuito sería:

1. **Compuerta AND:** $A=S \cdot C$
2. **Compuerta OR:** $L=M+A$

Diagrama:

- Conectar la entrada M directamente a una de las entradas de la compuerta OR.
- Conectar las entradas S y C a la compuerta AND, y su salida a la otra entrada de la compuerta OR.

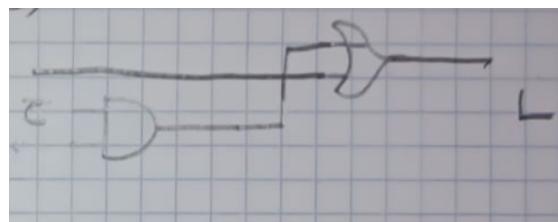
5. Simplificación con Mapas de Karnaugh

El siguiente paso es simplificar la función utilizando Mapas de Karnaugh.

Mapa de Karnaugh para la función:

S / M	0	1
00	0	1
01	0	1
11	1	1
10	0	1

$$S = M + S \underline{M} C$$



Agrupando los 1's en el Mapa de Karnaugh:

- Grupo de 4: (Cualquier combinación de M con S = 1 y C = 1) $\rightarrow S \cdot CS \cdotdot CS \cdot C$
- Grupo de 2: (M = 1) $\rightarrow MM$

$$\text{Función simplificada: } L = M + S \cdot CL = M + S \cdotdot CL = M + S \cdot C$$

Simulación Final:

- El diseño simplificado del circuito es idéntico al obtenido mediante álgebra de Boole, por lo que se mantiene la misma implementación con compuertas lógicas.

CONCLUSIONES

En esta práctica, logramos simplificar la función booleana que controla el encendido de la lámpara utilizando álgebra de Boole y Mapas de Karnaugh. La simplificación obtenida es equivalente a la función original, lo que nos permitió diseñar un circuito más simple con compuertas AND y OR, optimizando su implementación sin perder funcionalidad. Esto demuestra la utilidad de la simplificación de circuitos en el diseño lógico.