Lab Report #5

Digital Logic Design

Submitted By:

Haider Ali (FA21-BEE-053)

Malik Ammar Murtaza (FA21-BEE-078)

Muhammad Ahmad (FA21-BEE-095)

Submitted to:

Dr. Adnan Qureshi

LAB #5 Logic Minimization of complex functions using automated tools

Equipment Required:

- Xilinx
- Bread Board
- Logic Gate IC's

Background Theory:

The process of minimizing the tables has been made simpler by using automated tools. Automated tools are important when simplifying equations with high number of bits and can easily simplify any equation. Many minimization tools have been which implement different algorithms to solve k-maps.

Pre lab:

Lab # 05
Pre-Lab

1)

A \ BC	B'C' 00	B'C 01	BC 11	BC' 10
A' 00	m0	m1	m3	m2
A 01	m4	m5	m7	m6

A \ BC				
	A'B'C'	A'B'C	A'BC	A'BC'
	AB'C'	AB'C	ABC	ABC'

Truth table for 3-variables:

A	B	C	F	Minterm	
0	0	0	0	A'B'C m0	A'B'C'
0	0	1	0	m1	A'B'C
0	1	0	1	m2	A'BC'
0	1	1	1	m3	A'BC
1	0	0	1	m4	AB'C'
1	0	1	1	m5	AB'C
1	1	0	0	m6	ABC'
1	1	1	0	m7	ABC

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

$F = AB' + A'B$

$F = A \oplus B$

4- variable k-Map:

AB \ CD	C'D' 00	C'D 01	CD 11	CD' 10
A'B' 00	m0	m1	m3	m2
A'B 01	m4	m5	m7	m6
AB 11	m12	m13	m15	m14
AB' 10	m8	m9	m11	m10

A	B	C	D	F	Minterms
0	0	0	0	1	m0 A'B'C'D'
0	0	0	1	1	m1 A'B'C'D
0	0	1	0	1	m2 A'B'CD'
0	0	1	1	0	m3 A'B'CD
0	1	0	0	1	m4 A'BC'D'
0	1	0	1	1	m5 A'BC'D
0	1	1	0	1	m6 A'BCD'
0	1	1	1	0	m7 A'BCD
1	0	0	0	1	m8 AB'C'D'
1	0	0	1	1	m9 AB'C'D
1	0	1	0	0	m10 AB'CD'
1	0	1	1	0	m11 AB'CD
1	1	0	0	1	m12 ABC'D'
1	1	0	1	1	m13 ABC'D
1	1	1	0	1	m14 ABCD'
1	1	1	1	0	m15 ABCD

CD AB	00	01	11	10
00	A'B'C'D'	A'B'C'D	A'B'CD	A'B'CD'
01	A' B' CD'	A'BC'D	A'BCD	A'BCD'
11	ABC'D'	ABC'D	ABCD	ABCD'
10	AB' C'D'	AB'C'D	AB'CD	AB'CD'

CD AB	C'D'	C'D	CD	CD'
A'B'	1	1	0	1
A'B	1	1	0	1
AB	1	1	0	1
AB'	1	1	0	0

$F = C' + A'D' + BD'$

Lab Tasks:

Implement random 4 variable function using Xilinx

Procedure:

- Select the number of bits for which K-map will be made.
- Specify given inputs and output.
- Select the form of result. (SOP or POS)
- Create project in Xilinx.
- Implement given equation using gate functions in Verilog.
- Output variable will give the output of random function.
- An example for 4-bit is performed below.

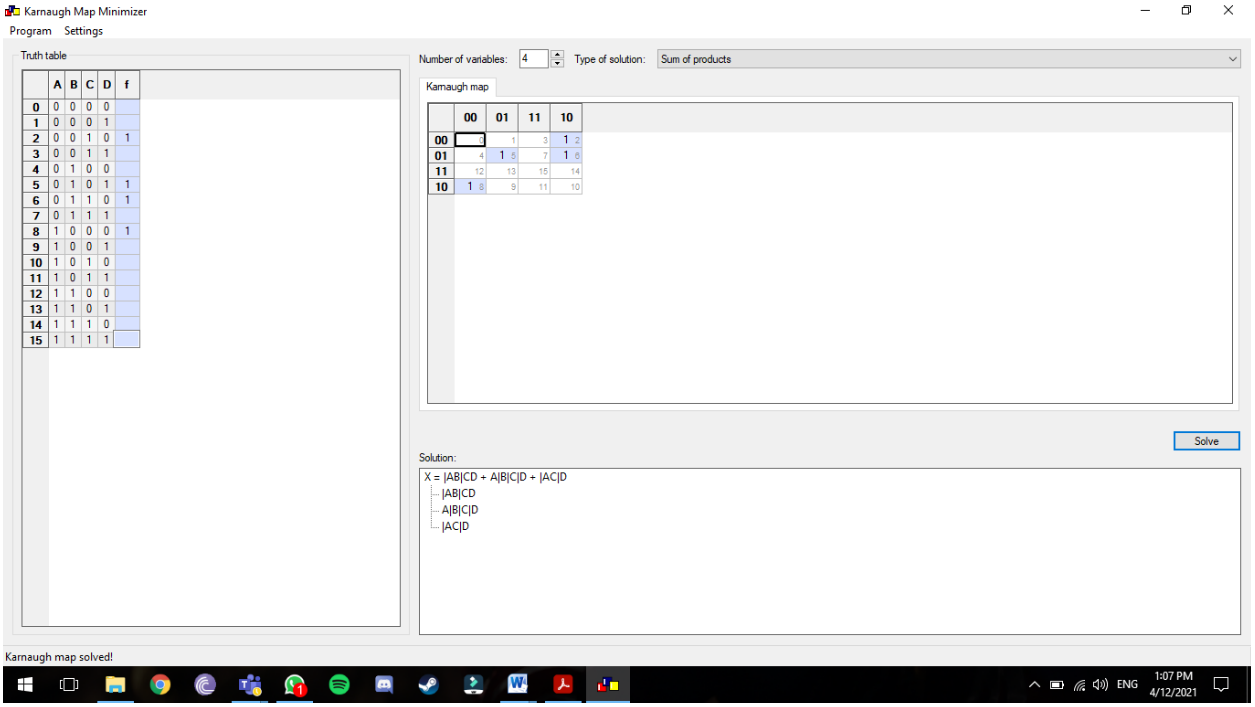
In lab Task 1:

F (A, B, C, D) = ∑ (2,5,6,8)

Functions in sum of Min-terms form:
A'B'CD'+A'BC'D+A'BCD'+AB'C'D'

Function in a simplified form using K-Maps:
A'CD'+AB'C'D'+A'BC'D

K-MAP:



Truth table

	A	B	C	D	f
0	0	0	0	0	
1	0	0	0	1	
2	0	0	1	0	1
3	0	0	1	1	
4	0	1	0	0	
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	
8	1	0	0	0	1
9	1	0	0	1	
10	1	0	1	0	
11	1	0	1	1	
12	1	1	0	0	
13	1	1	0	1	
14	1	1	1	0	
15	1	1	1	1	

Number of variables: 4

Type of solution: Sum of products

Karnaugh map

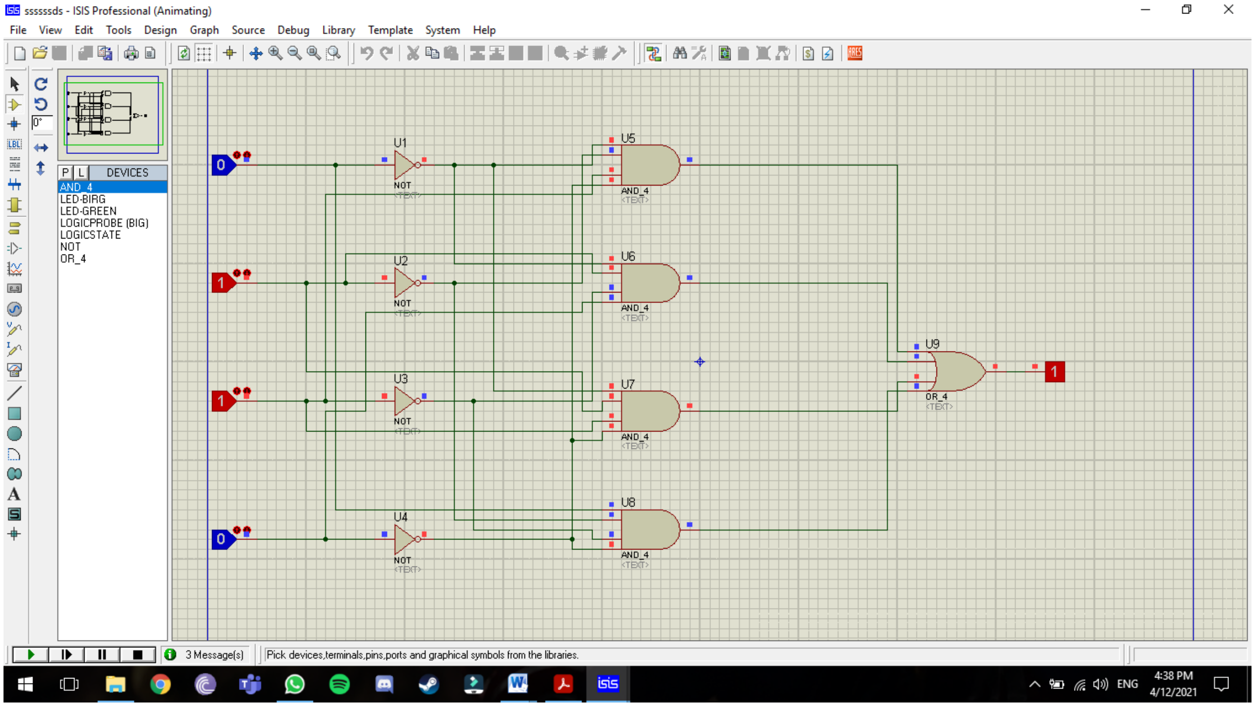
	00	**01**	**11**	**10**
00	0	1	3	1 2
01	4	1 5	7	1 6
11	12	13	15	14
10	1 8	9	11	10

Solution:

```
X = |AB|CD + A|B|C|D + |AC|D
  |AB|CD
  A|B|C|D
  |AC|D
```

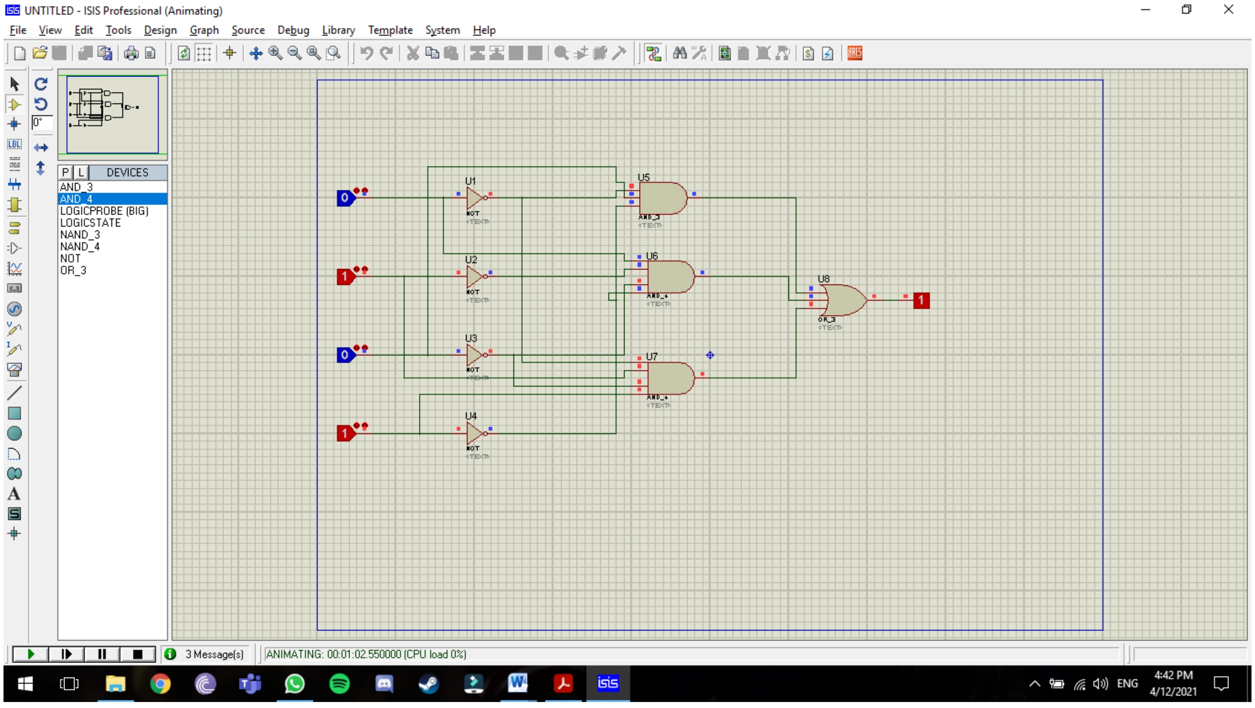
Karnaugh map solved!

Minterms Equation Circuit:



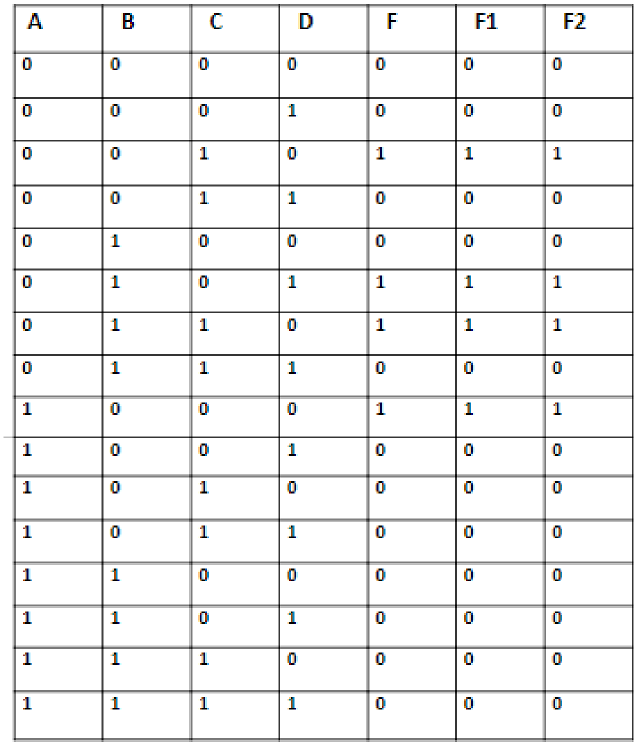
Pick devices,terminals,pins,ports and graphical symbols from the libraries.

Reduced Form Equation:



ANIMATING: 00:01:02.550000 (CPU load 0%)

Truth Table:

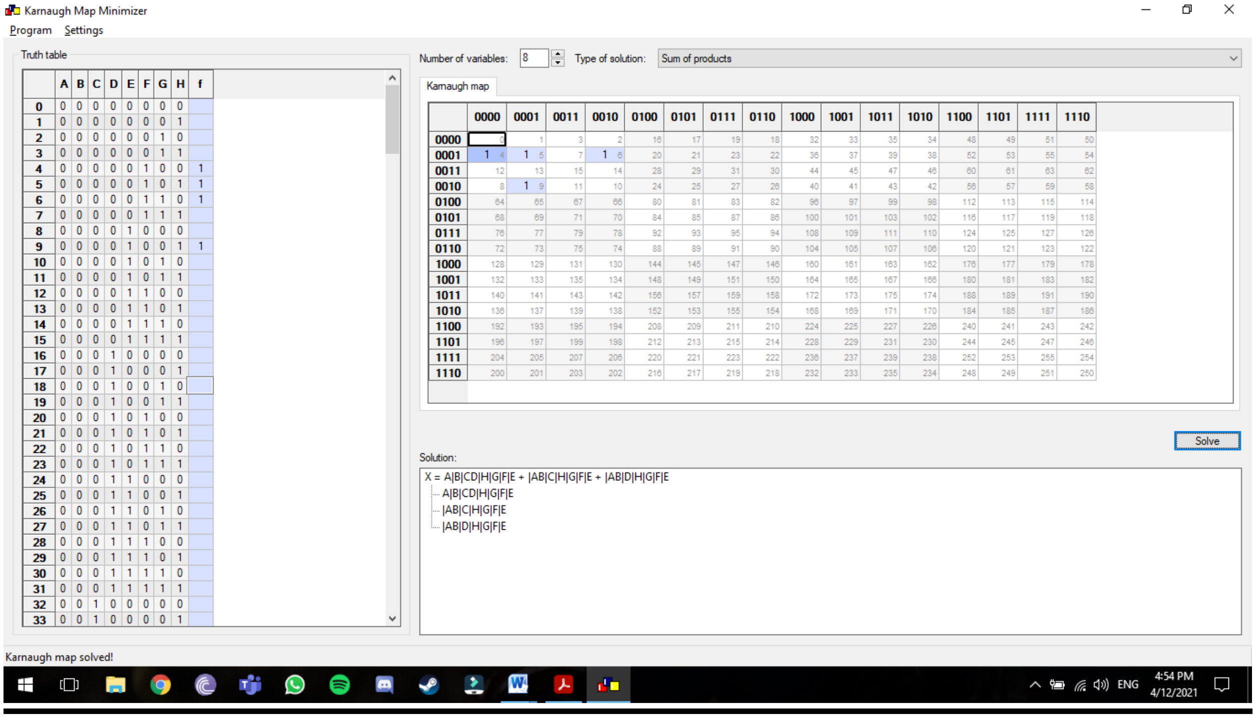


A	B	C	D	F	F1	F2
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	1	1	1
0	0	1	1	0	0	0
0	1	0	0	0	0	0
0	1	0	1	1	1	1
0	1	1	0	1	1	1
0	1	1	1	0	0	0
1	0	0	0	1	1	1
1	0	0	1	0	0	0
1	0	1	0	0	0	0
1	0	1	1	0	0	0
1	1	0	0	0	0	0
1	1	0	1	0	0	0
1	1	1	0	0	0	0
1	1	1	1	0	0	0

In Lab Task 2:

F (A, B, C, D, E, F, G, H) = $\sum$ (4,5,6,9)

K-Map:



Truth table

	A	B	C	D	E	F	G	H	f
0	0	0	0	0	0	0	0	0	
1	0	0	0	0	0	0	0	1	
2	0	0	0	0	0	0	1	0	
3	0	0	0	0	0	0	1	1	
4	0	0	0	0	0	1	0	0	1
5	0	0	0	0	0	1	0	1	1
6	0	0	0	0	0	1	1	0	1
7	0	0	0	0	0	1	1	1	
8	0	0	0	0	1	0	0	0	
9	0	0	0	0	1	0	0	1	1
10	0	0	0	0	1	0	1	0	
11	0	0	0	0	1	0	1	1	
12	0	0	0	0	1	1	0	0	
13	0	0	0	0	1	1	0	1	
14	0	0	0	0	1	1	1	0	
15	0	0	0	0	1	1	1	1	
16	0	0	0	1	0	0	0	0	
17	0	0	0	1	0	0	0	1	
18	0	0	0	1	0	0	1	0	
19	0	0	0	1	0	0	1	1	
20	0	0	0	1	0	1	0	0	
21	0	0	0	1	0	1	0	1	
22	0	0	0	1	0	1	1	0	
23	0	0	0	1	0	1	1	1	
24	0	0	0	1	1	0	0	0	
25	0	0	0	1	1	0	0	1	
26	0	0	0	1	1	0	1	0	
27	0	0	0	1	1	0	1	1	
28	0	0	0	1	1	1	0	0	
29	0	0	0	1	1	1	0	1	
30	0	0	0	1	1	1	1	0	
31	0	0	0	1	1	1	1	1	
32	0	0	1	0	0	0	0	0	
33	0	0	1	0	0	0	0	1	

Number of variables: 8 Type of solution: Sum of products

Karnaugh map

	0000	0001	0011	0010	0100	0101	0111	0110	1000	1001	1011	1010	1100	1101	1111	1110
0000	0	1	3	2	16	17	19	18	32	33	35	34	48	49	51	50
0001	1 4	1 5	7	1 6	20	21	23	22	36	37	39	38	52	53	55	54
0011	12	13	15	14	28	29	31	30	44	45	47	46	60	61	63	62
0010	8	1 9	11	10	24	25	27	26	40	41	43	42	56	57	59	58
0100	64	65	67	66	80	81	83	82	96	97	99	98	112	113	115	114
0101	68	69	71	70	84	85	87	86	100	101	103	102	116	117	119	118
0111	76	77	79	78	92	93	95	94	108	109	111	110	124	125	127	126
0110	72	73	75	74	88	89	91	90	104	105	107	106	120	121	123	122
1000	128	129	131	130	144	145	147	146	160	161	163	162	176	177	179	178
1001	132	133	135	134	148	149	151	150	164	165	167	166	180	181	183	182
1011	140	141	143	142	156	157	159	158	172	173	175	174	188	189	191	190
1010	136	137	139	138	152	153	155	154	168	169	171	170	184	185	187	186
1100	192	193	195	194	208	209	211	210	224	225	227	226	240	241	243	242
1101	196	197	199	198	212	213	215	214	228	229	231	230	244	245	247	246
1111	204	205	207	206	220	221	223	222	236	237	239	238	252	253	255	254
1110	200	201	203	202	216	217	219	218	232	233	235	234	248	249	251	250

Solution:

```
X = A|B|CD|H|G|F|E + |AB|C|H|G|F|E + |AB|D|H|G|F|E
  A|B|CD|H|G|F|E
  |AB|C|H|G|F|E
  |AB|D|H|G|F|E
```

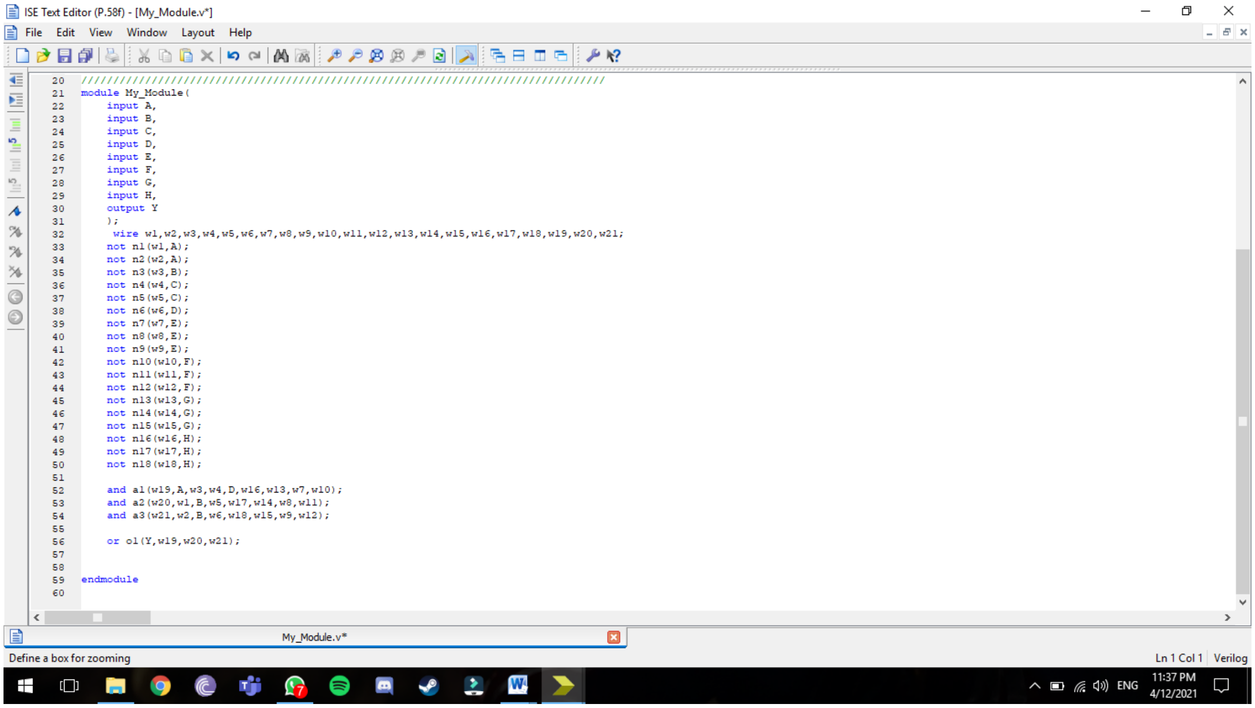
Karnaugh map solved!

Simulation and verifying the results by Xilinx ISE Tool

VeryLog Code:

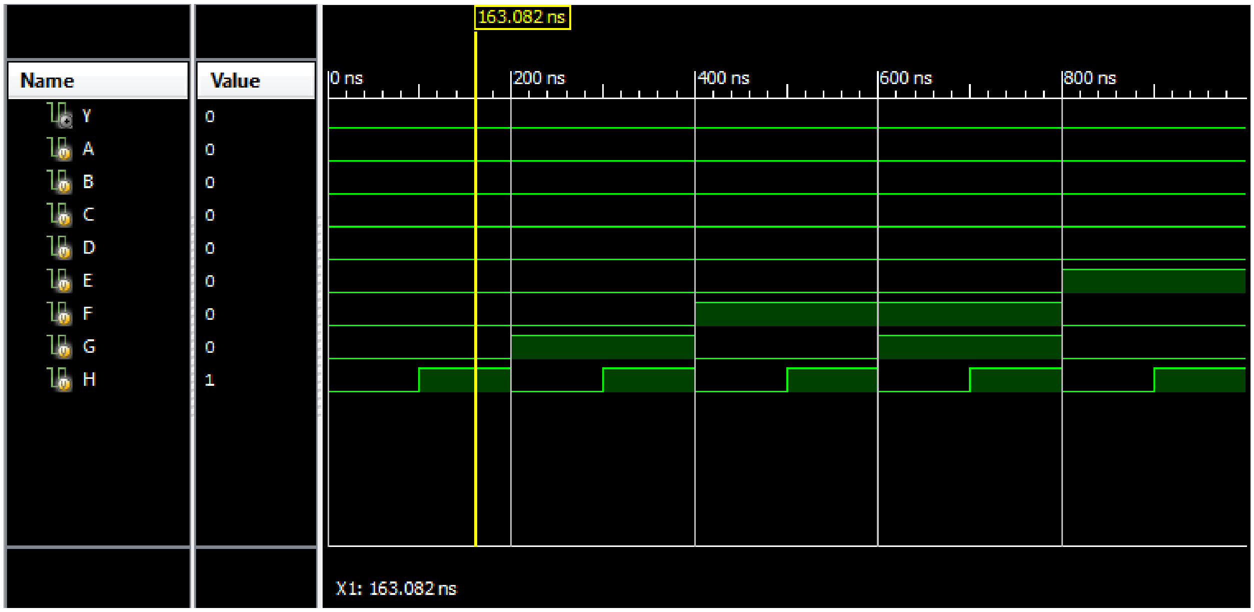
```
module My_Module(
  input A,
  input B,
  input C,
  input D,
  input E,
  input F,
  input G,
  input H,
  output Y
  );
          wire w1,w2,w3,w4,w5,w6,w7,w8,w9,w10,w11,w12,w13,w14,w15,w16,w17,w18,w19,w20,w21;
         not n1(w1,A);
         not n2(w2,A);
         not n3(w3,B);
         not n4(w4,C);
         not n5(w5,C);
         not n6(w6,D);
```

```
    not n7(w7,E);
    not n8(w8,E);
    not n9(w9,E);
    not n10(w10,F);
    not n11(w11,F);
    not n12(w12,F);
    not n13(w13,G);
    not n14(w14,G);
    not n15(w15,G);
    not n16(w16,H);
    not n17(w17,H);
    not n18(w18,H);

    and a1(w19,A,w3,w4,D,w16,w13,w7,w10);
    and a2(w20,w1,B,w5,w17,w14,w8,w11);
    and a3(w21,w2,B,w6,w18,w15,w9,w12);

    or o1(Y,w19,w20,w21);


endmodule
```



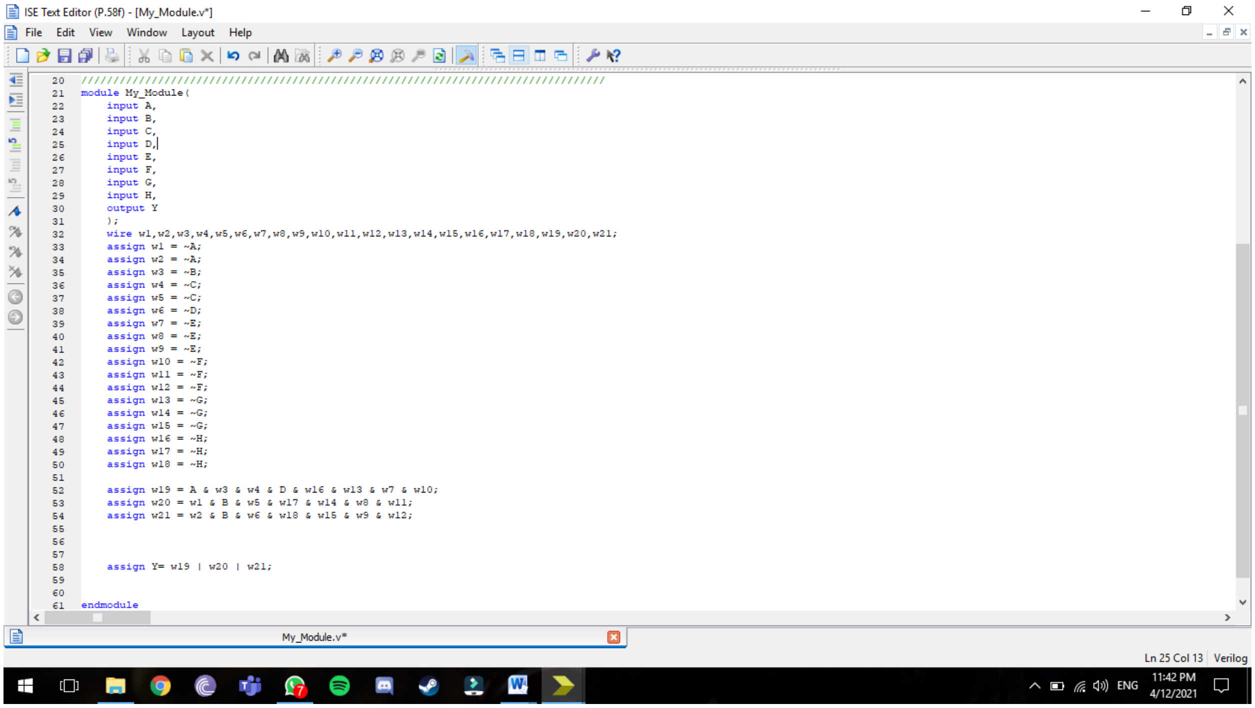
```
//////////////////////////////////////////////////////////////////////////////////
module My_Module(
    input A,
    input B,
    input C,
    input D,
    input E,
    input F,
    input G,
    input H,
    output Y
    );
     wire w1,w2,w3,w4,w5,w6,w7,w8,w9,w10,w11,w12,w13,w14,w15,w16,w17,w18,w19,w20,w21;
    not n1(w1,A);
    not n2(w2,A);
    not n3(w3,B);
    not n4(w4,C);
    not n5(w5,C);
    not n6(w6,D);
    not n7(w7,E);
    not n8(w8,E);
    not n9(w9,E);
    not n10(w10,F);
    not n11(w11,F);
    not n12(w12,F);
    not n13(w13,G);
    not n14(w14,G);
    not n15(w15,G);
    not n16(w16,H);
    not n17(w17,H);
    not n18(w18,H);

    and a1(w19,A,w3,w4,D,w16,w13,w7,w10);
    and a2(w20,w1,B,w5,w17,w14,w8,w11);
    and a3(w21,w2,B,w6,w18,w15,w9,w12);

    or o1(Y,w19,w20,w21);


endmodule

```

Simulation Wave Form



Post Lab

Verilog description by using DATA FLOW for 8-bit variables



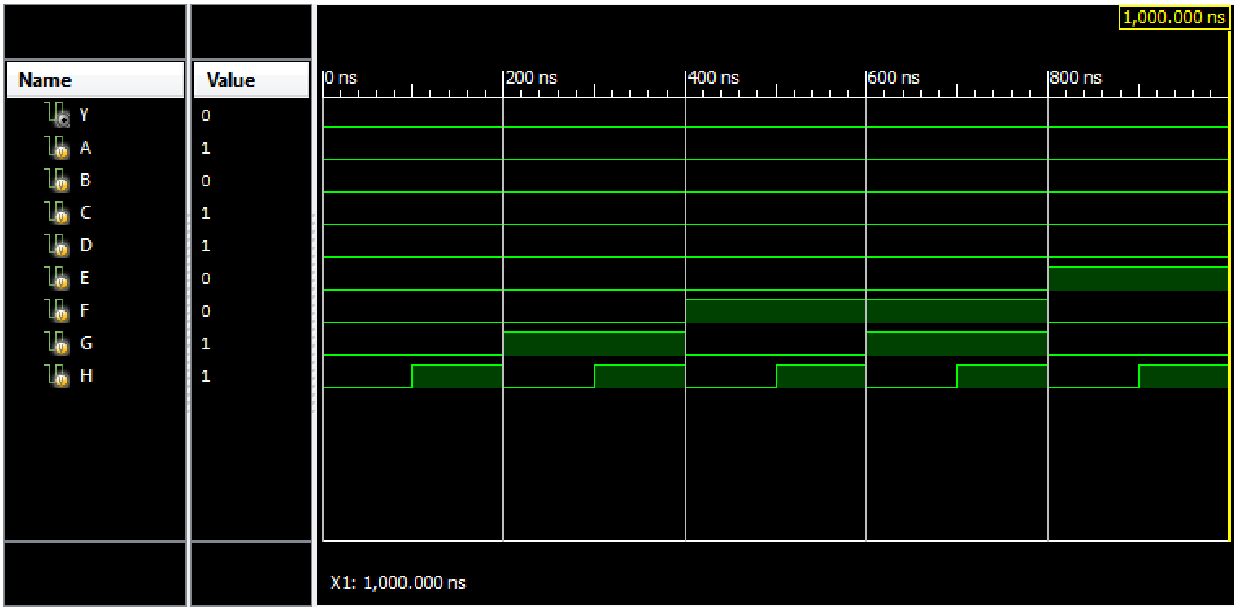
ISE Text Editor (P.58f) - [My_Module.v*]
File Edit View Window Layout Help
My_Module.v*
Ln 25 Col 13
Verilog
11:42 PM
4/12/2021
ENG

```
////////////////////////////////////////////////////////////////////////////////
module My_Module(
    input A,
    input B,
    input C,
    input D,
    input E,
    input F,
    input G,
    input H,
    output Y
    );
    wire w1,w2,w3,w4,w5,w6,w7,w8,w9,w10,w11,w12,w13,w14,w15,w16,w17,w18,w19,w20,w21;
    assign w1 = ~A;
    assign w2 = ~A;
    assign w3 = ~B;
    assign w4 = ~C;
    assign w5 = ~C;
    assign w6 = ~D;
    assign w7 = ~E;
    assign w8 = ~E;
    assign w9 = ~E;
    assign w10 = ~F;
    assign w11 = ~F;
    assign w12 = ~F;
    assign w13 = ~G;
    assign w14 = ~G;
    assign w15 = ~G;
    assign w16 = ~H;
    assign w17 = ~H;
    assign w18 = ~H;

    assign w19 = A & w3 & w4 & D & w16 & w13 & w7 & w10;
    assign w20 = w1 & B & w5 & w17 & w14 & w8 & w11;
    assign w21 = w2 & B & w6 & w18 & w15 & w9 & w12;



    assign Y= w19 | w20 | w21;


endmodule
```

```
module My_Module(
   input A,
   input B,
   input C,
   input D,
   input E,
   input F,
   input G,
   input H,
   output Y
   );
            wire w1,w2,w3,w4,w5,w6,w7,w8,w9,w10,w11,w12,w13,w14,w15,w16,w17,w18,w19,w20,w21;
            assign w1 = ~A;
            assign w2 = ~A;
            assign w3 = ~B;
            assign w4 = ~C;
            assign w5 = ~C;
            assign w6 = ~D;
            assign w7 = ~E;
            assign w8 = ~E;
            assign w9 = ~E;
```

```
        assign w10 = ~F;
        assign w11 = ~F;
        assign w12 = ~F;
        assign w13 = ~G;
        assign w14 = ~G;
        assign w15 = ~G;
        assign w16 = ~H;
        assign w17 = ~H;
        assign w18 = ~H;

        assign w19 = A & w3 & w4 & D & w16 & w13 & w7 & w10;
        assign w20 = w1 & B & w5 & w17 & w14 & w8 & w11;
        assign w21 = w2 & B & w6 & w18 & w15 & w9 & w12;



        assign Y= w19 | w20 | w21;


endmodule
```

Simulation Wave Form



Name	Value
Y	0
A	1
B	0
C	1
D	1
E	0
F	0
G	1
H	1

Critical Analysis:

In this lab we learnt how to minimize large variables expression just by using an automated tool known as K-map minimizer and later we verified the results bu using the Xilinx software with the help of Verilog code of both structural and data flow coding and simulated the wave forms as well.