

**Department of Computer Science**

**Faculty of Computing**

**UNIVERSITI TEKNOLOGI MALAYSIA**

|  |
| --- |
| SUBJECT : SCSR1013 DIGITAL LOGIC    SESSION/SEM: 2020-21-01 |
| **LAB 2 : COMBINATIONAL LOGIC CIRCUIT DESIGN**  **SIMULATION** |
| Group No: 04.  NAME 1 : ABU SAKIB BIN LUTFUL HASSAN (A20EC4007).  NAME 2 : LIM LI TING (A20EC5008).  NAME 3 : MD FARIDUL ISLAM (A20EC4030). |

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 (C+ C̅) +BC (A+A̅) +AC (B+B̅)

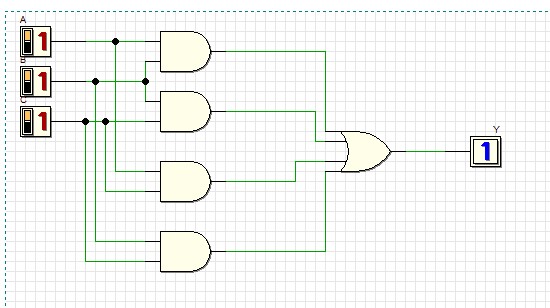
=ABC+ABC̅ +ABC+A̅BC+ABC+AB̅ C

1. 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 |

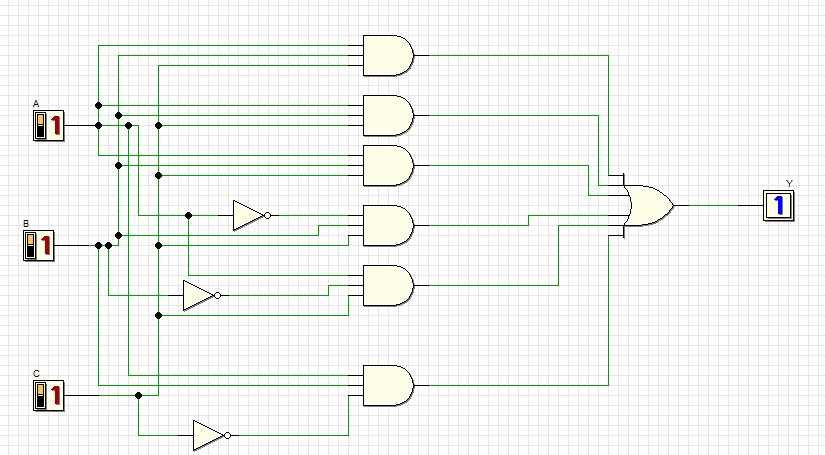
1. Using Deeds Simulator, draw the following circuits:

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

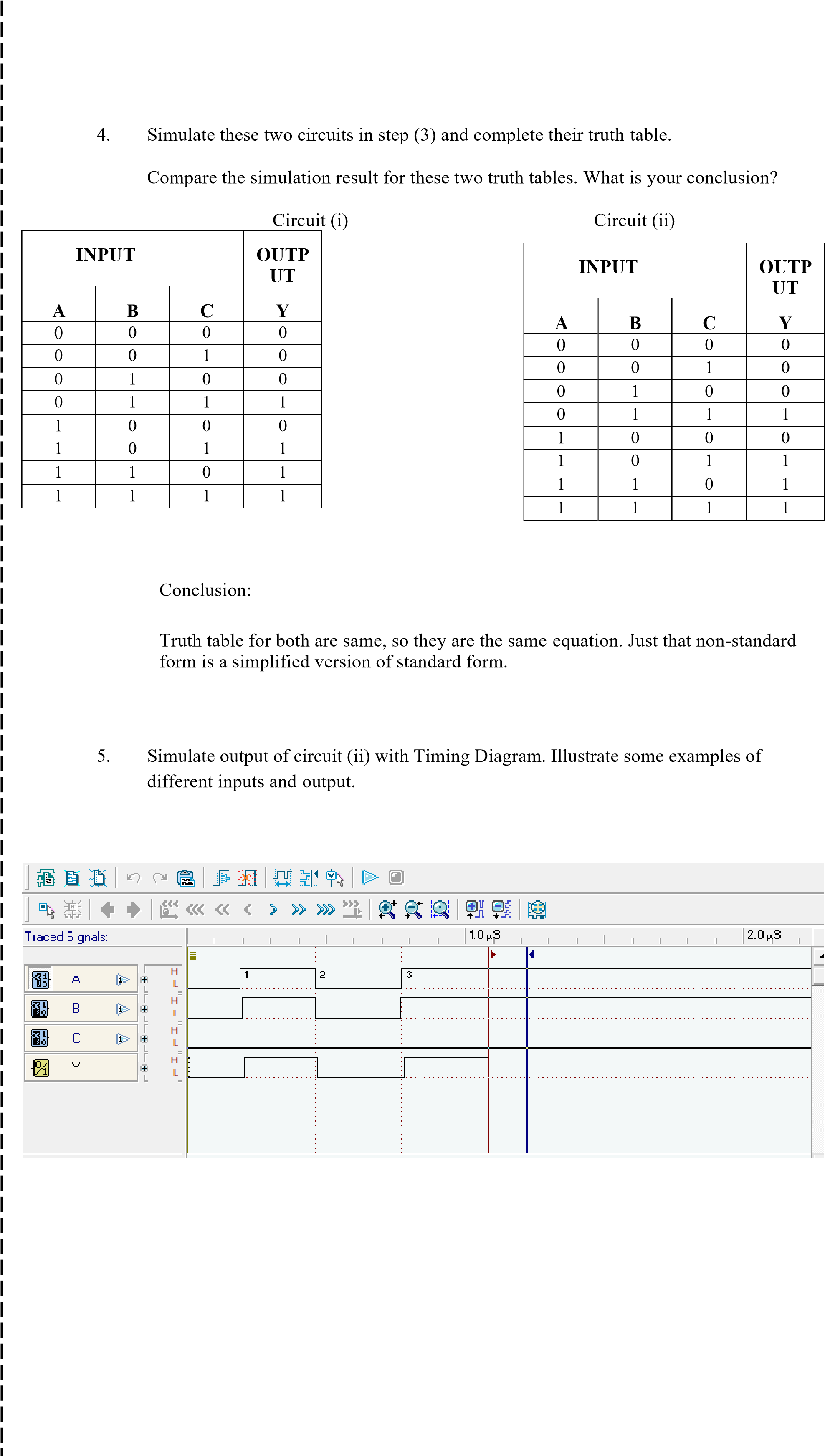


Circuit (i)

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



Circuit (ii)



# Part 2

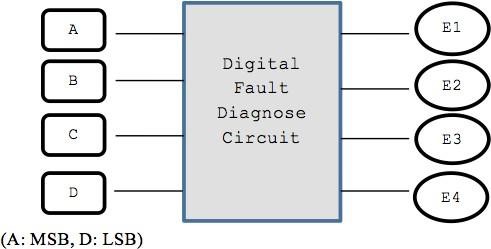
Combinational circuit design process and simulate with Deeds Simulator.

## Design Process

1. Determine Parameter Input / Output and their relations.
2. Construct Truth Table. iii) Using K-Map, get the SOP optimized form of all Boolean equation outputs.
3. Draw the circuit and use duality symbol; convert AND-OR circuit to NAND gates ONLY.
4. 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 and negative 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.

CD

00

01

11

10

AB

00

01

11

10

E2

)

i

(

E1

i

)

(

X

E2

E2

)

(

i

X

X

X

E3

(

i

)

E3

(1)

E4

i

)

(

0

X

E3

(

i

)

X

E1= A̅ B̅ C̅ D

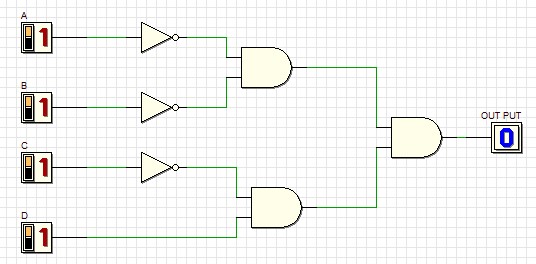
E2= AB̅ C̅ D̅ + A̅ B̅ CD̅

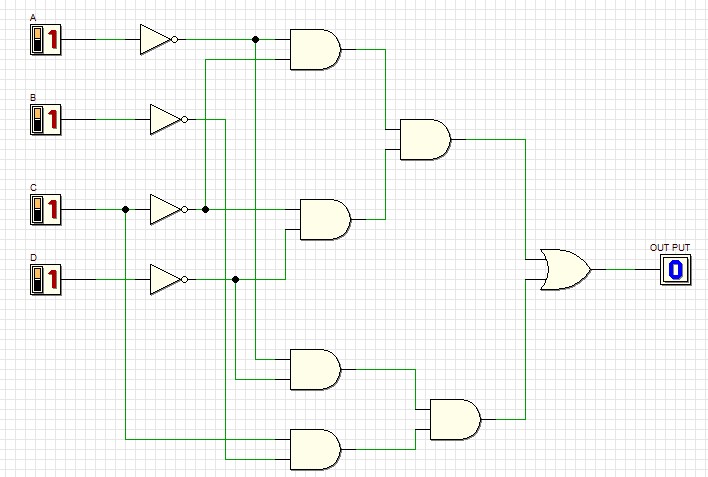
E3= ABD + AB̅ CD

E4= ABCD̅

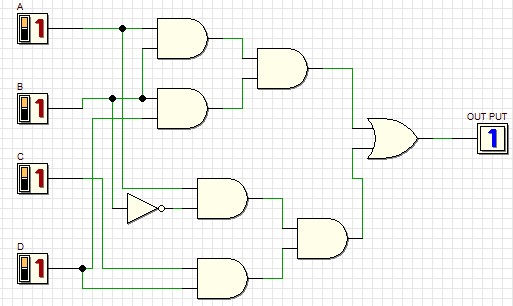
3.From the Boolean expression in the step (2), draw your final E1, E2, E3 and E4 circuits using 2 input basic gates (AND, OR, NOT). Use Deeds Simulator.

E1:

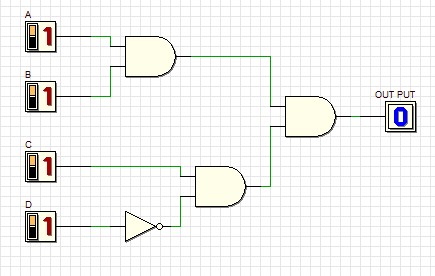


E2:

E3:



E4:



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

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

## Truth Table 2

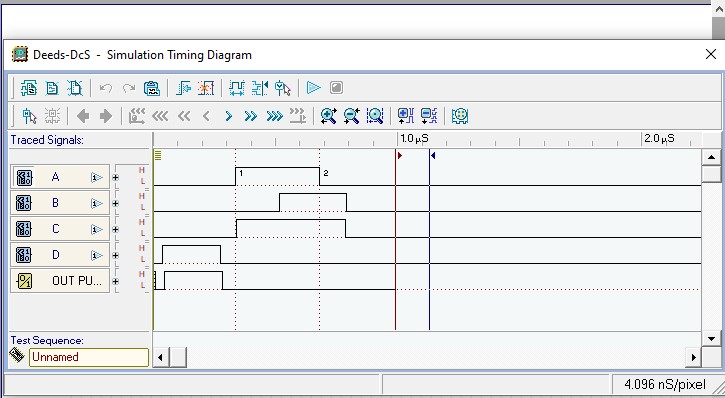
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **INPUTS** | |  |  | **OUTPUTS** | |  |
| **A** | **B** | **C** | **D** | **E1** | **E2** | **E3** | **E4** |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0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

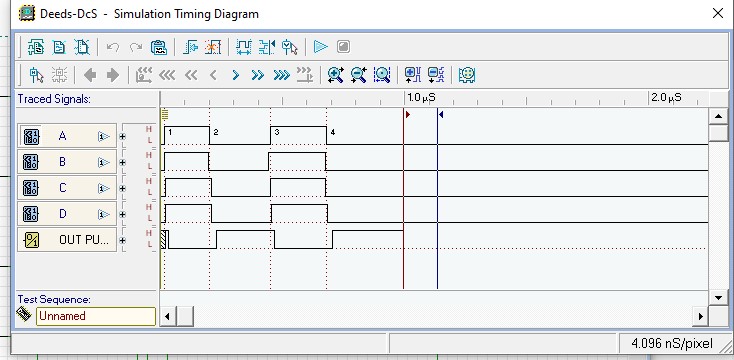
# Same result as the equations were derived from the previous table.

b) Timing Diagram:

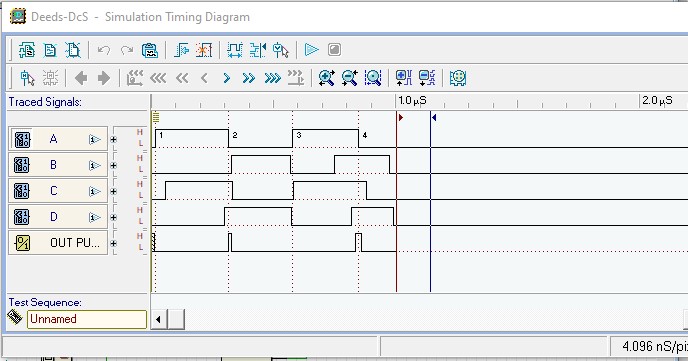
E1:



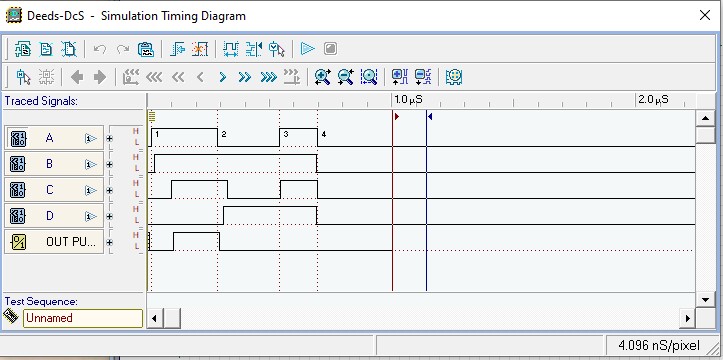
E2:



E3:



E4:



Explain some analysis values based on your timing diagram:

# In E1: only input D value is come out as an output; for E2: if all input is low

Then output will be high. And for E3: if all input is high then output will be high. And lastly For E4: if the input for A, B, C is high and D is low then the output will be

High.

5.

Using dual symbol concept, convert

your circuit in step (3) to NAND gates only. Use

Deeds

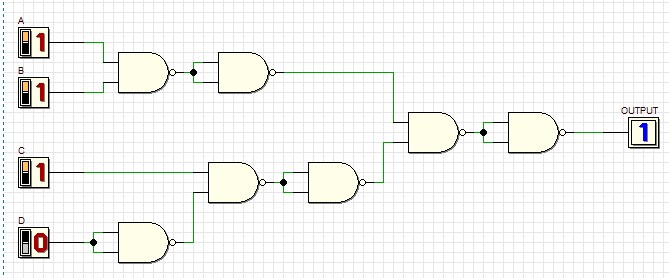
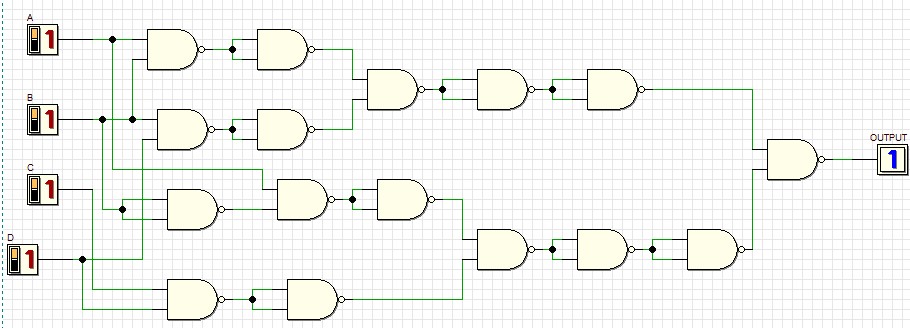
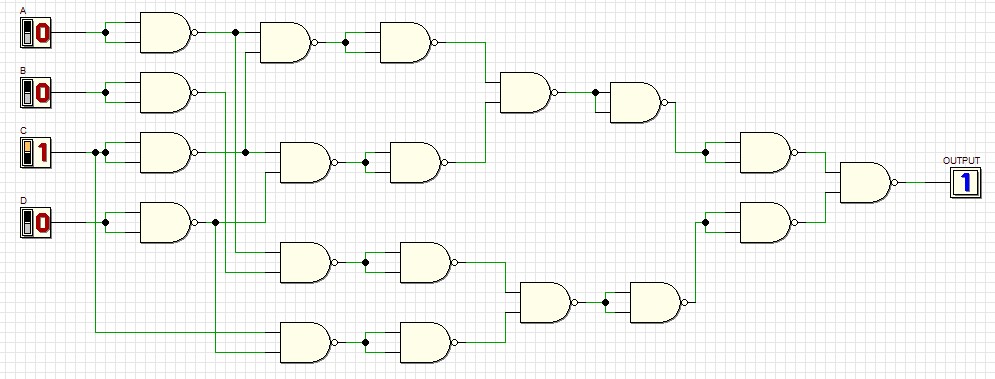
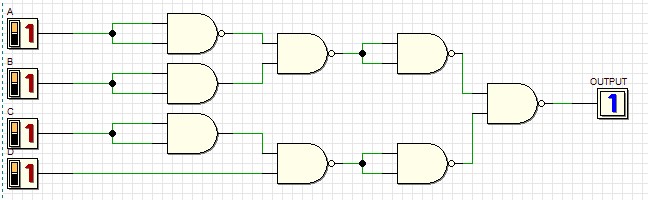
Simulator.

E1:

E2:

E3:

E4:



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 | 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 |

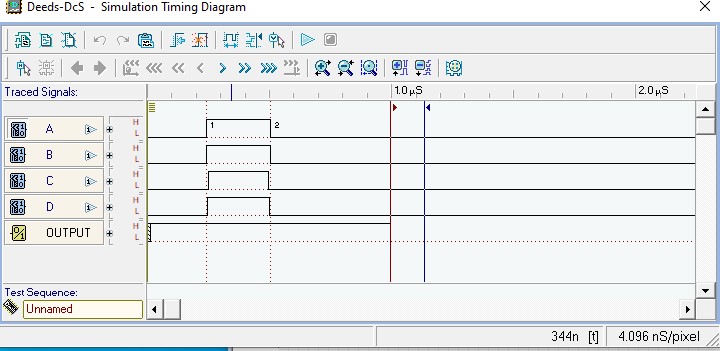
Compare the output results in Truth Table 3 with Truth Table 2. What is your conclusion?

Same because NAND gates are universal gates. It has been used to replace

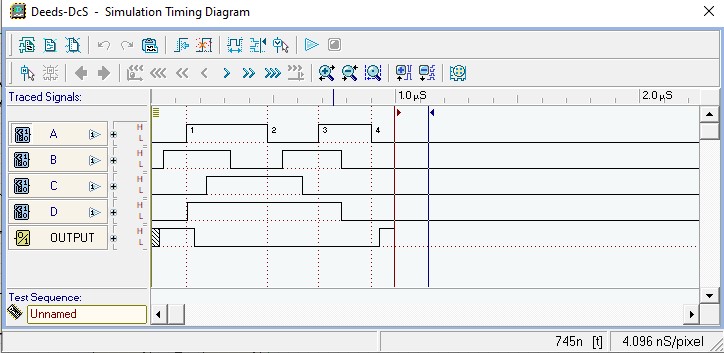
The basic gates here.

**b) Timing Diagram**

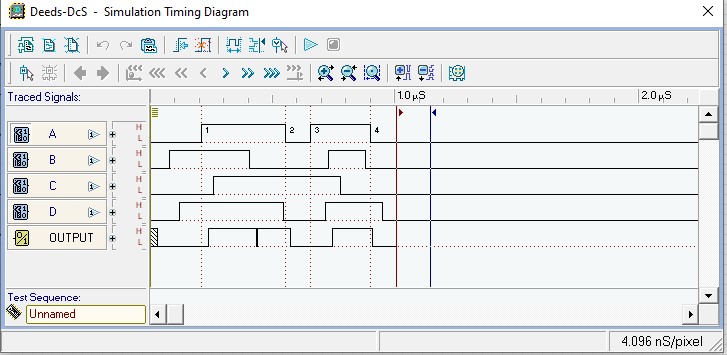
E1:



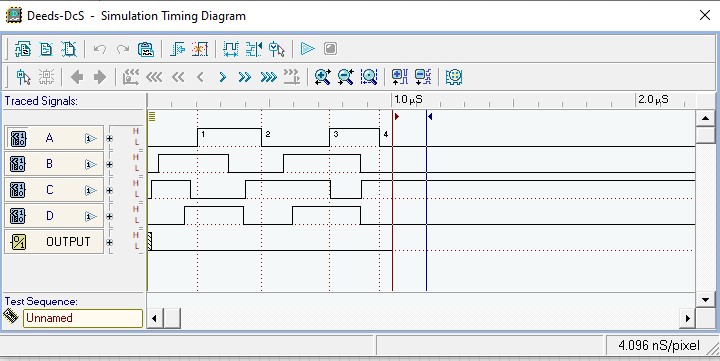
E2:



E3:



E4:



Explain some analysis values based on your timing diagram:

# The timing diagram shows that the input value for A, B, C, D for each circuit

E1, E2,E3 and E4 . And show the output as well.

Fully Partially



Completed Completed Checked by: