

Laboratório 3:



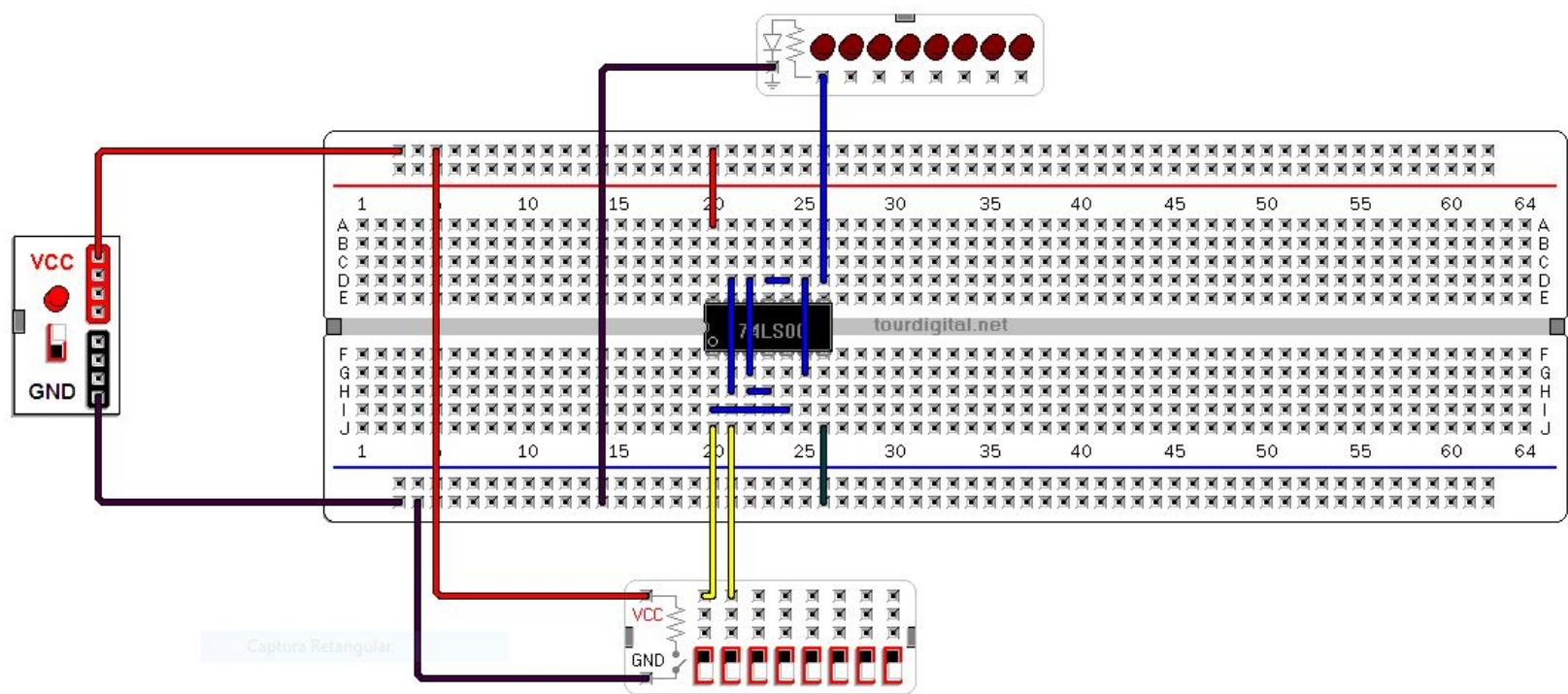
Projeto e montagem de redes iterativas

Professor:
Osamu Saotome

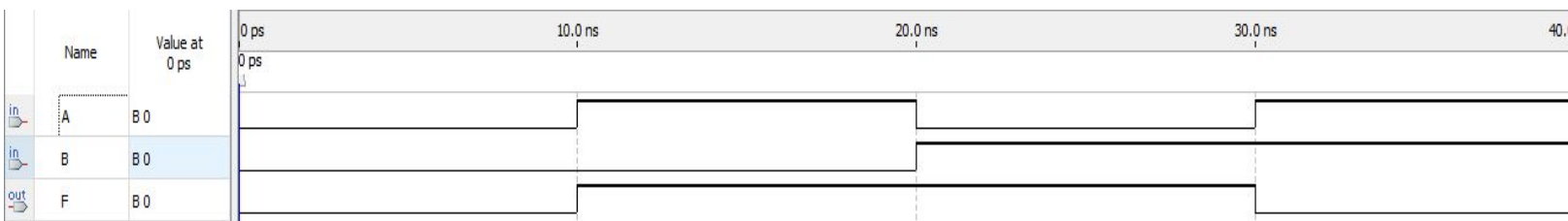
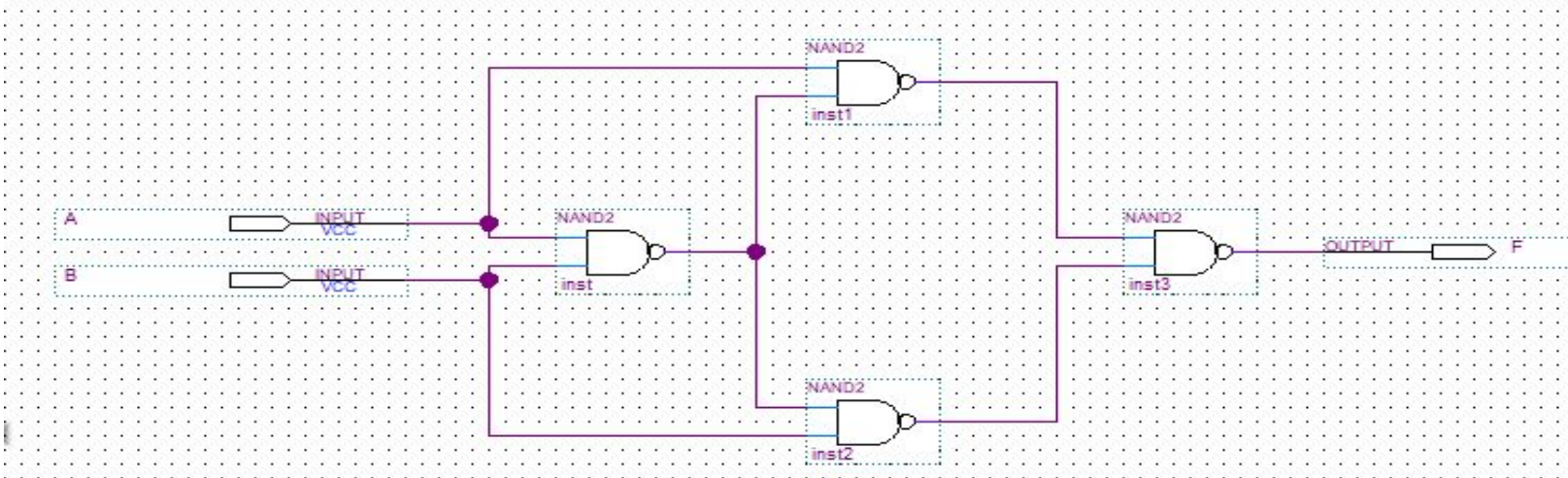
Alunos:
Rodrigo Alves de Almeida
(rodrigoalmeida@gmail.com)
Eduardo Menezes Moraes
(eduardomenezesm@msn.com)

Comp 22

5.1) Montagem do circuito:



Para obtenção da tabela verdade, foi montado o diagrama esquemático e de temporização:



A	B	F
0	0	0
1	0	1
0	1	1
1	1	0

$$F = A \oplus B$$

5.2) a) Tabela Verdade:

$Y_{1,i}$	$Y_{3,i}$	A_i	B_i	$Y_{1,i-1}$	$Y_{3,i-1}$
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	0	0
1	0	x	x	1	0
0	1	x	x	0	1

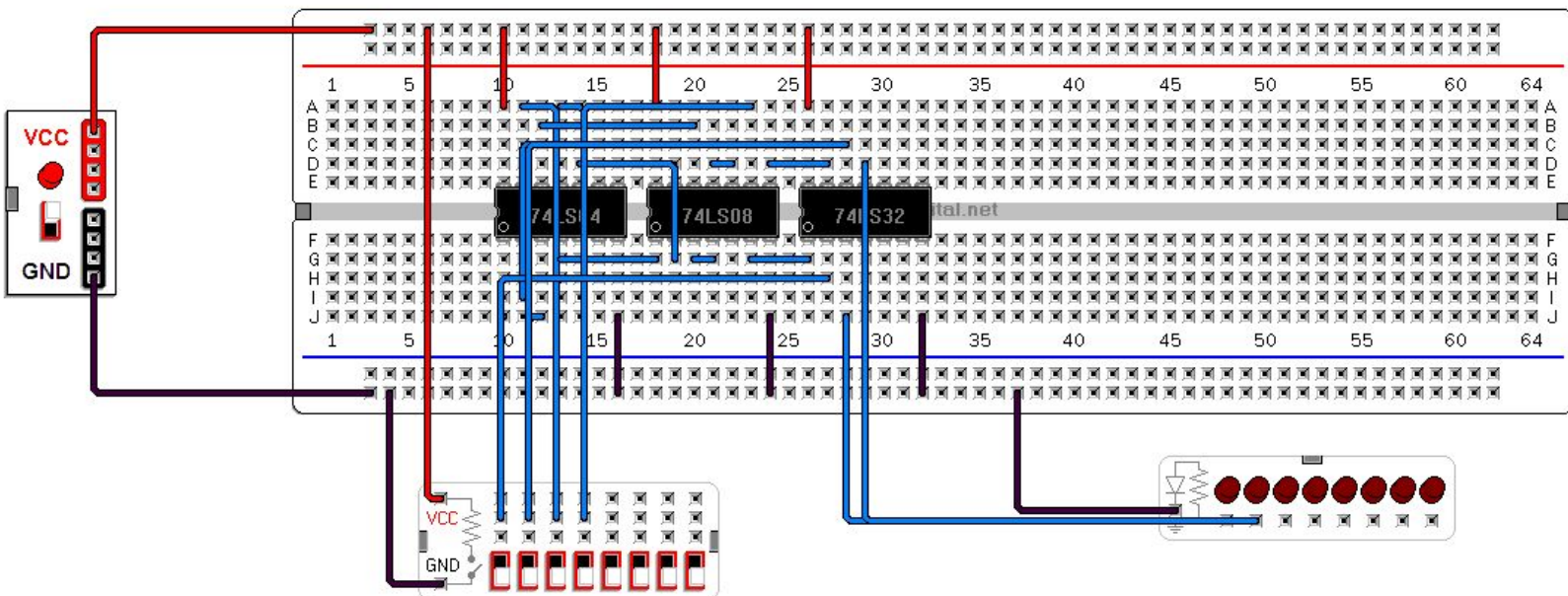
b) Funções Lógicas:

A partir do mapa de Karnaugh são extraídas as seguintes funções lógicas:

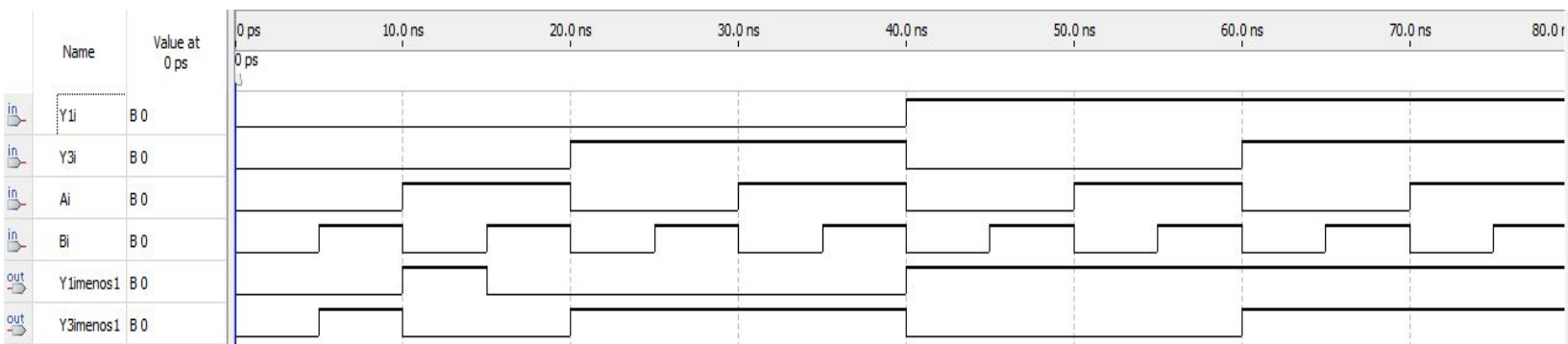
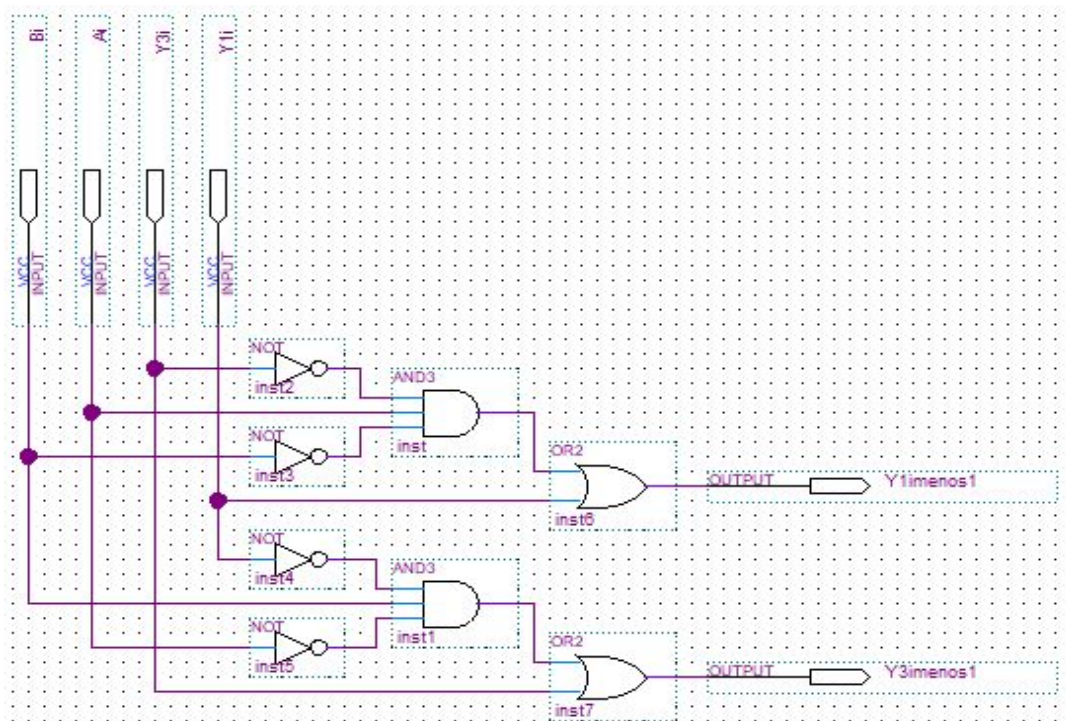
$$Y_{1,i-1} = Y_{1,i} + \overline{Y_{3,i}} A_i \overline{B_i}$$

$$Y_{3,i-1} = Y_{3,i} + \overline{Y_{1,i}} A_i B_i$$

c) Montagem:



d) Montando o circuito no Quartus e simulando:



É importante perceber que a tabela gerada pelo diagrama de temporização é exatamente igual à tabela do item a.

e) Estas variáveis são obtidas a partir do output da célula 1:

$$\begin{aligned}
 Y_1 &= Y_{1,0} \overline{Y_{3,0}} \\
 Y_2 &= \overline{Y_{1,0}} Y_{3,0} \\
 Y_3 &= \overline{Y_{1,0}} \overline{Y_{3,0}}
 \end{aligned}$$

5.3) a) Tabela Verdade:

A_i	B_i	E_{i-1}	S_i	E_i
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

b) Funções Lógicas:

A partir do mapa de Karnaugh são extraídas as seguintes funções lógicas:

$$S_i = A_i \oplus B_i \oplus E_{i-1}$$

$$E_i = B_i E_{i-1} + \overline{A_i} (B_i \oplus E_{i-1})$$

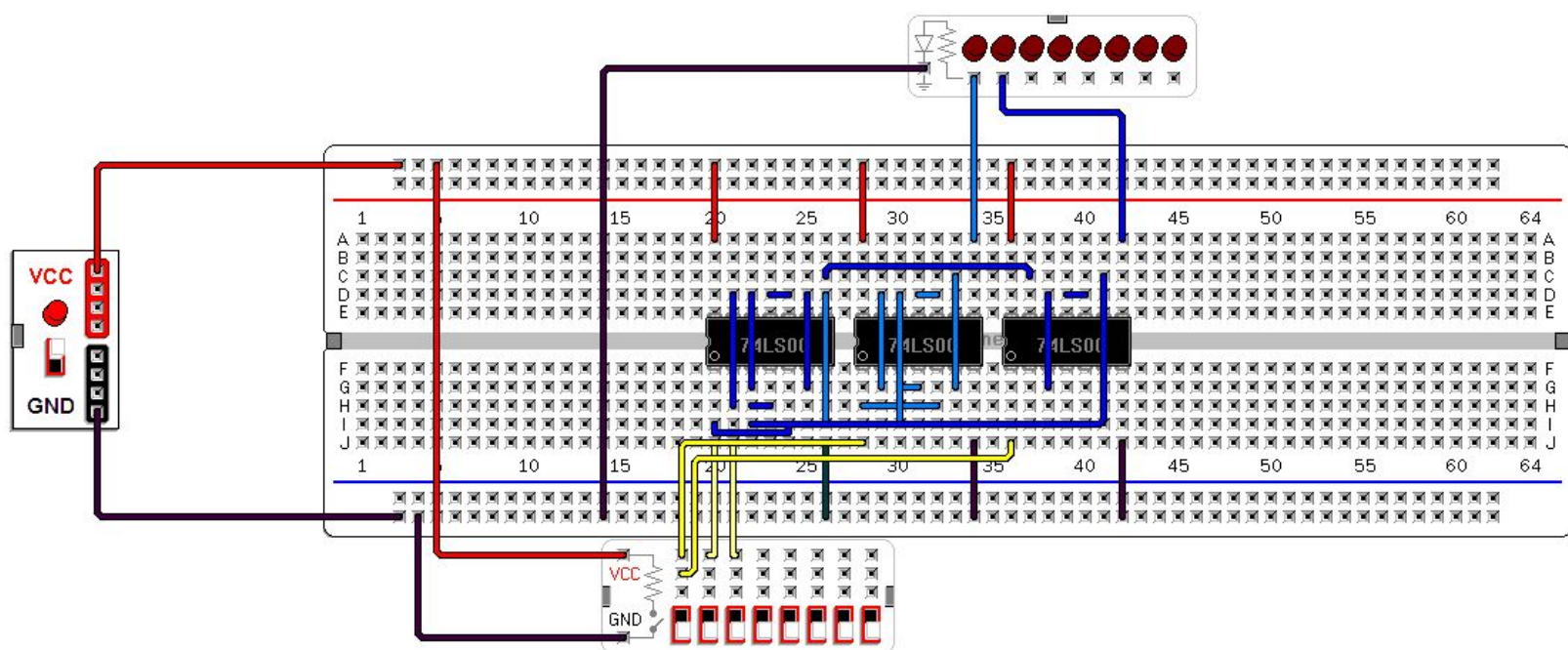
c) Para reescrever as funções, partiremos da ideia utilizada no exercício 5.1:

$$B_i \oplus E_{i-1} = \overline{\overline{B_i E_{i-1}} \overline{B_i E_{i-1}} E_{i-1}} = X$$

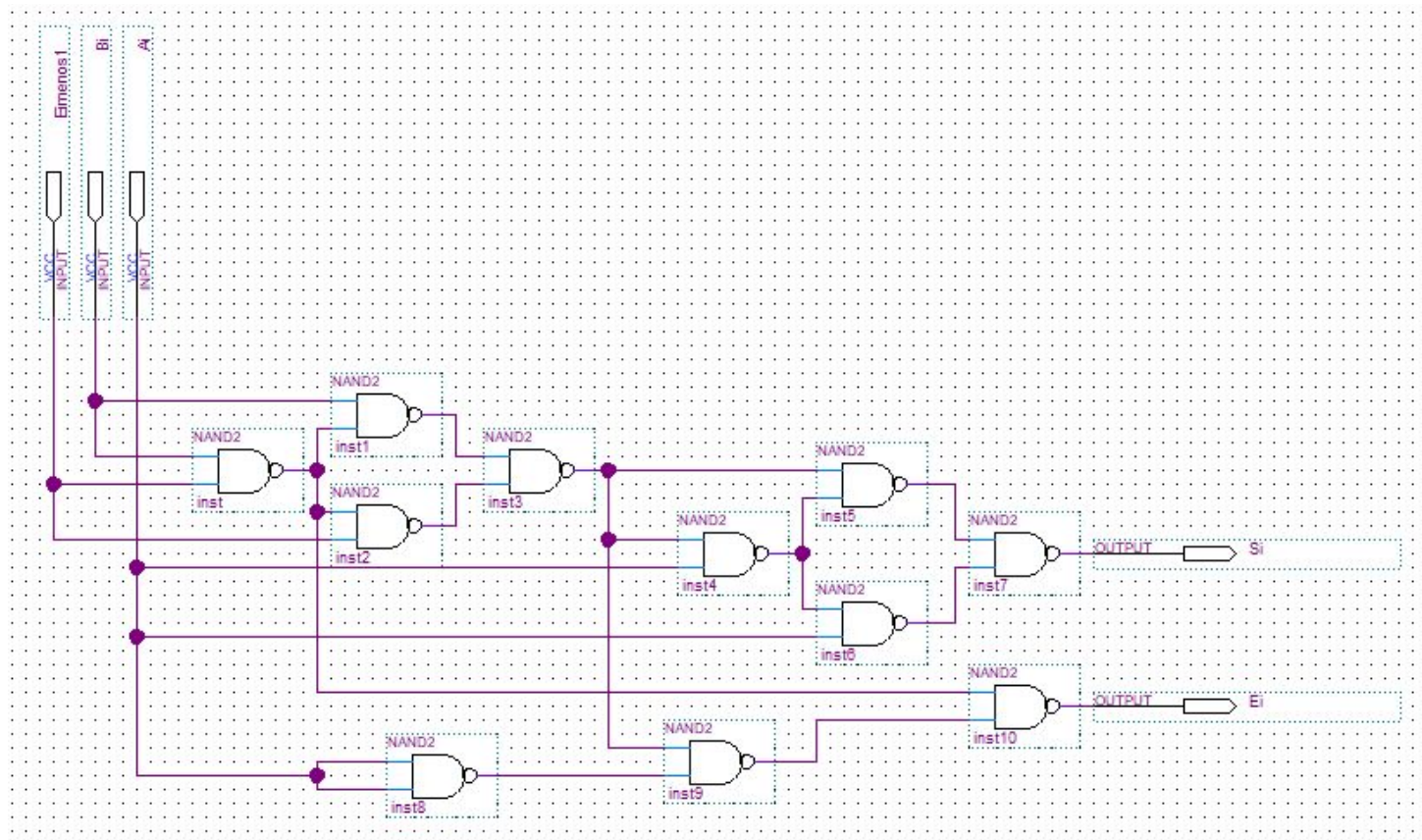
$$S_i = \overline{\overline{A_i X} \overline{A_i X} X}$$

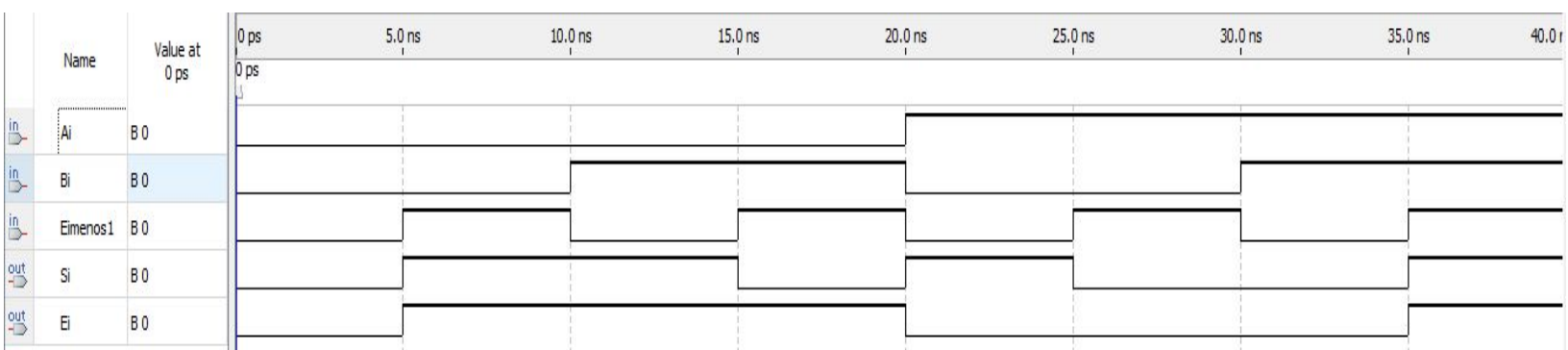
$$E_i = \overline{\overline{B_i E_{i-1}} \overline{A_i X}} = \overline{\overline{B_i E_{i-1}} \overline{A_i} \overline{A_i} X}$$

d) Montagem do circuito:



e) Montando o diagrama esquemático e o de temporização no Quartus:





Assim, notamos que o diagrama de temporização é exatamente a mesma representação da tabela verdade do item a, ou seja, a implementação com as portas NAND é correta.

5.4) a) Tabela Verdade:

A_i	B_i	C_{i-1}	S_i	C_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

b) Funções Lógicas:

A partir do mapa de Karnaugh são extraídas as seguintes funções lógicas:

$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

$$C_i = A_i C_{i-1} + B_i (A_i \oplus C_{i-1})$$

c) Para reescrever as funções, partiremos da ideia utilizada no exercício 5.1:

$$A_i \oplus C_{i-1} = \overline{\overline{A_i} \overline{C_{i-1}}} \overline{\overline{A_i} \overline{C_{i-1}}} \overline{A_i} \overline{C_{i-1}} = X$$

$$S_i = \overline{\overline{B_i} \overline{X}} \overline{\overline{B_i} \overline{X}} \overline{B_i} \overline{X} X$$

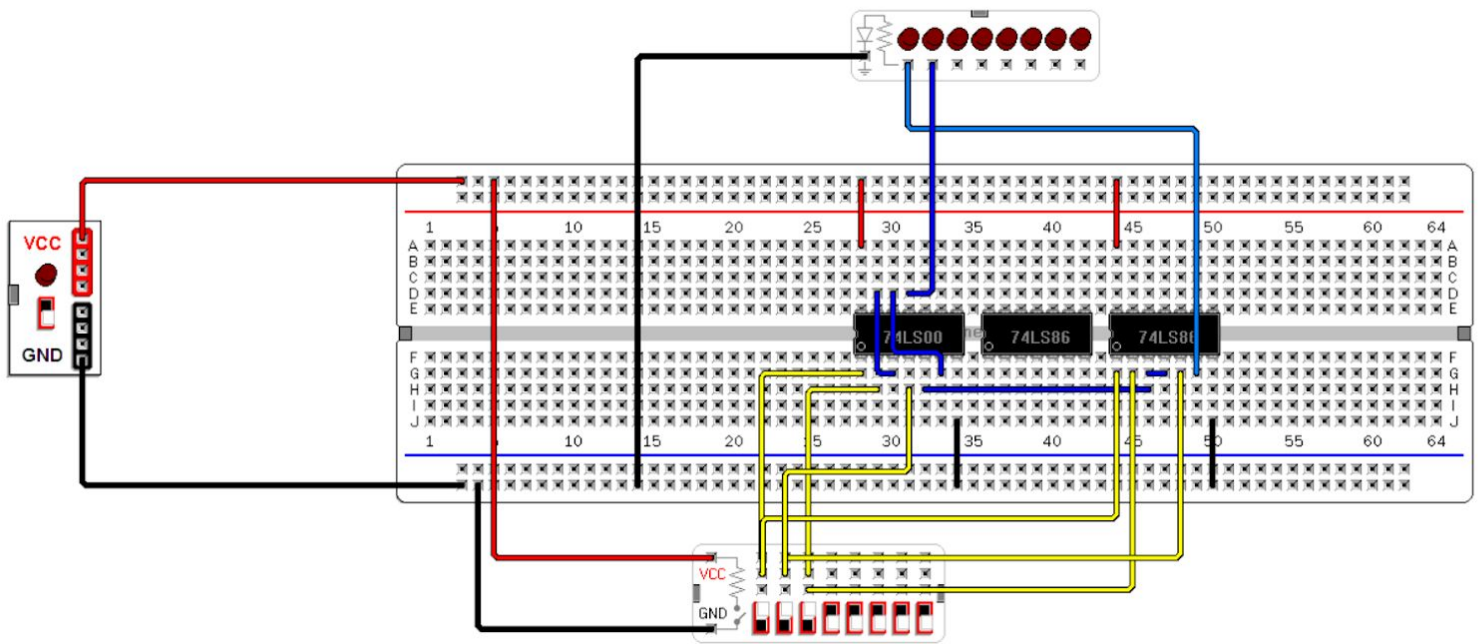
$$C_i = \overline{\overline{A_i} \overline{C_{i-1}}} \overline{\overline{B_i} \overline{X}} = \overline{\overline{A_i} \overline{C_{i-1}}} \overline{B_i} \overline{X}$$

d) Utilizando as portas solicitadas, podemos escrever as expressões do item c) novamente:

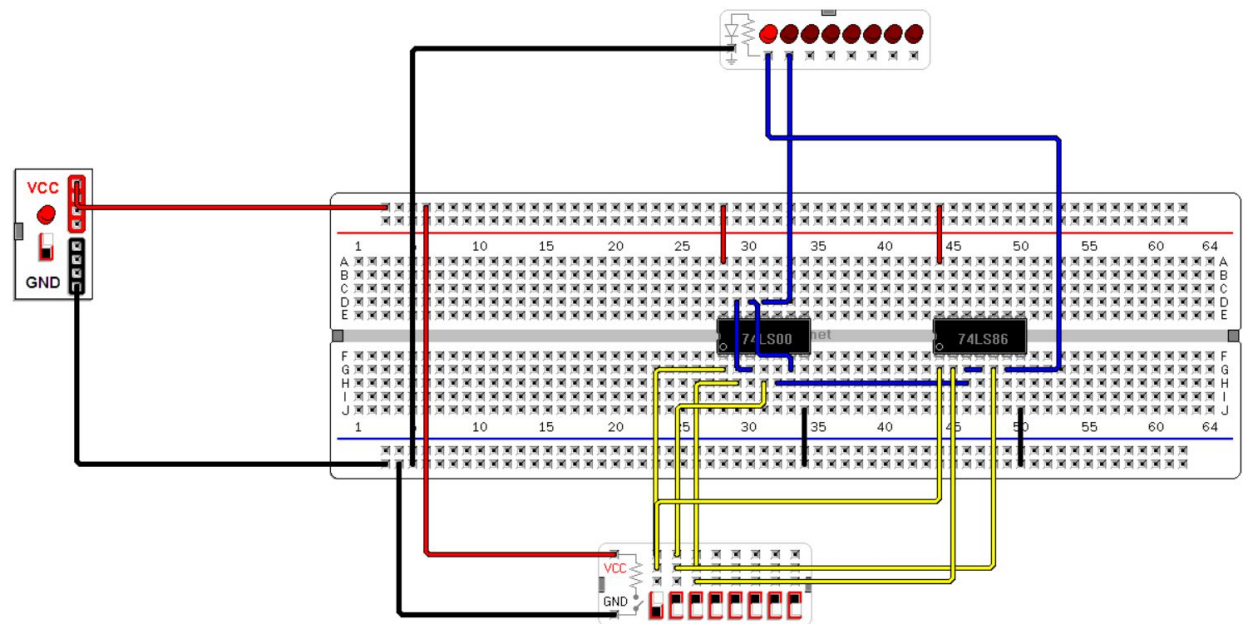
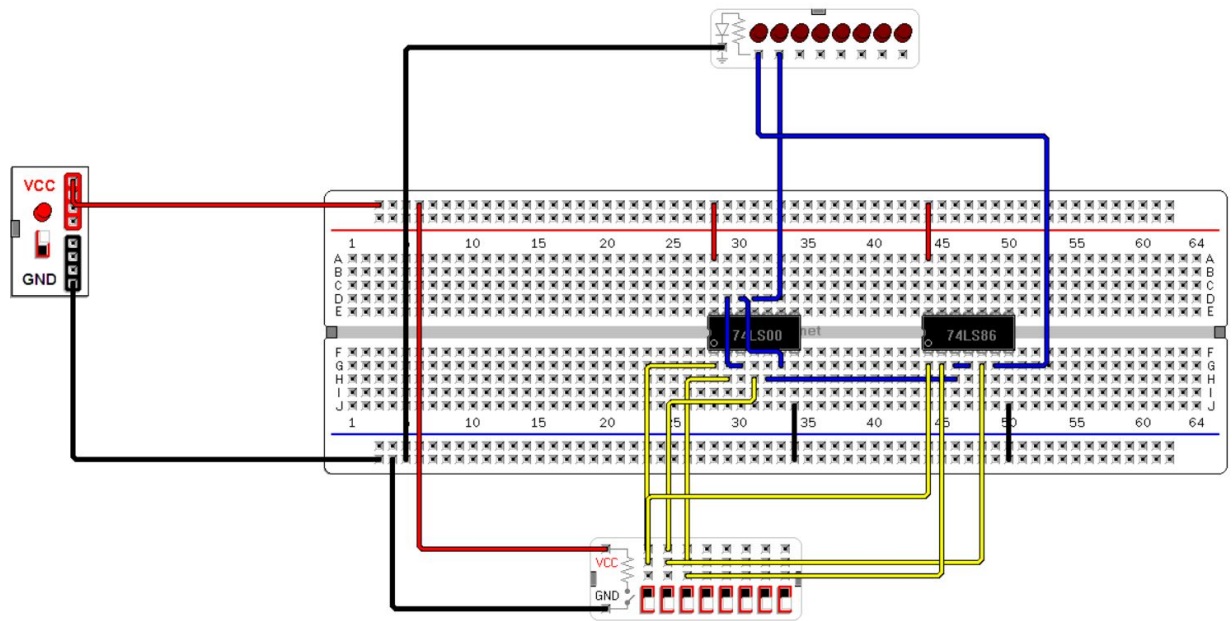
$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

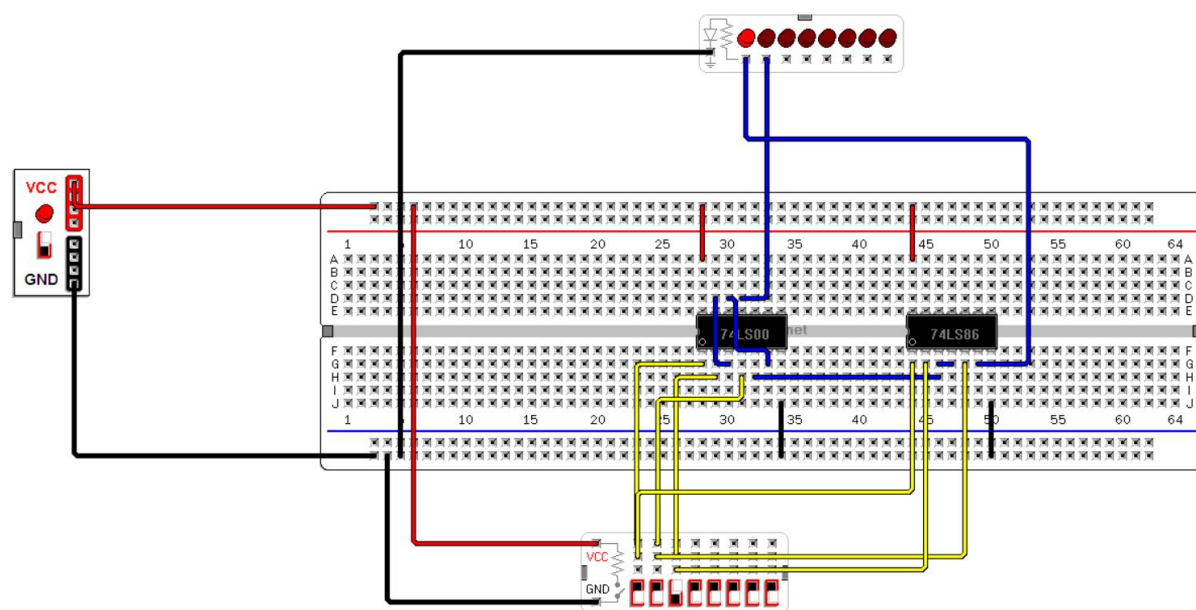
$$C_i = \overline{\overline{A_i} \overline{C_{i-1}}} \overline{B_i} \overline{(A_i \oplus C_{i-1})}$$

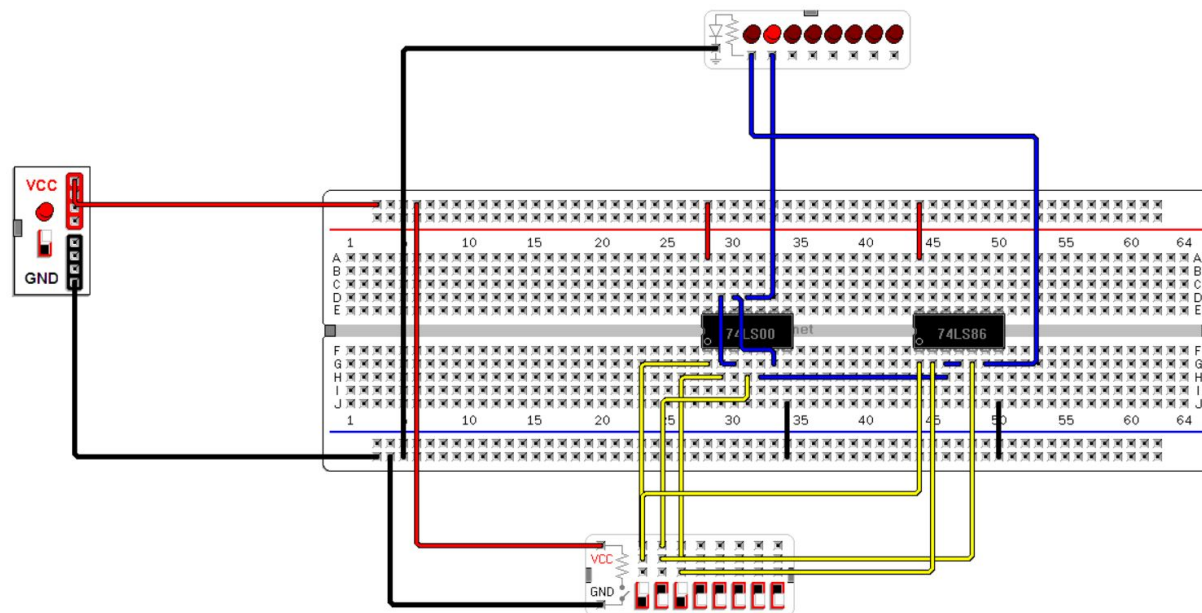
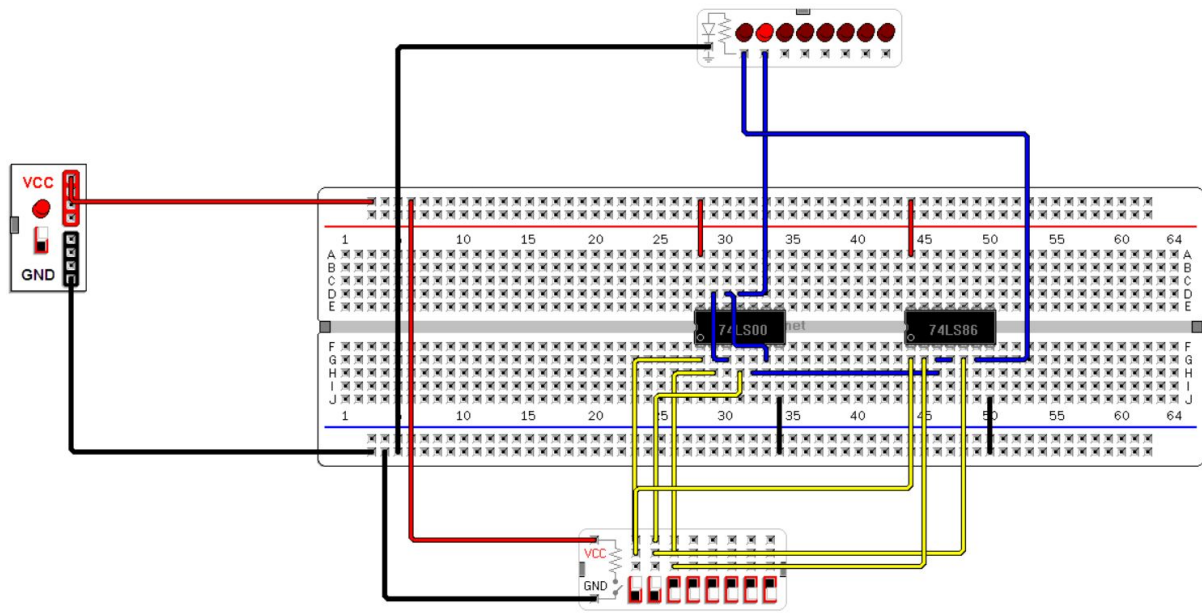
Implementação na protoboard:

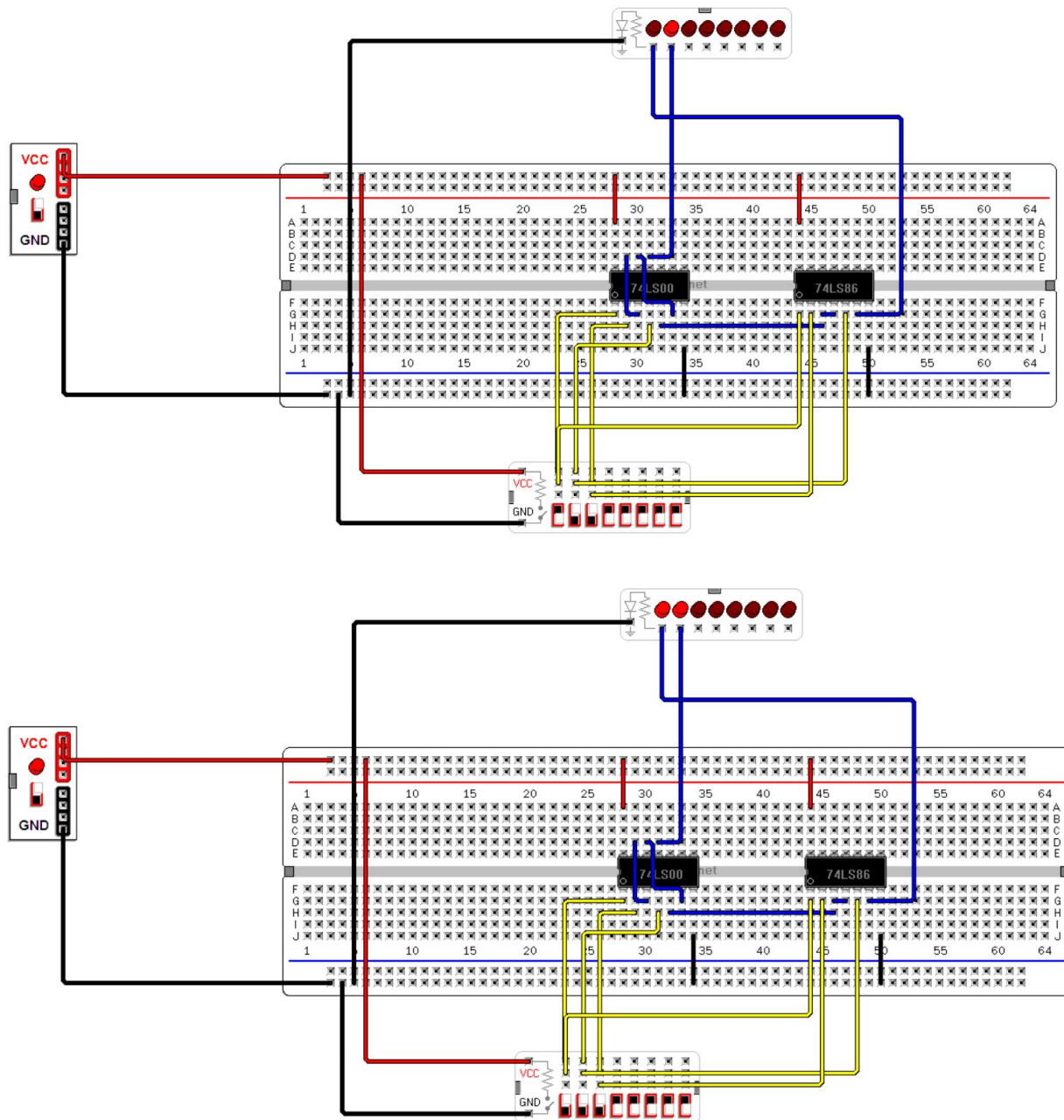


e) Com o intuito de verificar o funcionamento do circuito na protoboard, será colocado uma série de imagens, onde cada uma apresentará uma sequência de chaves diferentes. Vale ressaltar que as chaves são, respectivamente, A, B e C, da esquerda para a direita. Enquanto os LED's são S_i e C_{out} , da esquerda para a direita.









Note que, conforme o esperado, enquanto somente uma ou três chaves são acionadas, o primeiro LED, que representa o S_1 , é aceso, como já era previsto na tabela verdade. Além disso, o C_{out} ou Carry, o qual é representado pelo segundo LED, da esquerda para direita, ficou aceso apenas quando dois ou mais interruptores foram acionados, conforme também era esperado pela tabela verdade.

6.1) a) Primeiramente, montou-se a tabela verdade, de acordo com as condições do enunciado. Para tanto, o raciocínio basicamente foi criar uma função para *complemento* de B, e as detectoras.

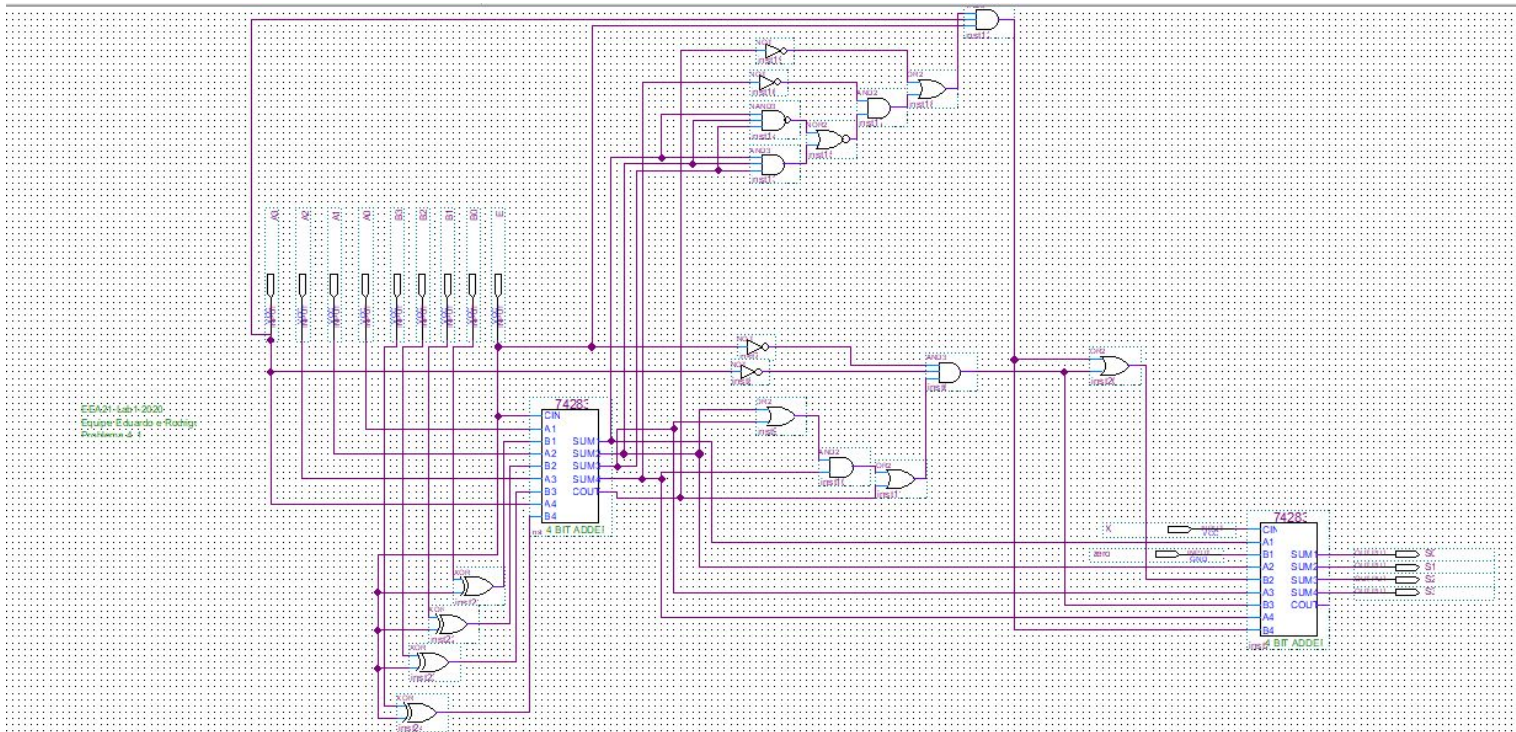
Para a função complemento de 2, a lógica foi se utilizar de portas XOR, conforme visto nas aulas de teoria da disciplina. Já quanto às detectoras, foi necessário criar uma tabela verdade à priori com o intuito de desenvolver mapas de Karnaugh. Contudo, devido a certa simplicidade das expressões booleanas, não foi necessário montar o mapa, bastou observar os padrões, os quais foram confirmados na simulação.

S_3	S_2	S_1	S_0	A_3	E	C	Y_3	Y_2	Y_1	Y_0
1	1	1	1	0	0	0	0	1	1	0
1	1	1	0	0	0	0	0	1	1	0
1	1	0	1	0	0	0	0	1	1	0
1	1	0	0	0	0	0	0	1	1	0
1	0	1	0	0	0	0	0	1	1	0
1	0	1	0	0	0	0	0	1	1	0
0	0	0	1	1	1	1	1	0	1	0
0	0	1	0	1	1	1	1	0	1	0
0	0	1	1	1	1	1	1	0	1	0
0	1	0	0	1	1	1	1	0	1	0
0	1	0	1	1	1	1	1	0	1	0
0	1	1	0	1	1	1	1	0	1	0
X	X	X	X	1	1	0	1	0	1	0
X	X	X	X	0	0	1	0	1	1	0

No caso da saída Y_2 , quando somador atua como adição, por exemplo, é fácil ver que ela só é 1 quando os números são do tipo: 1010, 1011, 1100, 1101, 1111, ou seja, os números maiores que 10, aqueles que irão dar problema. Percebeu-se que isso só ocorre quando o $A_3=1$ e (A_2 ou A_1) = 1. Utilizando várias vezes essa linha de raciocínio, obteve-se sem a necessidade de Karnaugh, as seguintes expressões booleanas:

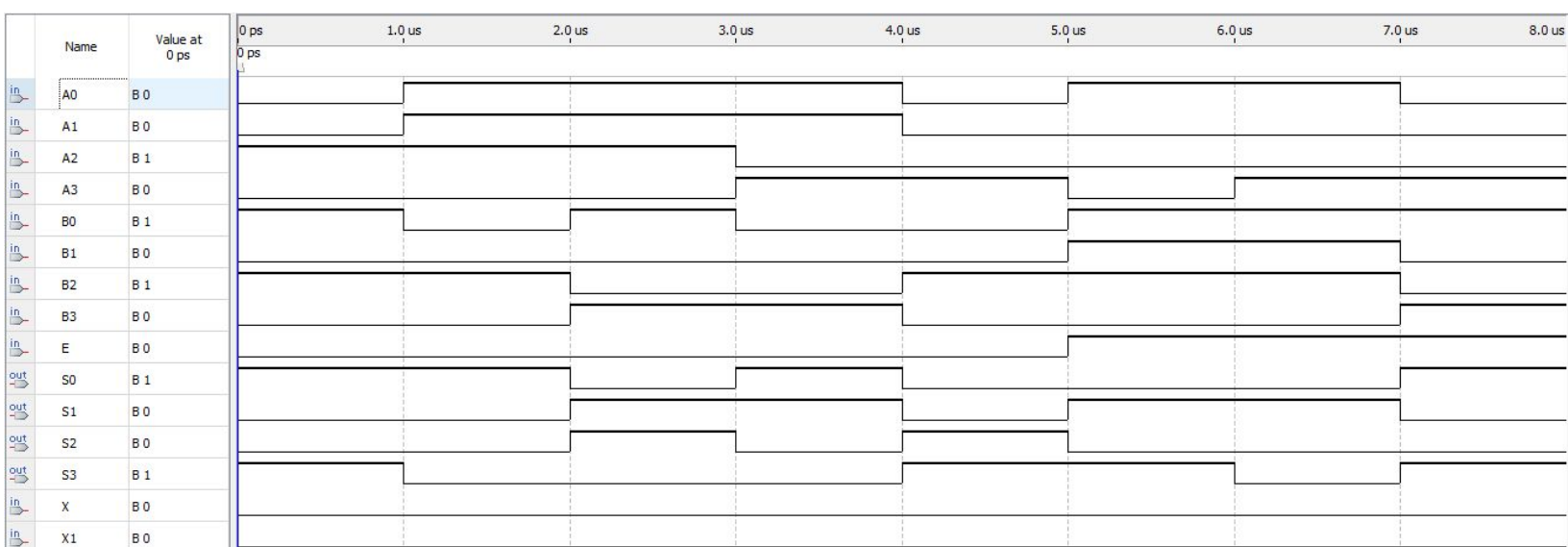
$$\begin{aligned}
 Y_3 &= E A_3 (C' + S_3' (S_2 S_1 S_0 + S_2' S_1' S_0')) \\
 Y_2 &= E' A_3' (C + S_3 (S_2 + S_1)) \\
 Y_1 &= Y_2 + Y_3 \\
 Y_0 &= 0
 \end{aligned}$$

A partir delas, obteve-se o diagrama esquemático:



b) Conforme pedido, foram realizadas as somas, de tal forma que numa mesma onda foram realizadas as 8 expressões. Para isso, reduziu-se o *grip size* e o *end time*, a fim de que todas expressões coubessem numa mesma figura.

Desse modo, temos para cada micro segundo uma expressão formada por $A_3A_2A_1A_0$ e $B_3B_2B_1B_0$, a qual resulta uma saída S, na mesma ordem pedida no arquivo de instruções para esse relatório.



Observe que da esquerda para direita temos para a saída, conforme o esperado, os seguintes valores:
 $(9)_d$ $(1)_d$ $(6)_d$ $(3)_d$ $(4)_d$ $(-6)_d$ $(-4)_d$ $(-7)_d$

6.2) a) A partir da equação do enunciado, chega-se em:

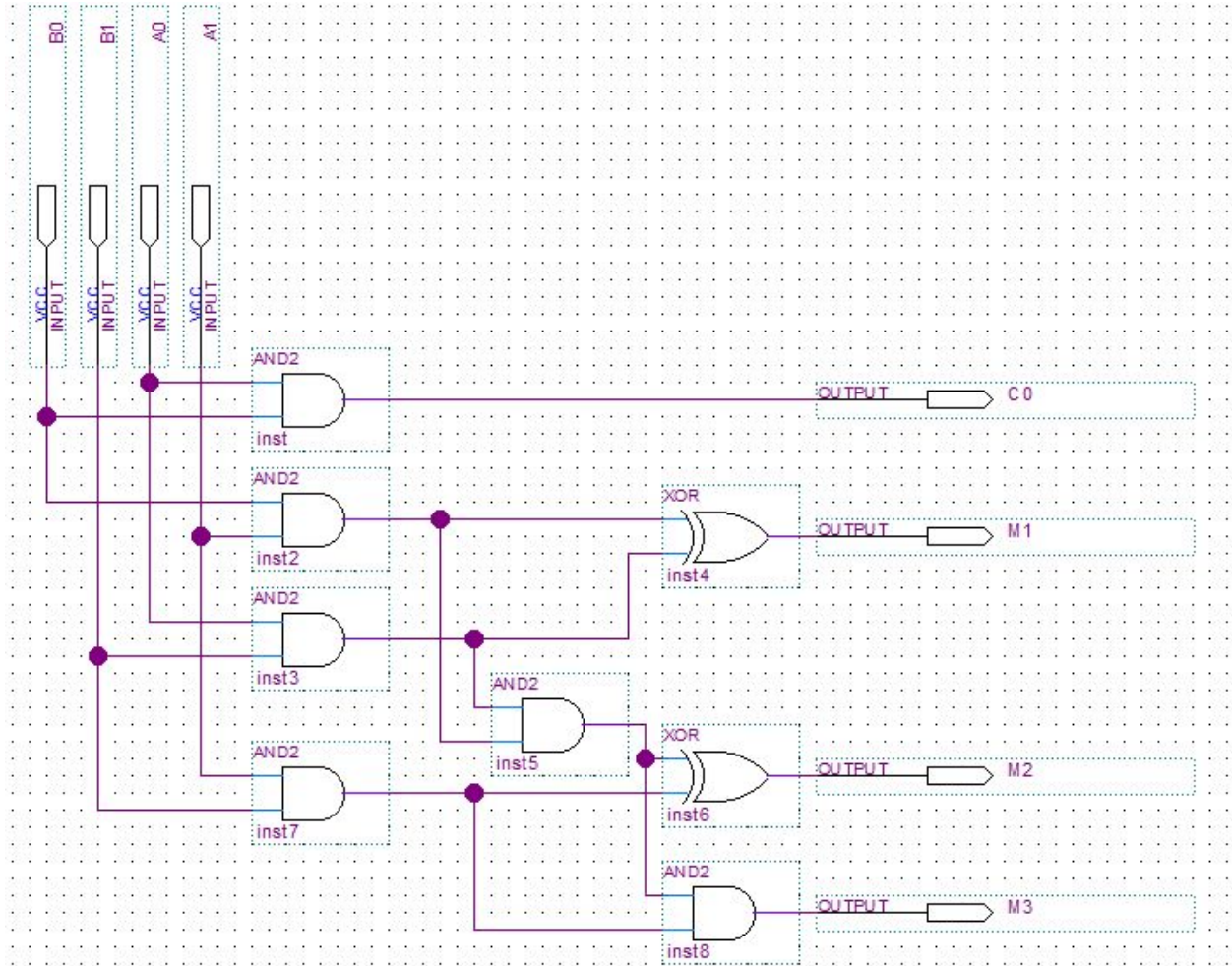
$$C_0 = B_0 A_0$$

$$M_1 = B_0 A_1 \text{ mais } B_1 A_0 ; M_1 = B_0 A_1 \oplus B_1 A_0 , C_{out} = B_0 A_1 B_1 A_0$$

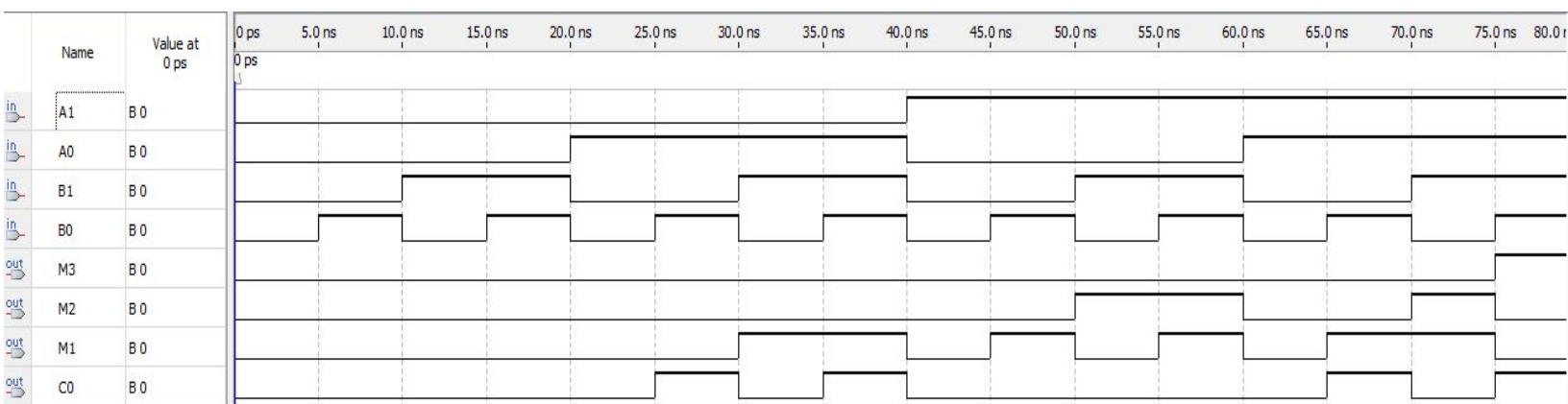
$$M_2 = C_{out} \text{ mais } B_1 A_1 ; M_2 = C_{out} \oplus B_1 A_1$$

$$M_3 = C_{out} B_1 A_1$$

Assim, basta montar o diagrama esquemático:



b) Obtendo o diagrama de temporização por simulação:



A partir dele, verificamos as seguintes operações:

$$(10)_b * (11)_b = (0110)_b ; (2)_d * (3)_d = (6)_d$$

$$(10)_b * (00)_b = (0000)_b ; (2)_d * (0)_d = (0)_d$$

$$(01)_b * (01)_b = (0001)_b ; (1)_d * (1)_d = (1)_d$$

$$(11)_b * (11)_b = (1001)_b ; (3)_d * (3)_d = (9)_d$$