



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA**

**CARRERA:** INGENIERÍA EN TELECOMUNICACIONES

**NRC:** 9434

**ASIGNATURA:** CIRCUITOS DIGITALES

**PRODUCTO DE UNIDAD #1**

**INTEGRANTES:**

- MONTIEL DÍAZ DEYDER ANDRÉS
- SALAZAR DUQUE DOMENICA ALEJANDRA
- VILLACIS NARVAEZ JONATHAN STEVEN

**DOCENTE:** ING. DARWIN ALULEMA

Junio, 2020

## ÍNDICE

|                                                      |           |
|------------------------------------------------------|-----------|
| <b>PLANTEAMIENTO DEL PROBLEMA</b>                    | <b>3</b>  |
| <b>OBJETIVOS</b>                                     | <b>3</b>  |
| <b>ESTADO DEL ARTE</b>                               | <b>3</b>  |
| <b>MARCO TEÓRICO</b>                                 | <b>4</b>  |
| <b>DIAGRAMAS</b>                                     | <b>10</b> |
| <b>LISTA DE COMPONENTES</b>                          | <b>19</b> |
| <b>MAPA DE VARIABLES</b>                             | <b>20</b> |
| <b>EXPLICACIÓN DEL CÓDIGO FUENTE</b>                 | <b>26</b> |
| <b>DESCRIPCIÓN DE PRERREQUISITOS Y CONFIGURACIÓN</b> | <b>35</b> |
| <b>CONCLUSIONES</b>                                  | <b>36</b> |
| <b>RECOMENDACIONES</b>                               | <b>37</b> |
| <b>CRONOGRAMA</b>                                    | <b>37</b> |
| <b>BIBLIOGRAFÍA</b>                                  | <b>37</b> |
| <b>ANEXOS</b>                                        | <b>38</b> |
| 15.1 MANUAL DE USUARIO                               | 38        |
| 15.2 HOJAS TÉCNICAS                                  | 44        |

## **1. PLANTEAMIENTO DEL PROBLEMA**

Los sistemas digitales permiten solucionar de manera rápida y eficiente distintos problemas que surgen en la vida cotidiana. Para la resolución de estos problemas mediante la utilización de sistemas digitales se necesita conocimiento del mismo. ¿Será que los estudiantes de Tercer semestre de la carrera de Ingeniería en Telecomunicaciones del período mayo-septiembre 2020 poseen los conocimientos necesarios para solucionar varios problemas planteados?

## **2. OBJETIVOS**

General: Realizar distintas aplicaciones de los circuitos digitales con la finalidad de aplicar lo aprendido en clases por parte de los estudiantes de Tercer semestre de la carrera de Ingeniería en Telecomunicaciones del período mayo- septiembre 2020, a través del planteamiento de varios problemas.

Específicos:

1. Diseñar un circuito de tres entradas que dé como resultado el valor mayoritario.
2. Diseñar un circuito que compara dos números binarios de 2 bits e identifique si el primero es mayor , menor o igual al segundo a través de displays de 7 segmentos mostrando por pantalla el número cero.
3. Diseñar un circuito que especifique los turnos de trabajo dado cierto número.

## **3. ESTADO DEL ARTE**

Título: Effective bug triage based on historical Bug-Fix information

Autores: Hao Hu, Hongyu Zhang, Jifeng Xuan, Weigang Sun.

Año: 2018

El artículo consiste en la descripción de un nuevo método de detección de errores basado en el distintos métodos antiguos. El método llamado Bug-Fix fue desarrollado por los autores con la finalidad de ayudar a los desarrolladores con el trabajo de la detección de errores. La investigación fue implementada en China, específicamente en tres proyectos abiertos a gran escala y dos proyectos industriales pequeños.

Título: High Speed Current Mode Threshold Logic Gates

Autores: Vinod Vijayan C, Ayoob Khan, Shahul Hammed.

Año: 2018

El artículo consiste en la implementación de compuertas lógicas de umbral, con la finalidad de ser implementadas en la corriente modelógica. El modelo diseñado posee dos partes principales que son la parte diferencial y la parte del sensor. La investigación fue implementada en India.

Título: A Novel Design Gate based Low-Cost Configurable RO PUF using Reversible Logic

Autores: Bappaditya Dey, Kasem Khalil , Ashok Kumar , Magdy Bayoumi.

Año: 2018

El artículo consiste en el estudio de las operaciones lógicas reversibles. Con esta lógica se puede preservar un bit adicional de información tanto de entrada como de salida. Esta tecnología al ser implementada permite minimizar la disipación de calor y el consumo energético. La investigación fue implementada en Luisiana, Estados Unidos.

## **4. MARCO TEÓRICO**

### ***4.1. Algebra de Boole:***

La utilización de variables que tienen dos posibles valores la sintetizó Shannon usando ideas que las expresó inicialmente un matemático llamado George Boole. A diferencia de las variables del álgebra común (que pueden tomar un número infinito de valores). Una variable booleana, que se simboliza con las letras mayúsculas A, B, C, etc., pueden tomar solamente dos valores 1 lógico y 0 lógico.

Para trabajar con variables booleanas se utilizan operadores similares al álgebra común. Estos operadores booleanos se los conoce comúnmente como conectivos lógicos.

Los conectivos lógicos son los operadores del álgebra de boole y representan a los circuitos digitales más fundamentales y su operación o funcionamiento se describe mediante el uso del álgebra de boole.

Como el álgebra de boole, al igual que el álgebra común, posee varios teoremas o leyes para su comprensión, que son expresados a continuación.

|              |                                             |                                               |
|--------------|---------------------------------------------|-----------------------------------------------|
| Idempotencia | $X * X = X$                                 | $X + X = X$                                   |
| Complementos | $X * X' = 0$                                | $X + X' = 1$                                  |
| Identidad    | $X * 1 = X$                                 | $X + 0 = X$                                   |
| ETOSNulos    | $X * 0 = 0$                                 | $X + 1 = 1$                                   |
| Involución   | $(X')' = X$                                 |                                               |
| Conmutativa  | $X * Y = Y * X$                             | $X + Y = Y + X$                               |
| Asociativa   | $(X * Y) * Z = X * (Y * Z)$                 | $(X + Y) + Z = X + (Y + Z)$                   |
| Distributiva | $X * (Y + Z) = X * Y + X * Z$               | $X + (Y * Z) = (X + Y) * (X + Z)$             |
| Absorción    | $X(X + Y) = X$                              | $X + X * Y = X$                               |
| Redundancia  | $X'(X + Y) = X' * Y$<br>$X(X' + Y) = X * Y$ | $X' + X * Y = X' + Y$<br>$X + X' * Y = X + Y$ |
| D'Morgan     | $(X * Y)' = X' + Y'$                        | $(X + Y)' = X' * Y'$                          |
| Combinación  | $(X + Y)(X + Y') = X$                       | $X * Y + X * Y' = X$                          |

Fuente:("Álgebra Booleana — MecatrónicaLATAM," n.d.)

#### 4.2. **Compuertas Lógicas:**

Las compuertas lógicas son los bloques básicos de cualquier circuito digital. Todos los aparatos digitales desde el más simple dispositivos hasta el más sofisticado computador, están formados por compuertas conectadas a una gran variedad de configuraciones.

Una compuerta digital es un circuito electrónico con dos o más líneas de entrada y una línea de salida que tiene la capacidad de tomar decisiones dependiendo de sus entradas y de la función lógica para la cual ha sido diseñada. En

electrónica digital existen siete compuertas lógicas designadas que son las siguientes.

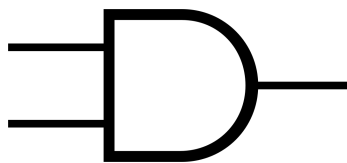
- And

Ecuación Lógica:  $A \cdot B = Y$

Tabla de verdad:

| A | B | Y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Símbolo:



Fuente: ("File:AND ANSI.svg - Wikimedia Commons," n.d.)

- Or

Ecuación Lógica:  $A + B = Y$

Tabla de verdad:

| A | B | Y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Símbolo:



Fuente: ("Archivo:Puerta OR.svg - Wikipedia, la enciclopedia libre," n.d.)

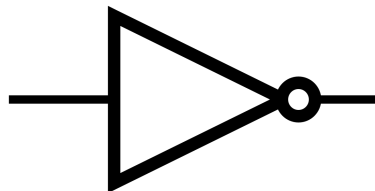
- Not

Ecuación Lógica:  $Y=A'$

Tabla de verdad:

| A | Y |
|---|---|
| 0 | 1 |
| 1 | 0 |

Símbolo:



Fuente: ("File:NOT ANSI.svg - Wikimedia Commons," n.d.)

- Nand

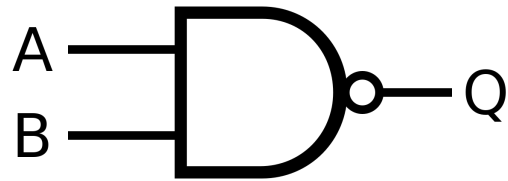
Ecuación Lógica:  $(A*B)'=Y$

Tabla de verdad:

| A | B | Y |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |

|   |   |   |
|---|---|---|
| 1 | 1 | 0 |
|---|---|---|

Símbolo:



Fuente: (“Lógica NAND - Wikipedia, la enciclopedia libre,” n.d.)

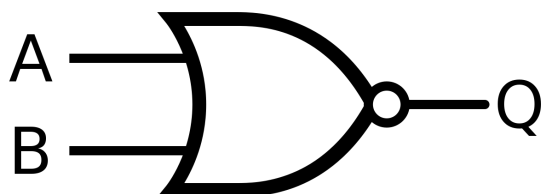
- Nor

Ecuación Lógica:  $(A+B)'=Y$

Tabla de verdad:

| A | B | Y |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 0 |

Símbolo:



Fuente: (“Lógica NOR - Wikipedia, la enciclopedia libre,” n.d.)

- Xor

Ecuación Lógica:  $A'B+AB'=Y$

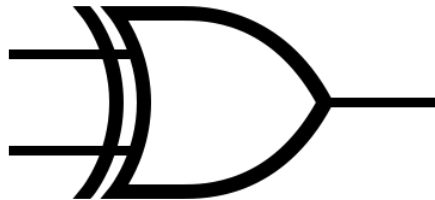
Tabla de verdad:

| A | B | Y |
|---|---|---|
|---|---|---|



|   |   |   |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

Símbolo:



Fuente:("Puerta XOR - Wikiwand," n.d.)

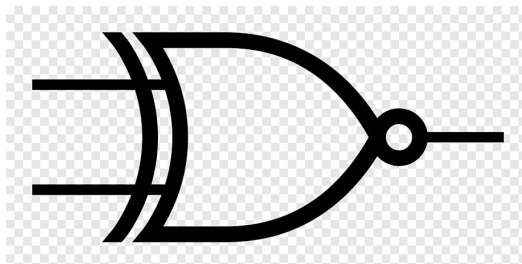
- Xnor

Ecuación Lógica:  $A'B' + AB = Y$

Tabla de verdad:

| A | B | Y |
|---|---|---|
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Símbolo:



Fuente: ("XNOR gate XOR gate NAND gate Logic gate, symbol, miscellaneous, angle, text png | PNGWing," n.d.)

#### **4.3. Thinkercad:**

Thinkercad es un software gratuito online creado por Autodesk, con la finalidad de enseñar el diseño y la implementación en formato 3D. El principal objetivo de este programa es que sea a primera vista atrayente para el usuario para que este empiece a realizar sus diseños de una manera fácil y atractiva. Esto se puede dar ya que su interfaz de trabajo es bastante sencillo de utilizar, pero el problema con este software es que una vez que el usuario domina los conceptos básicos Thinkercad deja de ser atrayente para realizar diseños más complejos.

Sus principales ventajas son su sencillez de uso, su aspecto atractivo para el usuario y el hecho que en pocas horas de entrenamiento ya se pueda adquirir mucha destreza.

Como desventajas, cabe señalar, primero que es necesario crear una cuenta de usuario para poder utilizar el software, además que de esta manera solo se puede usar de forma online por lo que el uso de conexión a internet es indispensable. La segunda desventaja es que cuando se requiera realizar un diseño más complejo el software que obsoleto.

(“Diseña con Tinkercad - EDUCACIÓN Y TECNOLOGÍA,” n.d.)

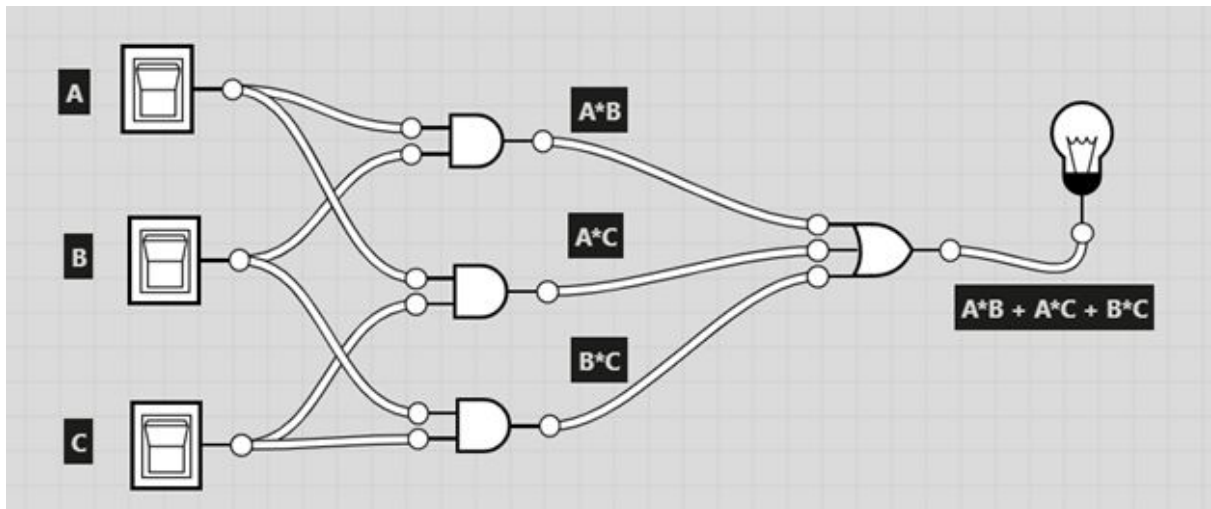
### **5. DIAGRAMAS**

Problemas a resolver:

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.

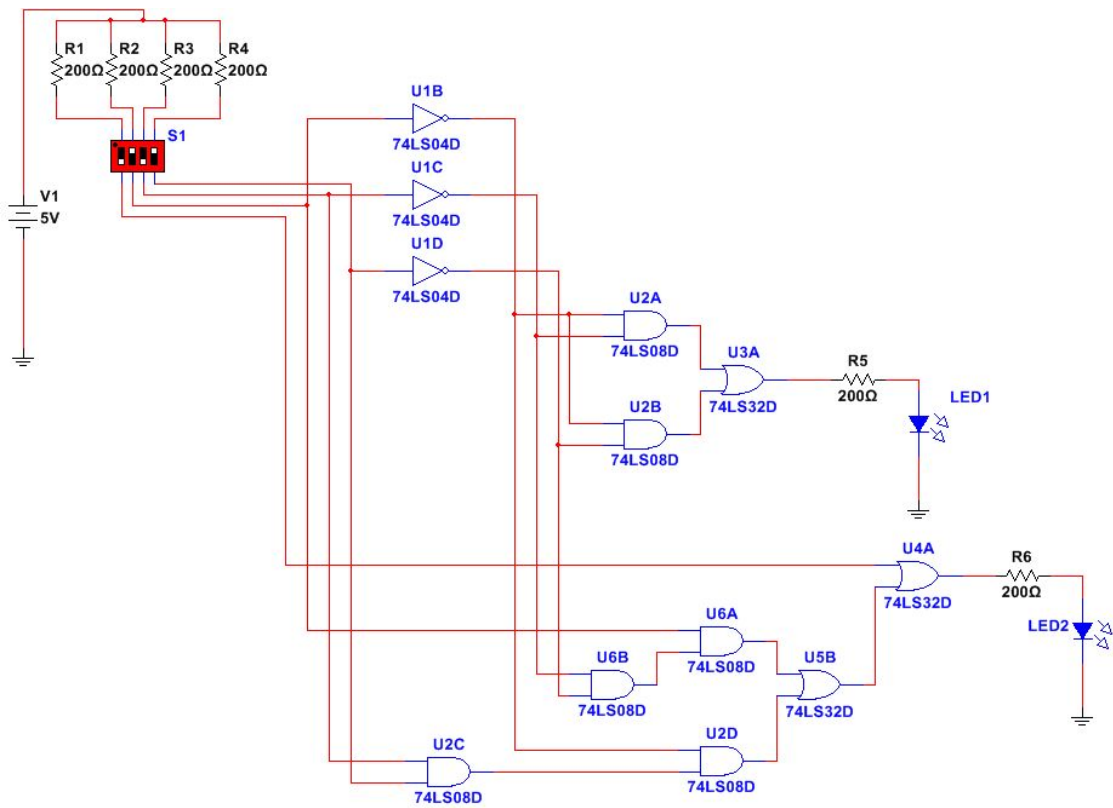
Implementado en Multisim

Implementado en ThinkerCad

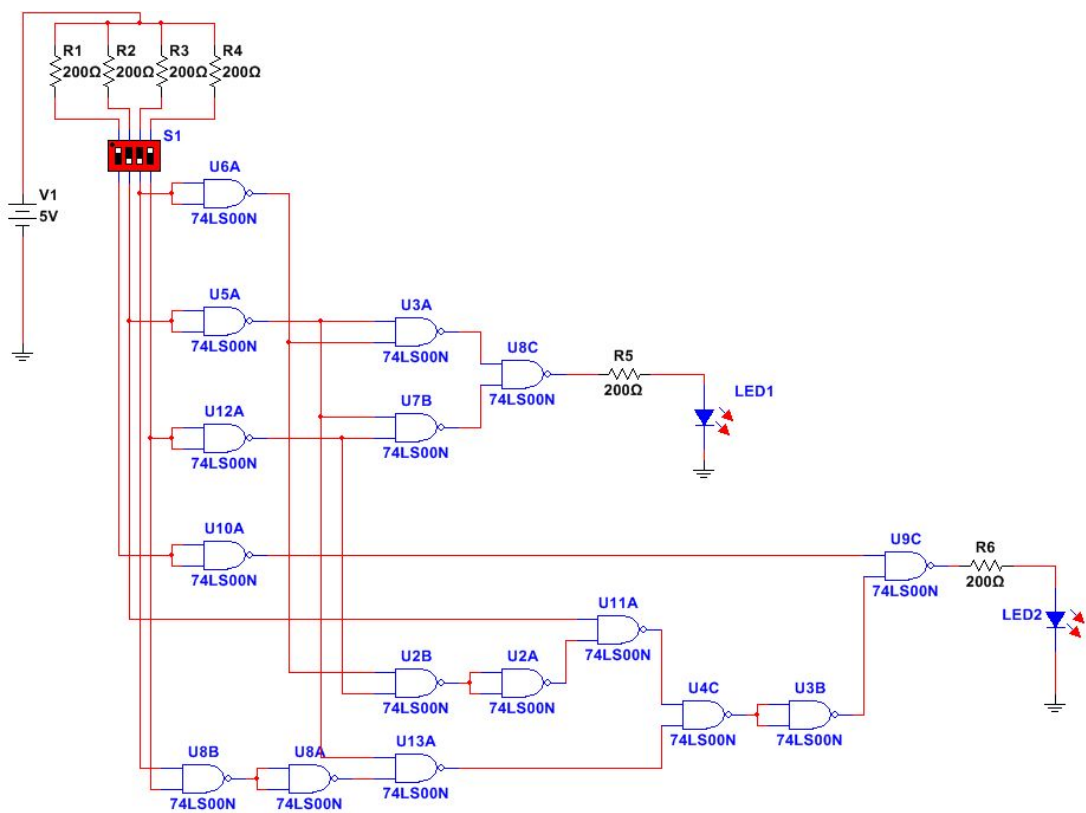


2.- El horario laboral de una factoría es de 8 horas diarias, divididas en tres turnos: de 8 a 11 (primer turno), de 11 a 13 (segundo turno), de 13 a 16 (descanso) y de 16 a 19 (tercer turno). Se pretende diseñar un circuito que tenga como entradas la representación binaria de la hora actual menos ocho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

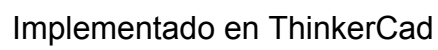
Implementado en Multisim

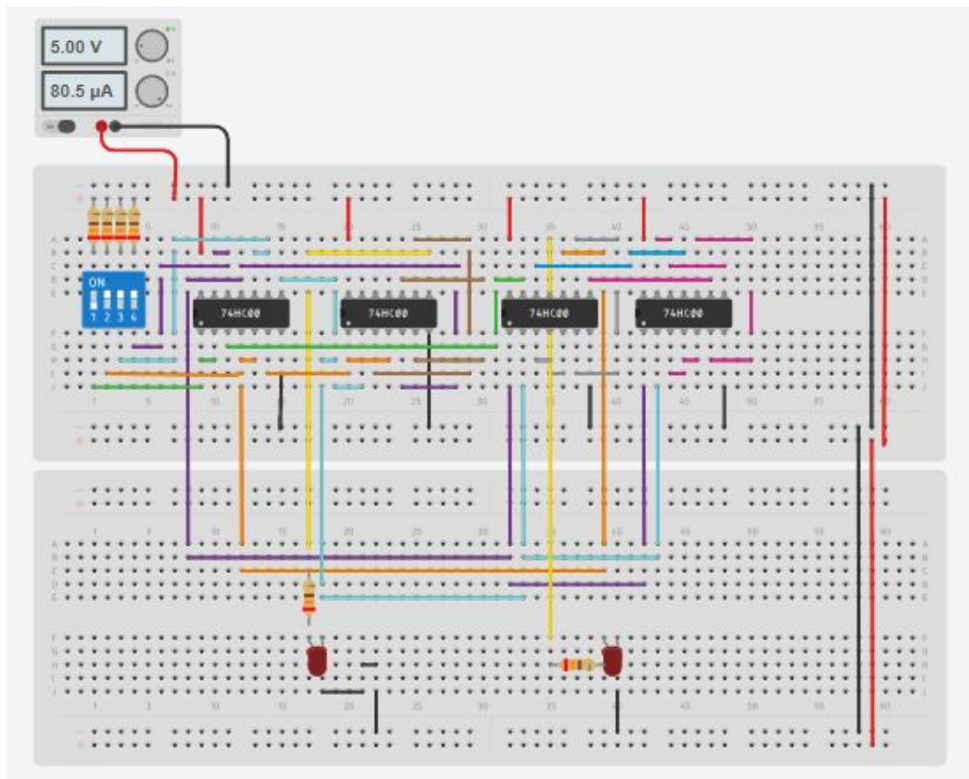


Simplificando a NAND se tiene que



-Implementando con el display de 7 segmentos se tiene que





**3.- Se pretende diseñar un circuito comparador de 2 números de 2 bits,  $A=(a_1, a_0)$  y  $B=(b_1, b_0)$ . Dicho circuito deberá tener tres salidas  $M$ ,  $I$ ,  $m$ , de tal forma que :  $M = 1$  si  $A > B$   $I = 1$  si  $A = B$   $m = 1$  si  $A < B$  Diseñada exclusivamente con puertas NOR .**

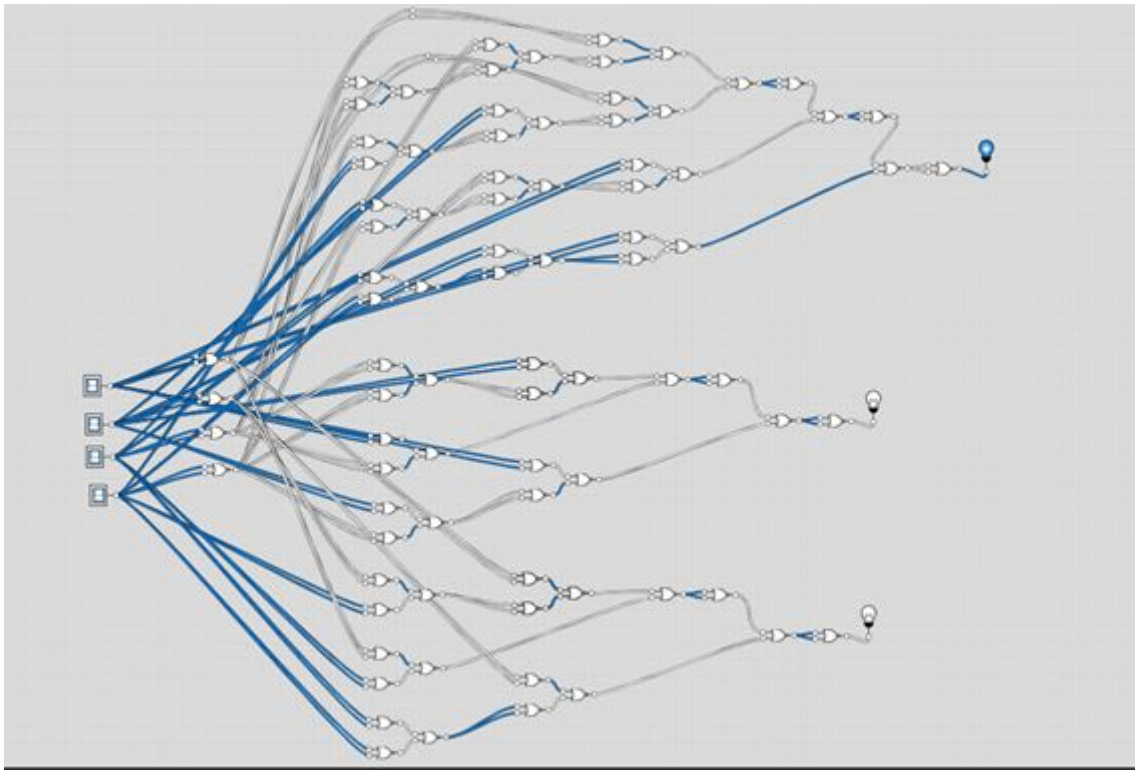
Implementado en LabCircuit

Implementado en Multisim

Implementado en ThinkerCad

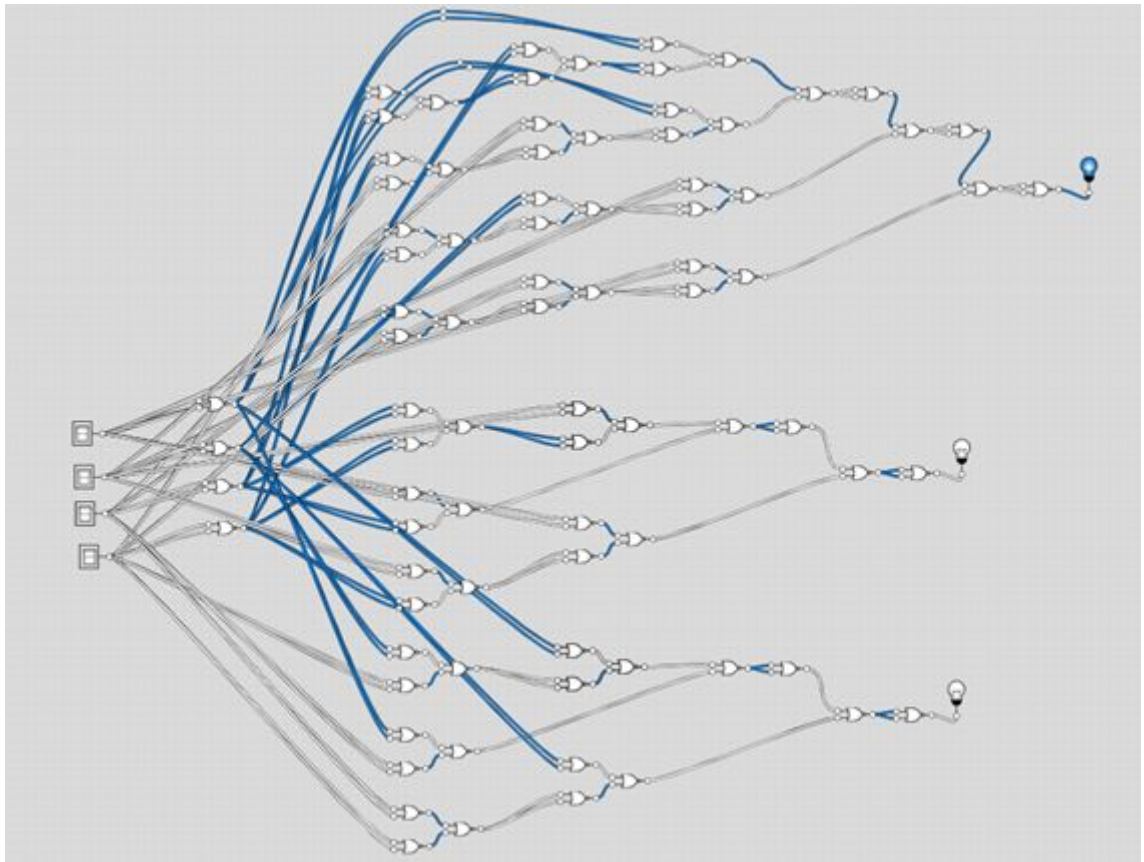
Vamos a implementar cuando el primero número de dos bits es igual al segundo número, es decir en binario tenemos:

Cuando  $A = 11$  y  $B = 11$



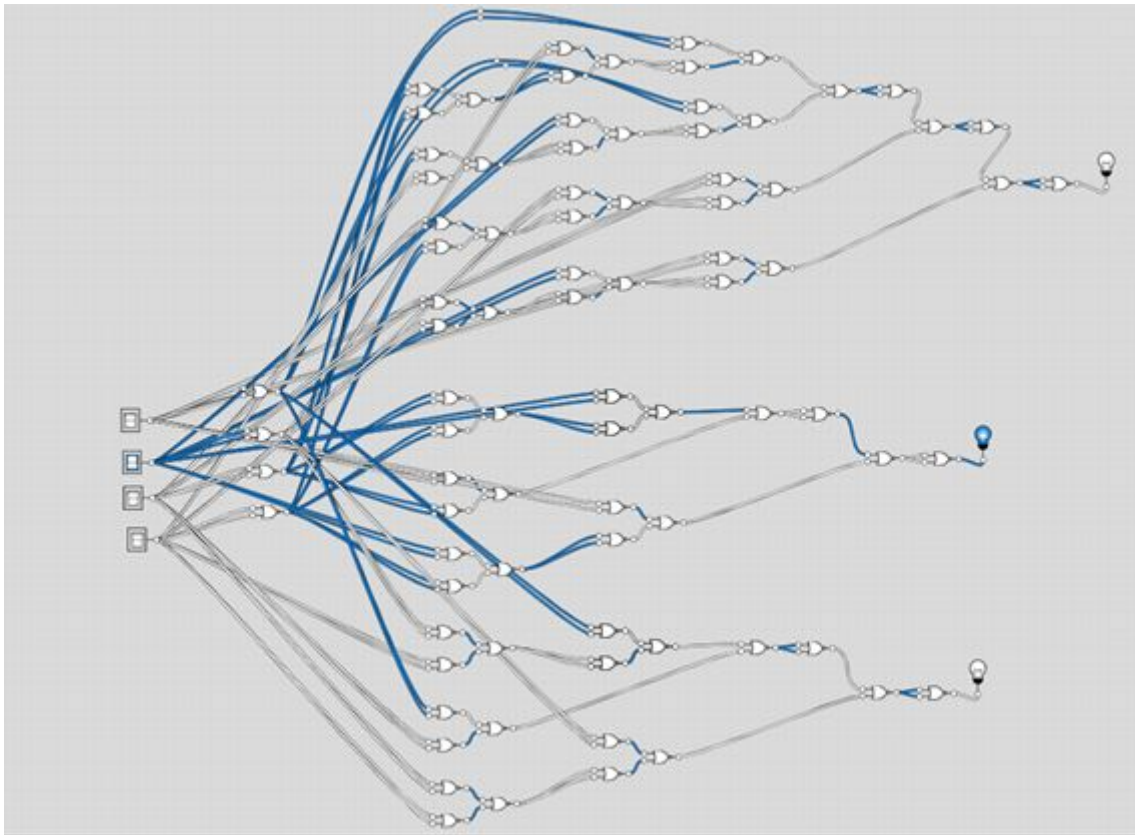
como podemos ver en la gráfica el foco 1 se prende solamente si es que A y B son iguales .

Cuando A= 00 y B= 00

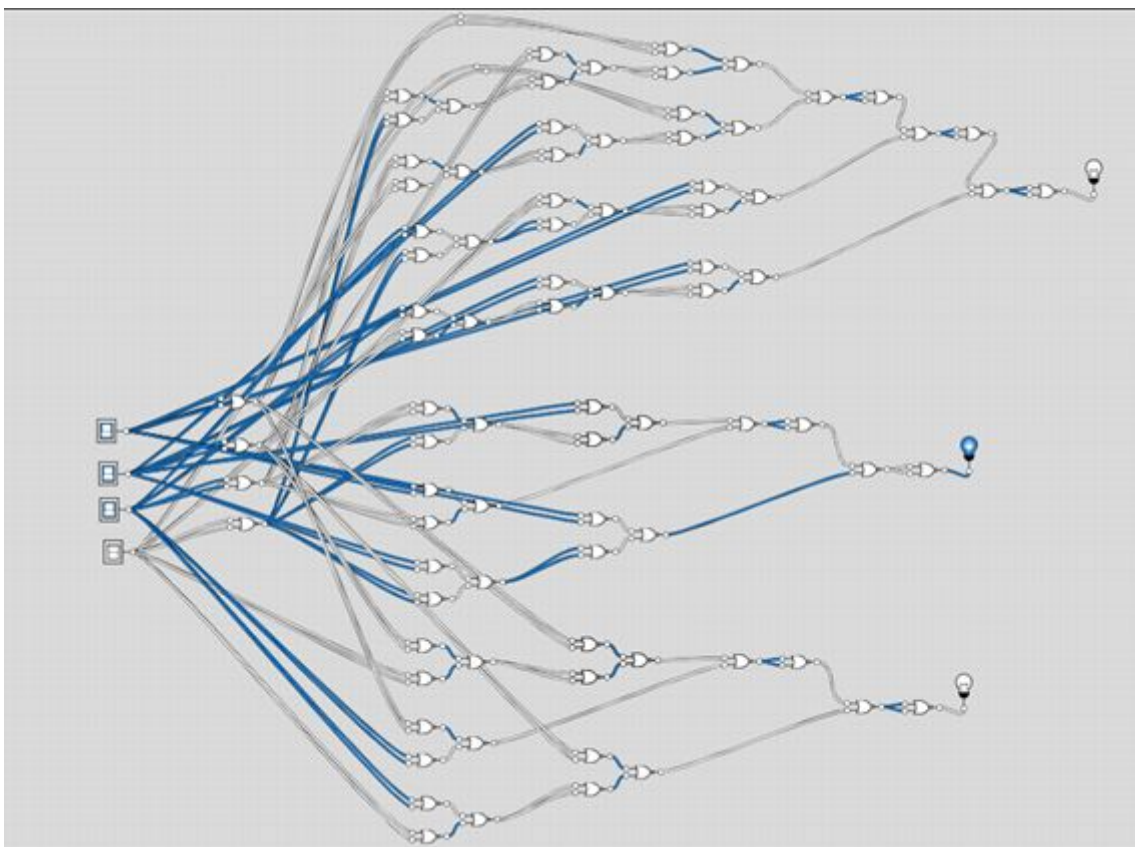


Para cuando  $A > B$  se cumple la siguiente función que permite prender el foco 2  
Cuando  $A = 01$  y  $B = 00$

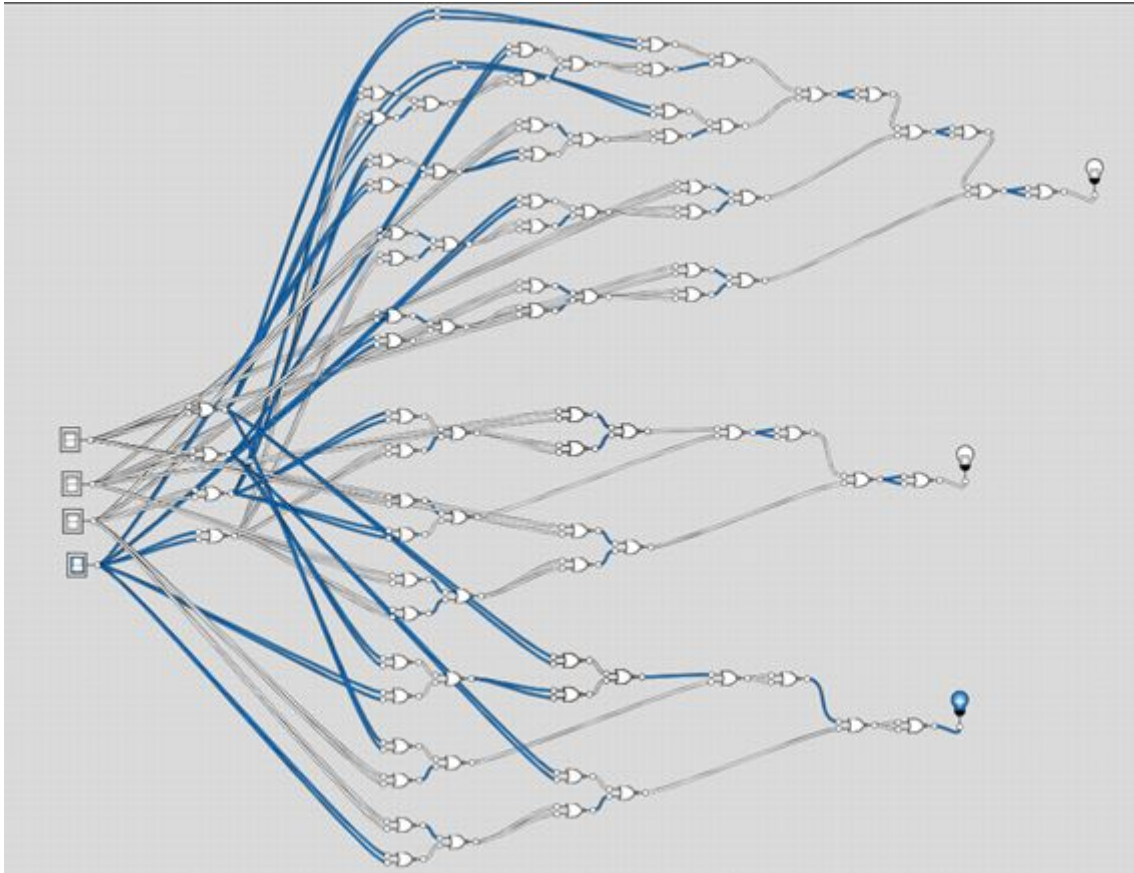




de igual manera tenemos otro ejemplo con el mismo caso para la salida M:  $A > B$   
 Cuando  $A = 11$  y  $B = 10$

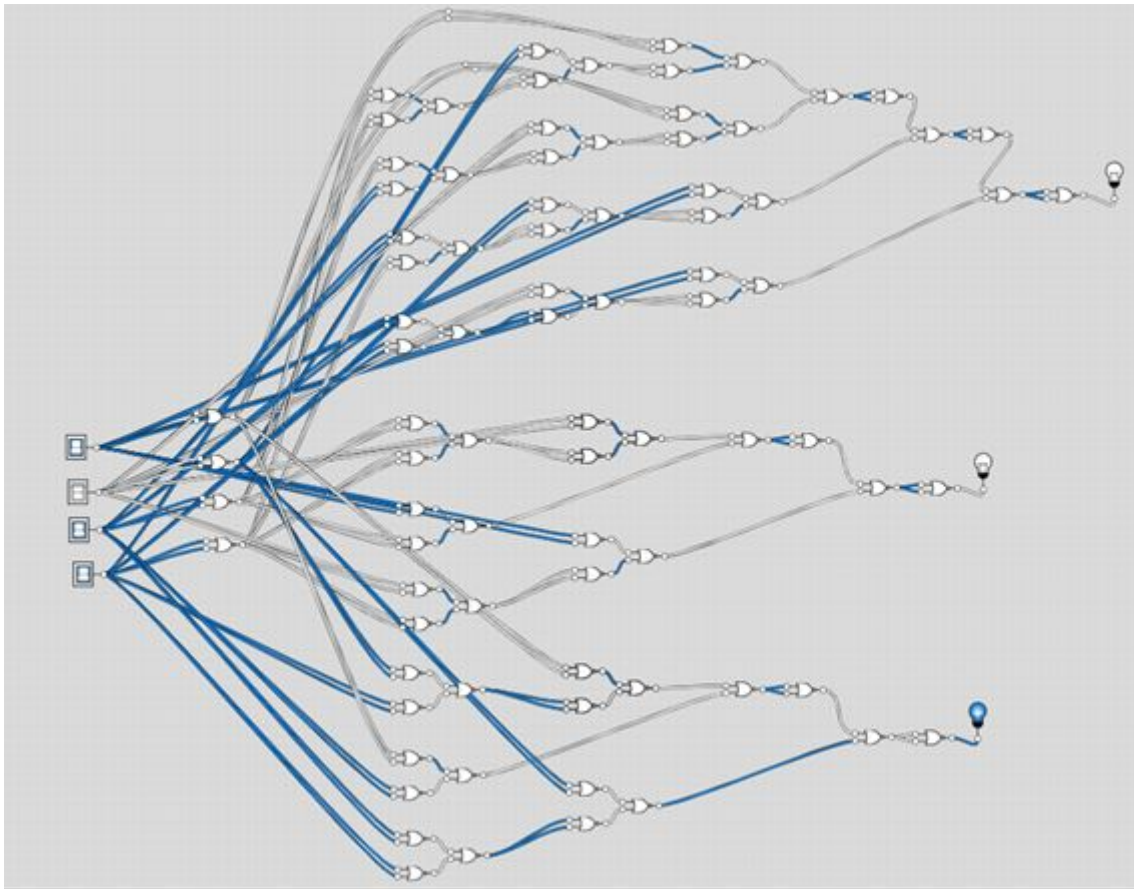


Cuando  $A = 00$  y  $B = 01$  es decir para la salida  $m$ :  $A < B$  se va a cumplir que se enciende el tercer foco



Aquí tenemos otro ejemplo de la misma relación con la salida  $m = A < B$  se prende el foco

Cuando  $A = 10$  y  $B = 11$



## 6. LISTA DE COMPONENTES

1. Para el circuito de valor mayoritario se utilizaron distintos componentes dependiendo del programado utilizado para su desarrollo.

| Programa                     | Componentes                                                                                                                                        |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| logic.ly (Simulador Virtual) | <ul style="list-style-type: none"> <li>• 3 Toggle Switch</li> <li>• 3 AND Gate 2 In</li> <li>• 1 Or Gate 3 In</li> <li>• 1 Light Bulb</li> </ul>   |
| Multisim 14.0                | <ul style="list-style-type: none"> <li>• 3 fuentes de continua 5V</li> <li>• 4 resistencias 330Ω</li> <li>• 3 switch</li> <li>• 74LS08D</li> </ul> |

|           |                                                                                                                                                                                                                                                                                    |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|           | <ul style="list-style-type: none"> <li>• 74LS32D</li> <li>• 1 diodo emisor de luz LED rojo</li> </ul>                                                                                                                                                                              |
| Tinkercad | <ul style="list-style-type: none"> <li>• Generador de voltaje</li> <li>• 1 dip switch de 4 entradas</li> <li>• 4 resistencias de 330Ω</li> <li>• 74HC08</li> <li>• 74HC32</li> <li>• 1 diodo emisor de luz LED rojo</li> <li>• Protoboard</li> <li>• Cables conductores</li> </ul> |

## 7. MAPA DE VARIABLES

1. El circuito a diseñar posee tres entradas, dando como resultado 8 posibles combinaciones. Se tomará una lógica positiva en la salida de las combinaciones.

| A | B | C | Y |
|---|---|---|---|
| 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 |

Como se está trabajando con lógica positiva, se procede a trabajar con minterminos. Visualizando la tabla se puede apreciar que existen cuatro salidas de alto voltaje; se procede a realizar la suma de productos de los términos subrayados, dando como resultado la siguiente expresión.

Expresión:  **$A'BC + AB'C + ABC' + ABC = Y$**

Implementar la expresión anterior consumiría mucho material, por tal motivo se procede a realizar la simplificación correspondiente aplicando los teoremas del álgebra de boole.

Simplificación:

$$A'BC + AB'C + ABC' + ABC = Y$$

$$A'BC + AB'C + AB(C' + C) = Y$$

$$A'BC + AB'C + AB = Y$$

$$A'BC + (AB'C + AB) = Y$$

$$A'BC + AC + AB = Y$$

$$(A'BC + AB) + AC = Y$$

$$AB + AC + BC = Y$$

La expresión simplificada a implementar:  **$AB + AC + BC = Y$**

2. Para ello se siguió los siguientes pasos

Para el circuito clasificador

-Tabla de verdad

| Entradas |      |      |      | Salidas |    |
|----------|------|------|------|---------|----|
| A=B3     | B=B2 | C=B1 | D=B0 | Z2      | Z1 |
| 0        | 0    | 0    | 0    | 0       | 1  |
| 0        | 0    | 0    | 1    | 0       | 1  |
| 0        | 0    | 1    | 0    | 0       | 1  |
| 0        | 0    | 1    | 1    | 1       | 0  |
| 0        | 1    | 0    | 0    | 1       | 0  |
| 0        | 1    | 0    | 1    | 0       | 0  |
| 0        | 1    | 1    | 0    | 0       | 0  |
| 0        | 1    | 1    | 1    | 0       | 0  |
| 1        | 0    | 0    | 0    | 1       | 1  |
| 1        | 0    | 0    | 1    | 1       | 1  |
| 1        | 0    | 1    | 0    | 1       | 1  |
| 1        | 0    | 1    | 1    | X       | X  |
| 1        | 1    | 0    | 0    | X       | X  |
| 1        | 1    | 0    | 1    | X       | X  |
| 1        | 1    | 1    | 0    | X       | X  |
| 1        | 1    | 1    | 1    | X       | X  |

-Construcción de las expresiones algebraicas por medio de minterminos

$$Z2 = A'B'CD + A'BC'D' + AB'C'D' + AB'C'D + AB'CD'$$

$$Z1 = A'B'C'D' + A'B'C'D + A'B'CD' + AB'C'D' + AB'C'D + AB'CD'$$

-Simplificación de expresiones

$$\begin{aligned}
 Z1 &= A'B'C'D' + A'B'C'D + A'B'CD' + AB'C'D' + AB'C'D + AB'CD' \\
 &= A'B'C'(D' + D) + B'CD'(A' + A) + AB'C'(D' + D) \\
 &= A'B'C' + B'CD' + AB'C'
 \end{aligned}$$

$$= B'C'(A'+A)+B'CD'$$

$$= B'C'+B'CD'$$

$$Z1 = B'(C'+D')$$

$$Z2=A'B'CD+A'BC'D'+AB'C'D'+AB'C'D+AB'CD'$$

Simplificando con Mapas de Karnaugh

|      | C'D' | C'D | CD | CD' |
|------|------|-----|----|-----|
| A'B' | 0    | 0   | 1  | 0   |
| A'B  | 1    | 0   | 0  | 0   |
| AB   | X    | X   | X  | X   |
| AB'  | 1    | 1   | X  | 1   |

$$Z2 = A+BC'D'+B'CD$$

Para el decodificador de 7 segmentos

-Tabla de verdad

| Entradas |    | Salidas |   |   |   |   |   |   | Dígito formado |
|----------|----|---------|---|---|---|---|---|---|----------------|
| Z2       | Z1 | A       | B | C | D | E | F | G |                |
| 0        | 1  | 0       | 1 | 1 | 0 | 0 | 0 | 0 | 1              |
| 1        | 0  | 1       | 1 | 0 | 1 | 1 | 0 | 1 | 2              |
| 0        | 0  | 1       | 1 | 1 | 1 | 1 | 1 | 0 | 0              |
| 1        | 1  | 1       | 1 | 1 | 1 | 0 | 0 | 1 | 3              |

-Construcción de expresiones algebraicas por medio de máx términos

$$A = Z2'+Z1$$

$$B = 1$$

$$C = Z2 + Z1'$$

$$D = A = Z2' + Z1$$

$$E = (Z2' + Z1)(Z2 + Z1)$$

$$F = (Z2' + Z1)(Z2 + Z1')(Z2 + Z1)$$

$$G = (Z2' + Z1')(Z2 + Z1)$$

### 3. Circuito

| NÚMERO | A | B | C | D | M (A>B) | m(A<B) | l(A=B) |
|--------|---|---|---|---|---------|--------|--------|
| 0      | 0 | 0 | 0 | 0 | 0       | 0      | 1      |
| 1      | 0 | 0 | 0 | 1 | 0       | 1      | 0      |
| 2      | 0 | 0 | 1 | 0 | 0       | 1      | 0      |
| 3      | 0 | 0 | 1 | 1 | 0       | 1      | 0      |
| 4      | 0 | 1 | 0 | 0 | 1       | 0      | 0      |
| 5      | 0 | 1 | 0 | 1 | 0       | 0      | 1      |
| 6      | 0 | 1 | 1 | 0 | 0       | 1      | 0      |
| 7      | 0 | 1 | 1 | 1 | 0       | 1      | 0      |
| 8      | 1 | 0 | 0 | 0 | 1       | 0      | 0      |
| 9      | 1 | 0 | 0 | 1 | 1       | 0      | 0      |
| 10     | 1 | 0 | 1 | 0 | 0       | 0      | 1      |
| 11     | 1 | 0 | 1 | 1 | 0       | 1      | 0      |
| 12     | 1 | 1 | 0 | 0 | 1       | 0      | 0      |
| 13     | 1 | 1 | 0 | 1 | 1       | 0      | 1      |
| 14     | 1 | 1 | 1 | 0 | 1       | 0      | 1      |
| 15     | 1 | 1 | 1 | 1 | 0       | 0      | 1      |

| B1 | B2 | NÚMERO |
|----|----|--------|
| 0  | 0  | 0      |
| 0  | 1  | 1      |



|   |   |   |
|---|---|---|
| 1 | 0 | 2 |
| 1 | 1 | 3 |

|    |    |    |    |
|----|----|----|----|
| A  | B  | C  | D  |
| A0 | A1 | B0 | B1 |

M

$$M = A'BC'D' + AB'C'D' + AB'C'D + ABC'D' + ABC'D + ABCD'$$

$$M = A'BC'D' + B'C'(AD' + AD) + AB(C'D' + C'D + CD')$$

$$M = A'BC'D' + B'C'(A) + AB(C'(D' + D) + CD')$$

$$M = A'BC'D' + AB'C' + AB(C'(1) + CD')$$

$$M = A'BC'D' + ABCD' + AB'C' + ABC'$$

$$M = (A'C' + AC)BD' + (AB' + AB)C'$$

$$M = (A'C' + AC)BD' + (A)C'$$

$$M = A'C'BD' + ACBD' + AC'$$

$$M = A'C'BD' + A(CBD' + C')$$

$$M = A'C'BD' + A(BD' + C')$$

$$M = A'C'BD' + ABD' + AC'$$

$$M = ABD' + C'(A + A'BD')$$

$$M = ABD' + C'(A + BD')$$

$$M = ABD' + AC' + BC'D'$$

$$M = AC' + ABD' + BC'D'$$

m

$$m = A'B'C'D + A'B'CD' + A'B'CD + A'BCD' + A'BCD + AB'CD$$

$$m = A'B'(C'D + CD' + CD) + A'BC(D' + D) + AB'CD$$

$$m=A'B'(C'D+C(D'+D))+A'BC(1)+AB'CD$$

$$m=A'B'(C'D+C(1))+A'BC(1)+AB'CD$$

$$m=A'B'(C'D+C)+A'BC+AB'CD$$

$$m=A'B'(D+C)+A'BC+AB'CD$$

$$m=A'B'D+A'B'C+A'BC+AB'CD$$

$$m=B'D(A'+AC)+A'C(B'+B)$$

$$m=B'D(A'+C)+A'C(1)$$

$$m=A'B'D+B'CD+A'C$$

I

$$I=A'B'C'D'+A'BC'D+AB'CD'+ABCD$$

## 8. EXPLICACIÓN DEL CÓDIGO FUENTE

En este punto se debe explicar cómo funcionan la implementación del programa, explicando los valores que requiere y los valores que devuelve.

### 1) Circuito Votador

#### 1.1 Simulación:

Para la implementación del diseño se utilizó los circuitos integrados 74LS08D y 74LS32D. La primera compuerta del integrado 74LS08D se encuentran conectadas a los switch A y B, la salida se encuentra conectada a la primera entrada de la primera compuerta del integrado 74LS32D.

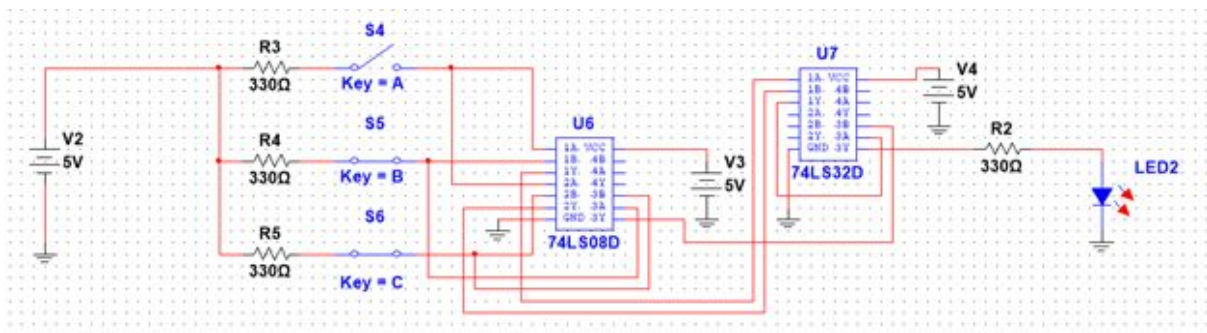
La segunda compuerta del integrado 74LS08D se encuentra conectado a los switch A y C, la salida se encuentra conectada a la segunda entrada de la primera compuerta del integrado 74LS32D. La salida de la primera compuerta del integrado 74LS32D se encuentra conectada a la primera entrada de la tercera compuerta del mismo integrado.

Por último la tercera compuerta del integrado 74LS08D se encuentra conectado a los switch B y C, y la salida se encuentra conectada a la segunda entrada de la tercera compuerta del integrado 74LS32D. La salida de la tercera compuerta se conecta a la resistencia de 330Ω en serie con el LED rojo.

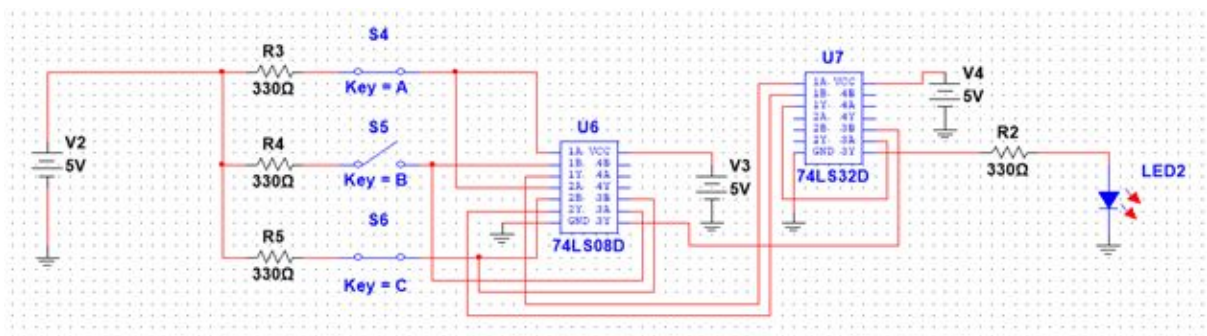
Se utilizó dos compuertas OR debido a que comercialmente no existe ningún integrado OR 3 In, por tanto primero se realiza la operación lógica de los primeros productos para que el resultado sea sumado con el otro producto restante. La utilización de las dos compuertas OR no afectó la lógica del diseño debido a lo realizado se justifica aplicando el teorema asociativo del álgebra de boole.

Se representará en la simulación el resultado que arrojan los valores de alto voltaje que daba la tabla de verdad del diseño.

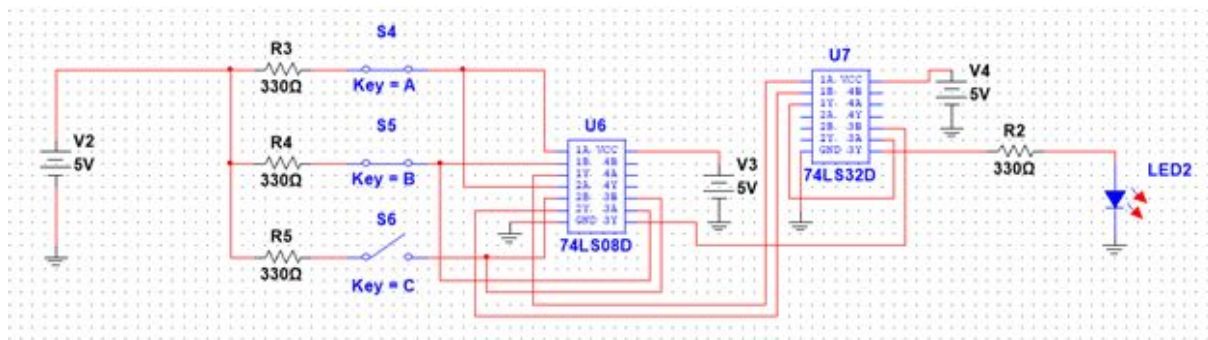
La imagen mostrada a continuación es el resultado de la simulación de la entrada **011**. Como se puede visualizar cuando B y C se encuentran cerrados, el LED rojo se enciende.



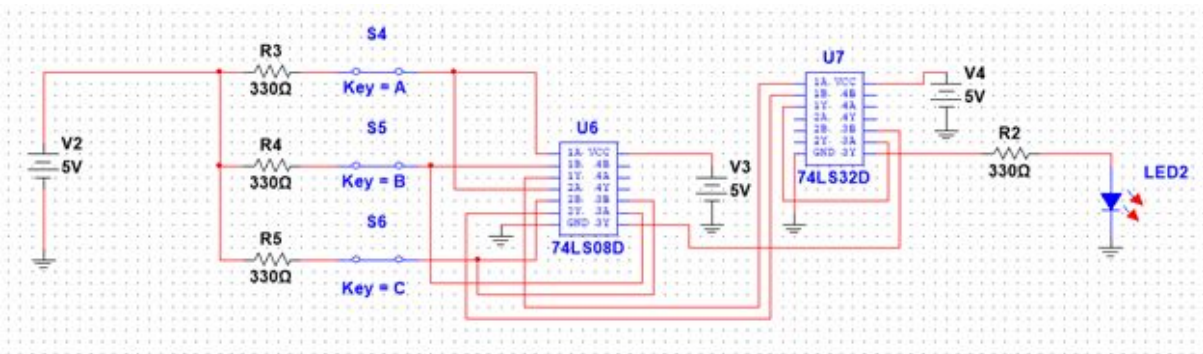
La imagen mostrada a continuación es el resultado de la simulación de la entrada **101**. Como se puede visualizar cuando A y C se encuentran cerrados, el LED rojo se enciende.



La imagen mostrada a continuación es el resultado de la simulación de la entrada **110**. Como se puede visualizar cuando A y B se encuentran cerrados, el LED rojo se enciende.



La imagen mostrada a continuación es el resultado de la simulación de la entrada **111**. Como se puede visualizar cuando A y B y C se encuentran cerrados, el LED rojo se enciende.



#### Laboratorio virtual:

Para la implementación del diseño en el laboratorio virtual primero se alimentó a la protoboard. Se conectaron tres resistencias de 330Ω en las filas donde se conectó el dip switch. Para el laboratorio virtual se tiene la siguiente consideración: A=1, B=2 y C=3, teniendo clara la consideración anterior se procederá a explicar la conexión del dip switch con los circuitos integrados utilizados.

Para las conexiones de la expresión  $A \cdot B$  se utilizó cables conductores de color verde. El switch 1 se encuentra conectado a la primera entrada de la primera compuerta del integrado 74HC08, mientras el switch 2 se encuentra conectado a la segunda entrada de la primera compuerta del integrado 74HC08. La salida se encuentra conectada a la primera entrada de la primera compuerta del integrado 74HC32.

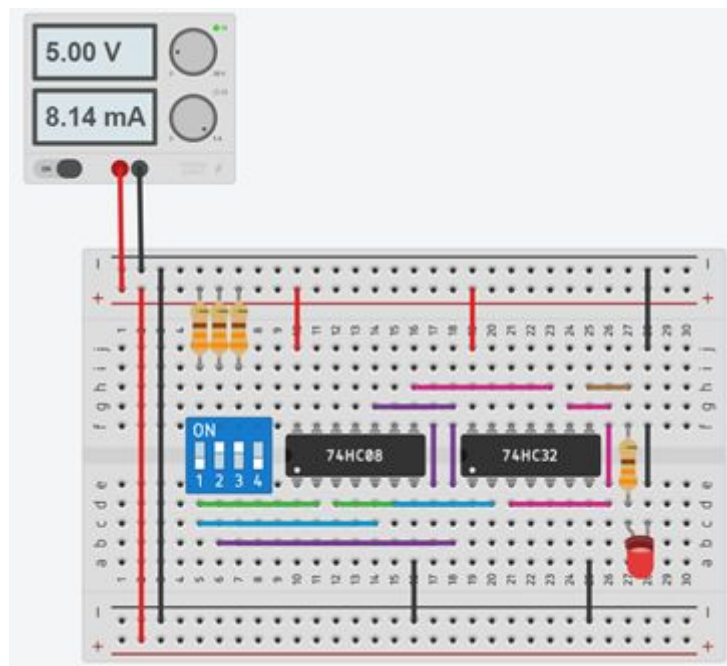
Para las conexiones de la expresión  $A \cdot C$  se utilizó cables conductores de color azul. El switch 1 se encuentra conectado a la primera entrada de la segunda compuerta del integrado 74HC08, mientras el switch 3 se encuentra conectado a la segunda entrada de la segunda compuerta del integrado 74HC08. La salida se encuentra conectada a la segunda entrada de la primera compuerta del integrado 74HC32.

Para las conexiones de la expresión B\*C se utilizó cables conductores de color morado. El switch 2 se encuentra conectado al pin 17b, mientras el switch 3 se encuentra conectado al pin 18b. Se realiza un puente entre las dos ranuras debido a que las entradas de la tercera compuerta del integrado 74HC08 se encuentran del otro lado. Realizado el puente entre ranuras se procede a conectar las cada uno de los switch con sus respectivas entradas. La salida se encuentra conectada a la segunda entrada de la tercera compuerta del integrado 74HC32.

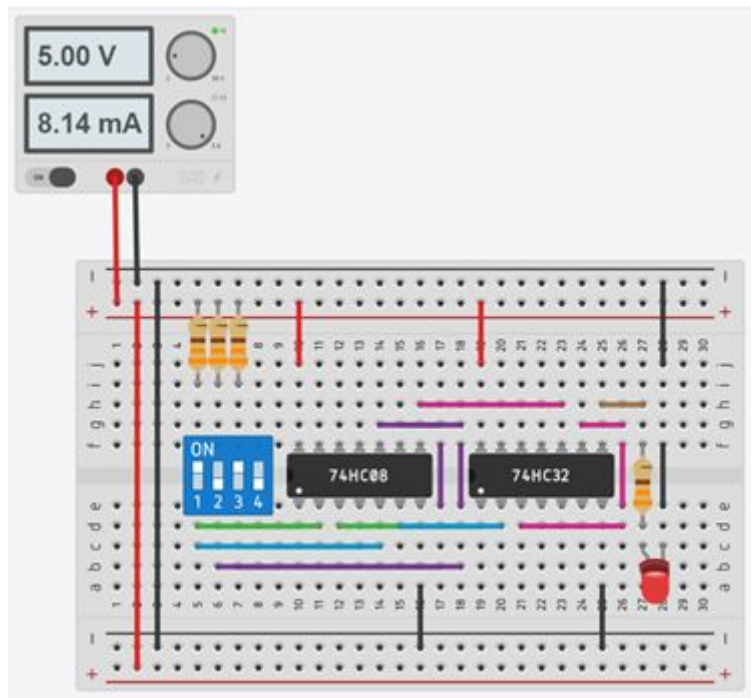
La salida de la primera compuerta del integrado 74HC32 es conectada al pin 26d, se vuelve a realizar un puente entre las ranuras para poder conectar la salida con la primera entrada de la tercera compuerta del mismo integrado. Esta conexión está representada con cables conectores de color rosado.

Por último la salida de la tercera compuerta del integrado 74HC32 se conecta a la resistencia de  $330\Omega$  en serie con el LED rojo. Para esta conexión se utilizó un cable conector de color café.

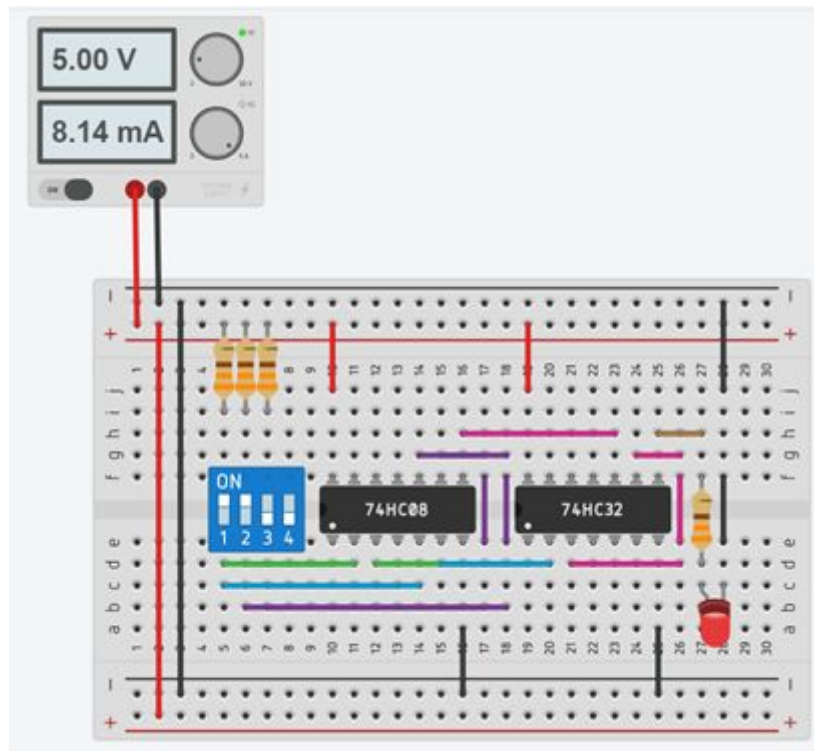
La imagen mostrada a continuación es el resultado de la simulación de la entrada **011**. Como se puede visualizar cuando los switch 2 y 3 se encuentran cerrados, el LED rojo se enciende.



La imagen mostrada a continuación es el resultado de la simulación de la entrada **101**. Como se puede visualizar cuando los switch 1 y 3 se encuentran cerrados, el LED rojo se enciende.

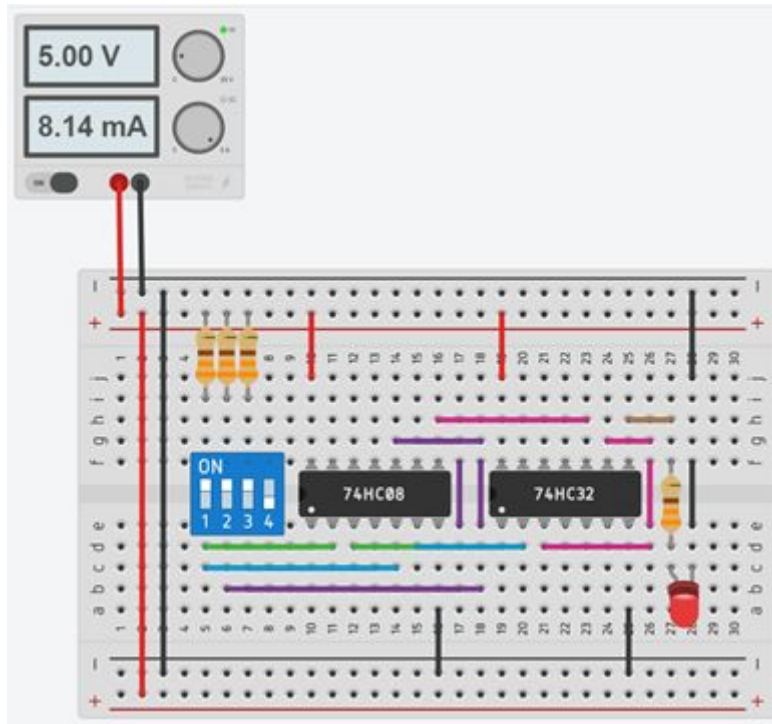


La imagen mostrada a continuación es el resultado de la simulación de la entrada **110**. Como se puede visualizar cuando los switch 1 y 2 se encuentran cerrados, el LED rojo se enciende.





La imagen mostrada a continuación es el resultado de la simulación de la entrada **111**. Como se puede visualizar cuando los switch 1, 2 y 3 se encuentran cerrados, el LED rojo se enciende.

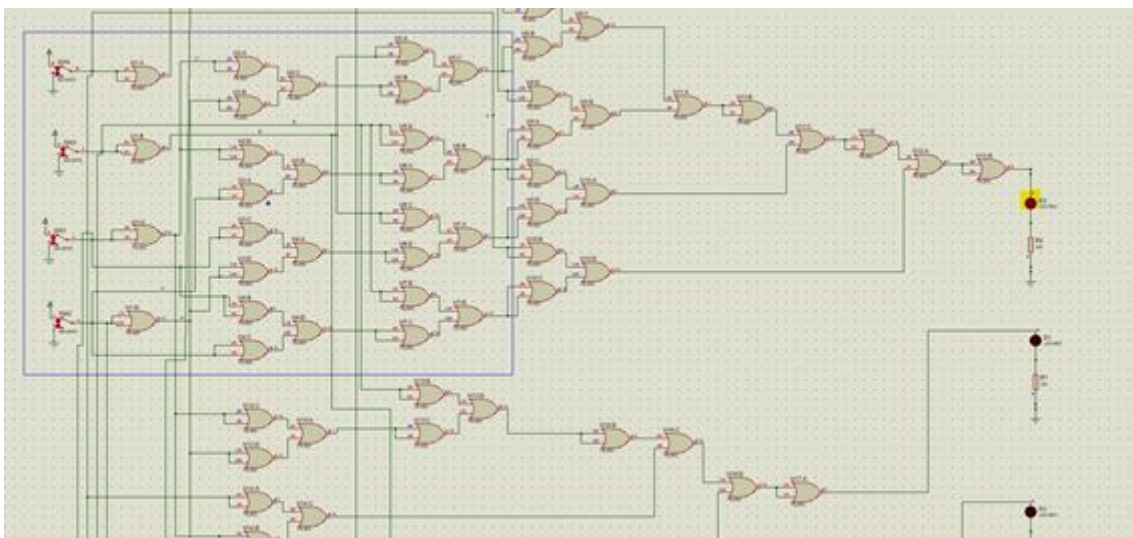


2. Circuito

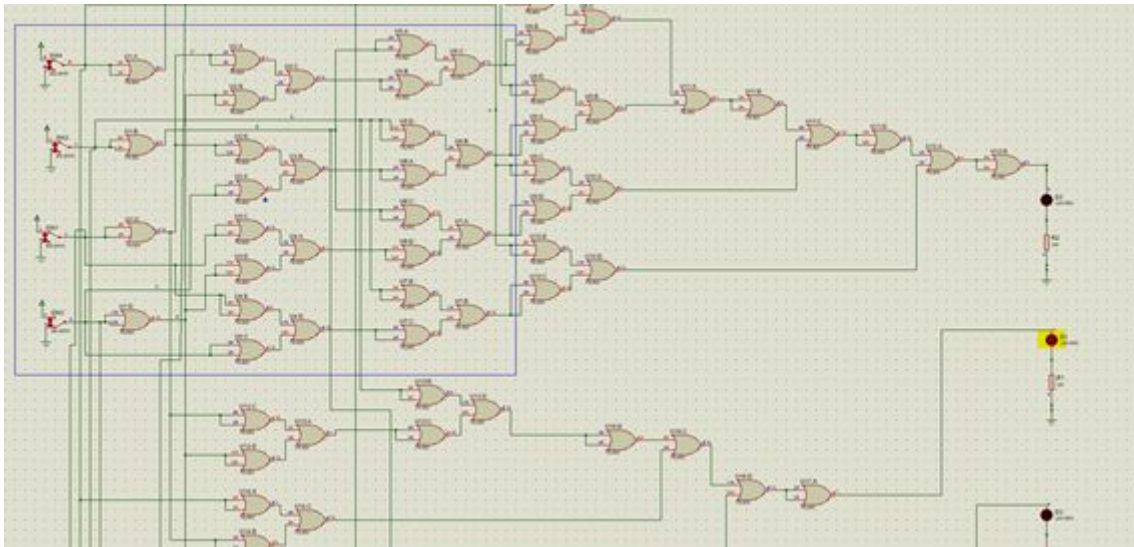
3. Circuito

Simulación:

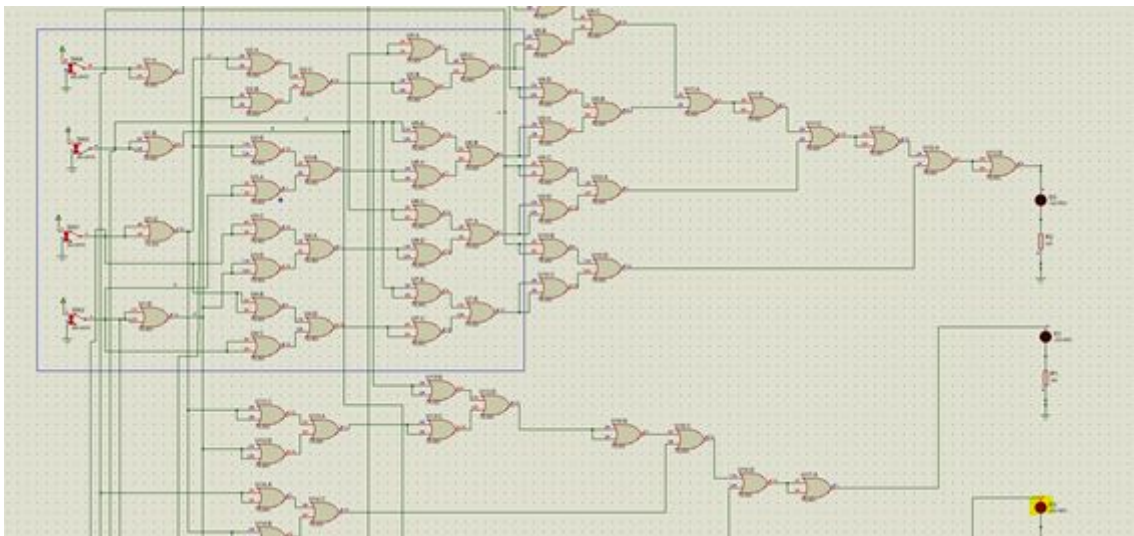
para mi salida I:  $A=B$  entonces el primer foco o en este caso LED se prendera  
Cuando  $A=11$  y  $B=11$



para mi salida M:  $A > B$  entonces el segundo foco o en este caso LED se prendera  
Cuando  $A=11$  y  $B=10$



para mi salida m:  $A < B$  entonces el tercer foco o en este caso LED se prendera  
Cuando  $A=10$  y  $B=11$



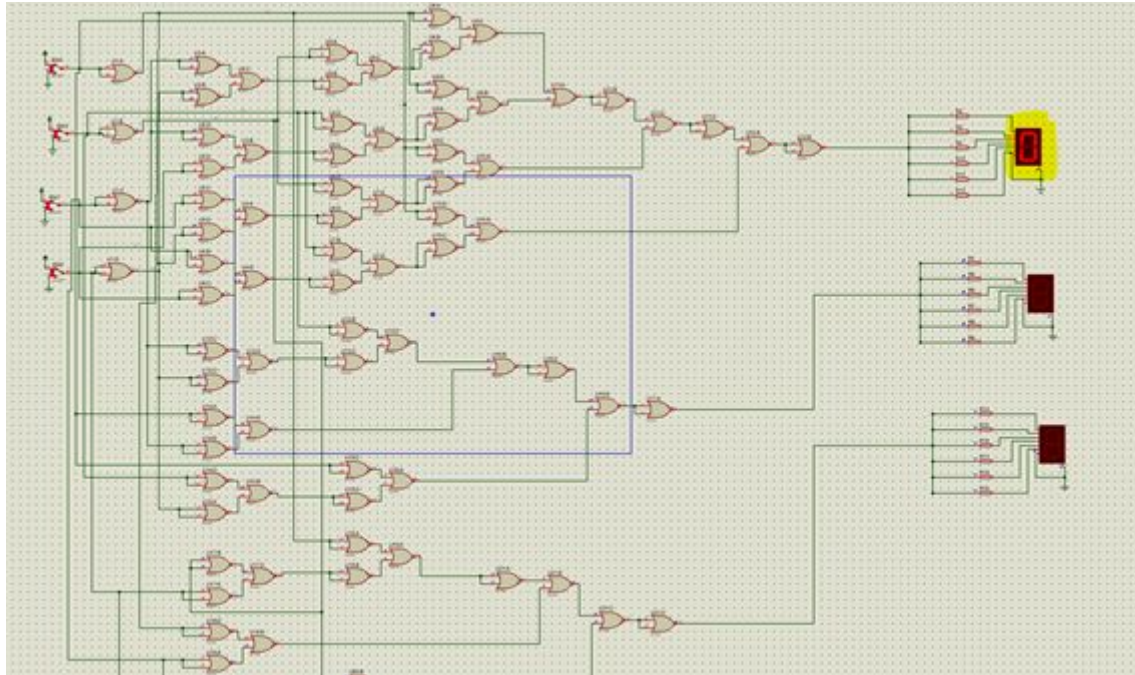
Para el tercer enunciado diseñe un módulo adicional que permita visualizar en un display de 7 segmentos conectado a cada salida el número 0 si han ocurrido uno de los 3 casos

Aplicando la misma metodología que el enunciado anterior



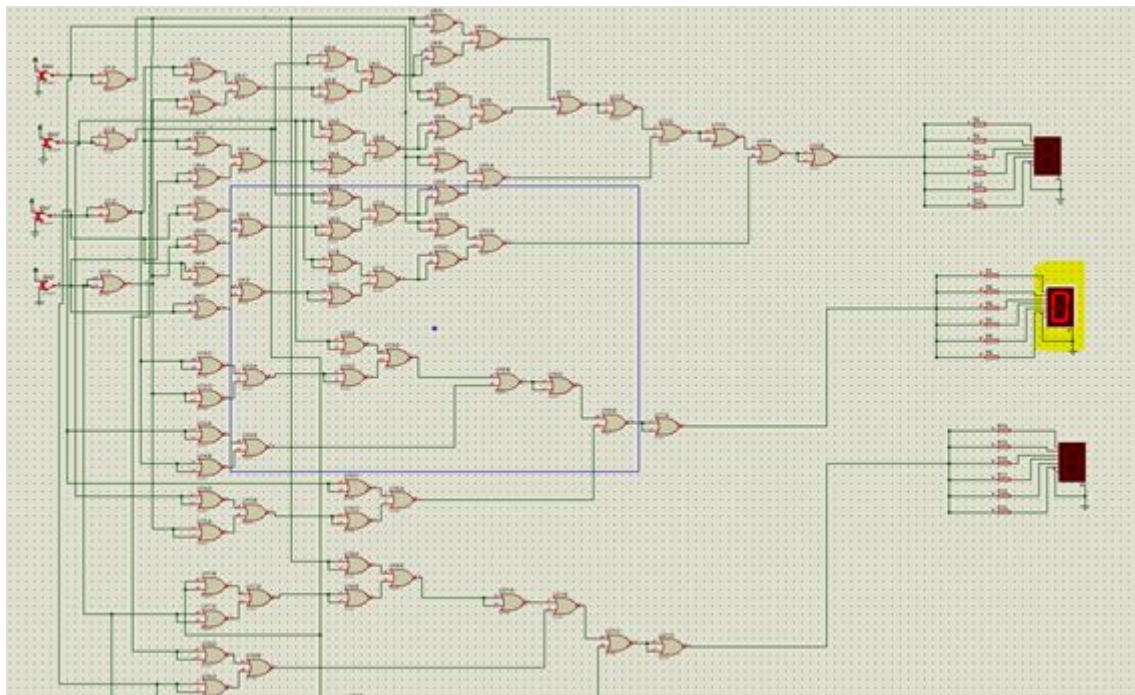
para mi salida I:  $A=B$  entonces el primer Display se prendera

Cuando  $A=11$  y  $B=11$

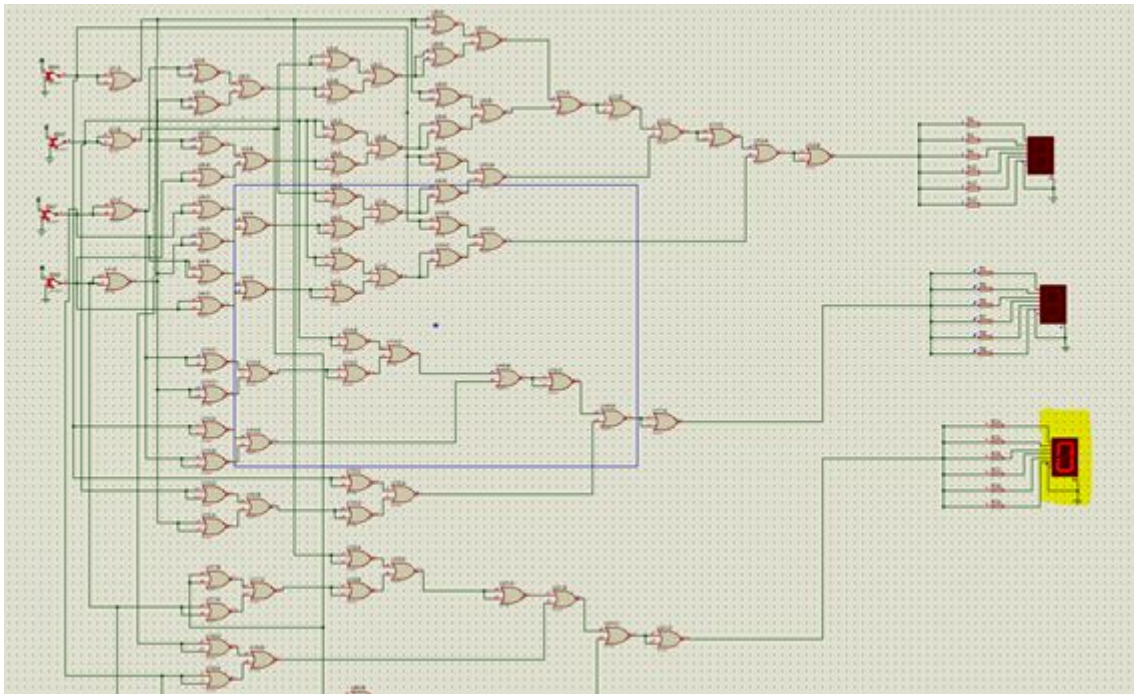


para mi salida M:  $A>B$  entonces el segundo Display se prendera

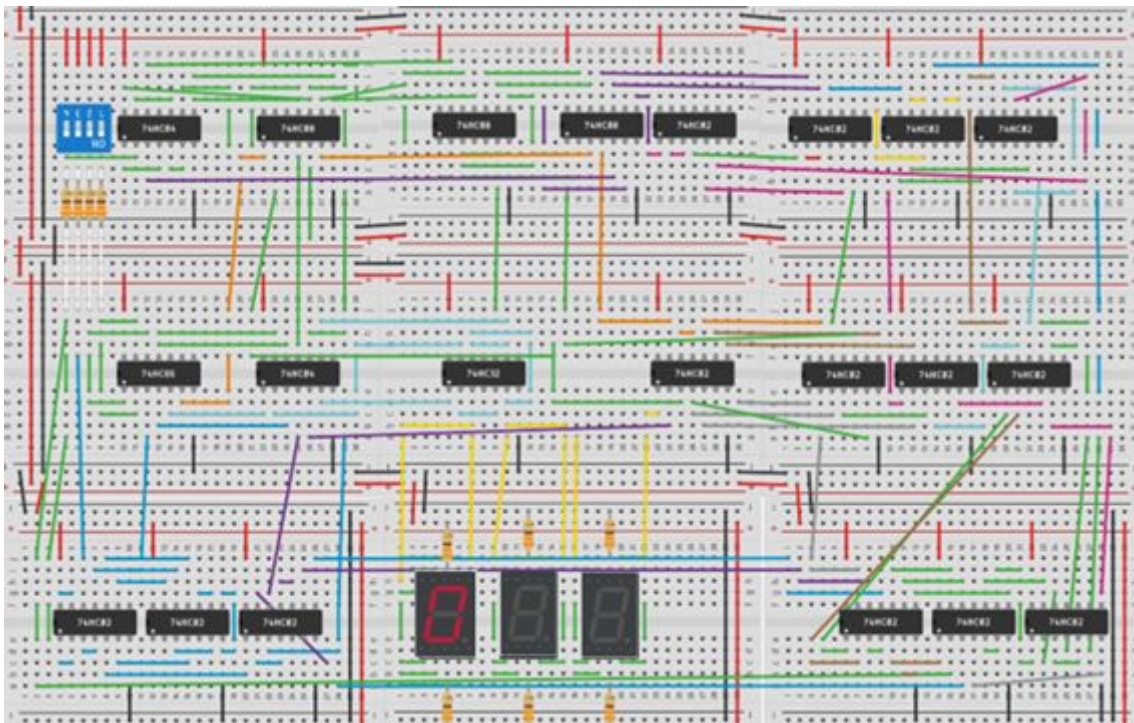
Cuando  $A=11$  y  $B=10$



Cuando  $A=10$  y  $B=11$

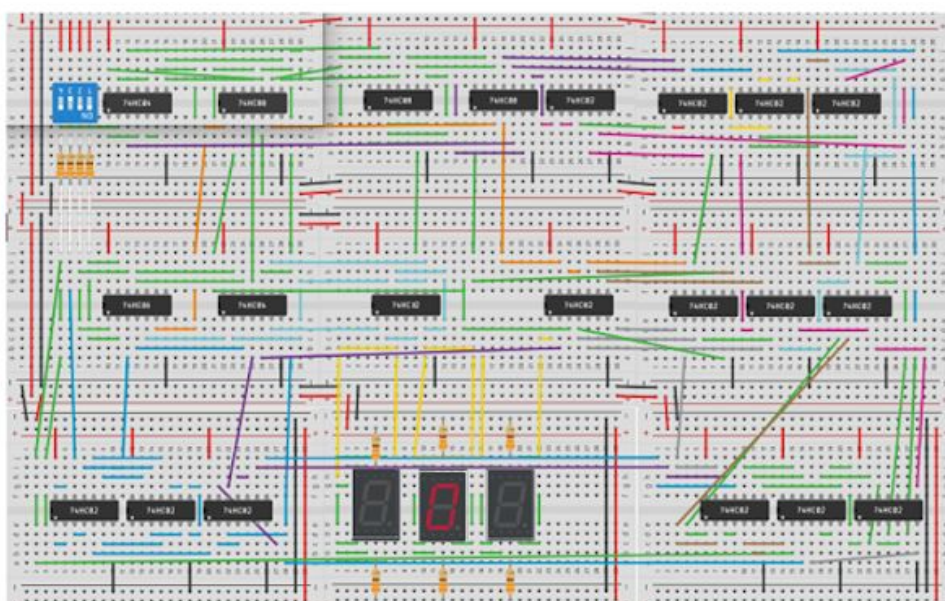


Laboratorio Virtual:  
 Cuando  $A=11$  y  $B=11$

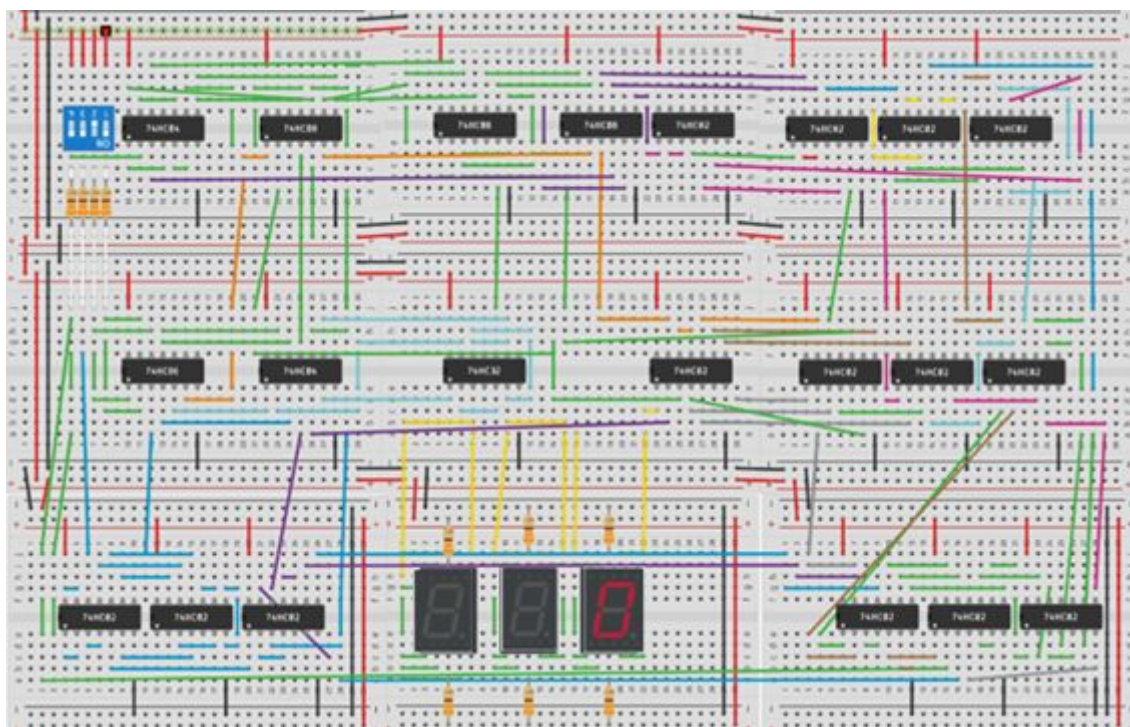


Cuando  $A=11$  y  $B=10$





Cuando A=10 y B=11



Desarrollo del ejercicio 2 en App Inventor

La pantalla de inicio se presenta de la siguiente forma:

## 9. DESCRIPCIÓN DE PRERREQUISITOS Y CONFIGURACIÓN

Los elementos necesarios para el proceso de diseño, codificación y pruebas son:

## Ordenador y el sistema operativo

- Macintosh (con procesador Intel): Mac OS X 10.5, 10.6
- De Windows: Windows XP, Windows Vista, Windows 7
- GNU / Linux: Ubuntu 8 +, Debian 5 +

## Navegador

- Mozilla Firefox 3.6 o superior
- Apple Safari 5.0 o superior
- Google Chrome 4.0 o alta
- App Inventor no soporta Microsoft Internet Explorer. Los usuarios de Windows deben utilizar Chrome o Firefox

## Teléfono

Las aplicaciones creadas con App Inventor pueden funcionar en cualquier teléfono Android, siempre que cuenten instalados con una tarjeta SD. El entorno de desarrollo y el propio software de configuración soportan directamente los siguientes teléfonos:

- Google: Nexus S, Nexus One, ADP1, ADP2, Ion
- T-Mobile: G1 \*, myTouch 3G \*
- Motorola: Droid \*
- hardware similar en otras compañías

## 10. CONCLUSIONES

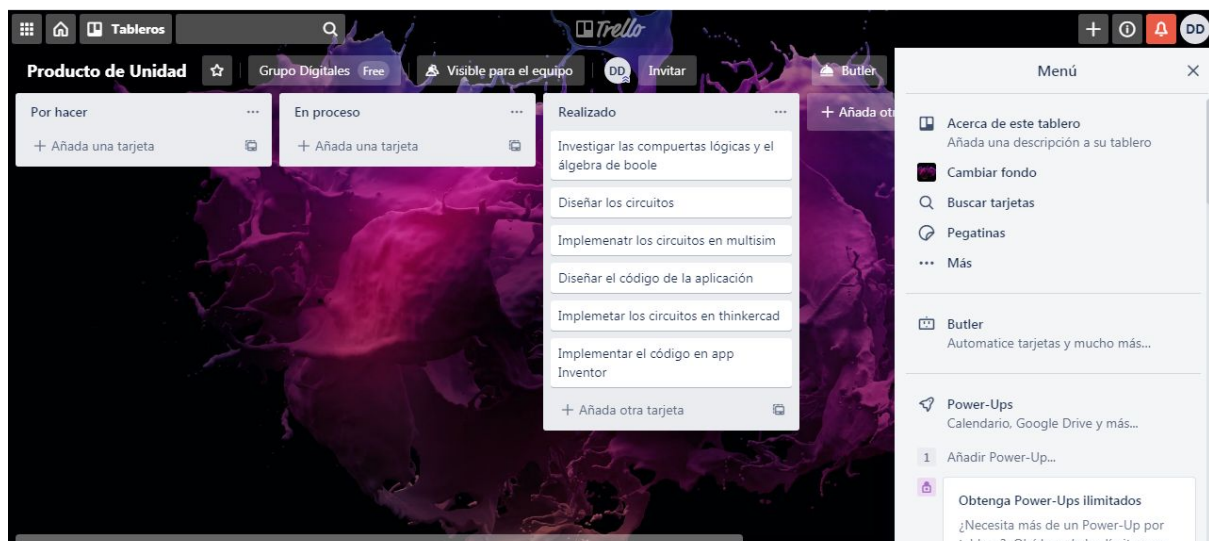
1. Los estudiantes de Tercer semestre de la carrera de Ingeniería en Telecomunicaciones del período mayo-septiembre 2020 han reforzado sus conocimientos sobre los circuitos digitales mediante la óptima resolución de los problemas planteados a lo largo del presente documento.
2. El circuito votador diseñado fue un circuito bastante sencillo de realizar, por lo que comprender su funcionamiento es ideal para tener claro los conceptos de los circuitos digitales tales como el álgebra de boole y las compuertas lógicas.

3. El circuito diseñado para la generación de turnos permite dar a conocer el estado actual por medio del ingreso un número expresado como switches. Sea solo con compuertas NAND, NOR o las fundamentales, AND, OR NOT, el circuito implementado quedaba en su máxima expresión.

## 11.RECOMENDACIONES

1. Se recomienda tener claro cuales van a ser las entradas que va a recibir la compuerta lógica, esto con la finalidad de reducir la cantidad de compuertas que se vaya a implementar.
2. Otro paso es la decodificación del display de 7 segmentos, es la generación de otro proceso, donde la salida del problema llega a ser la entradas del display. También se obtienen las ecuaciones características, siendo simplificado con el Álgebra de Boole y mapas de Karnaugh.

## 12.CRONOGRAMA



## 13.BIBLIOGRAFÍA

Álgebra Booleana — MecatrónicaLATAM. (n.d.). Retrieved June 22, 2020, from <https://www.mecatronicalatam.com/es/tutoriales/teoria/algebra-booleana/>

Archivo:Puerta OR.svg - Wikipedia, la enciclopedia libre. (n.d.). Retrieved June 22, 2020, from [https://es.wikipedia.org/wiki/Archivo:Puerta\\_OR.svg](https://es.wikipedia.org/wiki/Archivo:Puerta_OR.svg)

Diseña con Tinkercad - EDUCACIÓN Y TECNOLOGÍA. (n.d.). Retrieved June 22, 2020, from <https://www.educoteca.com/tinkercad.html>

File:AND ANSI.svg - Wikimedia Commons. (n.d.). Retrieved June 22, 2020, from [https://commons.wikimedia.org/wiki/File:AND\\_ANSI.svg](https://commons.wikimedia.org/wiki/File:AND_ANSI.svg)

File:NOT ANSI.svg - Wikimedia Commons. (n.d.). Retrieved June 22, 2020, from [https://commons.wikimedia.org/wiki/File:NOT\\_ANSI.svg](https://commons.wikimedia.org/wiki/File:NOT_ANSI.svg)

Lógica NAND - Wikipedia, la enciclopedia libre. (n.d.). Retrieved June 22, 2020, from [https://es.wikipedia.org/wiki/Lógica\\_NAND](https://es.wikipedia.org/wiki/Lógica_NAND)

Lógica NOR - Wikipedia, la enciclopedia libre. (n.d.). Retrieved June 22, 2020, from [https://es.wikipedia.org/wiki/Lógica\\_NOR](https://es.wikipedia.org/wiki/Lógica_NOR)

Puerta XOR - Wikiwand. (n.d.). Retrieved June 22, 2020, from [https://www.wikiwand.com/es/Puerta\\_XOR](https://www.wikiwand.com/es/Puerta_XOR)

XNOR gate XOR gate NAND gate Logic gate, symbol, miscellaneous, angle, text png | PNGWing. (n.d.). Retrieved June 22, 2020, from <https://www.pngwing.com/en/free-png-paxii>

## **14. ANEXOS**

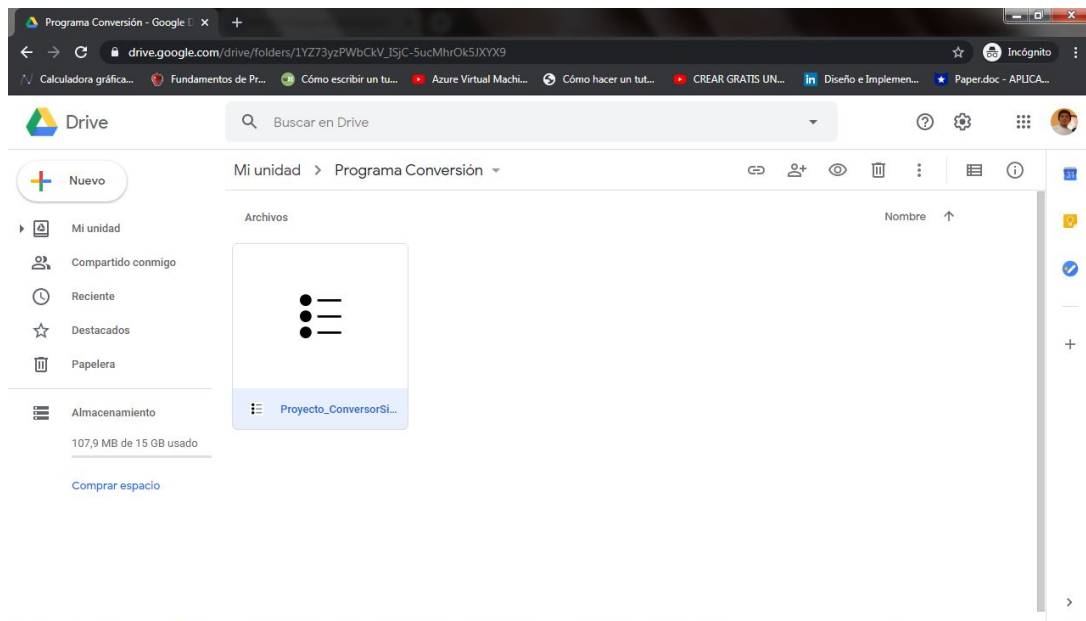
### **15.1 MANUAL DE USUARIO**

#### **15.1.1 Requerimientos**

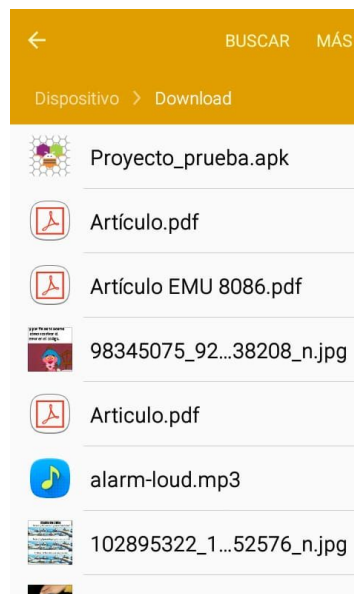
- 1) Conexión a Internet
- 2) 5 MB de memoria disponible
- 3) Teléfono con sistema operativo Android

#### **15.1.2 Instalación**

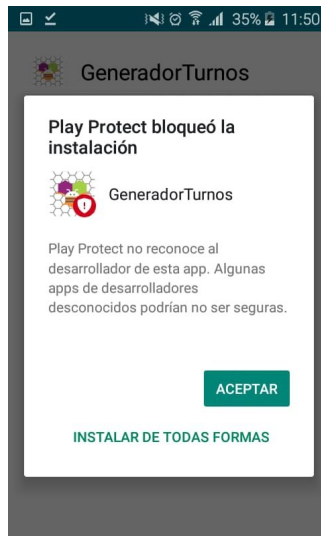
Esta aplicación no se encuentra en Google Play, por lo que se especifica la dirección de descarga.



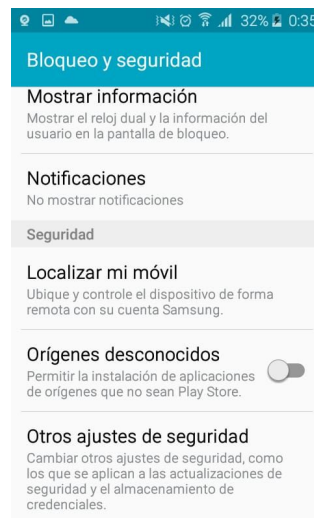
El link anterior permite acceder de forma directa al archivo de extensión APK que contiene la aplicación. Una vez descargado el archivo se procede a ejecutarlo.



Por defecto el sistema Android no permite la instalación de aplicaciones que no sean descargadas a través de la Play Store, por lo tanto cuando se ejecute el programa saltará un mensaje que avisa al usuario que la instalación, no es posible.

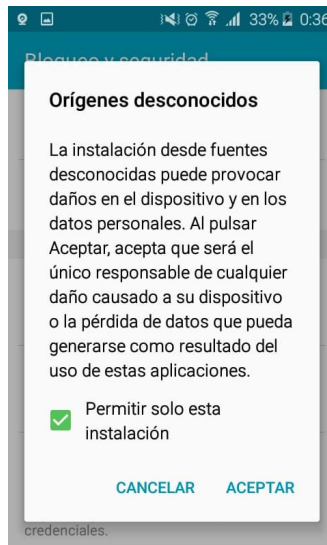


En la parte inferior se encuentran las opciones 'Cancelar' y 'Ajustes'. Se selecciona la opción 'Ajustes', seguido se muestra el menú de opciones de 'Bloqueo y Seguridad'.



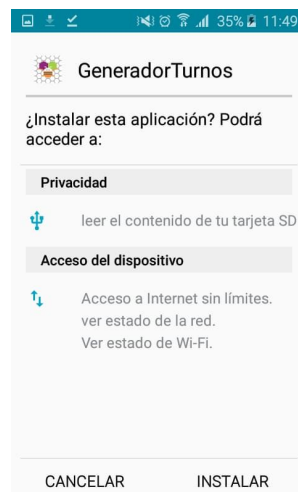
Se selecciona la opción 'Orígenes desconocidos', donde nos sale el siguiente mensaje:



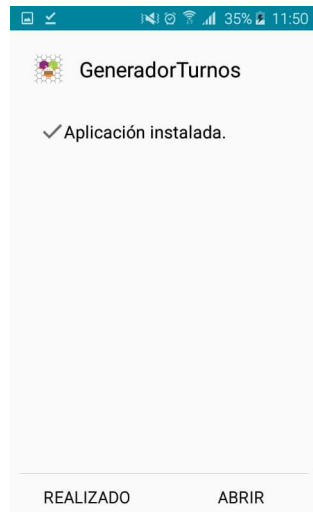
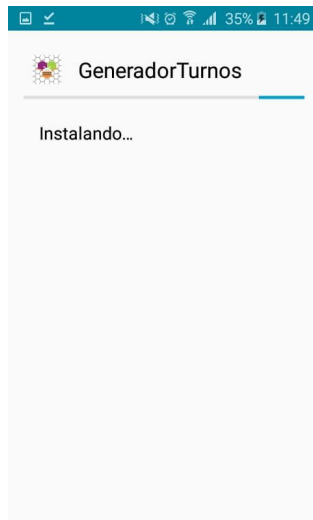


Se selecciona la opción 'Permitir sólo esta instalación' y se da click en el botón 'Aceptar'.

A continuación aparecerá el siguiente mensaje, donde se debe seleccionar la opción 'Instalar', para que dé comienzo a la instalación normal de la aplicación.

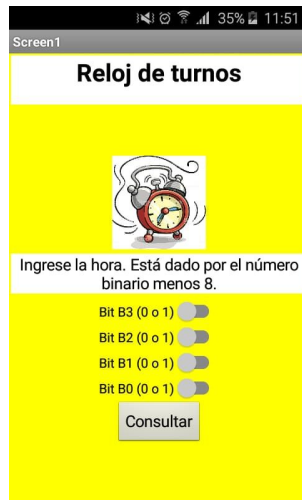


Una vez terminado todo el proceso de instalación aparecerá la siguiente pantalla donde se selecciona la opción 'Abrir', que nos conducirá a la pantalla de bienvenida de la aplicación.



### 15.1.3 Ejecución del programa

Lo primero en aparecer es la pantalla de Consulta donde se muestra el nombre de la aplicación 'Reloj de turnos'. A continuación se le pide al usuario que seleccione el horario en el que se encuentra, este se encuentra dado por el numero binario menos 8.



Seleccionado cualquiera de los cuatro campos existentes, como se muestra en la imagen, lo que aparece a continuación es la pantalla de resultado.



La pantalla de respuesta o pantalla de salida muestra el horario al que el usuario pertenece, además de eso aparecen los botones 'Realizar nueva acción' y 'Salir'.



Se encuentra en el turno

1

Realizar nueva acción

Salir

El primer botón permite al usuario regresar a la pantalla de consulta y el segundo botón permite al usuario salir definitivamente de la aplicación.



Se encuentra en el turno

2

Realizar nueva acción

Salir

## 15.2 HOJAS TÉCNICAS