BBM233: Logic Design Lab
2022 Fall
Lab Experiment #3 Report

Salih Eren Yüzbaşıoğlu,2220356040

November 9, 2023

1 Part 1 (Experiment 1)

A 7-segment display is a form of electronic display device that can represent decimal numerals and some basic characters using a combination of seven individually illuminated segments.

1.1 What is a 7-segment display and how it works?

The 7-segment display is the device containing seven different LEDs. It is usually used to show digits (0-9) or some letters (A, E, R...) All LEDs are connected to a boolean function.

1.2 How many types of 7-degment display are there?

there are two types of LED 7-segment display called: Common Cathode (CC) and Common Anode (CA). In the CC display, all the cathode connections of the LED segments are joined together to logic "0" or ground. The individual segments are illuminated by application of a "HIGH", or logic "1" signal via a current limiting resistor to forward bias the individual Anode terminals (a-g). In the CA display, all the anode connections of the LED segments are joined together to logic "1". The individual segments are illuminated by applying a ground, logic "0" or "LOW" signal via a suitable current limiting resistor to the Cathode of the particular segment (a-g).

1.3 Why do we need a decoder

To be able to turn on the desired segments of a 7-segment display and display specific numbers or characters, we often use a decoder.

1.4 If this assignment were about designing a common anode instead of common cathode, would there be any change in truth table and if yes what kind of change?

In the truth table of a decoder for a 7-segment display, the values of functions a-g, which represent the segments of the display, would be interchanged between 0s and 1s.

1.5 What happens if you apply inputs for which you used don't cares (X), i.e. 8-15? What is shown on the 7-segment display and why?

As the 7-segment display is designed to only show numbers from 0 to 9, any input within that range would not be affected. However, if a signal outside this range, specifically between 10 and 15, were provided, it would result in displaying nonsensical or unintended output.

2 Part 2

Table 1: Truth Table for Common Cathode Decoder

Inputs				**Outputs**						
A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	0	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	0	0	1	1

3 Part 3

$$a(A,B,C,D) = P(0,2,3,5,7,8,9) \tag{1}$$

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

$$a = A + \overline{B} \times \overline{D} + B \times D + \overline{B} \times C \tag{2}$$

$$b(A,B,C,D) = P(0,1,2,3,4,7,8,9) \tag{3}$$

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

$$b = \overline{B} + \overline{C} \times \overline{D} + C \times D \tag{4}$$

$$c(A,B,C,D) = P(0,1,3,4,5,6,7,8,9) \tag{5}$$

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

$$c = \overline{C} + D + B \tag{6}$$

$$d(A,B,C,D) = P(0,2,3,5,6,8) \tag{7}$$

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

$$d = \overline{B} \times \overline{D} + \overline{B} \times C + C \times \overline{D} + B \times \overline{C} \times D \tag{8}$$

$$e(A, B, C, D) = P(0, 2, 6, 8) \tag{9}$$

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

$$e = \overline{B} \times \overline{D} + C \times \overline{D} \tag{10}$$

$$f(A, B, C, D) = P(0, 4, 5, 6, 8, 9) \tag{11}$$

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

$$f = B \times \overline{D} + \overline{C} \times \overline{D} + B \times \overline{C} + A \tag{12}$$

$$g(A,B,C,D) = P(2,3,4,5,6,8,9) \tag{13}$$

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

$$g = \overline{C} \times B + \overline{B} \times C + C \times \overline{D} + A \tag{14}$$

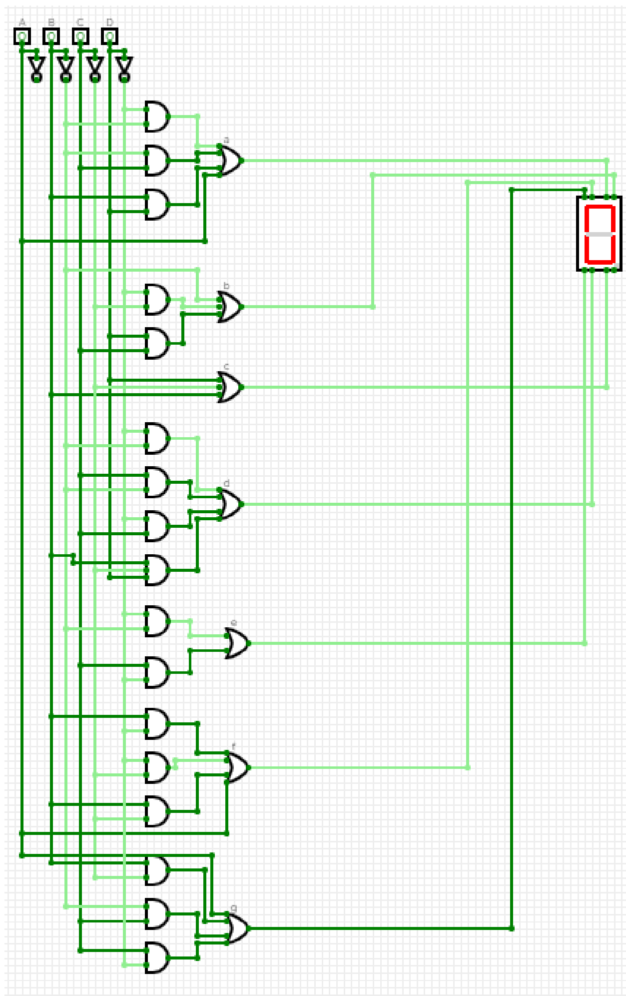

Figure 1: The BCD-to-7 Segment Display Decoder

4 Part 4 (Experiment 2)

A universal gate, also known as a functionally complete gate, is a logic gate that can be used to implement any other logic gate. In other words, with a single type of universal gate, you can construct circuits that perform any logical operation.

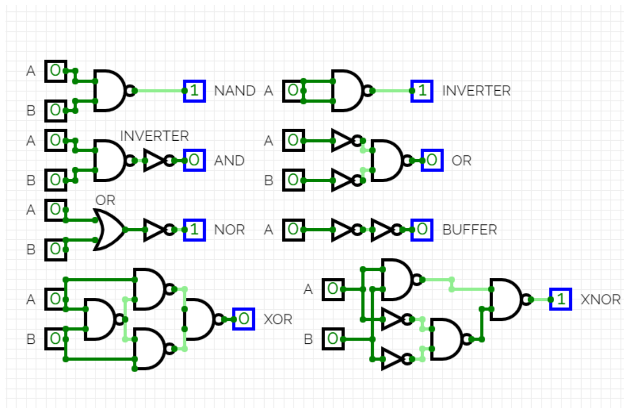

Figure 2: Implementing other gates using NAND

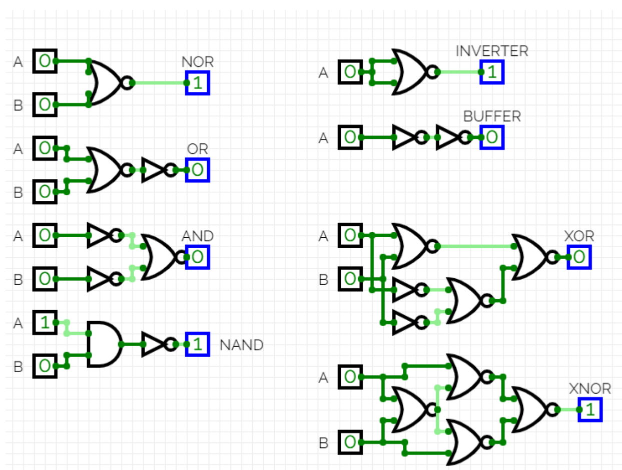

Figure 3: Implementing other gates using NOR

4.1 Use the formula you obtained from the K-Map and express it using only NAND gates.

$$a = \overline{A} \uparrow (A \uparrow B) \uparrow (\overline{B} \uparrow C) \uparrow (\overline{B} \uparrow \overline{D}) \tag{15}$$
$$b = B \uparrow (C \uparrow D) \uparrow (\overline{C} \uparrow \overline{D}) \tag{16}$$
$$c = \overline{B} \uparrow C \uparrow \overline{D} \tag{17}$$
$$d = (\overline{B} \uparrow D) \uparrow (C \uparrow \overline{D}) \uparrow (\overline{C} \uparrow D) \tag{18}$$
$$e = (\overline{B} \uparrow \overline{D}) \uparrow (C \uparrow \overline{D}) \tag{19}$$
$$f = \overline{A} \uparrow (\overline{B} \uparrow D) \uparrow C \tag{20}$$
$$g = \overline{A} \uparrow (B \uparrow \overline{C}) \uparrow (\overline{B} \uparrow C) \uparrow (C \uparrow \overline{D}) \tag{21}$$
$$\tag{22}$$

4.2 Using NOR gates

$$a = (A \downarrow \overline{B} \downarrow \overline{C} \downarrow \overline{D}) \downarrow (A \downarrow B \downarrow C) \tag{23}$$
$$b = (\overline{B} \downarrow \overline{C} \downarrow D) \downarrow (\overline{B} \downarrow C \downarrow \overline{D}) \tag{24}$$
$$c = \overline{(B \downarrow \overline{C} \downarrow D)} \tag{25}$$
$$d = (A \downarrow \overline{B} \downarrow C) \downarrow (A \downarrow \overline{B} \downarrow \overline{D}) \downarrow (A \downarrow C \downarrow \overline{D}) \tag{26}$$
$$e = (\overline{B} \downarrow C) \downarrow D \tag{27}$$
$$f = (A \downarrow \overline{B} \downarrow \overline{C}) \downarrow (A \downarrow \overline{C} \downarrow D) \tag{28}$$
$$g = (A \downarrow \overline{B} \downarrow \overline{C} \downarrow \overline{D}) \downarrow (A \downarrow B \downarrow C) \tag{29}$$
$$\tag{30}$$

References

- www.overleaf.com
- https://ctan.mines-albi.fr/graphics/pgf/contrib/karnaugh-map/karnaugh-map.pdf
- https://circuitverse.org/simulator
- https://www.dcode.fr/boolean-expressions-calculator