

Instituto Tecnológico de Costa Rica

Escuela de Ingeniería en Computadores



Fundamentos de arquitectura de computadores

Proyecto

Lógica combinatoria: Calculadora tomógrafo

Estudiante

Carné

Rodríguez Rojas Andrés

2019279722

Soto Varela Óscar

2020092336

Profesor:

Luis Alberto Chavarría Zamora

10 de abril de 2025

Proyecto Individual

Lógica Combinatoria: Calculadora tomógrafo

Fecha de asignación: 11 marzo 2025
Grupo: 1-2 personas

Fecha de entrega: 10 abril 2025
Profesores: Luis Chavarría Zamora

1. Descripción

En este punto el estudiante ya conoce los aspectos teóricos y prácticos preliminares de los circuitos lógicos digitales combinatorios como compuertas y sus diversas técnicas de diseño. Con este, la persona estudiante pondrá en práctica su perspicacia en diseño combinatorio tomando un sensor, mostrando el resultado al usuario y ejecutando un actuador.

2. Especificación

El circuito estará representado por tres etapas en la Figura 1. Este podrá usar fuentes de tensión o baterías como alimentación.

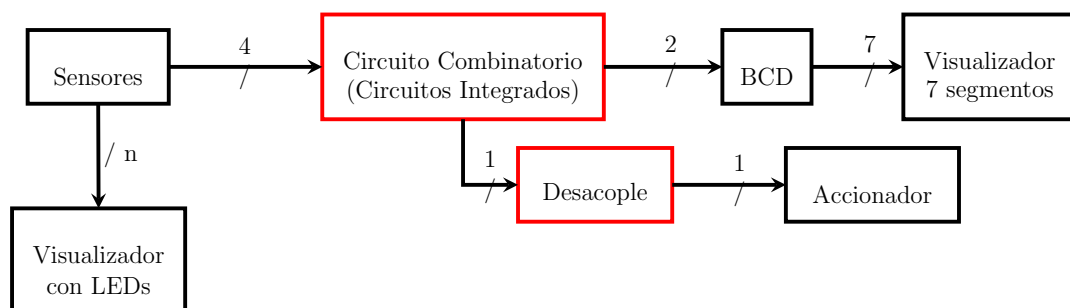


Figura 1: Diagrama de bloques del proyecto individual. Las etapas digitales diseñadas por el estudiante funcionarán con la disciplina estática de 0V a 5V

Las etapas se describen a continuación:

- **Sensores:** Estará constituido por un arreglo de sensores propuestos por el estudiante (al menos cuatro bits) que tengan alguna interacción humana no eléctrica. **Entre este módulo y el circuito combinatorio no hay ninguna interfaz adicional.** Este rubro requiere que no sean ni botones ni switches.
- **Visualizador con LED:** En este visualizador muestra de forma gráfica el comportamiento mediante n leds propuesto por los estudiantes.

- **Circuito Combinatorio:** Esta etapa estará conformada por únicamente circuitos integrados. Esta etapa requiere poner en práctica los conceptos teóricos del curso. Se observa que se tienen al menos 4 bits. Esta etapa tendrá dos submódulos:
 1. Decodificador 1: Es un módulo que tomará la cantidad binaria y comenzará a acumular el valor en código binario de forma circular de 0 a 3, sumando el acumulado (no use sumador, solo el decodificador). Esto será enviado al BCD. **Tome en cuenta que para el acumulado se debe usar un registro que se activa con un botón (será el único elemento secuencial).**
 2. Decodificador 2: Es un módulo que se encargará de habilitar un actuador en dos rangos de valores lógicos (por ejemplo que se active en el intervalo de 1 en decimal a 2 decimal y que luego se reactive en el rango de 0 y 3 en decimal, los intervalos de activación no deben ser contiguos y deben especificarse claramente).
- **Desacople:** Esta etapa tiene el propósito de desacoplar eléctricamente el **Circuito Combinatorio** y el Accionador. Esta etapa debe ser construida con componentes discretos, no únicamente mecánicos o magnéticos.
- Visualizador: Está conformado por un display de siete segmentos, que recibirá 7 bits. Estos deben representar los valores en hexadecimal.
- Accionador: El accionador será un motor de CD, que requiere una alimentación por separado al resto del circuito.
- Visualizador 7 segmentos: El visualizador debe mostrar el resultado actual en el display de 7 segmentos.

3. Metodología de trabajo

El estudiante deberá seguir los siguientes lineamientos:

1. Los circuitos deben ser diseñados solamente con compuertas CMOS o solamente con compuertas TTL (**¿por qué no es conveniente mezclarlos?, colocar en el documento cómo se conectan si se usarán con dos tecnologías diferentes**).
2. Los estudiantes pueden usar herramientas Electronic design automation (EDA) o de inteligencia artificial para simplificación. Debe documentar los procedimientos automatizados y los mismos deben ser validados por usted. **En este proyecto usted será un diseñador(a), no un ejecutador(a) sin criterio técnico crítico. Esto debe formar parte de conocimiento de ingeniería.**
3. Los circuitos deben ser simulados antes de ser implementados físicamente para que el estudiante tenga una referencia para comparar.

4. Evaluación y entregable

Se evaluarán los siguiente entregables:

1. Presentación proyecto 100 % funcional (70 %): Una defensa de 15 minutos donde el profesor evaluará el sistema.
2. Documentación (30 %): Esta se encuentra conformada por los siguientes documentos:
 - a) Artículo científico tipo *paper* (20 %): El paper a realizar deberá tener una extensión no mayor a 4 páginas completas (incluyendo bibliografías), deberá ser realizado con L^AT_EX, siguiendo un formato establecido (IEEE Transactions o ACM, por ejemplo). Debe ser redactado en tercera persona singular. Se les provee un ejemplo de paper en el [enlace](#). En general el *paper* deberá contar con las siguientes secciones:
 - 1) Abstract (en inglés): Un buen abstract tiene las siguientes características:
 - a' Un abstract permite a los lectores obtener la esencia o esencia de su artículo o artículo rápidamente, para decidir si leer el artículo completo.
 - b' Un abstract prepara a los lectores para seguir la información detallada, los análisis y los argumentos en su artículo completo.
 - c' Un abstract ayuda a los lectores a recordar puntos clave de su paper.
 - d' Un abstract es de entre 150 y 250 palabras.
 - 2) Palabras clave significativas (a lo sumo 6).
 - 3) Introducción: Una buena introducción muestra el contexto del problema o lo que se va a solucionar, introduce el tema al lector. Al final de la introducción se indica la organización del documento (primero se muestra el algoritmo, luego....).
 - 4) Algoritmo desarrollado.
 - 5) Resultados.
 - 6) Conclusiones escritas en prosa.
 - 7) Recomendaciones escritas en prosa.
 - 8) Bibliografía, en formato IEEE y referenciadas en el texto (usar cite). Referencia bien para evitar problemas de plagio. Una documento no referenciado en el texto no existe.
 - b) Bitácora (Atributo evaluado: Conocimiento de Ingeniería) (10 %): Es un documento donde el estudiante indica detalladamente los procedimientos que implementó en el diseño, tablas, ecuaciones booleanas, herramientas de automatización utilizadas y otras relacionadas. Debe existir una división clara de días en que se realizó. El estudiante debe desarrollarlo en PDF e irlo actualizando en un Git (**se revisará el historial de los commits, habrá penalidad si realiza pocos commits**).

Los documentos deben ser desarrollados \LaTeX , entregando el **PDF**. Se prohíbe el uso de referencias hacia sitios no confiables.

Si tienen dudas puede escribir al profesor al [correo electrónico de Luis Chavarría](#). **Los documentos serán sometidos a control de plagios.** La entrega se debe realizar a más tardar las 11:59 pm mediante TEC-Digital la fecha de entrega, cualquier entrega tardía después de este punto recibirá una penalización de un punto por minuto en la documentación.

Abstract—This project consists in the designing and creation of a 3 bits circular adder which inputs are given by 4 photoresistors and its calculus is made entirely through logical gates. Plus each input will be hold by a flip-flop and not processed until a button signal is given. Then, the result is sent to a BCD which shows the result in a 7-segment display. Furthermore in 2 specific results will also activate a dc motor.

Index Terms—BCD, Digital design, Logical Gates, combinatorial logic

I. INTRODUCCIÓN

En este documento se explica a detalle el proceso de diseño y ensamblaje de un circuito lógico digital con el cual se logra solventar el problema planteado, el cual consiste en un circuito combinatorio con disciplina estática de 5 V, capaz de procesar 4 bits de entrada dados por sensores como si fuera una calculadora circular de 2 bits, cuyos resultados se enviarán y mostrarán en un visualizador de 7 segmentos y será capaz de accionar un motor de corriente continua según determinados resultados. La solución implementa únicamente compuertas lógicas, componentes básicos y un BCD para controlar el 7 segmentos, no se permite el uso de microprocesadores. En la figura 1 se puede observar un diagrama base del circuito a construir.

Para el diseño de la solución se busca implementar diferentes herramientas como mapas de Karnaugh tablas de verdad y de ser necesario herramientas digitales para alcanzar la mayor simplificación y optimización del circuito final.

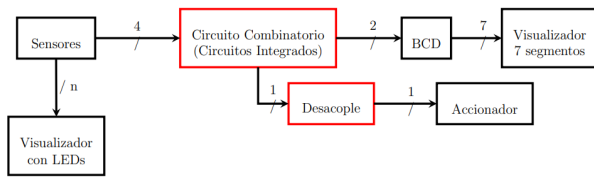


Fig. 1: Diagrama base del circuito planteado para diseñar

II. ALGORITMO DESARROLLADO

A. Sensores y visualizadores LED

Para este apartado se logró mediante la implementación de 4 fotoresistores como sensores. Su funcionamiento consiste en que al dejar de recibir alguna luz el fotoresistor activa un transistor conectado en una configuración de tipo pull

up, causando que se ilumine el LED y se envíe la señal a hacia el circuito combinatorio.

Para lograr el correcto funcionamiento de este circuito se realizaron diferentes simulaciones y cálculos de divisor tensión para determinar cuáles resistencias permitían obtener el resultado deseado con los fotoresistores que se disponían que tenían un valor de $10\text{ k}\Omega$. En la figura 2 puede observar la simulación de señal alta y baja respectivamente.

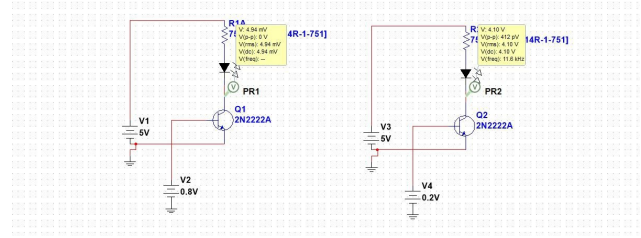


Fig. 2: Simulación de los sensores y LED's para señales altas y bajas respectivamente

En la figura 3 se puede observar el diagrama del circuito para que se utilizó para cada uno de los 4 sensores del circuito. Además, en la ecuación (1) con la que se calculó el mejor valor para la resistencia R2 mediante un divisor de tensión.

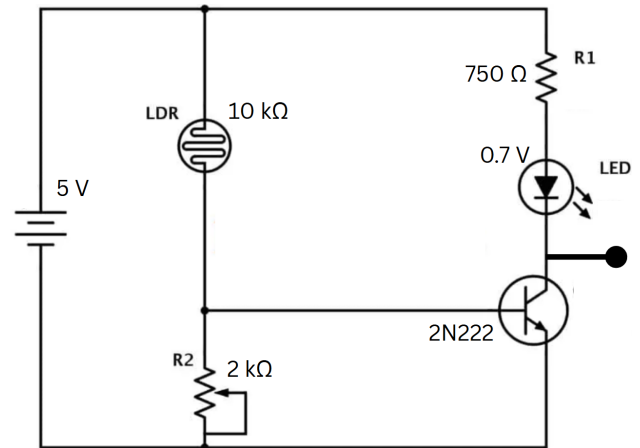


Fig. 3: Diagrama de circuito de sensores y LED's

$$0.8\text{ V} = \frac{5\text{ V}}{10\text{ V} + R2} \quad (1)$$

$$R2 = 2\text{ k}\Omega$$

B. Circuito combinatorio: encodificador

Para la primera etapa del circuito combinatorio se diseñó un encodificador [4:2] dado que

este pesar de tener 4 bits de entrada, solo tendría 4 combinaciones de entrada válidos, por lo que se empezó por definir los valores de entrada válidos tal que estos sería 1000 para 0, 1100 para 1, 1110 para 2 y 1111 para 3. Resultando en el cuadro I.

A	B	C	D	S_0	S_1
1	0	0	0	0	1
1	1	0	0	1	0
1	1	1	0	1	1
1	1	1	1	0	0

TABLE I: Encodificador [4:2]

Mediante la técnica de mapas de Karnaugh se desarrollaron los cuadros II y III para obtener las compuertas lógicas necesarias para calcular S_0 y S_1 :

CD\AB	00	01	11	10
00	X	X	1	0
01	X	X	X	X
11	X	X	0	X
10	X	X	1	X

TABLE II: Mapa de Karnaugh para S_0

A partir del cuadro II se obtuvo el resultado inicial para $S_0 = (\bar{A}) + \bar{D}) \cdot (\bar{A} + C)$, que sería simplificado hasta el resultado final 2:

$$S_0 = B\bar{D} \quad (2)$$

CD\AB	00	01	11	10
00	X	X	0	1
01	X	X	X	X
11	X	X	0	X
10	X	X	1	X

TABLE III: Mapa de Karnaugh para S_1

A partir del cuadro III se obtuvo el resultado inicial para $S_1 = \bar{A} + A\bar{B} + C\bar{D}$, que sería simplificado hasta el resultado final 3:

$$S_1 = \bar{D} \cdot (C + \bar{B}) \quad (3)$$

C. Circuito combinatorio: decodificador

Una vez listo el encodificador, se procedió a realizar el diseño del decodificador se planteó la siguiente tabla inicial IV:

S_0	S_1	S'_0	S'_1	Y_0	Y_1
0	1	0	1	1	0
1	0	0	1	1	1
1	1	0	1	0	0
0	0	0	1	0	1
0	1	1	0	1	1
1	0	1	0	0	0
1	1	1	0	0	1
0	0	1	0	1	0
0	1	1	1	0	0
1	0	1	1	0	1
1	1	1	1	1	0
0	0	1	1	1	1
0	1	0	0	0	1
1	0	0	0	1	0
1	1	0	0	1	1
0	0	0	0	0	0

TABLE IV: Decodificador [2:2]

A partir del cuadro IV se desarrollaron los cuadros para obtener las compuertas lógicas necesarias para calcular Y_0 y Y_1 :

$S'_0 S'_1 \setminus S_0 S_1$	00	01	11	10
00	0	0	1	1
01	0	1	0	1
11	1	0	1	0
10	1	1	0	0

TABLE V: Mapa de Karnaugh para Y_0

A partir del cuadro V se obtuvo el resultado inicial para $Y_0 = S_0 \bar{S}_1 \bar{S}'_0 + \bar{S}'_0 \bar{S}'_1 S'_0 + S_0 S_1 \bar{S}'_0 \bar{S}'_1 +$

$\bar{S}_0 S_1 \bar{S}_0 \bar{S}_1' + S_0 S_1 S_0' S_1' + \bar{S}_0 S_1 S_0' \bar{S}_1'$, dada la complejidad de este resultado, no podía ser simplificado a su máximo mediante las técnicas vistas hasta ahora en clase, por lo que se recurrió a el uso de calculadoras y otras herramientas online para llegar al resultado 4:

$$Y_0 = (S_0 \oplus S_1') \oplus S_0 S_1' \quad (4)$$

$S_0' S_1' \setminus S_0 S_1$	00	01	11	10
00	0	1	1	0
01	1	0	0	1
11	1	0	0	1
10	0	1	1	0

TABLE VI: Mapa de Karnaugh para Y_1

A partir del cuadro VI se obtuvo el resultado inicial para $Y_1 = (\bar{S}_1 + \bar{S}_1') \cdot (S_0 + S_1 + S_1') \cdot (\bar{S}_0 + S_1 + S_1')$, que sería simplificado hasta el resultado final 5:

$$Y_1 = S_1 \oplus S_1' \quad (5)$$

D. Diseño del decodificador del desacople del motor

Para el diseño del decodificador del desacople del motor se llevó a cabo un análisis con el cual se concluyó que este se podía reducir a prácticamente solo una compuerta XOR, resultando que el cálculo de este se representaría según; $S_0' \oplus S_1'$.

E. Funcionamiento del “flip flop” tipo D

Para el flip flop se llevaron a cabo simulaciones que concluyeron que el módulo SN74LS174 era el más adecuado para acoplar al circuito y lógica combinacional desarrollada. La función del flip flop en el circuito consiste en “sostener” la señal enviada por el encodificador hasta que se reciba la señal de clock que en este caso se simula mediante la activación de un botón común activado cuando una persona lo presiona.

F. Funcionamiento del decodificador “BCD” y la pantalla de siete segmentos

Como módulo BCD se escogió el modelo SN74LS175 con el cual al incorporarlo al resto del proyecto y lograr una correcta simulación se obtuvo el diagrama mostrado en la figura 4:

III. RESULTADOS

Para la presentación final del circuito se elaboró una simulación para corroborar el funcionamiento del mismo con los respectivos modelos de compuertas lógicas y componentes con los que se tenía a disposición para ensamblar el circuito físico, en la figura 4 se puede observar la simulación del circuito final, mientras que en la figura 5 se puede apreciar el circuito funcional ya ensamblado, cabe destacar que aunque en la imagen se observa un arduino, el mismo solo se usó única y exclusivamente como fuente de los 5 V y no tiene ninguna otra función ni conexión dentro del circuito.

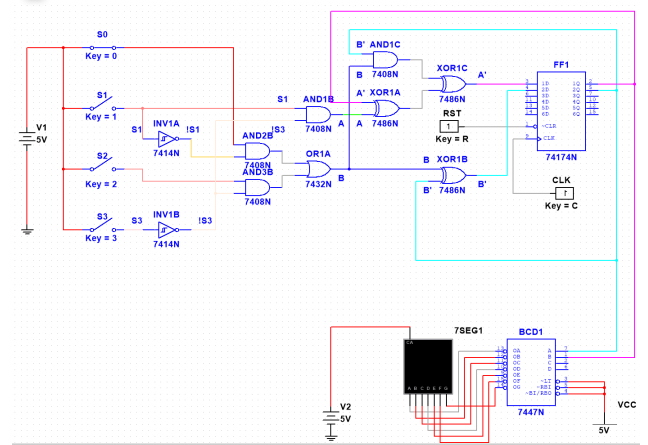


Fig. 4: Diagrama final del circuito.

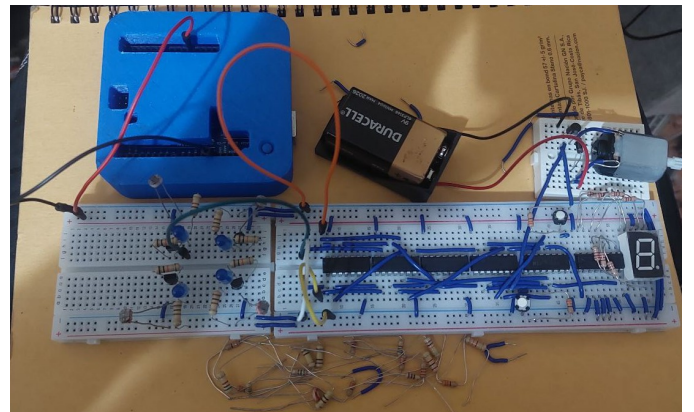


Fig. 5: Fotografía del circuito final ensamblado.

IV. CONCLUSIONES

Se logró diseñar exitosamente los circuitos combinatorios solicitados y simplificarlos al máximo para aumentar eficiencia y utilizar el menor número posible de compuertas lógicas.

Se consiguió un correcto funcionamiento del circuito final tanto en simulación como en funcionamiento real, con lo que se evidencia una correcta planificación y ejecución en el diseño y ensamblaje del proyecto.

Se comprobó que para alcanzar la más óptima configuración y simplificación de ciertos tipos de circuitos se necesitan de otras técnicas además de las vistas hasta el momento de simplificación de algebra booleana y mapas de Karnaugh.

V. RECOMENDACIONES

Se recomienda indagar más en las técnicas de simplificación y optimización de circuitos combinatorios para asegurar la configuración más reducida del mismo.