

**ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ**



ATHENS UNIVERSITY  
OF ECONOMICS  
AND BUSINESS

**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΑΚΑΔΗΜΑΪΚΟ ΕΤΟΣ 2020-2021**

**1<sup>η</sup> ΕΡΓΑΣΙΑ  
ΣΧΕΔΙΑΣΗ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ**

ΜΑΡΙΑ ΚΟΝΤΑΡΑΤΟΥ  
ΓΕΩΡΓΙΟΣ ΚΟΥΜΟΥΝΔΟΥΡΟΣ  
ΚΩΝΣΤΑΝΤΙΝΟΣ ΚΩΝΣΤΑΝΤΙΝΑΚΟΣ  
ΔΡ.ΓΕΩΡΓΙΟΣ ΠΟΛΥΖΟΣ

**ΑΘΗΝΑ  
ΑΠΡΙΛΙΟΣ 2021**

## ΣΤΟΙΧΕΙΑ ΕΠΙΚΟΙΝΩΝΙΑΣ

Όνοματεπώνυμο	Αριθμός Μητρώου	Ακαδημαϊκό e-mail
Μαρία Κονταράτου	3200078	p3200078@aueb.gr
Γεώργιος Κουμουνδούρος	3200083	p3200083@aueb.gr
Παναγιώτης Κωνσταντίνος	3200089	p3200089@aueb.gr

Για την εκπόνηση της εργασίας χρησιμοποιήσαμε την εφαρμογή Microsoft Teams για κοινή χρήση αρχείων.

## ΠΡΟΒΛΗΜΑ 1<sup>ο</sup>

-Μορφή SOP και POS ελαχίστου κόστους για την F με πίνακα Karnaugh

SOP

$x_3, x_4$ \ $x_1, x_2$		00	01	11	10
		00	01	11	10
00		1	0	0	0
01		0	d	d	0
11		0	1	1	0
10		1	0	0	0

$x_1'x_2'x_4'$  (blue box around 1 at 00,00)  
 $x_5=1$  (orange box around 1s at 01,11)  
 $x_2x_4x_5$  (orange box around 1s at 11,11)

$x_3, x_4$ \ $x_1, x_2$		00	01	11	10
		00	01	11	10
00		1	1	1	1
01		1	d	1	1
11		0	1	0	0
10		1	d	1	1

$x_1'x_2x_5'$  (red box around 1s at 00,01)  
 $x_5=0$  (yellow box around 1s at 00,10)  
 $x_4'x_5'$  (yellow box around 1s at 00,10)  
 $x_3'x_5'$  (green box around 1s at 00,01)

$$\text{SOP} = x_1'x_2x_5' + x_4'x_5' + x_3'x_5' + x_1'x_2'x_4' + x_2x_4x_5$$

POS

		x1,x2			
x3,x4		00	01	11	10
00		1	0	0	0
01		0	d	d	0
11		0	1	1	0
10		1	0	0	0

$x_2x_4'x_5'$  (points to cell 01,00)  
 $x_5=1$  (points to column 01)  
 $x_2'x_4x_5'$  (points to cell 11,10)  
 $x_1'x_2x_5'$  (points to cell 10,00)  
 $x_2x_3'x_4'$  (points to cell 11,00)

POS

		x1,x2			
x3,x4		00	01	11	10
00		1	1	1	1
01		1	d	1	1
11		0	1	0	0
10		1	d	1	1

$x_1'x_3'x_4'x_5$  (points to cell 11,11)  
 $x_5=0$  (points to column 11)

$$\text{POS} = (x_1' + x_3' + x_4' + x_5)(x_2 + x_4' + x_5')(x_2' + x_4 + x_5')(x_2 + x_3' + x_4')(x_1' + x_2 + x_5')$$

### Συνολικό κόστος SOP

2 πύλες AND 2 εισόδων

3 πύλες AND 3 εισόδων

1 πύλη OR 5 εισόδων

13 είσοδοι για πύλες AND

5 είσοδοι για πύλες OR

SOP = 24

### Συνολικό κόστος POS

4 πύλες OR 3 εισόδων

1 πύλη OR 4 εισόδων

1 πύλη AND 5 εισόδων

16 είσοδοι για πύλες OR

5 είσοδοι για πύλες AND

POS=27

Μικρότερο κόστος έχει το SOP

- Πρόγραμμα στη γλώσσα VHDL, το οποίο να υλοποιεί το παραπάνω κύκλωμα

```
basic_components.vhd
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4
5
6
7
8
9
10
11 package basic_components is
12
13
14
15     component product_of_2
16     port (in1, in2: in std_logic; out1: out std_logic);
17     end component;
18
19
20
21
22
23     component product_of_3
24     port (in1, in2, in3: in std_logic; out1: out std_logic);
25     end component;
26
27
28
29
30
31
32
33
34
35
36
37     component SOP
38     port (in1, in2, in3, in4, in5: in std_logic; out1: out std_logic);
39     end component;
40
41
42
43
44
45
46
47
48
49
50
```

```
basic_components.vhd (Read-Only)
52
53
54 end package basic_components;
55
56
57
58
59
60
61
62
63
64
65
66
67
68 -- product_of_2
69
70 library ieee;
71 use ieee.std_logic_1164.all;
72
73
74
75 entity product_of_2 is
76     port (in1, in2: in std_logic; out1: out std_logic);
77 end product_of_2;
78
79
80
81 architecture model_conc of product_of_2 is
82 begin
83     out1 <= in1 and in2;
84 end model_conc;
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99 -- product_of_3
100 library ieee;
101 use ieee.std_logic_1164.all;
```

```
basic_components.vhd (Read-Only)
103
104
105
106 entity product_of_3 is
107     port (in1, in2, in3: in std_logic; out1: out std_logic);
108 end product_of_3;
109
110
111 architecture model_conc2 of product_of_3 is
112 begin
113     out1 <= in1 and in2 and in3;
114 end model_conc2;
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137 -- SOP
138 library ieee;
139 use ieee.std_logic_1164.all;
140
141
142
143
144 entity SOP is
145     port (in1, in2, in3, in4, in5: in std_logic; out1: out std_logic);
146 end SOP;
147
148
149
150 architecture model_conc4 of SOP is
151 begin
152     out1 <= (in1 or in2) or (in3 or in4) or in5;
```

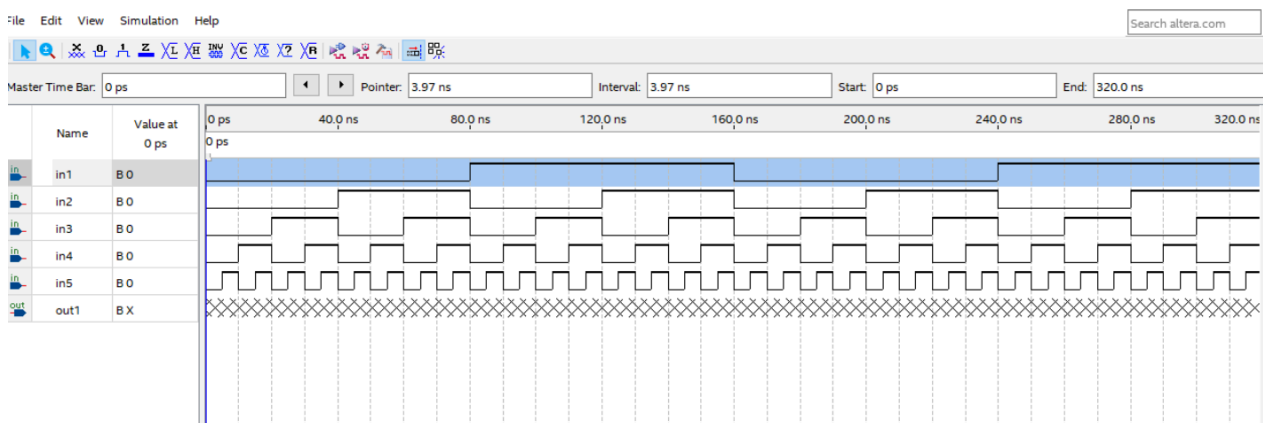
```
basic_components.vhd (Read-Only)
128
129
130
131
132
133
134
135
136
137 -- SOP
138 library ieee;
139 use ieee.std_logic_1164.all;
140
141
142
143
144 entity SOP is
145     port (in1, in2, in3, in4, in5: in std_logic; out1: out std_logic);
146 end SOP;
147
148
149
150 architecture model_conc4 of SOP is
151 begin
152     out1 <= (in1 or in2) or (in3 or in4) or in5;
153 end model_conc4;
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
```

```

1  library IEEE;
2  use IEEE.std_logic_1164.all;
3
4  library work;
5  use work.basic_components.all;
6
7
8
9
10
11
12
13
14
15  entity F_circuit is
16  port (x1, x2, x3, x4, x5: in std_logic;
17        f: out std_logic);
18  end F_circuit;
19
20
21
22
23
24
25
26
27
28
29  architecture structural of F_circuit is
30
31      signal product1, product2, product3, product4, product5:std_logic;
32
33  begin
34
35      Step0: product_of_3 port map (x2, x4, x5, product1);
36      Step1: product_of_3 port map (not x1, x2, not x5, product2);
37      Step2: product_of_2 port map (not x3, not x5, product3);
38      Step3: product_of_2 port map (not x4, not x5, product4);
39      Step4: product_of_3 port map (not x1, not x2, not x4, product5);
40      Step5: SOP port map (product1, product2, product3, product4, product5, f);
41
42
43
44
45  end structural;

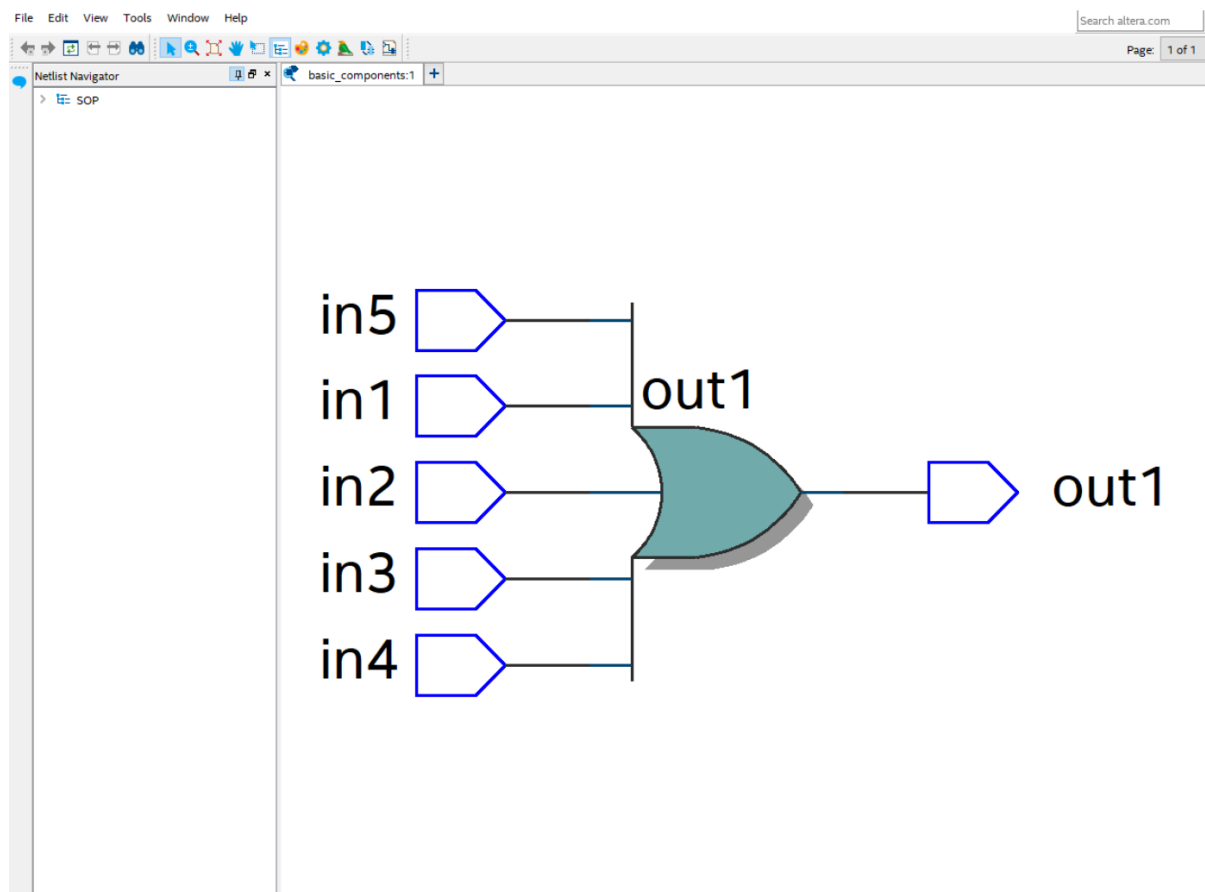
```

-Κυματομορφή προσομοίωσης λειτουργίας εξόδου για όλες τις δυνατές τιμές εισόδου





## -RTL διάγραμμα



## ΠΡΟΒΛΗΜΑ 2°

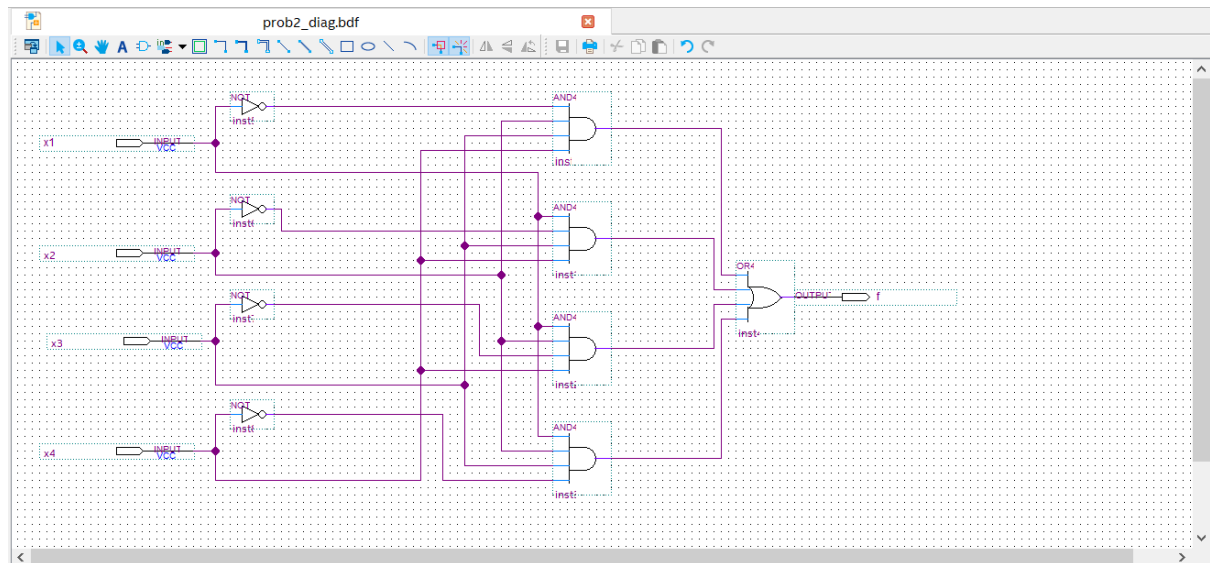
-Μορφή SOP ελαχίστου κόστους για την F με πίνακα Karnaugh

		x1-x2			
		00	01	11	10
x3-x4	00	0	0	0	0
	01	0	0	1	0
	11	0	1	0	1
	10	0	0	1	0

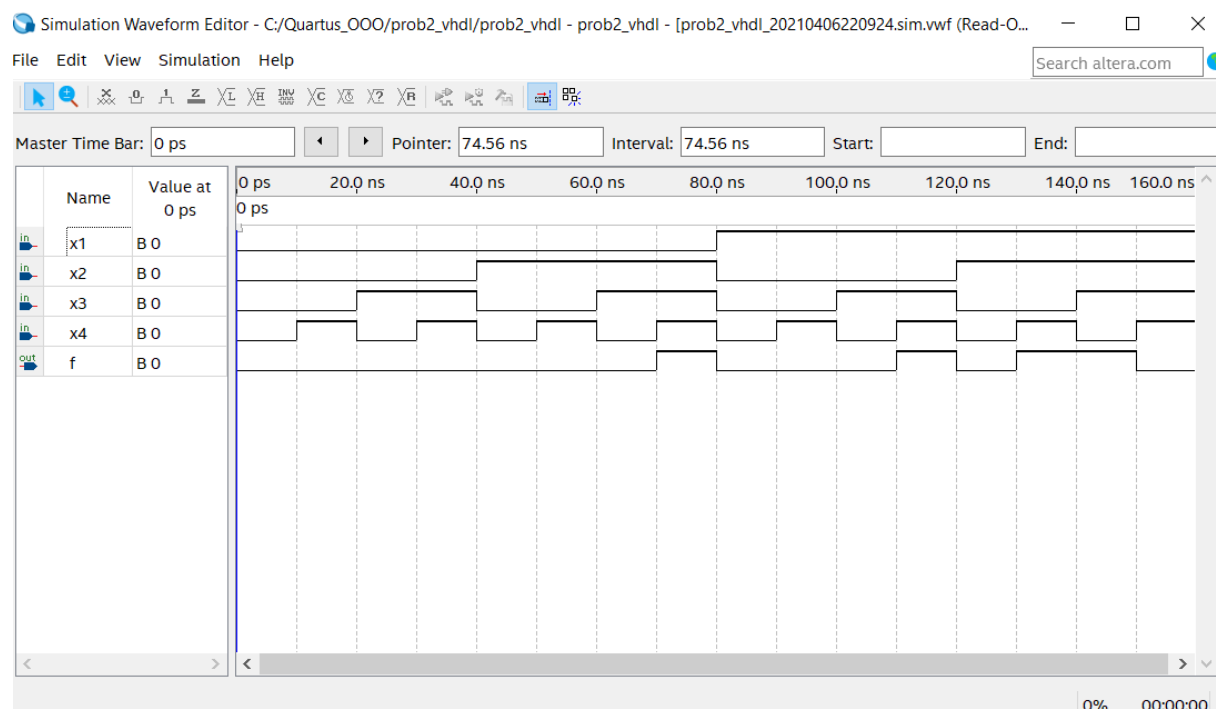
$$F(x_1, x_2, x_3, x_4) = \sum m(7, 11, 13, 14) = \\ x_1'x_2x_3x_4 + x_1x_2'x_3x_4 + x_1x_2x_3'x_4 + x_1x_2x_3x_4'$$

Το συνολικό ελάχιστο κόστος είναι 25. Έχουμε 4 πύλες AND 4 εισόδων, 1 πύλη OR τριών εισόδων, 16 εισόδους για τις πύλες AND καθώς και 4 εισόδους για τις πύλες OR.

-Σχηματικό διάγραμμα για κύκλωμα ελαχίστου κόστους του υποερωτήματος (α)



-Κυματομορφή προσομοίωσης λειτουργίας εξόδου για όλες τις δυνατές τιμές εισόδου (ανά 10ns)



### ΠΡΟΒΛΗΜΑ 3<sup>ο</sup>

-Πίνακας αληθείας

X1	X2	X3	X4	f	Minterm
0	0	0	0	0	$m_0 = x_1' * x_2' * x_3' * x_4'$
0	0	0	1	0	$m_1 = x_1' * x_2' * x_3' * x_4$
0	0	1	0	0	$m_2 = x_1' * x_2' * x_3 * x_4'$
0	0	1	1	1	$m_3 = x_1' * x_2' * x_3 * x_4$
0	1	0	0	1	$m_4 = x_1' * x_2 * x_3' * x_4'$
0	1	0	1	1	$m_5 = x_1' * x_2 * x_3' * x_4$
0	1	1	0	1	$m_6 = x_1' * x_2 * x_3 * x_4'$
0	1	1	1	1	$m_7 = x_1' * x_2 * x_3 * x_4$
1	0	0	0	0	$m_8 = x_1 * x_2' * x_3' * x_4'$
1	0	0	1	0	$m_9 = x_1 * x_2' * x_3' * x_4$
1	0	1	0	0	$m_{10} = x_1 * x_2' * x_3 * x_4'$
1	0	1	1	1	$m_{11} = x_1 * x_2' * x_3 * x_4$
1	1	0	0	0	$m_{12} = x_1 * x_2 * x_3' * x_4'$
1	1	0	1	0	$m_{13} = x_1 * x_2 * x_3' * x_4$
1	1	1	0	0	$m_{14} = x_1 * x_2 * x_3 * x_4'$
1	1	1	1	1	$m_{15} = x_1 * x_2 * x_3 * x_4$

-Συνάρτηση που δημιουργεί το διάγραμμα χρονισμού με την απλούστερη μορφή αθροίσματος γινομένων (SOP)

$x_1x_2 \backslash x_3x_4$	00	01	11	10
00	0	1	0	0
01	0	1	0	0
11	1	1	1	1
10	0	1	0	0

$x_1' * x_2$

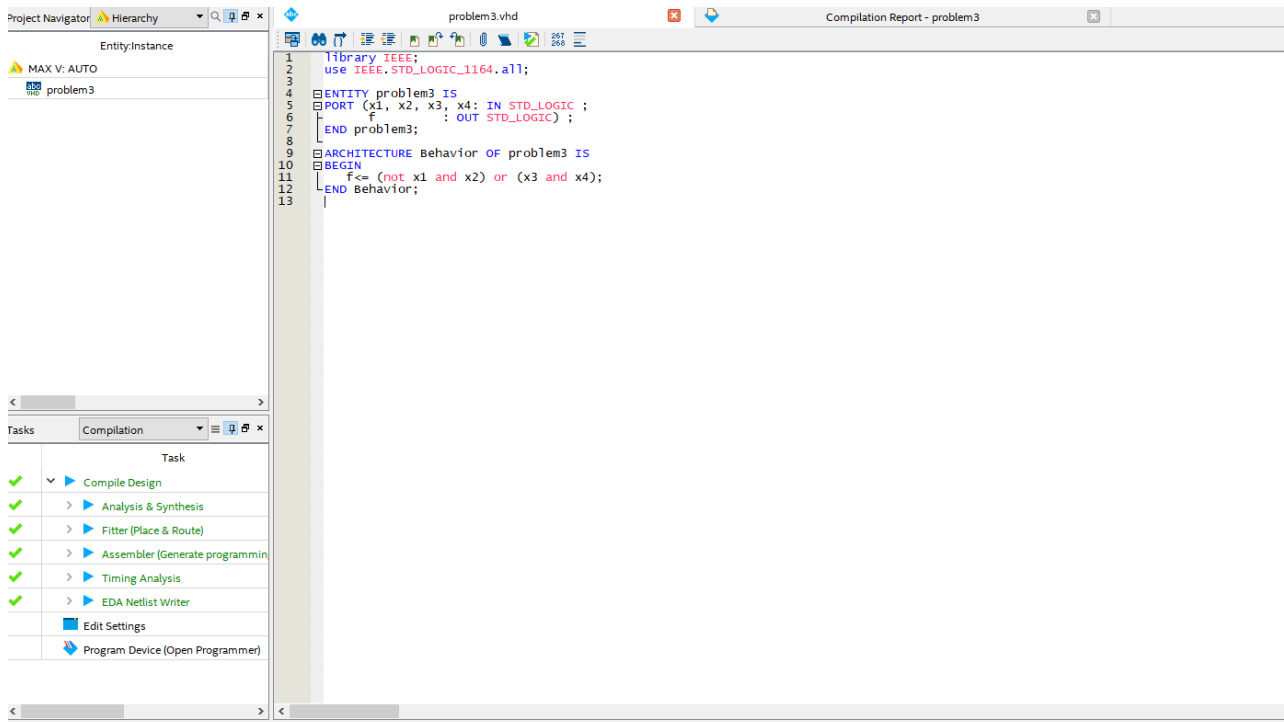
$x_3 * x_4$

$$f(x_1, x_2, x_3, x_4) = x_1' * x_2 + x_3 * x_4$$

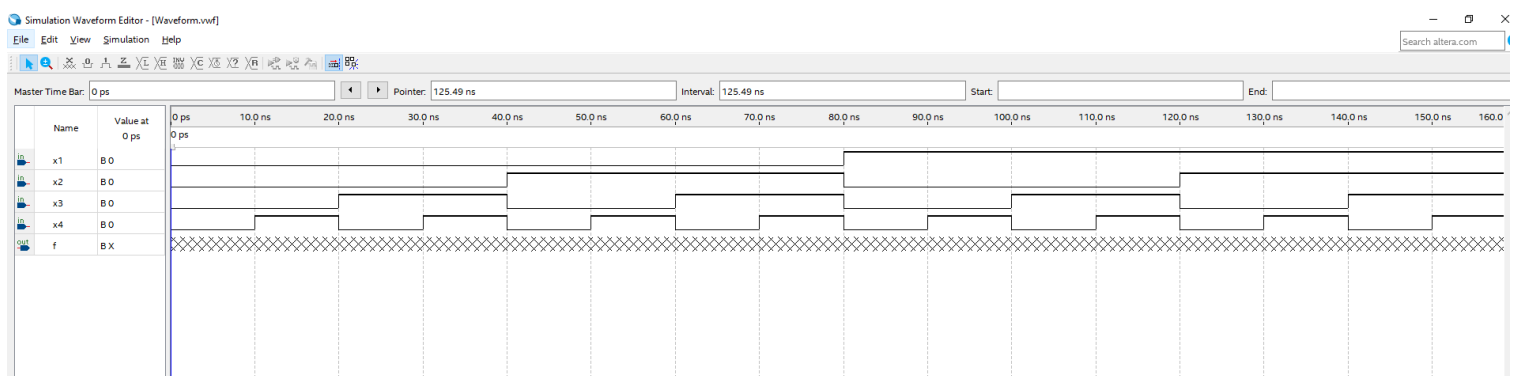
-2ος τρόπος εύρεσης απλούστερης μορφής αθροίσματος γινομένου

$$\begin{aligned}
 \Sigma m(3,4,5,6,7,11,15) &= f(x_1, x_2, x_3, x_4) = \\
 m_3 + m_4 + m_5 + m_6 + m_7 + m_{11} + m_{15} &= \\
 x_1' * x_2' * x_3 * x_4 + x_1' * x_2 * x_3' * x_4' + x_1' * x_2 * x_3' * x_4 + \\
 x_1' * x_2 * x_3 * x_4' + x_1' * x_2 * x_3 * x_4 + x_1 * x_2' * x_3 * x_4 + \\
 x_1 * x_2 * x_3 * x_4 &= \\
 x_1' * x_2 * x_3' * (x_4' + x_4) + x_1' * x_2 * x_3 * (x_4' + x_4) + \\
 (x_1' + x_1) * x_2' * x_3 * x_4 + (x_1' + x_1) * x_2 * x_3 * x_4 &= \\
 x_1' * x_2 * x_3' + x_1' * x_2 * x_3 + x_2' * x_3 * x_4 + x_2 * x_3 * x_4 &= \\
 x_1' * x_2 * (x_3' + x_3) + (x_2' + x_2) * x_3 * x_4 &= \\
 x_1' * x_2 + x_3 * x_4
 \end{aligned}$$

- Πρόγραμμα στη γλώσσα VHDL, το οποίο να υλοποιεί το παραπάνω κύκλωμα



-Κυματομορφή προσομοίωσης λειτουργίας εξόδου για όλες τις δυνατές τιμές εισόδου (ανά 10ns)



-RTL διάγραμμα

