Department of Computer Science
Faculty of Computing
UNIVERSITI TEKNOLOGI MALAYSIA

SUBJECT	:	SCSR1013 DIGITAL LOGIC
SESSION/SEM	:	07 / 1
LAB 2	**:**	**COMBINATIONAL LOGIC CIRCUIT DESIGN SIMULATION**
NAME 1	:	Eunice Lim Xian Ni
NAME 2	:	
DATE	:	11/1/2021

REMARKS :

MARKS:

Lab # 2

Combinational Digital Circuit Design Simulation Using Deeds Simulator

A. Objective

i) To expose student with producing digital logic circuit, generating truth table and Timing Diagram with Deeds Simulator.

ii) To expose student with a complete cycle process of a combinatorial circuit design and simulate with Deeds Simulator.

B. Material

Install Deeds Software for Windows.

C. Introduction

Deeds Simulator

The Digital Circuit Simulator ***d-DcS*** appears to the user as a graphical schematic editor, with a library of simplified logic components, specialized toward pedagogical needs and not describing specific commercial products. As described before, the schematic editor allows building a simple digital networks composed of gates, flip-flops, pre-defined combinational and sequential circuits and custom-defined components (defined as Finite state machine).

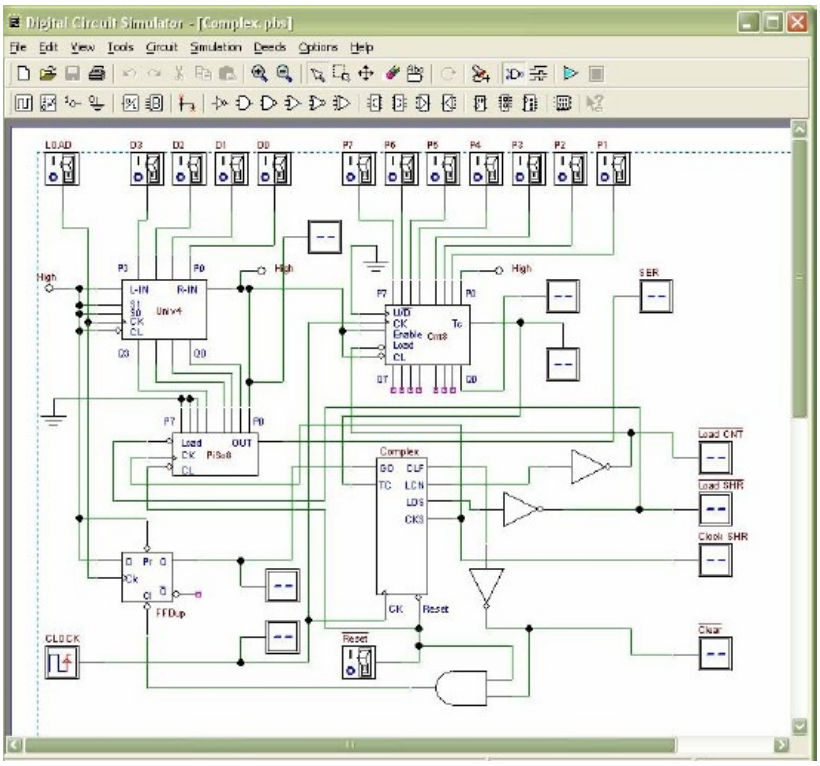


Figure 1 Circuit Editor of Digital Circuit Simulator (d-DcS)

Simulation can be interactive or in timing-mode. In **interactive mode**, the student can *"animate"* the digital system in the editor, controlling its inputs and observing the results. This is the simplest mode to examine a digital network, and this way of operation can be useful for the beginners. In **timing mode**, the behavior of the circuit can be analyzed by a timing diagram window, in which the user can define graphically an input signal sequence and observe the simulation results.

Digital Circuit Simulator (d-DcS): A Simple Example

In the following screen shots (Figure 2a, 2b, and 2c), student can see the circuit during the drawing and then simulate it by animation with following simple steps:

a) student picks-up components from the *Component Tool Bar*.
b) connects them using *Wires*.
c) student activates the animation.

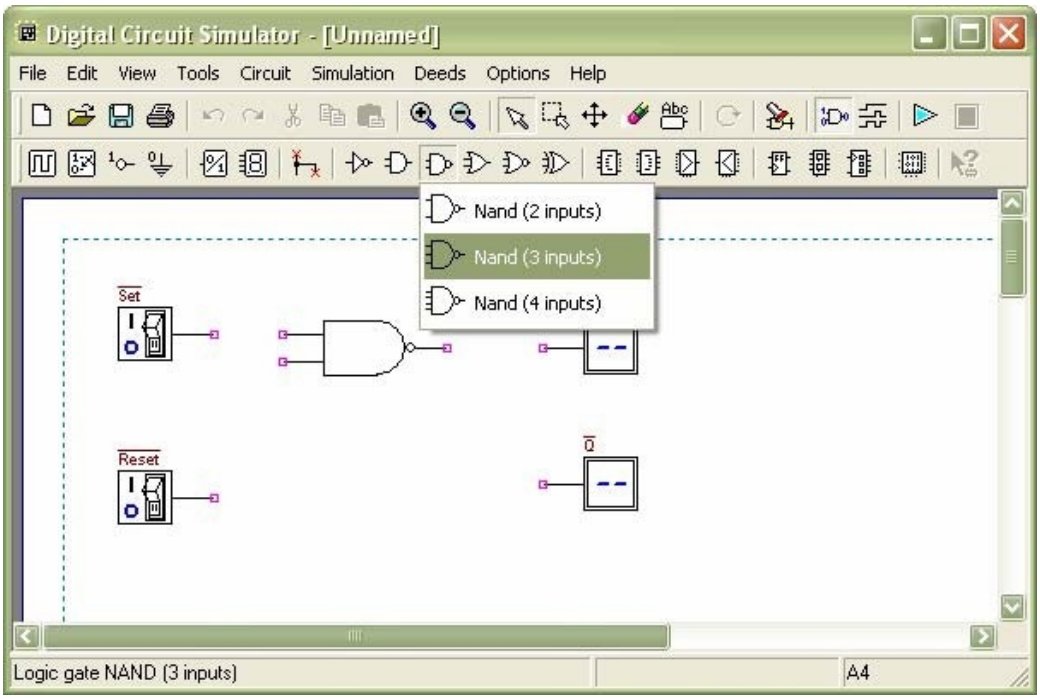


Figure 2a Drawing Phase of the Digital Circuit Editor: Insertion of Components

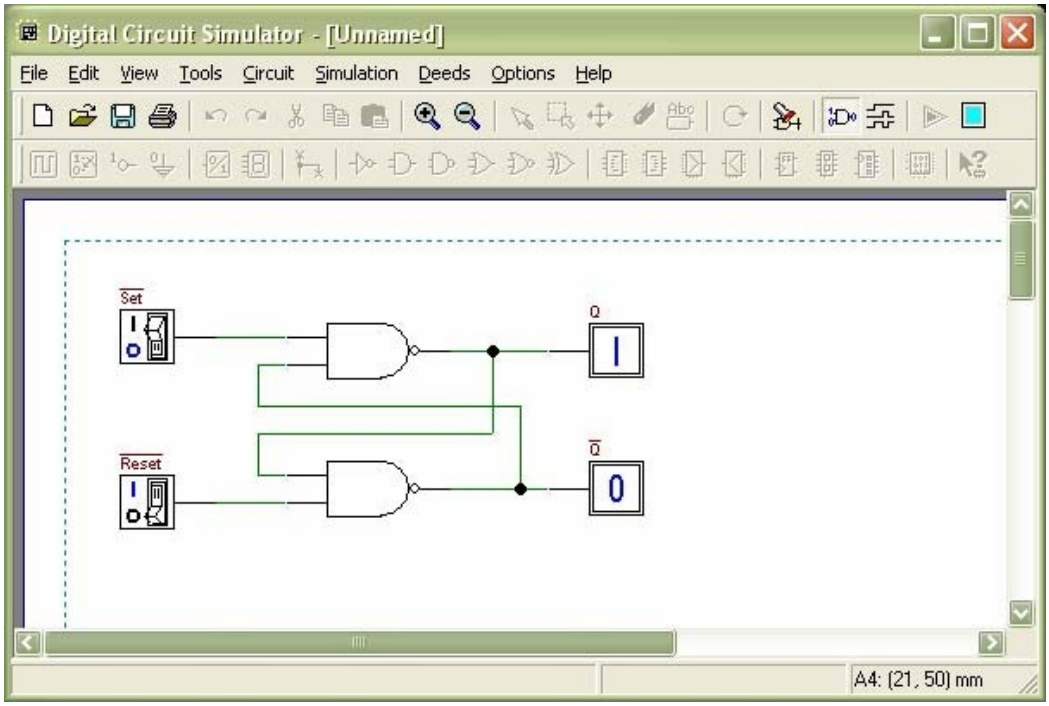


Figure 2b Next Phase of the Work: Connection of Components using *Wires*

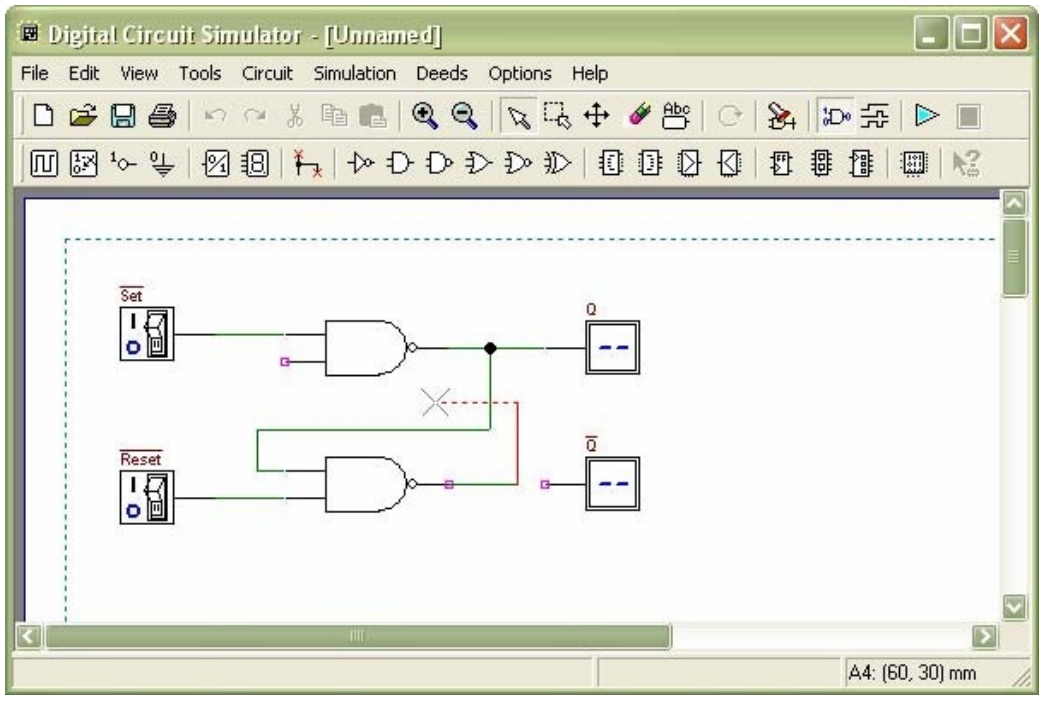


Figure 2c Animation: User Switches Inputs and the Circuit Shows Changes on Outputs

To exit the animation, it is necessary to click on the square *Stop* button. Instead, if the timing simulation is to be performed, student should click on the *Timing Simulation* button. This will show the Timing Diagram simulation window (Figure 3).

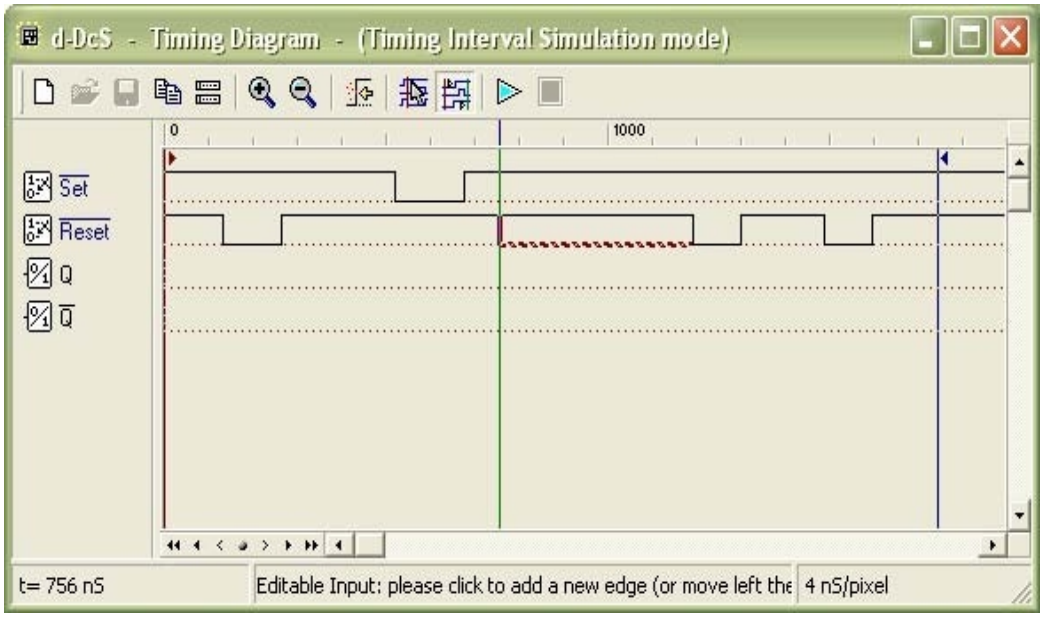


Figure 3 Timing Diagram Simulation Window

In this window, first student should define the timing of the input signals, drawing them on the diagram with the mouse. A vertical line cursor permits to define the 'end time' of the simulation. When student clicks on the triangular *play* button on the toolbar, the simulation is executed, and its results are displayed in the same window (Figure 4).

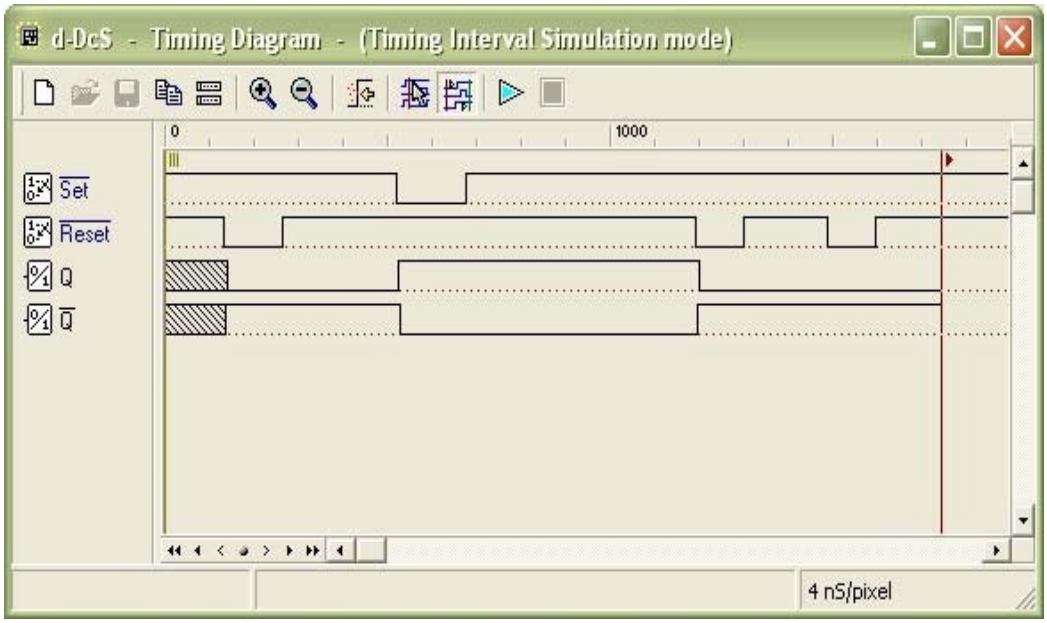


Figure 4 Timing Simulation Results, Displayed in Timing Diagram Window

Student can verify the correct behavior of the digital circuit, comparing simulation results with reasoning and theory concepts.

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.

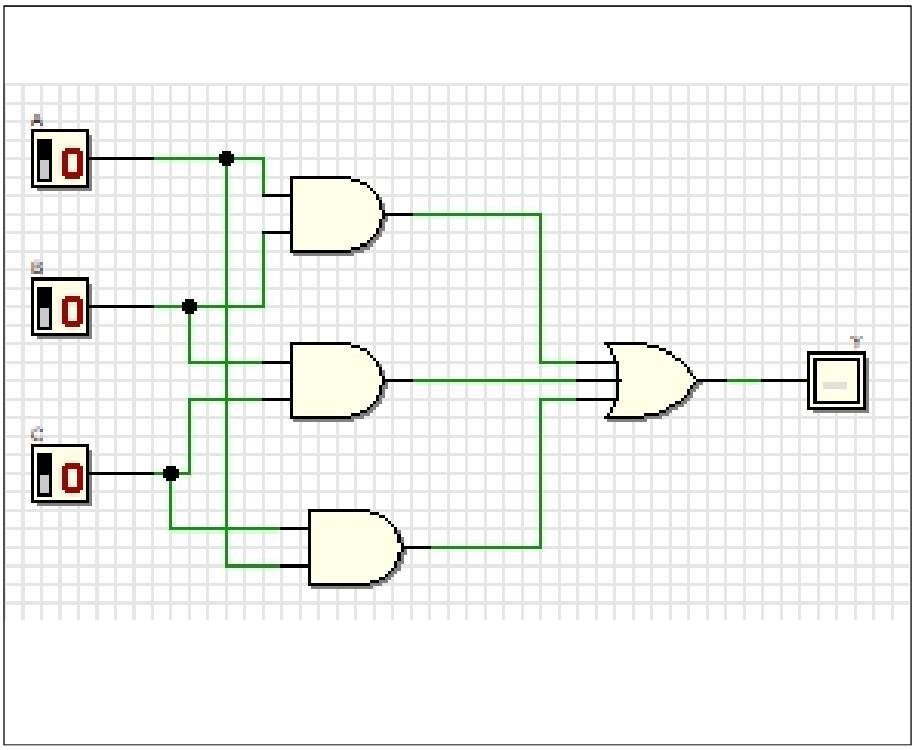
$$\begin{aligned} Y &= AB + BC + AC \\ &= AB\,(C+C') + BC\,(A+A') + AC\,(B+B') \\ &= ABC + ABC' + ABC + ABC + A'BC + ABC + AB'C \\ &= ABC + ABC' + A'BC + AB'C \end{aligned}$$

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

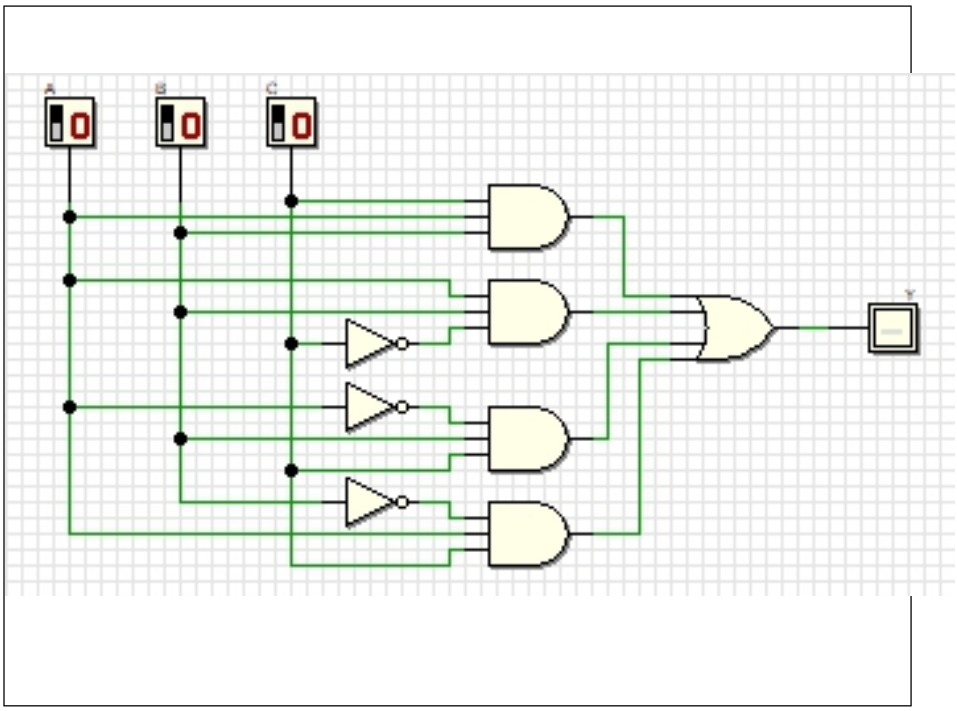
3. Using Deeds Simulator, draw the following circuits:

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

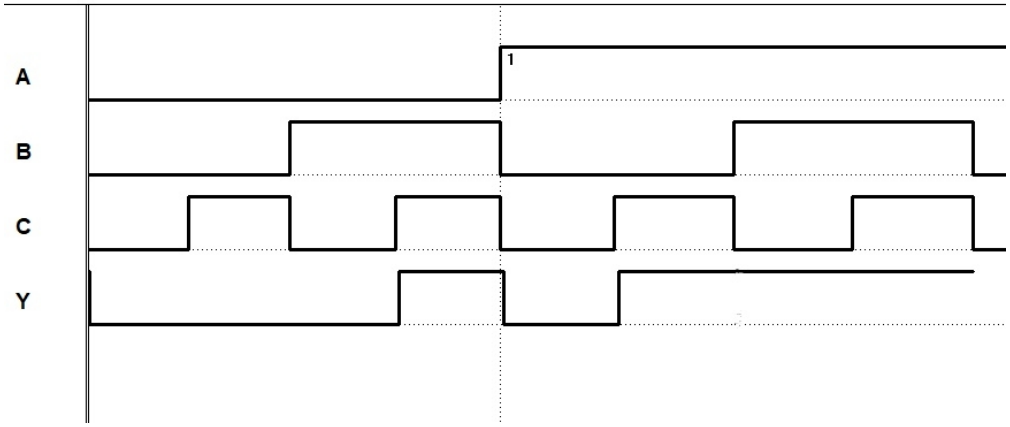
Circuit (ii)

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:

Circuit (i) is a simplified circuit of circuit (ii). Standardization does allow us to easily make evaluation,simplification and implementation of Boolean expression but it makes the circuit more complicated and costly because of using more gates and wiring. However, the outputs will not be affected. Hence, circuit (i) and circuit (ii) both produce the same outputs in their truth table.

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



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.

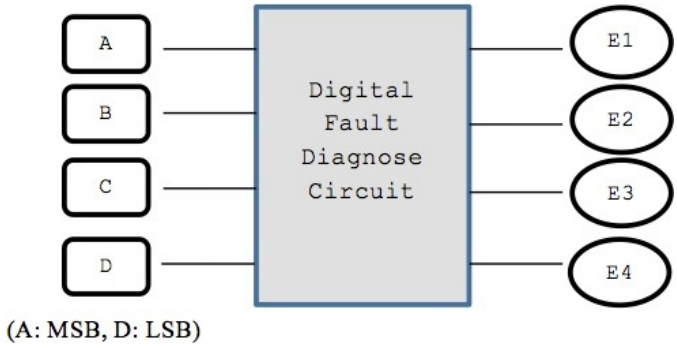


Figure 5

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 and negative EVEN is negative numbers that are even).

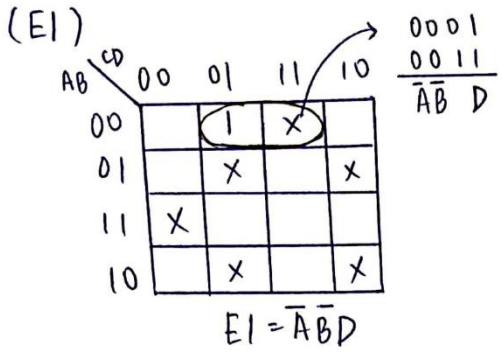
Experimental Steps

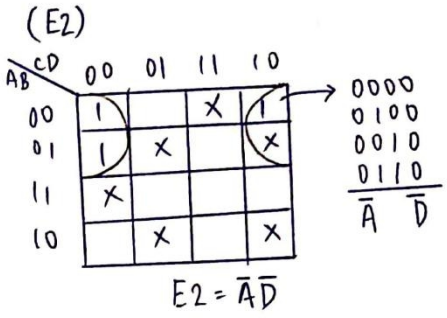
1. Complete Truth Table 1 for Digital Fault Diagnose Circuit. Use variables A, B, C and D as inputs; E1, E2, E3 and E4 as outputs.

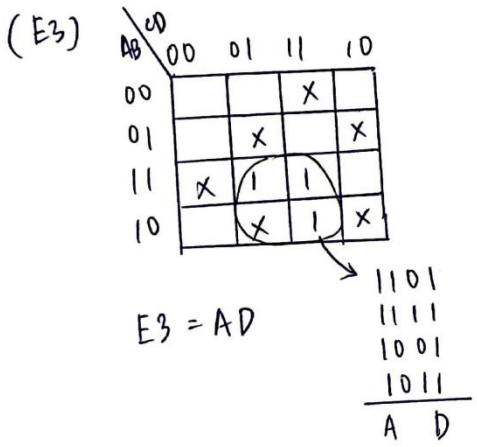
Truth Table 1

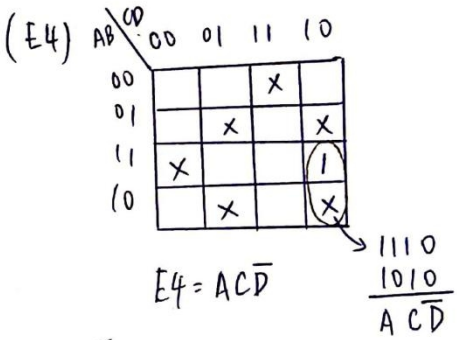
INPUTS				OUTPUTS			
A	**B**	**C**	**D**	**E1**	**E2**	**E3**	**E4**
0	0	0	0	0	1	0	0
0	0	0	1	1	0	0	0
0	0	1	0	0	1	0	0
0	0	1	1	X	X	X	X
0	1	0	0	0	1	0	0
0	1	0	1	X	X	X	X
0	1	1	0	X	X	X	X
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	1	X	X	X	X
1	0	1	0	X	X	X	X
1	0	1	1	0	0	1	0
1	1	0	0	X	X	X	X
1	1	0	1	0	0	1	0
1	1	1	0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

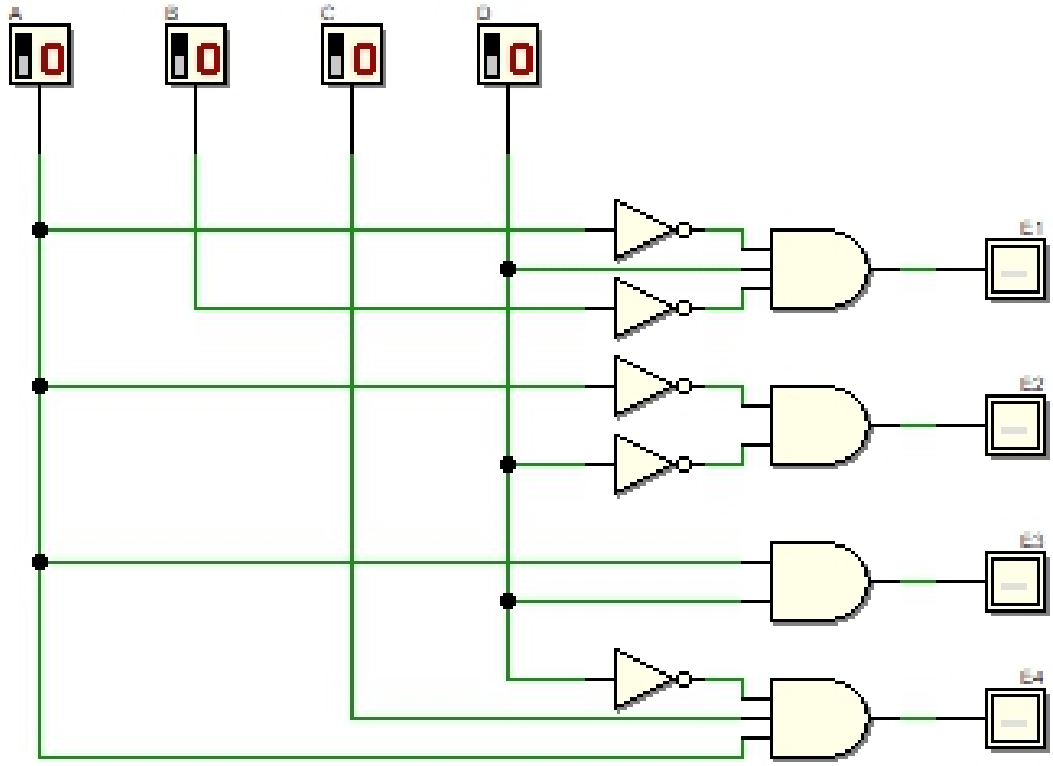








3. From the Boolean expression in the step (2), draw your final E1, E2, E3 and E4 circuits using 2 input basic gates (AND, OR, NOT). Use Deeds Simulator.



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

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

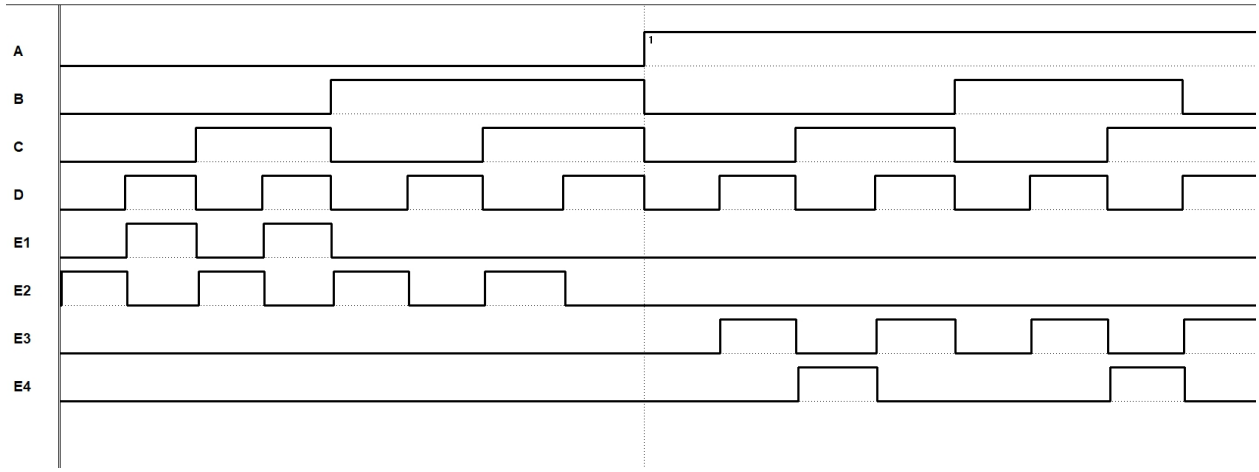
Truth Table 2

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?

Truth Table 1 and Truth Table 2 both produce the same outputs. Truth Table 2 is the simplified circuit of Truth Table 1 using K-MAP. Truth Table 1 and Truth Table 2 contain 'don't care' conditions. Although Truth Table 1 shows the 'don't care' conditions by labelling 'X' and Truth Table 2 does not show the label 'X' , but the outputs for 'don't care' conditions in Truth Table 2 will never be considered as they are invalid outputs which means it will never occur. Thus, Truth Table 1 and Truth Table 2 have the same outputs.

b) Timing Diagram

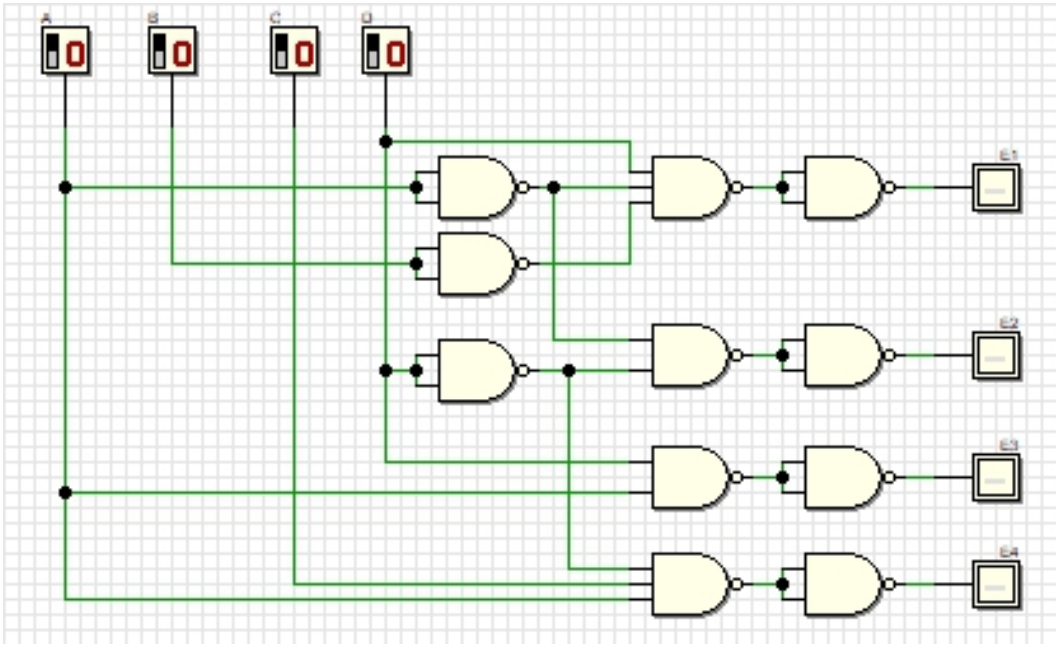


Explain some analysis values based on your timing diagram:

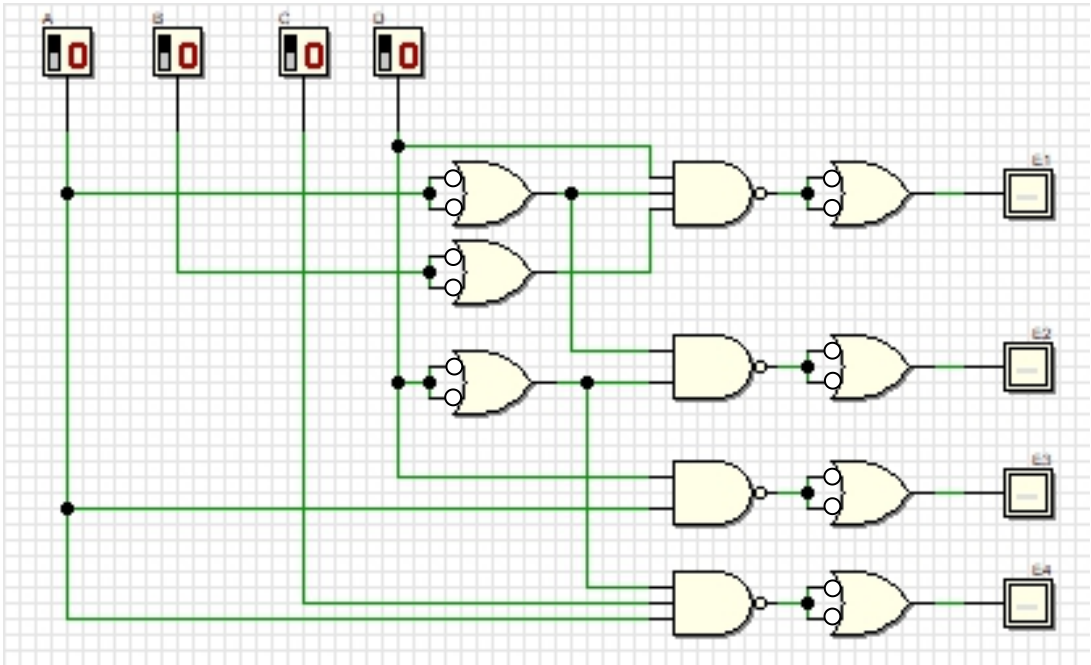
The timing diagram is based on Truth Table 2. When input D is HIGH, only E1 produces HIGH output while E2, E3 and E4 produces LOW outputs. When inputs A, B and C are HIGH, only E4 produces HIGH outputs while E1, E2 and E3 produces LOW outputs. When the inputs have equal number of bits which are 0011, 0101, 0110, 1001, 1010 and 1100, their outputs are not considered as they are don't care conditions.

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

Convert to NAND gates first:



Then convert to its dual symbol:



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

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

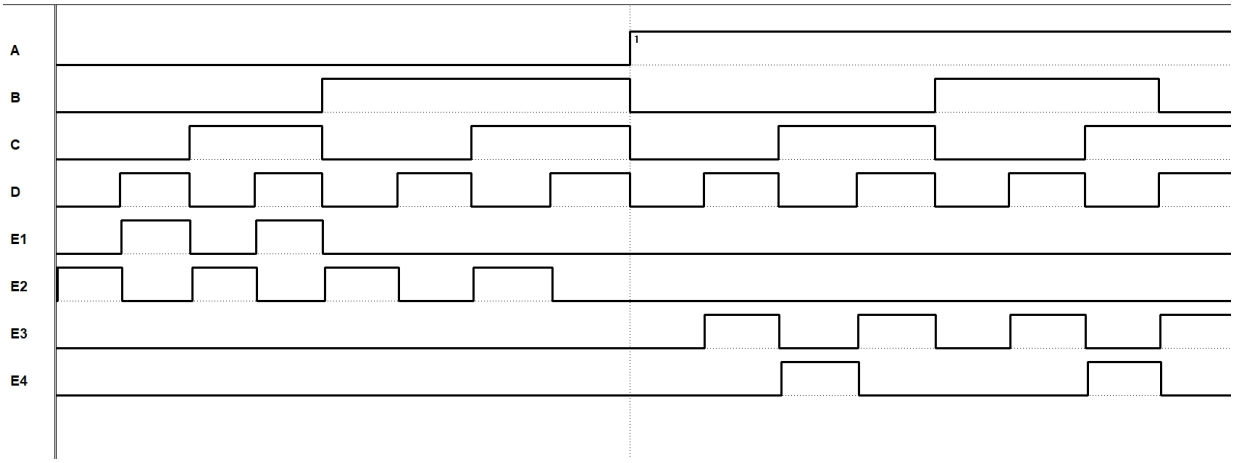
Truth Table 3

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?

Truth Table 2 and Truth Table 3 both produce the same outputs. Circuit in step (5) is a NAND gates only circuit with its dual symbol which is Negative-OR. NAND gate is a universal gate because it can be converted and carries out the function of basic gates. Dual symbol is used to simplify the schematic diagram so that they are easier to be read. Hence, the outputs of Truth Table 3 will not be affected and have the same outputs as Truth table 2.

b) Timing Diagram



Explain some analysis values based on your timing diagram:

The timing diagram is based on Truth Table 3. When input D is HIGH, only E1 produces HIGH output while the rest produce LOW outputs. When inputs A, B and C are HIGH, only E4 produces HIGH outputs while E1, E2 and E3 produce LOW outputs. When inputs have equal number of bits which are 0011, 0101, 0110, 1001, 1010 and 1100, their outputs are not considered as they are don't care conditions. We observe that the timing diagram based on Truth Table 3 produces the same outputs as timing diagram based on Truth Table 2. This proves that the outputs are not affected although the circuit in step (5) is converted to using only NAND gates with its dual symbol.

Fully Completed ☐ Partially Completed ☐ Checked by: _____