

Jonathan Emanuel Du Aguilera
carne 19249

Fecha: 03.08.2020
sección lab. 11.

Laboratorio #4

Ejercicio 01.

Tabla 1.

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

A	BC	00	01	11	10
0	1	0	0	0	1
1	1	1	1	1	0

A	BC	00	01	11	10
0	$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}BC$	$\bar{A}B\bar{C}$	$\bar{A}B\bar{C}$
1	$\bar{A}B\bar{C}$	$A\bar{B}C$	$A\bar{B}C$	ABC	$AB\bar{C}$

$$Y = \bar{B}\bar{C} + AC + \bar{A}\bar{C}$$

Tabla 2.

A	B	C	Y
0	0	0	1
0	0	1	X
0	1	0	0
0	1	1	0
1	0	0	X
1	0	1	1
1	1	0	0
1	1	1	0

A	BC	00	01	11	10
0	1	X	0	0	0
1	X	1	0	0	0

A	BC	00	01	11	10
0	$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}BC$	$\bar{A}B\bar{C}$	$\bar{A}B\bar{C}$
1	$\bar{A}B\bar{C}$	$A\bar{B}C$	$A\bar{B}C$	ABC	$AB\bar{C}$

$$Y = \bar{B}$$

Tabla 3.

A	B	C	D	Y
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

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

AB	CD	00	01	11	10
00	ABC̄D	ABC̄D	AB̄CD	AB̄C̄D	
01	AB̄CD	AB̄CD	ABC̄D	ABC̄D	
11	AB̄C̄D	AB̄C̄D	ABCD	AB̄D	
10	AB̄CD	AB̄CD	AB̄D	AB̄D	

Tabla 4.

A	B	C	D	Y
0	0	0	0	X
0	0	0	1	X
0	0	1	0	X
0	0	1	1	0
0	1	0	0	0
0	1	0	1	X
0	1	1	0	0
0	1	1	1	X
1	0	0	0	1
1	0	0	1	0
1	0	1	0	X
1	0	1	1	1
1	1	0	1	1
1	1	1	0	X
1	1	1	1	1

AB	CD	00	01	11	10
00	X	X	0	X	
01	0	X	X	0	
11	1	H	A	X	
10	1	D	1	X	

AB	CD	00	01	11	10
00	ABC̄D	X	ABC̄D	ABC̄D	
01	ABC̄D	X	X	ABC̄D	
11	ABC̄D	ABC̄D	ABC̄D	ABC̄D	
10	ABC̄D	ABC̄D	ABC̄D	ABC̄D	

$$\begin{aligned}
 Y = & \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}C\bar{D} \\
 & + A\bar{B}C\bar{D} + A\bar{B}C\bar{D} + A\bar{B}C\bar{D} + A\bar{B}C\bar{D}
 \end{aligned}$$

$$Y = AB + AC + \bar{A}\bar{B}\bar{D}$$

Ejercicio 02

$$1. Y = A \cdot B \cdot C \cdot \bar{D} + A \cdot \bar{B} \cdot C \cdot D + (\bar{A} + B + C + D)$$

A	B	C	D	$A \cdot B \cdot C \cdot \bar{D}$	$A \cdot \bar{B} \cdot C \cdot D$	$\bar{A} + B + C + D$	OR $\Sigma = Y$
1	1	1	1	0	0	0	0
1	1	1	0	1	1	0	1
1	1	0	1	0	1	0	1
1	1	0	0	0	1	0	1
1	0	1	1	0	1	0	1
1	0	1	0	0	1	0	1
1	0	0	1	0	1	0	1
1	0	0	0	0	1	0	1
0	1	1	1	0	0	0	0
0	1	1	0	0	0	0	0
0	1	0	1	0	0	0	0
0	1	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	0	1	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	0	1	1	1

AB	CD			
	00	01	11	10
00	(1)	0	0	0
01	0	0	0	0
11	(1)	1	0	(1)
10	(1)	1	1	(1)

AB	CD			
	00	01	11	10
00	$\bar{A}\bar{B}\bar{C}\bar{D}$	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}B\bar{C}\bar{D}$	$\bar{A}B\bar{C}D$
01	$\bar{A}B\bar{C}\bar{D}$	$\bar{A}B\bar{C}D$	$\bar{A}BC\bar{D}$	$\bar{A}BCD$
11	$A\bar{B}\bar{C}\bar{D}$	$A\bar{B}C\bar{D}$	$AB\bar{C}\bar{D}$	$ABC\bar{D}$
10	$A\bar{B}\bar{C}\bar{D}$	$A\bar{B}C\bar{D}$	$AB\bar{C}\bar{D}$	$ABC\bar{D}$

A	B	C	D	Y
X	0	0	0	1
1	X	0	X	1
1	0	X	X	1
1	X	X	0	1

$$Y = \bar{B}\bar{C}\bar{D} + AC + A\bar{B} + AD$$

Tabla reducida

$$2. Y = \bar{A} \cdot B \cdot C + B \cdot \bar{C} + B \cdot C$$

A	B	C	$\bar{A} \cdot B \cdot C$	$B \cdot \bar{C}$	$B \cdot C$	OR = Y
1	1	1	0	1	1	1
1	1	0	0	0	0	0
1	0	1	0	1	0	1
1	0	0	0	1	0	1
0	1	1	1	1	1	1
0	1	0	0	0	0	0
0	0	1	0	1	0	1
0	0	0	0	1	0	1

A	BC			
	00	01	11	10
0	1	1	1	0
1	1	1	1	0

A	BC			
	00	01	11	10
0	$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}BC$	$\bar{A}B\bar{C}$
1	$A\bar{B}\bar{C}$	$A\bar{B}C$	ABC	$A\bar{B}\bar{C}$

$$Y = \bar{B} + C$$

A	B	C	Y
X	0	X	1
X	X	1	1

$$3. Y = (A+B+C+D) + A \cdot D + B$$

A	B	C	D	$A+B+C+D$	AD	$OR = Y$
1	1	1	1	1	0	1
1	1	1	0	1	0	1
1	1	0	1	0	1	1
1	1	0	0	0	0	1
1	0	1	1	1	0	1
1	0	1	0	0	0	0
1	0	0	1	0	1	1
1	0	0	0	0	0	0
0	1	1	1	1	0	1
0	1	1	0	0	0	1
0	1	0	1	0	0	1
0	1	0	0	0	0	1
0	0	1	1	0	0	0
0	0	1	0	0	0	0
0	0	0	1	1	0	1
0	0	0	0	0	0	0

AB	CD	00	01	11	10
00	D	1	0	0	0
01		1	1	1	1
11		1	1	1	1
10		0	1	1	0

AB	CD	00	01	11	10
00		$\bar{A}\bar{B}C\bar{D}$	$\bar{A}B\bar{C}D$	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}\bar{B}C\bar{D}$
01		$\bar{A}B\bar{C}\bar{D}$	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}B\bar{C}D$	$\bar{A}\bar{B}C\bar{D}$
11		$A\bar{B}\bar{C}\bar{D}$	$A\bar{B}\bar{C}D$	$ABC\bar{D}$	$ABC\bar{D}$
10		$A\bar{B}\bar{C}\bar{D}$	$A\bar{B}\bar{C}D$	$ABC\bar{D}$	$\bar{A}BC\bar{D}$

$$Y = \bar{C}D + B + AD$$

Tabla compactada

A	B	C	D	Y
X	X	0	1	1
X	1	X	X	1
1	X	X	1	1

$$4. Y = B \cdot C + \bar{A} \cdot \bar{B} \cdot \bar{C} + B \cdot \bar{C}$$

A	B	C	BC	$\bar{A}\bar{B}\bar{C}$	$B\bar{C}$	Y
1	1	1	1	0	0	1
1	1	0	0	0	1	1
1	0	1	0	0	0	0
1	0	0	0	0	0	0
0	1	1	1	0	0	1
0	1	0	0	1	1	1
0	0	1	0	0	0	0
0	0	0	0	1	0	1

BC		00	01	11	10
A		0	1	0	1
		0	0	1	1
		1	0	0	1

BC		00	01	11	10
A		0	$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}BC$
		1	$A\bar{B}\bar{C}$	$A\bar{B}C$	ABC
		1	$A\bar{B}\bar{C}$	$A\bar{B}C$	ABC

$Y = \bar{A}\bar{C} + B$

Tabla compacta

A	B	C	Y
0	X	0	1
X	1	X	1

Ejercicio 05

A = Armado

M = Movimiento

P = Ventana/Puerta

L = Alarma y luces encendidas

	O	1	
	apagado	encendido	
	/	X	
	X	/	
	/	X	← Salida

A	M	P	L
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

} no importa que pase porque

no se armó la alarma

} no importa si el P no se activa.

ya uno está mal, se activa la alarma

SOP

$$A \cdot \bar{M} \cdot \bar{P} + A \cdot M \cdot \bar{P} + A \cdot M \cdot P = L$$

POS

$$(A + \bar{M} + P) \cdot (A + M + \bar{P}) \cdot (A + \bar{M} + P) \cdot (A + \bar{M} + \bar{P}) \cdot (\bar{A} + M + \bar{P}) = L$$

A	MP			
	00	01	11	10
0	0	0	0	0
1	1	0	1	1

A	NP			
	00	01	11	10
0	$\bar{A} \bar{M} \bar{P}$	$\bar{A} M \bar{P}$	$\bar{A} M P$	$\bar{A} \bar{M} \bar{P}$
1	$A \bar{M} \bar{P}$	$A \bar{M} P$	$A M \bar{P}$	$A M P$

$$L = AM + A\bar{P}$$

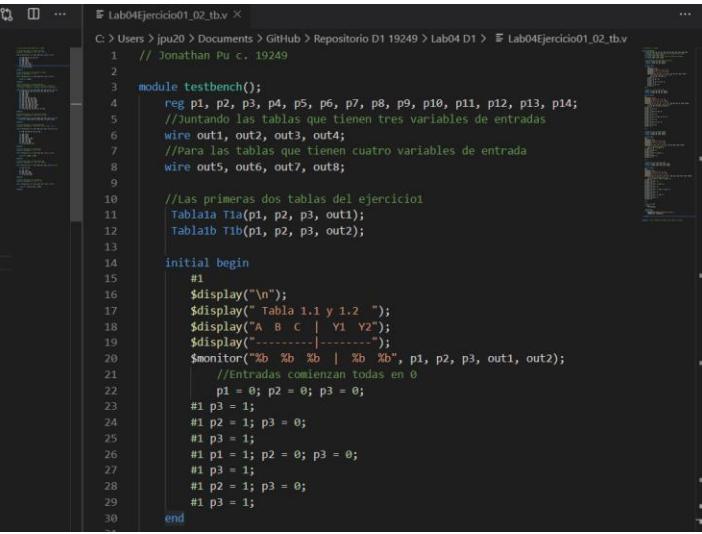
A	M	P	L
1	1	X	1
1	X	0	1

Universidad del Valle de Guatemala
 Digital1
 Jonathan Emanuel Pu Aguilera
 Carné 19249

Fecha de entrega:
 09.08.2020

LABORATORIO 04

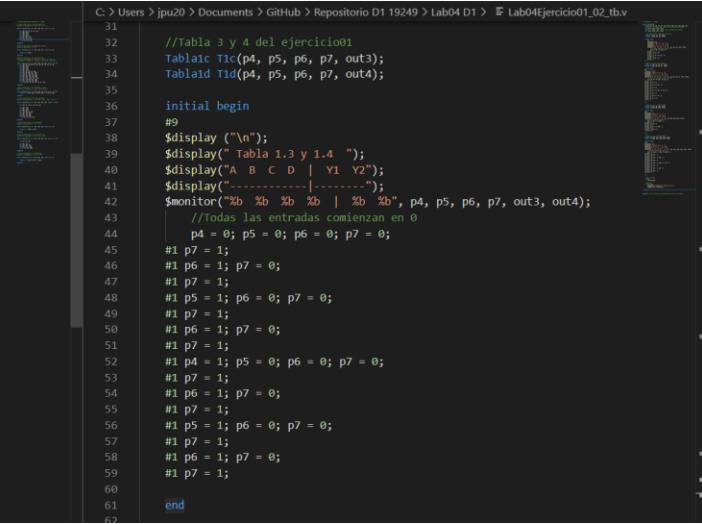
Capturas del código Ejercicio 03



```

Lab04Ejercicio01_02.v
C:\> Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Lab04Ejercicio01_02.v
1 // Ejercicio04 Lab04 Jonathan Pu c. 19249
2
3 /* Ejercicio01 Tabla 1 Con gate modelling
4 Ecuacion a implementar Y = B'C' + AC + A'C' */
5
6 module Tabla1(input wire inA, inB, inC, output wire Y);
7
8     wire NA, NB, NC, A1, A2, A3;
9     not (NA, inA);
10    not (NB, inB);
11    not (NC, inC);
12    and (A1, NB, NC);
13    and (A2, inA, inC);
14    and (A3, NA, NC);
15    or (Y, A1, A2, A3);
16
17 endmodule
18
19 /*Ejercicio01 Tabla 2 con Operadores logicos
20 Funcion a implementar Y = B*/
21
22 module Tabla2(input wire inA, inB, inC, output wire Y);
23
24     assign Y = (~inB);
25
26 endmodule
27
28 /*Ejercicio01 Tabla 3 con gate modelling
29 Funcion a implementar Y = A'B'C'D' + A'B'C*D + A'*B*C'*D
30 + A'*B*C*D' + A*B'C*D' + A*B'*C*D + A*B'*C*D'*/
31
32

```



```

Lab04Ejercicio01_02_tb.v
C:\> Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Lab04Ejercicio01_02_tb.v
1 // Jonathan Pu c. 19249
2
3 module testbench();
4
5 reg p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11, p12, p13, p14;
6 //Juntando las tablas que tienen tres variables de entradas
7 wire out1, out2, out3, out4;
8 //Para las tablas que tienen cuatro variables de entrada
9 wire out5, out6, out7, out8;
10
11 //Las primeras dos tablas del ejercicio
12 Tabla1 T1a(p1, p2, p3, out1);
13 Tabla1b T1b(p1, p2, p3, out2);
14
15 initial begin
16
17 #1
18 $display("\n");
19 $display(" Tabla 1.1 y 1.2 ");
20 $display("A B C | Y1 Y2");
21 $display("-----|-----");
22 $monitor("%b %b %b | %b %b", p1, p2, p3, out1, out2);
23
24 //Entradas comienzan todas en 0
25 p1 = 0; p2 = 0; p3 = 0;
26 #1 p3 = 1;
27 #1 p2 = 1; p3 = 0;
28 #1 p3 = 1;
29 #1 p2 = 1; p3 = 0;
30 #1 p3 = 1;
31
32 //Tabla 3 y 4 del ejercicio01
33 Tabla1c T1c(p4, p5, p6, p7, out3);
34 Tabla1d T1d(p4, p5, p6, p7, out4);
35
36 initial begin
37
38 #1
39 $display("\n");
40 $display(" Tabla 1.3 y 1.4 ");
41 $display("A B C D | Y1 Y2");
42 $display("-----|-----");
43 $monitor("%b %b %b %b | %b %b", p4, p5, p6, p7, out3, out4);
44
45 //Todas las entradas comienzan en 0
46 p4 = 0; p5 = 0; p6 = 0; p7 = 0;
47 #1 p7 = 1;
48 #1 p6 = 1; p7 = 0;
49 #1 p5 = 1; p6 = 0; p7 = 0;
50 #1 p6 = 1; p7 = 0;
51 #1 p7 = 1;
52 #1 p4 = 1; p5 = 0; p6 = 0; p7 = 0;
53 #1 p7 = 1;
54 #1 p6 = 1; p7 = 0;
55 #1 p7 = 1;
56 #1 p5 = 1; p6 = 0; p7 = 0;
57 #1 p7 = 1;
58 #1 p6 = 1; p7 = 0;
59 #1 p7 = 1;
60
61 end
62

```

```

C:\> Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Lab04Ejercicio01_02.v
  53     wire NB, NC, ND, A1, A2, A3, A4;
  54
  55     not (NB, inB);
  56     not (NC, inC);
  57     not (ND, inD);
  58     and (A1, NB, NC, ND);
  59     and (A2, inA, NC);
  60     and (A3, inA, NB);
  61     and (A4, inA, ND);
  62     or (Y, A1, A2, A3, A4);
  63
  64 endmodule
  65
  66 /* Ejercicio02 Tabla 2 con operadores
  67 Ecuacion a implementar Y = B' + C */
  68
  69 module Tabla2b(input wire inA, inB, inC, output wire Y);
  70
  71     assign Y = (~inB) | (inC);
  72
  73 endmodule
  74
  75 /* Ejercicio02 Tabla 3 con gate modelling
  76 Ecuacion a implementar Y = C*D + B + A*D */
  77
  78 module Tabla2c(input wire inA, inB, inC, inD, output wire Y);
  79
  80     wire NC, A1, A2;
  81
  82     not (NC, inC);
  83     and (A1, NC, inD);
  84     and (A2, inA, inB);
  85     or (Y, A1, A2, inB);
  86
  87 endmodule
  88
  89
  90
  91
  92
  93
  94
  95
  96
  97
  98
  99 /* Ejercicio02 Tabla 4 con operadores
 100 Ecuacion a implementar Y = A'*C' + B */
 101
 102 module Tabla2d(input wire inA, inB, inC, output wire Y);
 103
 104     assign Y = (~inA & ~inC) | (inB);
 105
 106 endmodule
 107
 108
 109
 110
 111
 112
 113
 114
 115
 116
 117
 118
 119
 120
 121
 122
 123
 124
 125
 126
 127
 128
 129

```

Screenshot of the Quartus II software showing the Verilog code for the four tables (Tabla2b, Tabla2c, Tabla2d) and their corresponding testbench (Lab04Ejercicio01_02_tb.v). The code includes logic gates and system tasks for displaying and monitoring the results.

```

C:\> Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Lab04Ejercicio01_02_tb.v
  62
  63 //Tabla 2.2 y 2.4 (ejercicio02)
  64 Tabla2b T2b(p8, p9, p10, out5);
  65 Tabla2d T2d(p8, p9, p10, out6);
  66
  67 initial begin
  68 #25
  69 $display("\n");
  70 $display(" Tabla 2.b y 2.d ");
  71 $display("A B C | Y1 Y2");
  72 $display("-----|-----");
  73 $monitor("%b %b | %b %b", p8, p9, p10, out5, out6);
  74 //Entradas comienzan todas en 1
  75 #1 p8 = 1; p9 = 1; p10 = 1;
  76 #1 p10 = 0;
  77 #1 p9 = 0; p10 = 1;
  78 #1 p10 = 0;
  79 #1 p8 = 0; p9 = 1; p10 = 1;
  80 #1 p10 = 0;
  81 #1 p9 = 0; p10 = 1;
  82 #1 p10 = 0;
  83
  84 end
  85
  86 //Tabla 2.1 y 2.3 (ejercicio02)
  87 Tabla2a T2a(p11, p12, p13, p14, out7);
  88 Tabla2c T2c(p11, p12, p13, p14, out8);
  89
  90 initial begin
  91 #33
  92 $display("\n");
  93 $display(" Tabla 2.a y 2.c ");
  94 $display("A B C D | Y1 Y2");
  95 $display("-----|-----");
  96 $monitor("%b %b | %b %b", p11, p12, p13, p14, out7, out8);
  97 //Entradas comienzan todas en 1
  98
  99
 100
 101
 102
 103
 104
 105
 106
 107
 108
 109
 110
 111
 112
 113
 114
 115
 116
 117
 118
 119
 120
 121
 122
 123
 124
 125
 126
 127
 128
 129

```

Screenshot of the Quartus II software showing the Verilog code for the testbench (Lab04Ejercicio01_02_tb.v), which includes stimulus generation and waveform dumping.

Capturas de los circuitos Ejercicio 03 (circuitverse)
 Tablas de verdad de los códigos

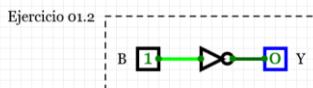
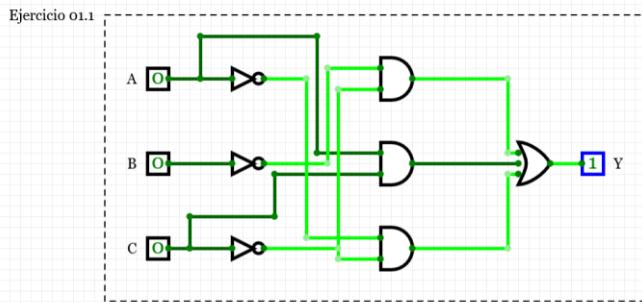


Tabla 1.1 y 1.2				
A	B	C	Y1	Y2
0	0	0	1	1
0	0	1	0	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	1	0

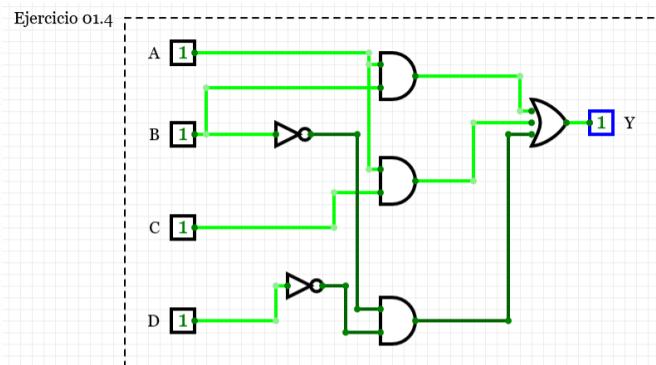
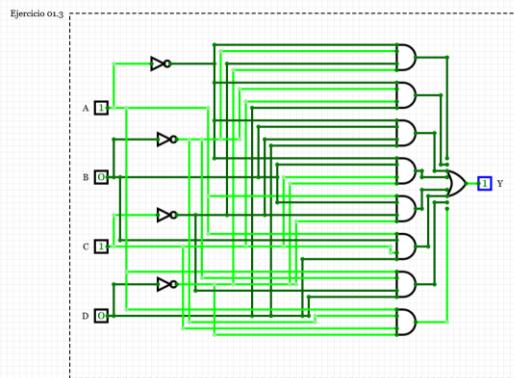


Tabla 1.3 y 1.4					
A	B	C	D	Y1	Y2
0	0	0	0	1	1
0	0	0	1	0	0
0	0	1	0	0	1
0	0	1	1	1	0
0	1	0	0	0	0
0	1	0	1	1	0
0	1	1	0	1	0
0	1	1	1	0	0
1	0	0	0	0	1
1	0	0	1	1	0
1	0	1	0	1	1
1	0	1	1	0	1
1	1	0	0	1	1
1	1	0	1	0	1
1	1	1	0	0	1
1	1	1	1	1	1

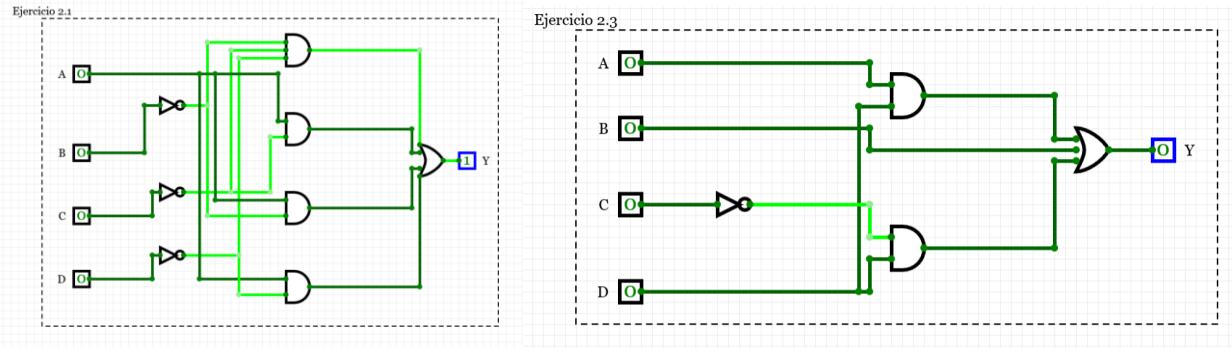


Tabla 2.a y 2.c				Y1	Y2
A	B	C	D		
1	1	1	1	0	1
1	1	1	0	1	1
1	1	0	1	1	1
1	1	0	0	1	1
1	0	1	1	1	1
1	0	1	0	1	0
1	0	0	1	1	1
1	0	0	0	1	0
0	1	1	1	0	1
0	1	1	0	0	1
0	1	0	1	0	1
0	1	0	0	0	1
0	0	1	1	0	0
0	0	1	0	0	0
0	0	0	1	0	1
0	0	0	0	1	0

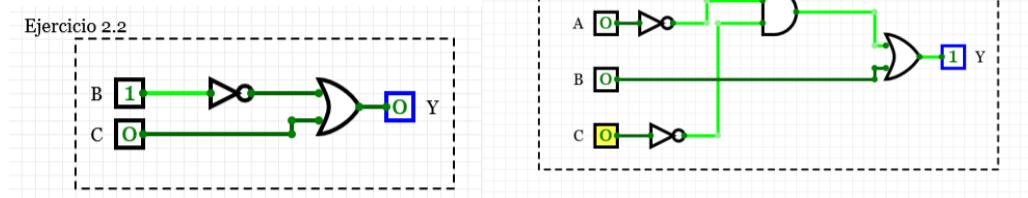
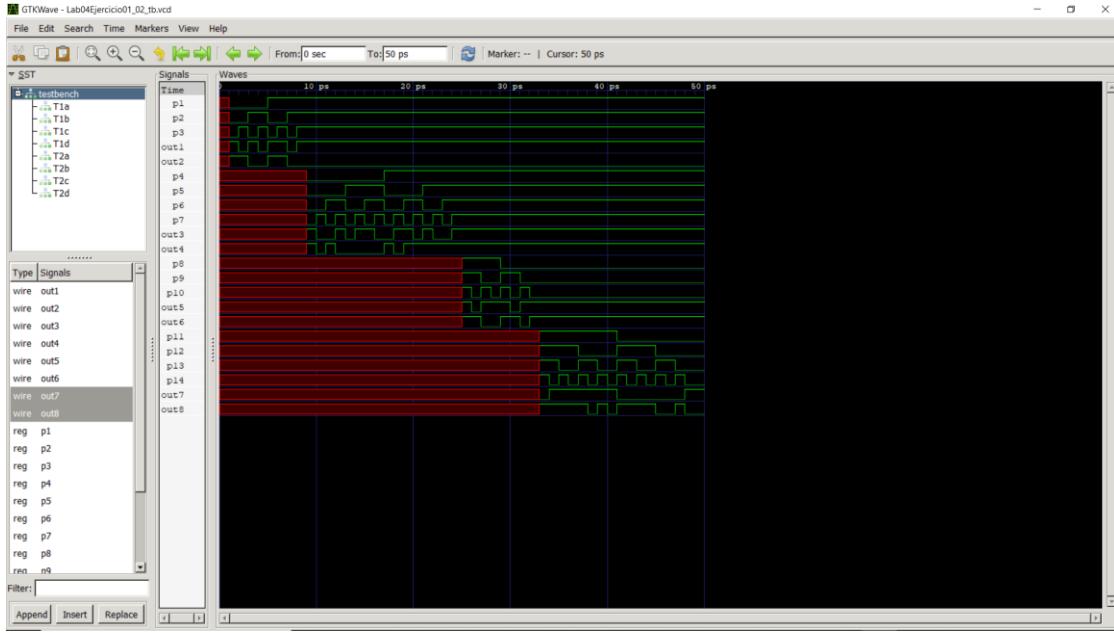


Tabla 2.b y 2.d			Y1	Y2
A	B	C		
1	1	1	1	1
1	1	0	0	1
1	0	1	1	0
1	0	0	1	0
0	1	1	1	1
0	1	0	0	1
0	0	1	1	0
0	0	0	1	1



Capturas Código ejercicio 05

```
c > Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Ejercicio05(Alarma) > Lab05Ejercicio05.v
1 // Ejercicio05 Lab05 Jonathan Pu c. 19249
2
3 // Gate modelling
4 //SOP L = A*M'*P' + A*M*P' + A*M*P
5
6 module SOP(input wire inA, inM, inP, output wire L);
7
8     wire NM, NP, A1, A2, A3;
9     not (NM, inM);
10    not (NP, inP);
11    and (A1, inA, NM, NP);
12    and (A2, inA, inM, NP);
13    and (A3, inA, inM, inP);
14    or (L, A1, A2, A3);
15
16 endmodule
17
18 //POS L = (A + M + P)*(A + M + P')*(A + M' + P)*(A + M' + P')*(A' + M + P')
19
20 module POS(input wire inA, inM, inP, output wire L);
21
22     wire NA, NM, NP, o1, o2, o3, o4, o5;
23     not (NA, inA);
24     not (NM, inM);
25     not (NP, inP);
26     or (o1, inA, inM, inP);
27     or (o2, inA, inM, NP);
28     or (o3, inA, NM, inP);
29     or (o4, inA, NM, NP);
30     or (o5, NA, inM, NP);
31     and (L, o1, o2, o3, o4, o5);
32
33 endmodule
```

```
c > Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Ejercicio05(Alarma) > Lab05Ejercicio05_tb.v
1 // Jonathan Pu c. 19249
2
3 module testbench();
4
5     reg p1, p2, p3;
6     wire out1, out2, out3;
7
8     //SOP
9     SOP S1(p1, p2, p3, out1);
10
11 initial begin
12
13     #1
14     $display("\n");
15     $display("Tabla SOP1 GM");
16     $display("A M P | L");
17     $display("-----|---");
18     $monitor("%b %b %b | %b", p1, p2, p3, out1);
19
20     //Entradas comienzan en 0
21     p1 = 0; p2 = 0; p3 = 0;
22     #1 p3 = 1;
23     #1 p2 = 1; p3 = 0;
24     #1 p3 = 1;
25     #1 p1 = 1; p2 = 0; p3 = 0;
26     #1 p3 = 1;
27     #1 p2 = 1; p3 = 0;
28
29 end
30
31 //POS
32 POS P1(p1, p2, p3, out2);
33
```

```

c: > Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Ejercicio05(Alarma) > Lab05Ejercicio05.v
34
35 //Simplificada L = A'M + A'P'
36
37 module SIMP(input wire inA, inM, inP, output wire L);
38
39   wire NP, A1, A2;
40   not (NP, inP);
41   and (A1, inA, inM);
42   and (A2, inA, inP);
43   or (L, A1, A2);
44
45 endmodule
46
47 //Behavioral modelling
48 module SOP1(input wire inA, inM, inP, output wire L);
49
50   assign L = (inA & ~inM & ~inP) | (inA & inM & ~inP)
51   | (inA & inM & inP);
52
53 endmodule
54
55
56 module POS1(input wire inA, inM, inP, output wire L);
57
58   assign L = (inA | inM | inP) & (inA | inM | ~inP)
59   & (inA | ~inM | inP) & (inA | ~inM | ~inP) & (~inA | inM | ~inP);
60
61 endmodule
62
63
64 module SIMP1C(input wire inA, inM, inP, output wire L);
65
66   assign L = (inA & inM) | (inA & ~inP);
67
68 endmodule
69

```

```

c: > Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Ejercicio05(Alarma) > Lab05Ejercicio05_tb.v
33
34 initial begin
35
36   #9
37   $display("\n");
38   $display("Tabla POS1 BM");
39   $display("A M P | L");
40   $display("-----|---");
41   $monitor("%b %b %b | %b", p1, p2, p3, out2);
42
43   //Entradas comienzan en 0
44   #1 p1 = 0; p2 = 0; p3 = 0;
45   #1 p3 = 1;
46   #1 p2 = 1; p3 = 0;
47   #1 p1 = 1;
48   #1 p2 = 0; p3 = 0;
49   #1 p3 = 1;
50
51 end
52
53 //Simplificada
54 SIMP1C(p1, p2, p3, out3);
55
56 initial begin
57
58   #17
59   $display("\n");
60   $display("Tabla SIMP 1 ");
61   $display("A M P | L");
62   $display("-----|---");
63   $monitor("%b %b %b | %b", p1, p2, p3, out3);
64
65   //Entradas comienzan en 0
66   #1 p1 = 0; p2 = 0; p3 = 0;
67   #1 p3 = 1;
68   #1 p2 = 1; p3 = 0;
69   #1 p1 = 1; p2 = 0; p3 = 0;
70   #1 p3 = 1;
71   #1 p2 = 1; p3 = 0;
72

```

```

c: > Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Ejercicio05(Alarma) > Lab05Ejercicio05_tb.v
71   #1 p3 = 1;
72
73 end
74
75 //behavioral modelling
76 reg p4, p5, p6;
77 wire out4, out5, out6;
78
79 //SOP1
80
81 SOP1 S2(p4, p5, p6, out4);
82
83 initial begin
84   #25
85   $display("\n");
86   $display("Tabla SOP2 BM");
87   $display("A M P | L");
88   $display("-----|---");
89   $monitor("%b %b %b | %b", p4, p5, p6, out4);
90
91   //Entradas comienzan en 0
92   p4 = 0; p5 = 0; p6 = 0;
93   #1 p6 = 1;
94   #1 p5 = 1; p6 = 0;
95   #1 p6 = 1;
96   #1 p4 = 1; p5 = 0; p6 = 0;
97   #1 p6 = 1;
98   #1 p5 = 1; p6 = 0;
99   #1 p6 = 1;
100
101 end
102
103 //POS1
104 POS1 P2(p4, p5, p6, out5);
105 initial begin
106   #33
107   $display("\n");
108   $display("Tabla POS2 BM");
109   $display("A M P | L");

```

```

c: > Users > jpu20 > Documents > GitHub > Repositorio D1 19249 > Lab04 D1 > Ejercicio05(Alarma) > Lab05Ejercicio05_tb.sv
110      $display("-----|---");
111      $monitor("%b %b %b | %b", p4, p5, p6, out5);
112          //Entradas comienzan en 0
113          p4 = 0; p5 = 0; p6 =0;
114          #1 p6 = 1;
115          #1 p5 = 1; p6 = 0;
116          #1 p6 = 1;
117          #1 p4 = 1; p5 = 0; p6 = 0;
118          #1 p6 = 1;
119          #1 p5 = 1; p6 =0;
120          #1 p6 = 1;
121
122      end
123
124      //SIMP1
125
126      SIMP1 SO2(p4, p5, p6, out6);
127      initial begin
128          #41
129          $display("\n");
130          $display("Tabla SIMP 2 ");
131          $display("A M P | L");
132          $display("-----|---");
133          $monitor("%b %b %b | %b", p4, p5, p6, out6);
134          //Entradas comienzan en 0
135          p4 = 0; p5 = 0; p6 =0;
136          #1 p6 = 1;
137          #1 p5 = 1; p6 = 0;
138          #1 p6 = 1;
139          #1 p4 = 1; p5 = 0; p6 = 0;
140          #1 p6 = 1;
141          #1 p5 = 1; p6 =0;
142          #1 p6 = 1;
143
144      end
145      //Fin de codigo
146      initial
147          #49 $finish;
148

```

```

133      $monitor("%b %b %b | %b", p4, p5, p6, out6);
134          //Entradas comienzan en 0
135          p4 = 0; p5 = 0; p6 =0;
136          #1 p6 = 1;
137          #1 p5 = 1; p6 = 0;
138          #1 p6 = 1;
139          #1 p4 = 1; p5 = 0; p6 = 0;
140          #1 p6 = 1;
141          #1 p5 = 1; p6 =0;
142          #1 p6 = 1;
143
144      end
145      //Fin de codigo
146      initial
147          #49 $finish;
148
149      //GTK Wave
150      initial begin
151          $dumpfile("Lab05Ejercicio05_tb.vcd");
152          $dumpvars(0, testbench05);
153
154      end
155
156  endmodule
157  //Fin de modulo de testbench para todas las formas del ejercicio05

```

Capturas circuitos ejercicio05 (circuitverse)

Capturas tablas de verdad del código

Ejercicio 05 SOP

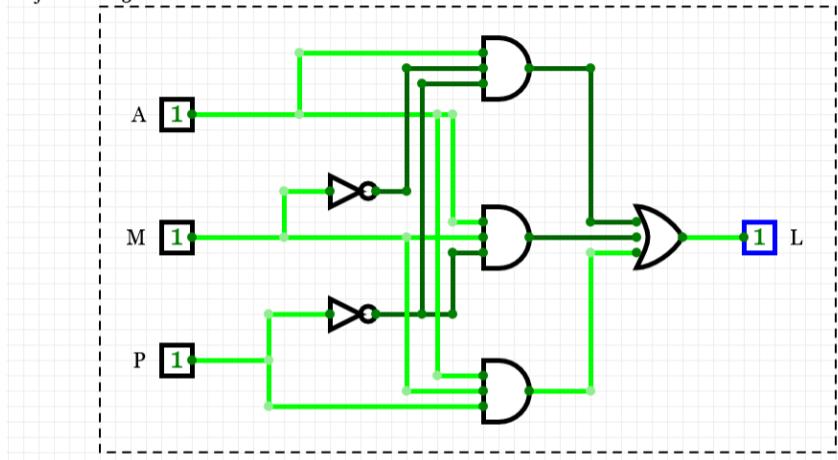


Tabla SOP1 GM			
A	M	P	L
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Tabla SOP2 BM			
A	M	P	L
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Ejercicio 05 POS

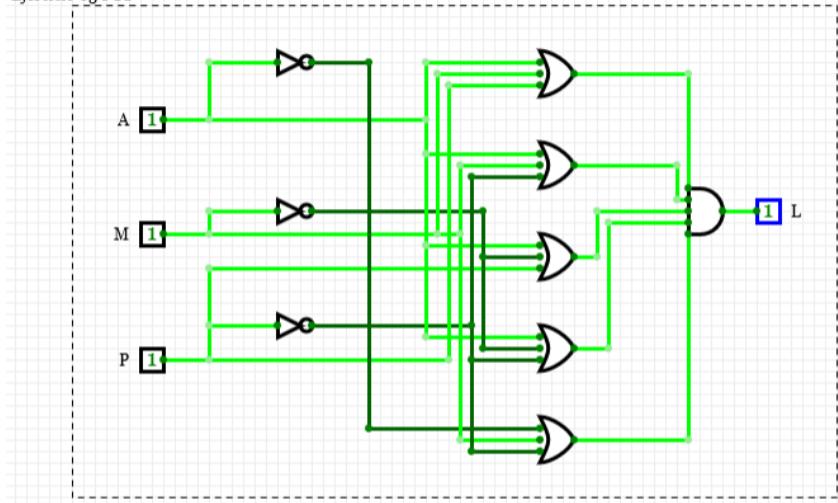


Tabla POS1 GM				Tabla POS2 BM					
A	M	P		L	A	M	P		L
0	0	0		0	0	0	0		0
0	0	1		0	0	0	1		0
0	1	0		0	0	1	0		0
0	1	1		0	0	1	1		0
1	0	0		1	1	0	0		1
1	0	1		0	1	0	1		0
1	1	0		1	1	1	0		1
1	1	1		1	1	1	1		1

Ejercicio 05. Simplificado

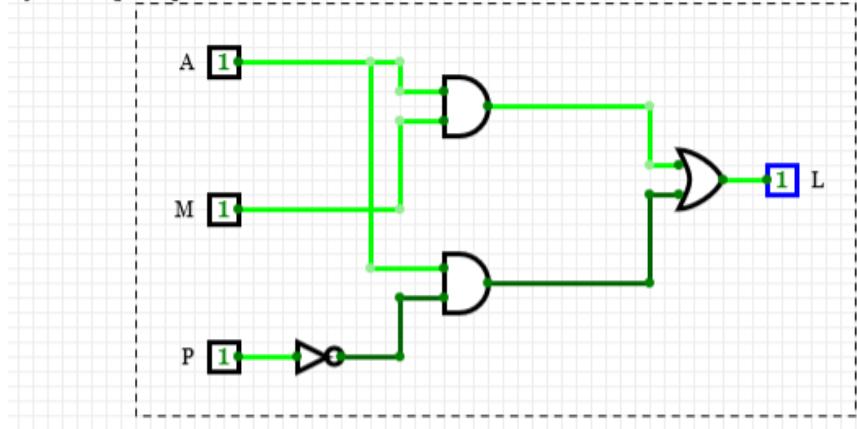
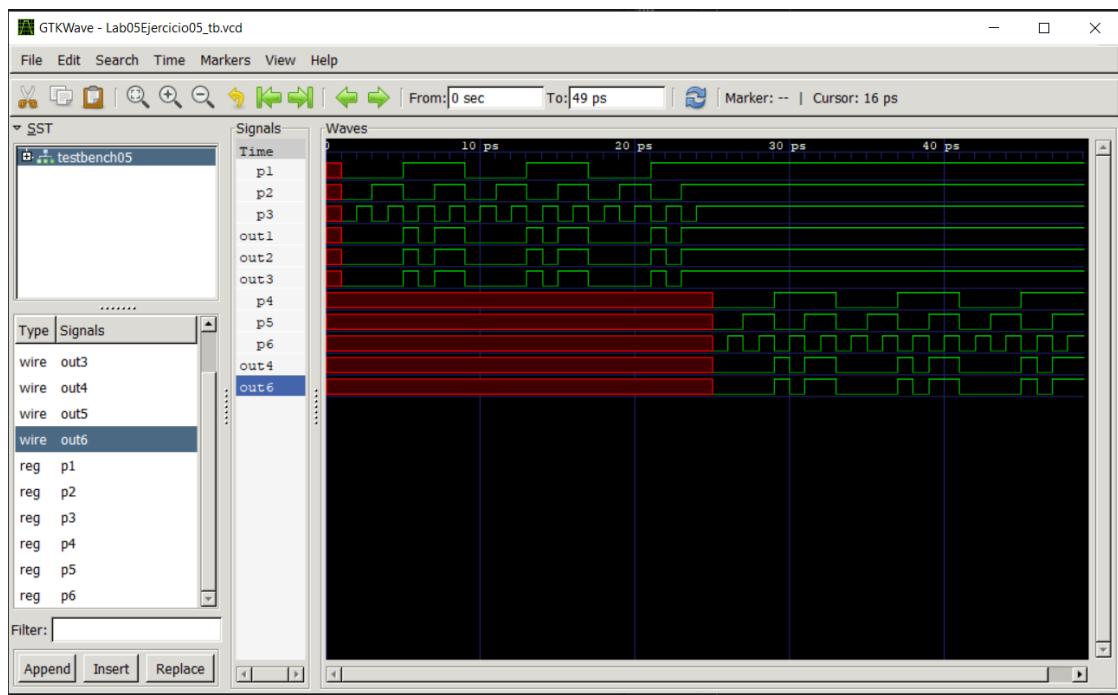


Tabla SIMP 1				
A	M	P		L
0	0	0		0
0	0	1		0
0	1	0		0
0	1	1		0
1	0	0		1
1	0	1		0
1	1	0		1
1	1	1		1

Tabla SIMP 2				
A	M	P		L
0	0	0		0
0	0	1		0
0	1	0		0
0	1	1		0
1	0	0		1
1	0	1		0
1	1	0		1
1	1	1		1

gtkwave Lab05Ejercicio05_tb.vcd Lab05Ejercicio05_tb.gtkw



LINK REPOSITORIO

<https://github.com/pu19249/Repositorio-D1-19249.git>