

Trabajo práctico N°1

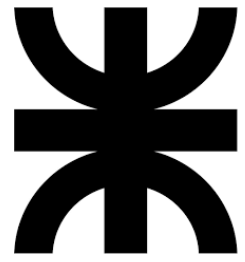
■ **Autores:**

- Gonzalo Ezequiel Filsinger - Leg. 403797
- Ignacio Ismael Perea - Leg. 406265
- Manuel Leon Parfait - Leg. 406599

■ **Curso:** 3R1

■ **Asignatura:** Tecnicas Digitales.

■ **Institución:** Universidad Tecnológica Nacional - Facultad Regional de Córdoba.



U
T
N

F
R
C

Índice

1. Actividad 2.1: Conversor de BCD a Exceso-3	1
1.1. Tabla de verdad	1
1.2. Funciones lógicas de salida	1
1.3. Minimización por mapas de Karnaugh	1
1.4. Circuito lógico en Falstad	2
2. Actividad 2.2: Comparador Binario	2
2.1. Tabla de verdad	2
2.2. Funciones lógicas de salida	3
2.3. Minimización por mapas de Karnaugh	3
2.4. Circuito lógico en Falstad	3
3. Imágenes de Circuitos	4
3.1. Circuito 2.1	4
3.2. Circuito 2.2	4

1. Actividad 2.1: Conversor de BCD a Exceso-3

Diseñar y armar un conversor de código BCD a XS3 (exceso 3). Realizar:

- Tabla de verdad.
- Obtener las funciones lógicas de salidas con circuitos combinacionales.
- Minimizar las funciones canónicas obtenidas de la tabla de verdad.
- Armar el circuito y verificar su funcionamiento en el MiniLab.
- Armar el circuito y verificar su funcionamiento en el simulador "falstad.com"

1.1. Tabla de verdad

A	B	C	D	W	X	Y	Z
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	x	x	x	x
1	0	1	1	x	x	x	x
1	1	0	0	x	x	x	x
1	1	0	1	x	x	x	x
1	1	1	0	x	x	x	x
1	1	1	1	x	x	x	x

Tabla de verdad para el conversor BCD a Exceso-3

1.2. Funciones lógicas de salida

- $W = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot D + \bar{A} \cdot \bar{B} \cdot C \cdot \bar{D} + \bar{A} \cdot B \cdot C \cdot D + A \cdot \bar{B} \cdot \bar{C} \cdot \bar{D} + A \cdot \bar{B} \cdot C \cdot D$
- $X = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot D + \bar{A} \cdot \bar{B} \cdot C \cdot \bar{D} + \bar{A} \cdot B \cdot C \cdot D + \bar{A} \cdot B \cdot \bar{C} \cdot \bar{D} + A \cdot \bar{B} \cdot \bar{C} \cdot D$
- $Y = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot \bar{D} + \bar{A} \cdot \bar{B} \cdot C \cdot D + \bar{A} \cdot B \cdot \bar{C} \cdot \bar{D} + \bar{A} \cdot B \cdot C \cdot D + A \cdot \bar{B} \cdot \bar{C} \cdot \bar{D}$
- $Z = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot \bar{D} + \bar{A} \cdot \bar{B} \cdot C \cdot \bar{D} + \bar{A} \cdot B \cdot \bar{C} \cdot \bar{D} + \bar{A} \cdot B \cdot C \cdot \bar{D} + A \cdot \bar{B} \cdot \bar{C} \cdot \bar{D}$

1.3. Minimización por mapas de Karnaugh

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

Salida W

$$W = B \cdot D + B \cdot C + A$$

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

Salida X

$$X = B \cdot \bar{C} \cdot \bar{D} + \bar{B} \cdot D + \bar{B} \cdot C$$

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

Salida Y

$$Y = \bar{C} \cdot \bar{D} + C \cdot D$$

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

Salida Z

$$W = \bar{D}$$

1.4. Circuito lógico en Falstad

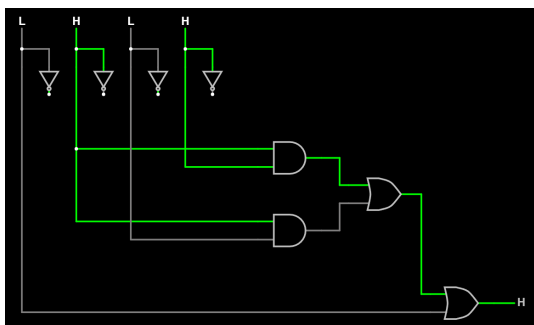


Figura 1: Circuito lógico implementado en Falstad para W
Bits usados para ejemplo 0101.

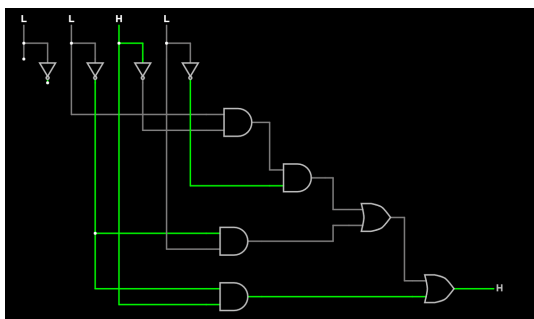


Figura 2: Circuito lógico implementado en Falstad para X
Bits usados para ejemplo 0010.

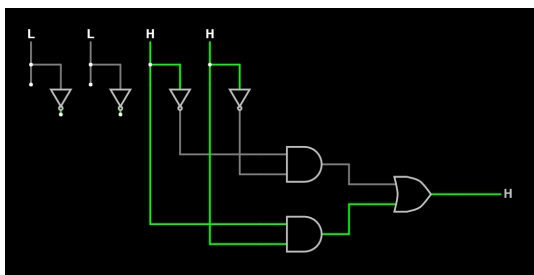


Figura 3: Circuito lógico implementado en Falstad para Y
Bits usados para ejemplo 0011.

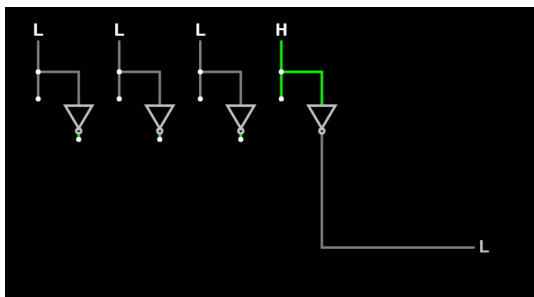


Figura 4: Circuito lógico implementado en Falstad para Z
Bits usados para ejemplo 0001.

2. Actividad 2.2: Comparador Binario

El circuito de la figura es un comparador binario de dos números (A y B) de dos bits cada uno. Las salidas (S_0 , S_1 y S_2) representan la salida del comparador y $S_0 = 1$ cuando $A > B$, $S_1 = 1$ cuando $A < B$ y $S_2 = 1$ $A = B$, en caso de no darse la condición la salida permanece en cero.

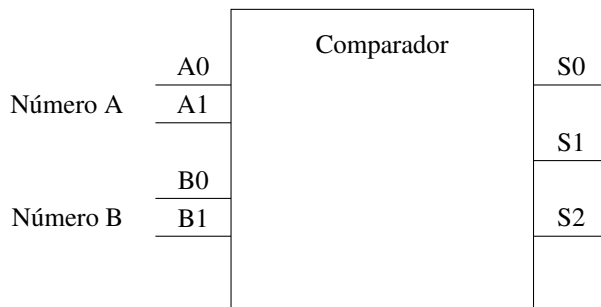


Figura 5: Comparador binario de dos bits

Realizar:

- Tabla de verdad.
- Obtener las funciones lógicas de salidas con circuitos combinacionales.
- Minimizar utilizando mapas de Karnaugh.
- Minimizar utilizando los teoremas y postulados del álgebra de Boole.
- Armar el circuito y verificar su funcionamiento en el MiniLab.
- Armar el circuito y verificar su funcionamiento en el simulador “falstad.com”

2.1. Tabla de verdad

A1	A0	B1	B0	S0	S1	S2
0	0	0	0	0	0	1
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	1	0	0
0	1	0	1	0	0	1
0	1	1	0	0	1	0
0	1	1	1	0	1	0
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	0	1	0
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	0	1

Tabla de verdad del comparador binario de dos bits

2.2. Funciones lógicas de salida

- $S_0 = \overline{A_1} \cdot \overline{A_0} \cdot \overline{B_1} \cdot \overline{B_0} + \overline{A_1} \cdot \overline{A_0} \cdot \overline{B_1} \cdot B_0 + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot \overline{B_0} + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot B_0 + \overline{A_1} \cdot A_0 \cdot \overline{B_1} \cdot \overline{B_0} + \overline{A_1} \cdot A_0 \cdot \overline{B_1} \cdot B_0 + \overline{A_1} \cdot A_0 \cdot B_1 \cdot \overline{B_0} + \overline{A_1} \cdot A_0 \cdot B_1 \cdot B_0$
- $S_1 = \overline{A_1} \cdot \overline{A_0} \cdot \overline{B_1} \cdot B_0 + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot \overline{B_0} + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot B_0 + \overline{A_1} \cdot A_0 \cdot \overline{B_1} \cdot \overline{B_0} + \overline{A_1} \cdot A_0 \cdot \overline{B_1} \cdot B_0 + \overline{A_1} \cdot A_0 \cdot B_1 \cdot \overline{B_0} + \overline{A_1} \cdot A_0 \cdot B_1 \cdot B_0$
- $S_2 = \overline{A_1} \cdot \overline{A_0} \cdot \overline{B_1} \cdot B_0 + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot \overline{B_0} + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot B_0 + \overline{A_1} \cdot A_0 \cdot \overline{B_1} \cdot \overline{B_0} + \overline{A_1} \cdot A_0 \cdot \overline{B_1} \cdot B_0 + \overline{A_1} \cdot A_0 \cdot B_1 \cdot \overline{B_0} + \overline{A_1} \cdot A_0 \cdot B_1 \cdot B_0$

2.3. Minimización por mapas de Karnaugh

$A_1A_0 \backslash B_1B_0$	00	01	11	10
00	0	0	0	0
01	1	0	0	0
11	1	1	0	1
10	1	1	0	0

Mapa de Karnaugh para S_0 con la celda 1100 mostrando los tres colores

- **Grupo rojo:** 1100, 1101, 1000, 1001.
- **Grupo cyan:** 0100 y 1100.
- **Grupo verde:** 1110 y 1100.
- $S_0 = \overline{A_1} \cdot \overline{B_1} + \overline{A_0} \cdot \overline{B_1} \cdot \overline{B_0} + \overline{A_1} \cdot A_0 \cdot \overline{B_0}$

$A_1A_0 \backslash B_1B_0$	00	01	11	10
00	0	1	1	1
01	0	0	1	1
11	0	0	0	0
10	0	0	1	0

Mapa de Karnaugh para S_1 con agrupaciones

- **Grupo rojo:** 0011, 0010, 0111, 0110.
- **Grupo cyan:** 0001 y 0011.
- **Grupo verde:** 1011.
- $S_1 = \overline{A_1} \cdot B_1 + \overline{A_1} \cdot \overline{A_0} \cdot B_0 + B_1 \cdot B_0 \cdot \overline{A_0}$

$A_1A_0 \backslash B_1B_0$	00	01	11	10
00	1	0	0	0
01	0	1	0	0
11	0	0	1	0
10	0	0	0	1

Mapa de Karnaugh para S_2

- $S_2 = \overline{A_1} \cdot \overline{A_0} \cdot \overline{B_1} \cdot \overline{B_0} + \overline{A_1} \cdot \overline{A_0} \cdot \overline{B_1} \cdot B_0 + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot \overline{B_0} + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot B_0$

En este caso, Karnaugh no es útil para simplificar, por lo que debemos hacerlo mediante los teoremas y postulados del álgebra de Boole.

- $S_2 = \overline{A_1} \cdot \overline{A_0} \cdot \overline{B_1} \cdot \overline{B_0} + \overline{A_1} \cdot \overline{A_0} \cdot \overline{B_1} \cdot B_0 + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot \overline{B_0} + \overline{A_1} \cdot \overline{A_0} \cdot B_1 \cdot B_0$
- $(\overline{A_1} \cdot \overline{B_1}) \cdot (\overline{A_0} \cdot \overline{B_0} + A_0 \cdot B_0) + (\overline{A_1} \cdot B_1) \cdot (\overline{A_0} \cdot B_0 + \overline{A_0} \cdot \overline{B_0})$
- $\overline{A_1} \cdot \overline{B_1} \cdot (\overline{A_0} \oplus B_0) + \overline{A_1} \cdot B_1 \cdot (\overline{A_0} \oplus B_0)$
- $(\overline{A_0} \oplus B_0) \cdot (\overline{A_1} \oplus B_1)$

2.4. Circuito lógico en Falstad

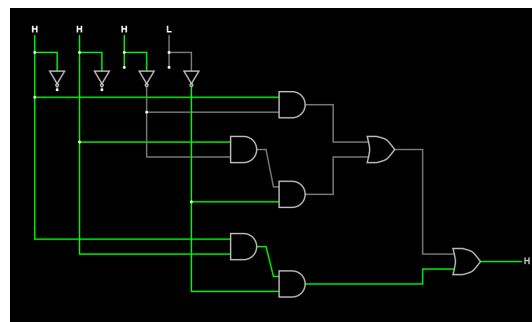


Figura 6: Circuito lógico implementado en Falstad para S_2
Bits usados para ejemplo 1110.

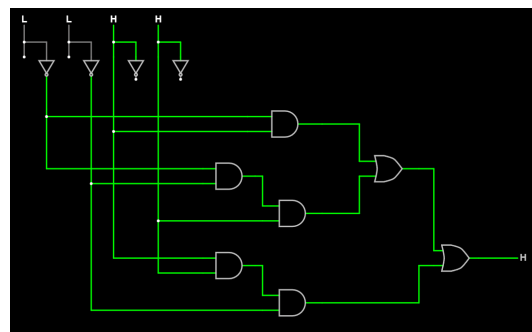


Figura 7: Circuito lógico implementado en Falstad para S_2
Bits usados para ejemplo 0011.

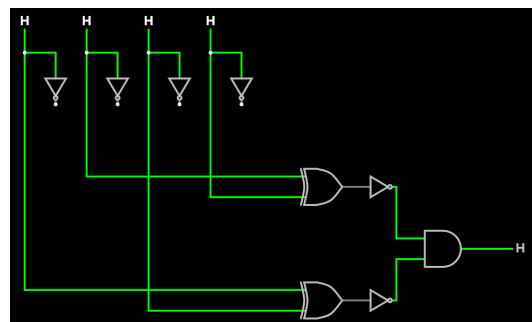
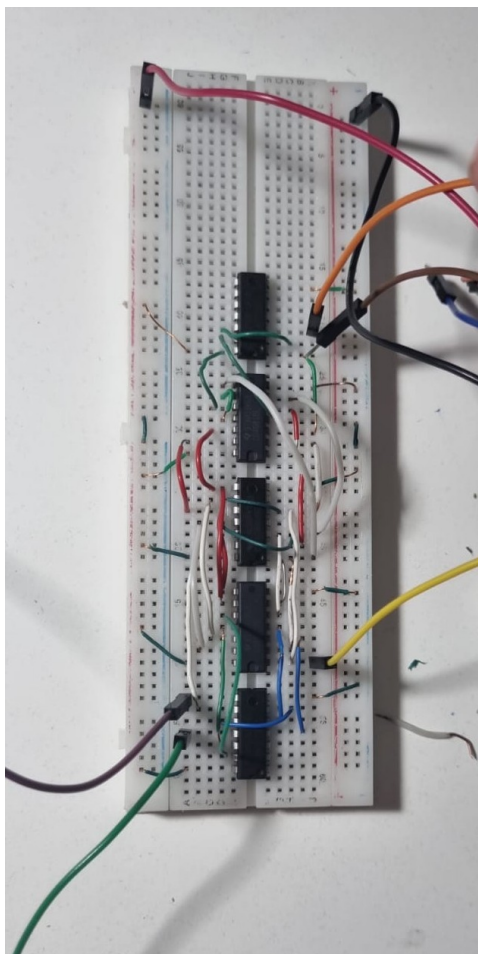


Figura 8: Circuito lógico implementado en Falstad para S_2
Bits usados para ejemplo 1111.

3. Imágenes de Circuitos

3.1. Circuito 2.1

Circuito lógico implementado en protoboard para la actividad 2.1



3.2. Circuito 2.2

Circuito lógico implementado en protoboard para la actividad 2.2

