

# Diseño de circuito combinatorio mediante el uso de un sensor ultrasónico, actuador y visualizador

Ludwin Ramos-Briceño

email: [lujorabri@estudiantec.cr](mailto:lujorabri@estudiantec.cr)

*Escuela de Ingeniería en Computadores*

*Instituto Tecnológico de Costa Rica*

**Abstract**—Combinational circuits are those which outputs depends only of input value combinations at that exact moment, no memory. These are crucial for systems where it's necessary to have an instant output in front of an input action, they are characterized for being able to describe through truth tables and Boolean equations. In this work a design of a system with combinational logic will be carried out to using like input the distances of obstacles got them by ultrasonic sensor, a logic section, that will consist of a combinational circuit, an actuator which depends of the output values of the above combinational circuit and a visualizer, that will be a sampling section for the user. This work seeks to put combinatorial design into practice based on theoretical concepts about combinatorial digital logic circuits, using truth tables and simplification methods such as Karnaugh maps and Boolean equations. Additionally, through the Tinkercad software, the combinational logic system was simulated as verification of the design, which was then built and tested with physical components. Finally, the result was the correct activation of the actuator as well as the sampling of results in the displays, as expected from the design carried out in the work.

**Palabras clave**—Arduino, Sensor ultrasónico, actuador, visualizador, mapas de Karnaugh, Algebra Booleana.

## I. INTRODUCCIÓN

Dentro del diseño digital se encuentra la lógica combinatoria, el cual no es más que una manera de diseñar un circuito digital, que se caracteriza porque las salidas solo dependen de los valores de las entradas, combinando estos para determinar una salida. Estos circuitos digitales no tienen memoria o no dependen de un valor o respuesta anterior, puede ocurrir que una salida de un circuito sea la misma que otro diferente, a su vez se caracterizan porque su funcionalidad puede ser especificada, descrita mediante una tabla de verdad, o mediante una ecuación Booleana [1].

La **Figura 1** muestra el diagrama de bloques del sistema a implementar, el cual se conforma de cuatro principales etapas. El Arduino, que tendrá como función convertir la señal de analógica a digital proveniente del sensor ultrasónico. El circuito combinatorio, estará conformado a su vez por dos diferentes partes: un encodificador y un decodificador. El encodificador tomará 3 bits en código Gray provenientes del Arduino y tendrá como salida 3 bits en código binario en exceso 3 circular. El decodificador por su parte se encargará de habilitar un actuador en dos rangos diferentes del código Gray (más adelante serán especificados estos rangos). Un desacople y un accionador, que tienen como función desacoplar

electricamente el sistema y a su vez será un elemento de potencia respectivamente, estos dos conformarán un actuador. Por último, se tendrá dos visualizadores en diferentes partes del sistema, que permitirá mostrar un resultado al usuario, uno ubicado en el Arduino que mostrará un valor decimal correspondiente al número binario que se retroalimenta del circuito combinatorio al Arduino (dado por el encodificador) y un segundo de igual forma que representará el número binario proveniente del circuito combinatorio por medio de leds.

Para este trabajo el circuito combinatorio es creado mediante el uso de circuitos integrados de compuertas lógicas, donde se estará utilizando la tecnología de la familia TTL. En este caso el uso de una sola familia de circuitos integrados es de considerar, ya que el tiempo de respuesta o salida pueden llegar a variar para dichos circuitos, lo que podría tener impacto en el resultado final en caso de combinar las dos familias. Esto debido a que la familia TTL implementa transistores en su interior BJT, mientras que la familia CMOS utiliza transistores MOSFET, que tiene como principal diferencia el uso del efecto de campo, teniendo una velocidad de conmutación mayor que los BJT, lo que podría conllevar a resultados inesperados si se combinaran [2].

Por otra parte, los actuadores son elementos o circuitos que reciben una señal eléctrica o digital, para efectuar una acción en concreto; es decir, que depende de la señal de entrada para ejecutar cierta tarea, accionar o intervenir en el proceso aportando la energía necesaria al sistema para modificar su estado o el valor de la magnitud física a controlar. Los actuadores en general depende del contexto de aplicación, sin embargo para este trabajo se utilizará como representación de un actuador, un motor en DC que más adelante será especificado [3].

Por último, los visualizadores son elementos del circuito que permiten el muestreo de algún dato o salida referente a lo que se esté realizando en el circuito, para que pueda ser leído de forma legible y entendible por el usuario acorde a la entrada proporcionada por este, sin necesidad de conocer el funcionamiento interno del sistema. En este caso se estará utilizando un display de siete segmentos que permitirá la visualización de números, del mismo modo se tendrá un conjunto de tres leds que permitirá representar una cadena de tres bits de un número binario [4].

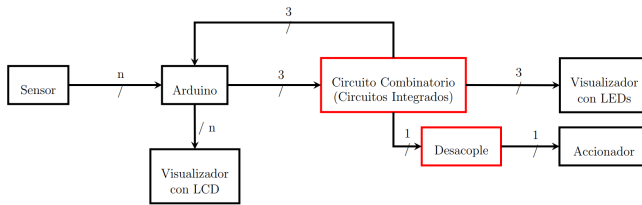


Figura 1: Diagrama de bloques del sistema.

Con lo anterior, en este trabajo se presentará como primera instancia el proceso de diseño del sistema, donde se mostrará los materiales utilizados así como el proceso de desarrollo matemático, reducción y construcción. Posteriormente se presentará el análisis de los resultados obtenidos y finalmente se expondrán las conclusiones con base a los resultados finales y se describirán las recomendaciones asociadas a dichos resultados.

## II. ALGORITMO DESARROLLADO

La **Tabla I** muestra un enlistado de los principales componentes requeridos para realización los procedimientos que se describirán más adelante.

| Cantidad | Descripción              |
|----------|--------------------------|
| 1        | Arduino                  |
| 2        | Protoboard               |
| 1        | Sensor ultrasónico       |
| 1        | Integrado SN74LS04 (NOT) |
| 1        | Integrado SN74LS08 (AND) |
| 1        | Integrado SN74LS32 (OR)  |
| 1        | Integrado SN74LS86 (XOR) |
| 3        | Leds                     |
| 1        | Transistor 2N222A        |
| 1        | Motor DC                 |
| 1        | Display siete segmentos  |

Tabla I: Principal equipo requerido

Los procedimientos se clasificarán en dos partes, el primero será un preanálisis donde se realizará el procedimiento matemático y reducción, mientras que el segundo será la simulación así como la construcción del sistema.

### A. Preanálisis

De acuerdo al contexto presentado anteriormente, se procederá a realizar la reducción del circuito combinatorio (encodificador y decodificador), el cual constituyen la lógica combinatoria del sistema:

#### A1. Encodificador

Este submódulo del sistema tiene como entrada 3 bits en código de Gray (provenientes del Arduino), y la idea fundamental es dar como salida un código binario en exceso tres circular; es decir, que según el número de 3 bits dado en código Gray (que tiene correspondencia a un numero decimal), se suma 3 a dicho valor de entrada y se proporciona como salida el resultado en código binario, tal como se puede observar en la **Tabla II**.

| Referencia |         | Entradas |   |   | Salidas  |    |    |
|------------|---------|----------|---|---|----------|----|----|
| Decimal    | Binario | Gray     |   |   | Exceso 3 |    |    |
| -          | -       | A        | B | C | A'       | B' | C' |
| 0          | 0 0 0   | 0        | 0 | 0 | 0        | 1  | 1  |
| 1          | 0 0 1   | 0        | 0 | 1 | 1        | 0  | 0  |
| 2          | 0 1 0   | 0        | 1 | 1 | 1        | 0  | 1  |
| 3          | 0 1 1   | 0        | 1 | 0 | 1        | 1  | 0  |
| 4          | 1 0 0   | 1        | 1 | 0 | 1        | 1  | 1  |
| 5          | 1 0 1   | 1        | 1 | 1 | 0        | 0  | 0  |
| 6          | 1 1 0   | 1        | 0 | 1 | 0        | 0  | 1  |
| 7          | 1 1 1   | 1        | 0 | 0 | 0        | 1  | 0  |

Tabla II: Tabla de verdad del encodificador.

Ahora bien, mediante la **Tabla II** es posible obtener las ecuaciones booleanas que describen cada una de las salidas, por medio del método de simplificación de los mapas de Karnaugh.

|   |   |     |    |    |    |
|---|---|-----|----|----|----|
|   |   | B C |    |    |    |
|   |   | 00  | 01 | 11 | 10 |
| A | 0 | 0   | 1  | 1  | 1  |
|   | 1 | 0   | 0  | 0  | 1  |

Figura 2: Mapa de Karnaugh para la salida A' (más significativa).

|   |   |     |    |    |    |
|---|---|-----|----|----|----|
|   |   | B C |    |    |    |
|   |   | 00  | 01 | 11 | 10 |
| A | 0 | 1   | 0  | 0  | 1  |
|   | 1 | 1   | 0  | 0  | 1  |

Figura 3: Mapa de Karnaugh para la salida B'.

|   |   |     |    |    |    |
|---|---|-----|----|----|----|
|   |   | B C |    |    |    |
|   |   | 00  | 01 | 11 | 10 |
| A | 0 | 1   | 0  | 1  | 0  |
|   | 1 | 0   | 1  | 0  | 1  |

Figura 4: Mapa de Karnaugh para la salida C' (menos significativa).

Utilizando mintérminos en los mapas de Karnaugh anteriores, se obtienen las ecuaciones que describen el comportamiento de cada salida con respecto a los valores de entrada, tal como se observan en las **Eq.1**, **Eq.2** y **Eq.3**.

$$A' = \overline{A}C + B\overline{C} \quad (1)$$

$$B' = \overline{C} \quad (2)$$

$$C' = \overline{A} \cdot \overline{B} \cdot \overline{C} + \overline{A}BC + \overline{A}B\overline{C} + ABC \quad (3)$$

Por medio del teorema de distributividad, a partir de la **Eq.3** es posible realizar una simplificación obteniendo así la **Eq.4**.

$$C' = \overline{C}(\overline{A} \cdot \overline{B} + AB) + C(\overline{A}\overline{B} + \overline{A}B) \quad (4)$$

La **Eq.4** es posible reducirla a la ecuación **Eq.5**.

$$C' = \overline{C} \cdot \overline{(A \oplus B)} + C(A \oplus B) \quad (5)$$

Por último, de la **Eq.5** se puede obtener una expresión final, para describir la salida  $C'$  del sistema, **Eq.6**.

$$C' = \overline{A \oplus B \oplus C} \quad (6)$$

Por tanto, de esta manera se obtiene el comportamiento de cada salida del encodificador de acuerdo a cada salida proveniente del Arduino (entradas del encodificador).

## A2. Decodificador

Este submódulo del sistema tiene como entrada 3 bits en código Gray (provenientes del Arduino) y la función de este será habilitar un actuador en dos rangos, para este caso se estará tomando como activo los rangos de 2 a 3 (en código Gray) y el intervalo de 6 a 7. Para realizar dicho comportamiento se crea una tabla de verdad utilizando mintérminos, pues se quiere que dichos rangos sean los activos, por lo que se asumirá como activo un 1 (lógica positiva), generando así la **Tabla III**.

| Referencia (decimal) | Entradas (Gray) |   |   | Salida |
|----------------------|-----------------|---|---|--------|
| -                    | A               | B | C | Y      |
| 0                    | 0               | 0 | 0 | 0      |
| 1                    | 0               | 0 | 1 | 0      |
| 2                    | 0               | 1 | 1 | 1      |
| 3                    | 0               | 1 | 0 | 1      |
| 4                    | 1               | 1 | 0 | 0      |
| 5                    | 1               | 1 | 1 | 0      |
| 6                    | 1               | 0 | 1 | 1      |
| 7                    | 1               | 0 | 0 | 1      |

Tabla III: Tabla de verdad del decodificador.

Al igual como antes, es posible utilizar mapas de Karnaugh para encontrar la ecuación booleana que describe la salida del decodificador.

|   |   | B C |    |    |    |
|---|---|-----|----|----|----|
|   |   | 00  | 01 | 11 | 10 |
| A | 0 | 0   | 0  | 1  | 1  |
|   | 1 | 1   | 1  | 0  | 0  |

Figura 5: Mapa de Karnaugh para la salida del decodificador.

Utilizando mintérminos en el mapa de Karnaugh anterior, se obtiene la **Eq.7** que describe la salida del decodificador.

$$Y = A\overline{B} + \overline{A}B \quad (7)$$

Finalmente, la expresión de la **Eq.7** es posible reducirla a la expresión de la **Eq.8**.

$$Y = A \oplus B \quad (8)$$

Por tanto, de las ecuaciones anteriores encontradas en esta sección es posible describir el circuito combinacional del sistema reducido en términos matemáticos, para así proceder al diseño del circuito en simulación y luego físicamente.

## B. Construcción

Una vez encontradas las ecuaciones booleanas necesarias que describen el circuito combinacional del sistema, es posible proseguir con el diseño del mismo en simulación.

Antes de pasar al montaje del circuito, es necesario codificar o programar el Arduino para obtener 3 bits en código de Gray como salida (entrada para el circuito combinacional) que estarán determinados por los rangos de distancia dados por el sensor ultrasónico. Así mismo, se necesita obtener 3 bits de entrada en el Arduino (código binario), provenientes del circuito combinacional (salida del encodificador) para ser mostrados en un visualizador en forma decimal (se estará utilizando un siete segmentos cátodo común), de igual forma utilizando esos 3 bits, en otro visualizador se debe representar dichos 3 bits en binario con leds, tal como se indica en el diagrama de la **Figura 1**.

Por otra parte, el circuito combinacional, concretamente el decodificador, habilitará el actuador (desacople y accionador), que estará constituido por un transistor, que servirá como un interruptor (desacople) y un motor DC (accionador) que estará alimentado con una fuente externa (Batería 9V).

Con los especificado anteriormente es posible montar el circuito en simulación, obteniendo así el esquema de la **Figura 6** y **Figura 7**.

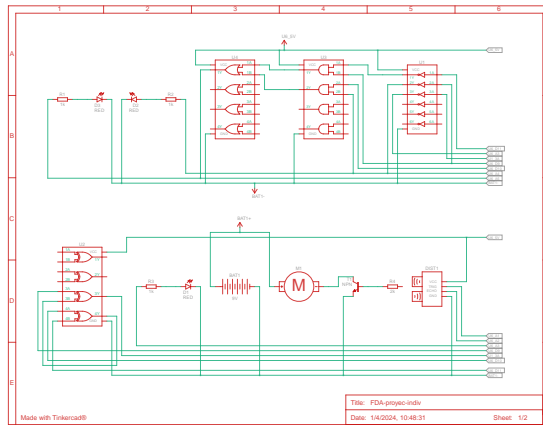


Figura 6: Primera parte del esquemático del sistema.

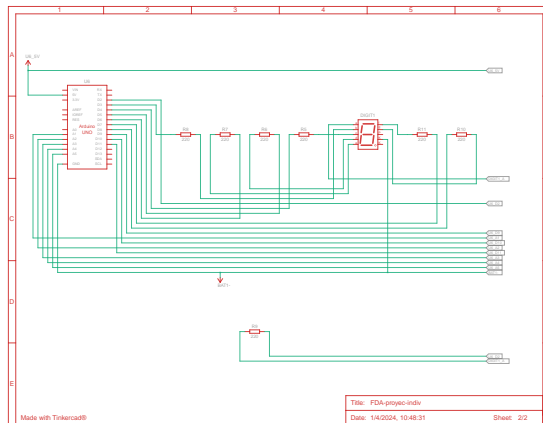


Figura 7: Segunda parte del esquemático del sistema.

Al obtener el plano o esquemático claro del sistema, es posible realizar el montaje con componentes físicos, siguiendo los esquemáticos antes originados del sistema, **Figura 6** y **Figura 7**.

### III. RESULTADOS

Una vez armado el circuito en protoboard físicamente, con base a la sección anterior, fue posible realizar ciertas pruebas para obtener así que el sistema se comporta según lo esperado, donde de acuerdo a los rangos establecidos de distancias (captadas por el sensor), se obtiene el código de Gray correspondiente a dicho intervalo en la salida del Arduino, tal como se pueden observar en la **Figura 8**.

A su vez en protoboard, es posible notar el comportamiento que se esperaba, los dos visualizadores evidencian que el sistema funciona correctamente, además del muestreo en consola de los diferentes valores. Sin dejar de lado a su vez la correcta activación del actuador acorde a los rangos escogidos en el decodificador, ver **Figura 8**.

```

67:   OutGray[2] = 0; OutGray[1] = 0; OutGray[0] = 0; // 0
68:
69:   if(distancia > 12 && distancia <= 24) {
70:     OutGray[2] = 0; OutGray[1] = 0; OutGray[0] = 1; // 1
71:
72:   if(distancia > 24 && distancia <= 36) {
73:     OutGray[2] = 0; OutGray[1] = 1; OutGray[0] = 1; // 2
74:
75:   if(distancia > 36 && distancia <= 48) {
76:     OutGray[2] = 0; OutGray[1] = 1; OutGray[0] = 0; // 3
77:
78:   if(distancia > 48 && distancia <= 60) {
79:     OutGray[2] = 1; OutGray[1] = 1; OutGray[0] = 0; // 4
80:
81:   if(distancia > 60 && distancia <= 72) {
82:     OutGray[2] = 1; OutGray[1] = 1; OutGray[0] = 1; // 5
83:
84:   if(distancia > 72 && distancia <= 84) {
85:     OutGray[2] = 1; OutGray[1] = 0; OutGray[0] = 1; // 6
86:
Output: Serial Monitor X
Message (Enter to send message to Arduino IDE on COM3)
La distancia es: 41 cm | Gray --> 010 | Decimal --> 6
La distancia es: 42 cm | Gray --> 010 | Decimal --> 6
La distancia es: 42 cm | Gray --> 010 | Decimal --> 6
La distancia es: 42 cm | Gray --> 010 | Decimal --> 6
La distancia es: 42 cm | Gray --> 010 | Decimal --> 6
La distancia es: 43 cm | Gray --> 010 | Decimal --> 6
La distancia es: 43 cm | Gray --> 010 | Decimal --> 6
La distancia es: 43 cm | Gray --> 010 | Decimal --> 6
La distancia es: 41 cm | Gray --> 010 | Decimal --> 6
La distancia es: 41 cm | Gray --> 010 | Decimal --> 6
La distancia es: 41 cm | Gray --> 010 | Decimal --> 6
La distancia es: 42 cm | Gray --> 010 | Decimal --> 6
La distancia es: 41 cm | Gray --> 010 | Decimal --> 6
La distancia es: 41 cm | Gray --> 010 | Decimal --> 6

```

Figura 8: Distancia, código Gray y valor en decimal en Arduino.

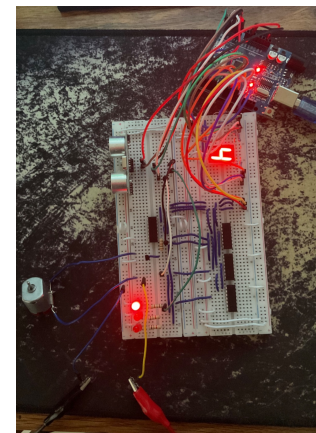


Figura 9: Prueba en protoboard del sistema.

### IV. CONCLUSIONES

Una vez concluido con el diseño, construcción del circuito lógico combinatorio, fue posible observar que la descripción de los módulos del sistema en términos de tablas de verdad y la utilización de mapas de Karnaugh, fueron fundamental para el diseño del circuito digital del sistema, obteniendo la **Eq.1**, **Eq.2**, **Eq.6** y **Eq.8**, el cual describen las salidas del encodificador y decodificador respectivamente.

A su vez el por medio del uso de tecnologías de simulación para circuitos digitales, fue posible realizar una previsualización del funcionamiento del sistema, ayudando a la corrección de errores en la construcción del sistema y evitando daños en componentes físicos, así como el desperdicio de recursos, obteniendo así la **Figura 6** y **Figura 7**.

Finalmente luego del diseño del sistema combinatorio, fue posible realizar la construcción del mismo, obteniendo así la **Figura 8** y **Figura 9**, el cual muestran los resultados esperados según el análisis y diseño realizado a lo largo del trabajo.

### V. RECOMENDACIONES

Aunque para la obtención de la **Eq.5** y **Eq.6** no se haya utilizado ninguna regla de inferencias lógicas y axiomas en concreto, estas se simplificaron por medio de un análisis de tablas de verdad a partir de la **Eq.4**, el cual no fue descrito en

este trabajo, por lo que se recomienda realizar dichos procesos para su comprobación.

Para el actuador antes mencionado, es necesario tener en cuenta el voltaje máximo del motor DC a utilizar, pues dado el caso se tendrá que colocar una resistencia para regular el paso de corriente en el transistor, pues en simulación dicho aspecto no fue considerado.

#### REFERENCIAS

- [1] D. Harris and S. Harris, *Digital design and computer architecture: Arm Edition*. MORGAN KAUFMANN Pub, 2015.
- [2] M. Morris, *DISEÑO DIGITAL*. PEARSON EDUCACIÓN Pub, 2003.
- [3] R. Sanchis, J. A. Romero and C. V. Ariño, *Automatización industrial*. Universitat Jaume I. Pub, 2010.
- [4] M. Á. Bañuelos, M. Pacheco and R. G. Villegas, *MÓDULO 7- SEGMENTOS*. Universidad Nacional Autónoma de México Pub, 2021.
- [5] Autodesk, *TinkerCad*, 2024.