Task 4

In this case we need to convert a 4-bit number into its complement to two. A truth table is built first with four outputs corresponding to the four input bits of the number complemented, as shown below.

4-Bit In				2-Comp. Out			
d	c	b	a	f_d	f_c	f_b	f_a
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1
1	0	0	0	1	0	0	0
1	0	0	1	0	1	1	1
1	0	1	0	0	1	1	0
1	0	1	1	0	1	0	1
1	1	0	0	0	1	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	1

Table 1: Outputs with complement to two.

The output functions are expressed based on the minternms. They are simplified using Karnaugh's Maps. Starting with f_d function:

$$f_d = \sum(m_1, m_2, m_3, m_4, m_5, m_6, m_7, m_8)$$

dc \ ba	00	01	11	10
00	0	1	1	1
01	1	1	1	1
11	0	0	0	0
10	1	0	0	0

Solving the map with the indicated groups, the simplified function remains as:

$$\boxed{f_d = (d\cdot\overline{c}\cdot\overline{b}\cdot\overline{a}) + (\overline{d}\cdot c) + (\overline{d}\cdot\overline{c}\cdot a) + (\overline{d}\cdot b\cdot\overline{a})}$$

Now, taking the f_c function, we do the same:

$$f_c = \sum(m_1, m_2, m_3, m_4, m_9, m_{10}, m_{11}, m_{12})$$

dc \ ba	00	01	11	10
00	0	1	1	1
01	1	0	0	0
11	1	0	0	0
10	0	1	1	1

With the indicated groups, we get the simplified funcion:

$$\boxed{f_c = (c \cdot \overline{b} \cdot \overline{a}) + (\overline{c} \cdot a) + (\overline{c} \cdot b \cdot \overline{a})}$$

Next, with the f_b function:

$$f_b = \sum(m_1, m_2, m_5, m_6, m_9, m_{10}, m_{13}, m_{14})$$

dc \ ba	00	01	11	10
00	0	1	0	1
01	0	1	0	1
11	0	1	0	1
10	0	1	0	1

Solving the map, we get:

$$\boxed{f_b = (\overline{b} \cdot a) + (b \cdot \overline{a})}$$

For the last function f_a:

$$f_a = \sum m_1, m_3, m_5, m_7, m_9, m_{11}, m_{13}, m_{15}$$

In the table it can be observed that the output depends directly from input a. We can write the simplified function without making the Karnaugh's map:

$$\boxed{f_a = a}$$

Having already the four output functions, the implementation can be carried out in a circuit with AND, OR and NOT logic gates, as shown in the next page.

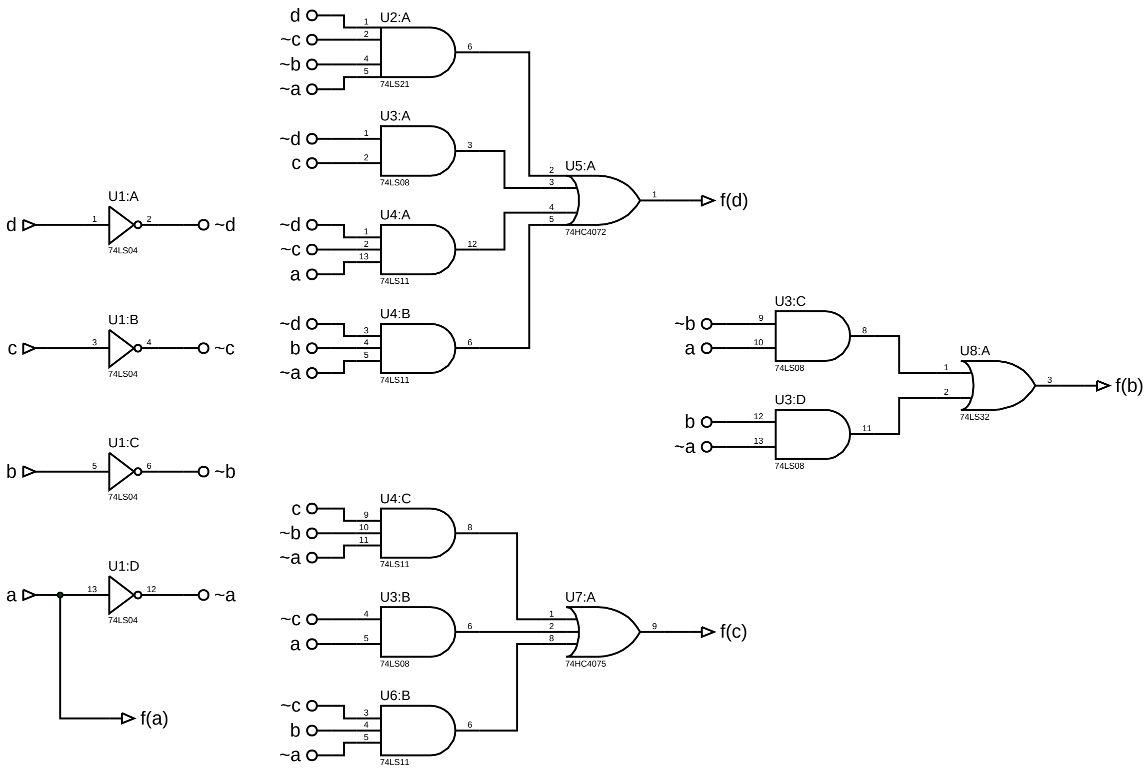

Figure 1: Implementation of 2-complement circuit for a 4-bit input number - Designed in Proteus 7.8