Department of Computer Science
Faculty of Computing
UNIVERSITI TEKNOLOGI MALAYSIA

SUBJECT : SCSR1013 DIGITAL LOGIC

SESSION/SEM : 20202021 / 1

LAB 2 : COMBINATIONAL LOGIC CIRCUIT DESIGN SIMULATION

NAME 1 : SAYANG ELYIANA AMIERA BINTI HELMEY (A20EC0143)

NAME 2 : YASSER EHAB MOHIELDIN (A17CS3013)

REMARKS :

MARKS:

D. Lab Activities

Part 1

Simulating logic circuit, construct truth table and timing diagram with Deeds.

Given Boolean expression as follow:

$$Y = AB + BC + AC$$

1. Convert the non-standard Boolean expression into standard form.

Y = AB + BC + AC

Y = AB (C + C') + A (B + B') C + (A + A') BC

Y = (ABC + ABC') + (ABC + AB'C) + (ABC + A'BC)

Y = A'BC + AB'C + ABC' + ([ABC + ABC)] + ABC)

Y = A'BC + AB'C + ABC' + (ABC + ABC)

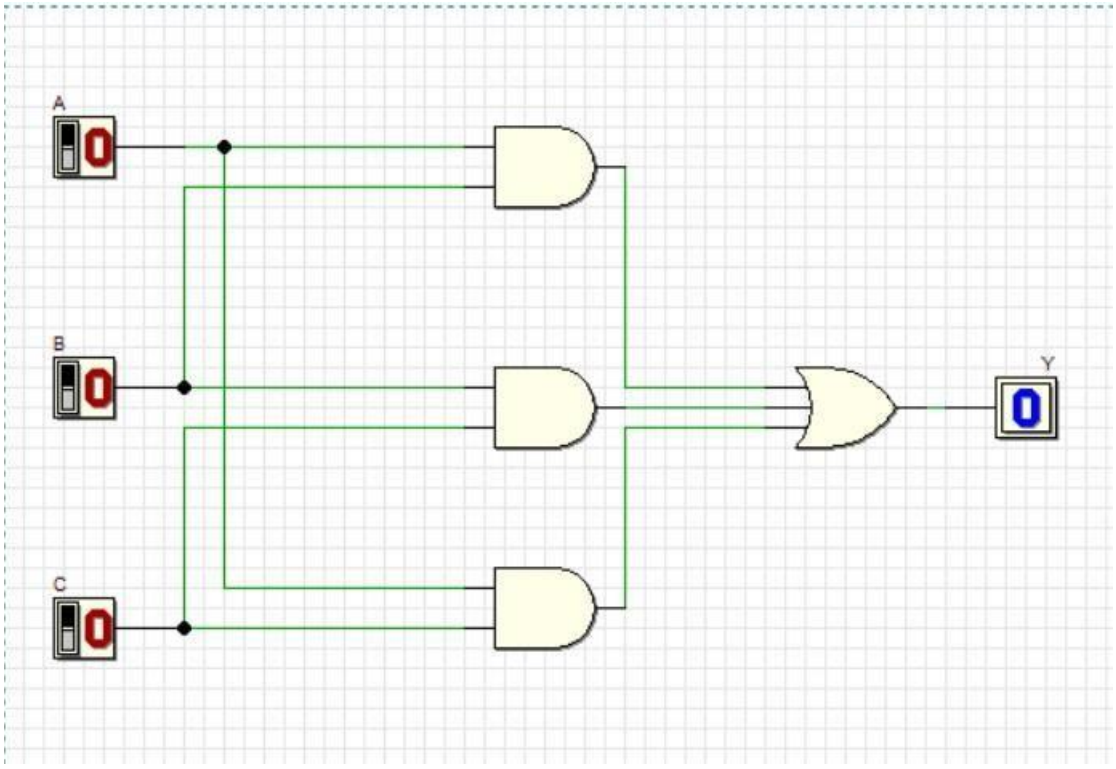
Y = A'BC + AB'C + ABC' + ABC

2. Based on standard form expression, complete the following truth table.

INPUT			OUTPUT
A	**B**	**C**	**Y**
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

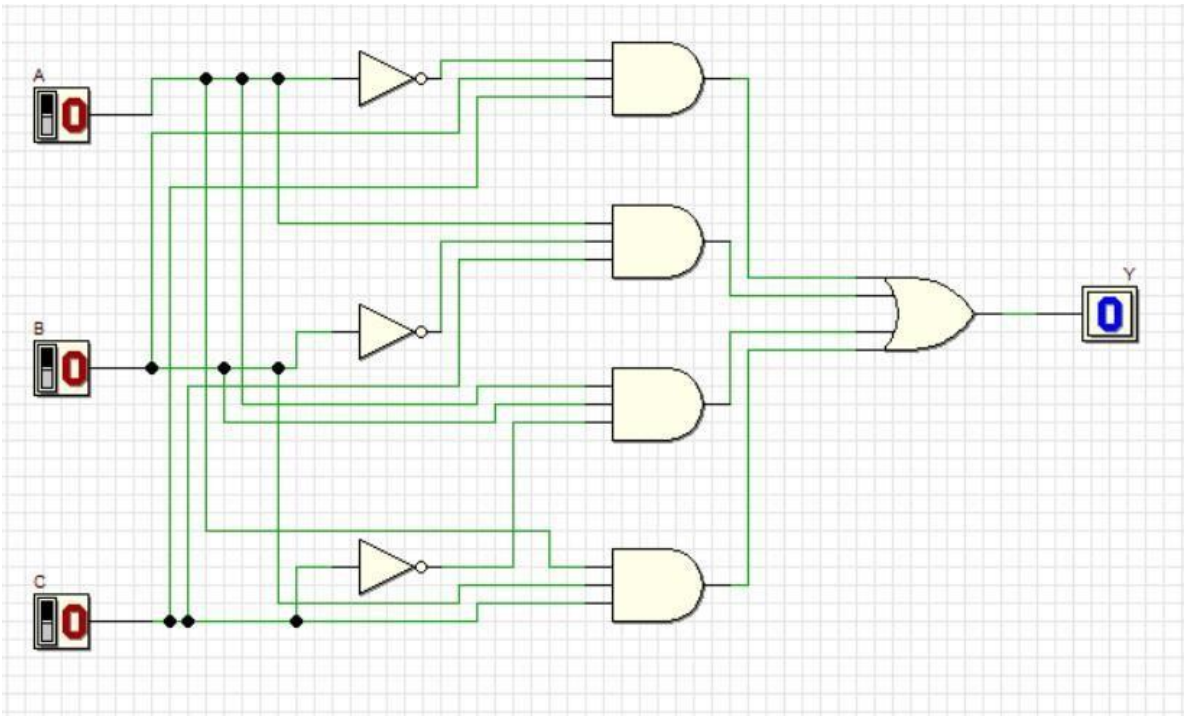
3. Using Deeds Simulator, draw the following circuits:

a) Circuit (i) for non-standard form (based on the given expression).



Circuit (i)

b) Circuit (ii) for standard form (from your answer in question (1)).



Circuit (ii)

4. Simulate these two circuits in step (3) and complete their truth table.

Compare the simulation result for these two truth tables. What is your conclusion?

Circuit (i)

INPUT			OUTPUT
A	**B**	**C**	**Y**
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Circuit (ii)

INPUT			OUTPUT
A	**B**	**C**	**Y**
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Conclusion:

The result of both truth table of standard form and non-standard form equation are the same. The non-standard form equation is the simplest form of an expression which makes it more convenient for efficient implementation and produce a simplest circuit.

5. Simulate output of circuit (ii) with Timing Diagram. Illustrate some examples of different inputs and output.

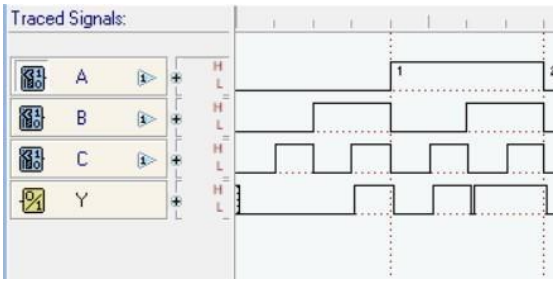


Figure 1: Timing diagram of output circuit (ii)

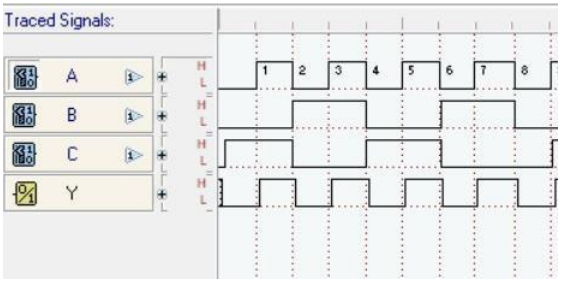


Figure 2: Examples with different inputs and outputs

Part 2

Combinational circuit design process and simulate with Deeds Simulator.

Design Process

i) Determine Parameter Input / Output and their relations.
ii) Construct Truth Table.
iii) Using K-Map, get the SOP optimized form of all Boolean equation outputs.
iv) Draw the circuit and use duality symbol; convert AND-OR circuit to NAND gates ONLY.
v) Simulate the design using Deeds Simulator. Check the results according to Truth Table and Timing Diagram Operation.

Problem Situation

A new digital fault diagnoses circuit is requested to be designed for analyzing four bit 2's complement input binary number from sensors A, B, C, and D. Sensor A represents input MSB and sensor D represents input LSB. As shown in the following Figure 5, bit pattern analysis from input sensors A, B, C, and D will trigger four different output errors (active HIGH) of type E1, E2, E3, and E4.

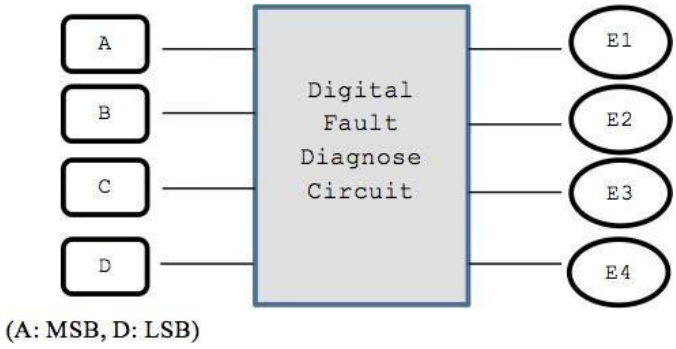


Figure 5

The following rules are used to activate the error's signal type:

RULE 1: E1 is activated if the input number is positive ODD and the majority of the bits is '0'.
RULE 2: E2 is activated if the input number is positive EVEN and the majority of the bits is '0'.
RULE 3: E3 is activated if the input number is negative ODD and the majority of the bits is '1'.
RULE 4: E4 is activated if the input number is negative EVEN and the majority of the bits is '1'.
RULE 5: The output of error signal is invalid if the input has equal bit '0' and bit '1'.
(**NOTE:** Positive ODD is positive numbers that are odd andnegative EVEN is negative numbers that are even).

Experimental Steps

1. Complete Truth Table 1 for Digital Fault Diagnose Circuit. Use variables A, B, C and D as inputs; E1, E2, E3 and E4 as outputs.

Truth Table 1

INPUTS				OUTPUTS			
A	**B**	**C**	**D**	**E1**	**E2**	**E3**	**E4**
0	0	0	0	0	1	0	0
0	0	0	1	1	0	0	0
0	0	1	0	0	1	0	0
0	0	1	1	X	X	X	X
0	1	0	0	0	1	0	0
0	1	0	1	X	X	X	X
0	1	1	0	X	X	X	X
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	X	X	X	X
1	0	1	0	X	X	X	X
1	0	1	1	0	0	1	0
1	1	0	0	X	X	X	X
1	1	0	1	0	0	1	0
1	1	1	0	0	0	0	1
1	1	1	1	0	0	1	0

2. Using K-MAP, get minimized SOP Boolean expressions for E1, E2, E3 and E4 circuits.

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

E1 = $\bar{A}BD$

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

E2 = $\bar{A}D$

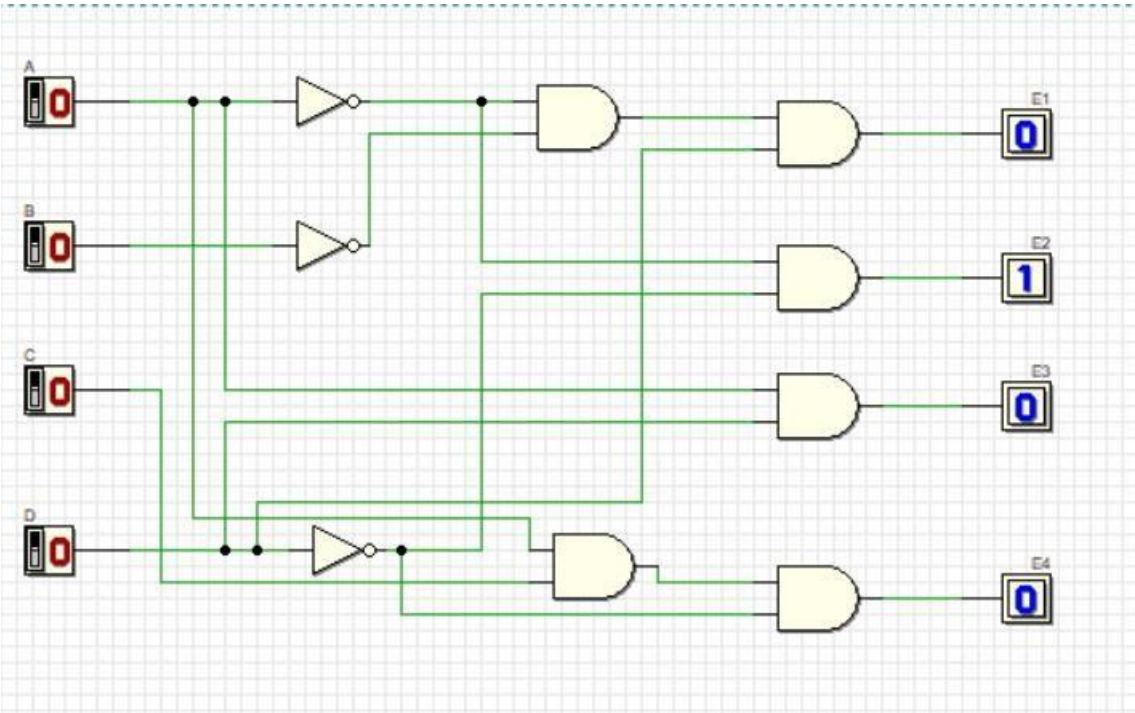
AB \ CD	00	01	11	10
00	0	0	X	0
01	0	X	0	0
11	X	1	1	0
10	0	X	1	X

E3 = $A\,D$

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

E4 = $A\,C\,\bar{D}$

3. From the Boolean expression in the step (2), draw your final E1, E2, E3 and E4 circuits using 2 input basic gates (AND, OR, NOT). Use Deeds Simulator.



4. Simulate the Deeds circuit in step (3):

a) Update Truth Table 2 based on the simulation result.

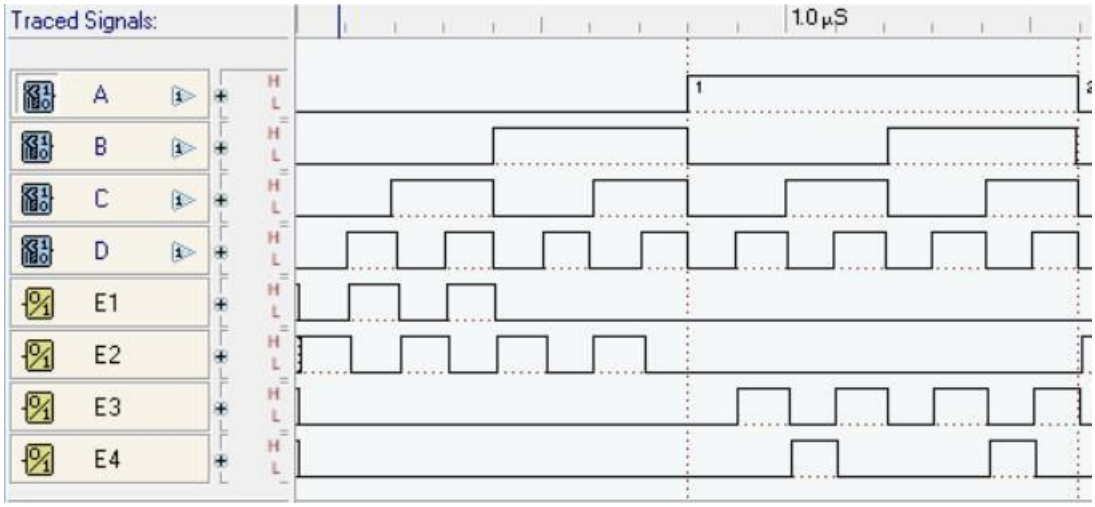
Truth Table 2

INPUTS				OUTPUTS			
A	**B**	**C**	**D**	**E1**	**E2**	**E3**	**E4**
0	0	0	0	0	1	0	0
0	0	0	1	1	0	0	0
0	0	1	0	0	1	0	0
0	0	1	1	1	0	0	0
0	1	0	0	0	1	0	0
0	1	0	1	0	0	0	0
0	1	1	0	0	1	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	0	0	1
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	0
1	1	1	0	0	0	0	1
1	1	1	1	0	0	1	0

Compare the output results in Truth Table 2 with Truth Table 1. What is your conclusion?

By comparing Truth Table 1 and 2, the output for Truth Table 1 has don’t care terms, while Truth Table 2 does not have don’t care terms. By including don’t care terms, the use of logic gates and variables is minimized and the simplest logic circuit with fewer variables and logic gates is produced.

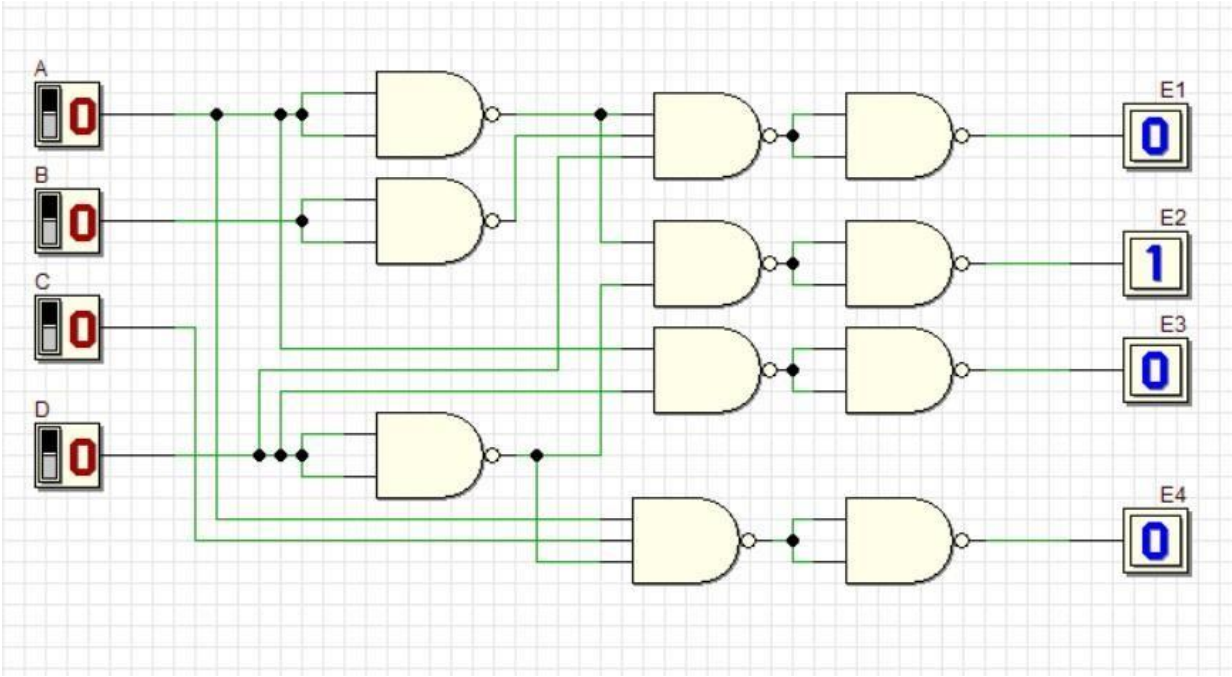
b) Timing Diagram



Explain some analysis values based on your timing diagram:

From the timing diagram above, we could analyze that for output E1, when the inputs A=0, B = 0, D = 1 and C = 0 or 1, the output is 1 (high). For output E2, when the inputs A = 0, D = 0, B and C = 0 or 1, the output is 1 (high). For output E3, when the inputs A=1, D = 1, B and C = 0 or 1, the output is 1 (high). For output E4, when the inputs A=1, C = 1, D = 0 and B = 0 or 1, the output is 1 (high).

5. Using dual symbol concept, convert your circuit in step (3) to NAND gates only. Use Deeds Simulator.



6. Simulate the Deeds circuit in step (5):

a) Update Truth Table 3 based on the simulation result.

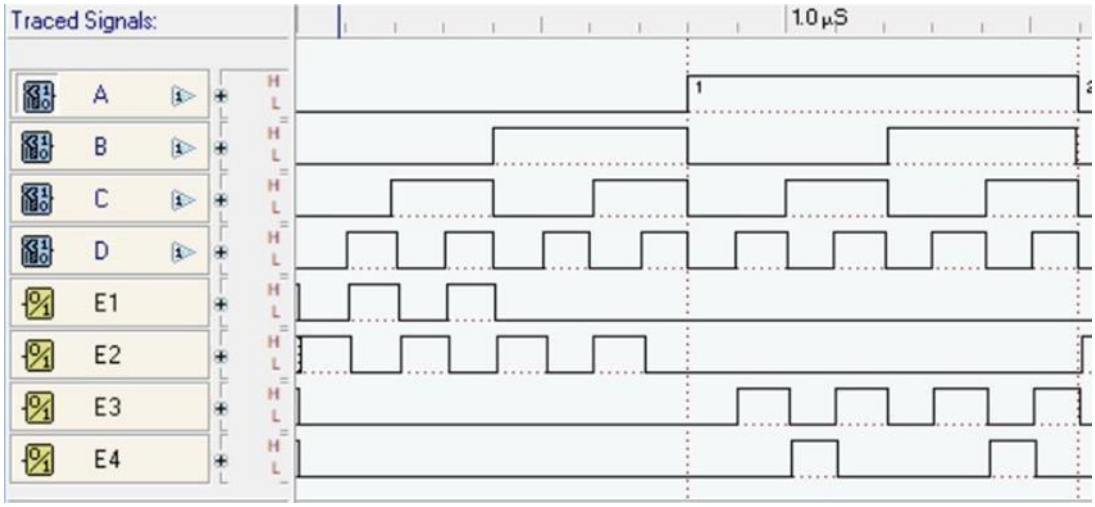
Truth Table 3

INPUTS				OUTPUTS			
A	**B**	**C**	**D**	**E1**	**E2**	**E3**	**E4**
0	0	0	0	0	1	0	0
0	0	0	1	1	0	0	0
0	0	1	0	0	1	0	0
0	0	1	1	1	0	0	0
0	1	0	0	0	1	0	0
0	1	0	1	0	0	0	0
0	1	1	0	0	1	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	0	0	1
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	0
1	1	1	0	0	0	0	1
1	1	1	1	0	0	1	0

Compare the output results in Truth Table 3 with Truth Table 2. What is your conclusion?

By comparing Truth Table 2 and 3, the input and output for both Truth Table are the same. Thus, this shows that NAND gate is a universal gate as it could perform functions of basic gates that are used in circuit 2 (Truth Table 2) which are AND gate and NOT gate.

b) Timing Diagram



Explain some analysis values based on your timing diagram:

From the timing diagram above, we could analyze that for output E1, when the inputs A=0, B = 0, D = 1 and C = 0 or 1, the output is 1 (high). For output E2, when the inputs A = 0, D = 0, B and C = 0 or 1, the output is 1 (high). For output E3, when the inputs A=1, D = 1, B and C = 0 or 1, the output is 1 (high). For output E4, when the inputs A=1, C = 1, D = 0 and B = 0 or 1, the output is 1 (high).

Fully Completed ☐ Partially Completed ☐ Checkedby