



Digital Logic Gates & Boolean Algebra

Experiment – 1 & 2

DIGITAL LOGIC DESIGN

Exp # 1 - DIGITAL LOGIC GATES

OBJECTIVE:

- To study the basic logic gates: AND, OR, INVERT, NAND, and NOR.
- To study the representation of these functions by truth tables, logic diagrams and Boolean algebra.
- To observe the pulse response of logic gates.
- To measure the propagation delay of logic gates.

APPARATUS:

- IC Type 7400 Quadruple 2-input NAND gates
- IC Type 7402 Quadruple 2-input NOR gates
- IC Type 7404 Hex Inverters
- IC Type 7408 Quadruple 2-input AND gates
- IC Type 7432 Quadruple 2-input OR gates
- IC Type 7486 Quadruple 2-input XOR gate
- IC Type 7493 4-bit ripple counter
- Digi-Designer Logic Board
- Dual-trace oscilloscope

THEORY:

AND A multi-input circuit in which the output is 1 only if all inputs are 1. The symbolic representation of the AND gate is shown in Fig. 1a.

OR A multi-input circuit in which the output is 1 when any input is 1. The symbolic representation of the OR gate is shown in Fig. 1b.

INVERT The output is 0 when the input is 1, and the output is 1 when the input is 0. The symbolic representation of an inverter is shown in Fig. 1c.

NAND AND followed by INVERT. The symbolic representation of the NAND gate is shown in Fig 1d.

NOR

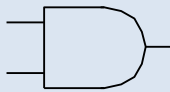
OR followed by INVERT as shown in Fig 1e.

EX-OR

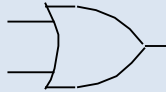
The output of the Exclusive –OR gate, is 0 when it's two inputs are the same and it's output is 1 when its two inputs are different.

Truth Table

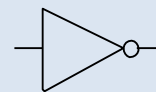
Representation of the output logic levels of a logic circuit for every possible combination of levels of the inputs. This is best done by means of a systematic tabulation.



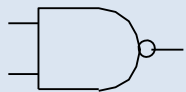
a. Two input AND gate



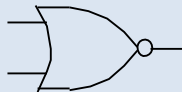
b. Two input OR gate



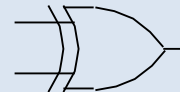
c. Inverter



d. Two input NAND gate



e. Two input NOR gate



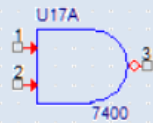
f. Two input XOR gate

Fig.1 Symbols for digital logic gates

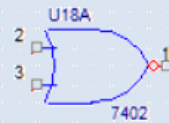
Part 1: Logic Functions

Exercises:

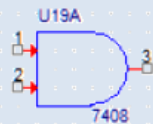
1. Use one gate for each IC 7400 (NAND), 7402 (NOR), 7408 (AND), 7432 (OR), 7486 (XOR). Each has input pins, 1 and 2, and output pin 3.



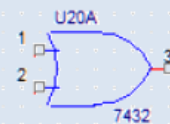
7400 (NAND gate)



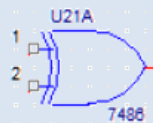
7402 (NOR gate)



7408 (AND gate)

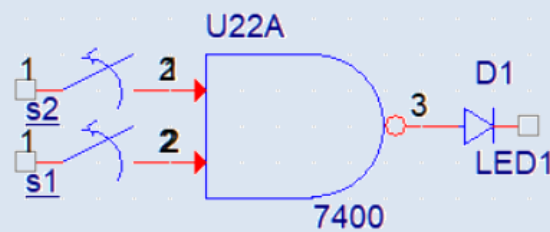


7432 (OR gate)



7486 (XOR gate)

2. Connect pin 1 to switch S1-1, pin 2 to switch S1-2, and pin 3 to LED-1 for every gate as shown in Fig 2 as an example for the NAND gate.



■ 7400 (NAND gate) connected to switches and LED ■

3. Simulation and Truth Table of AND, OR, NAND, NOR and XOR gates:

❖ AND gate:

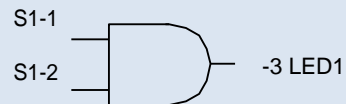
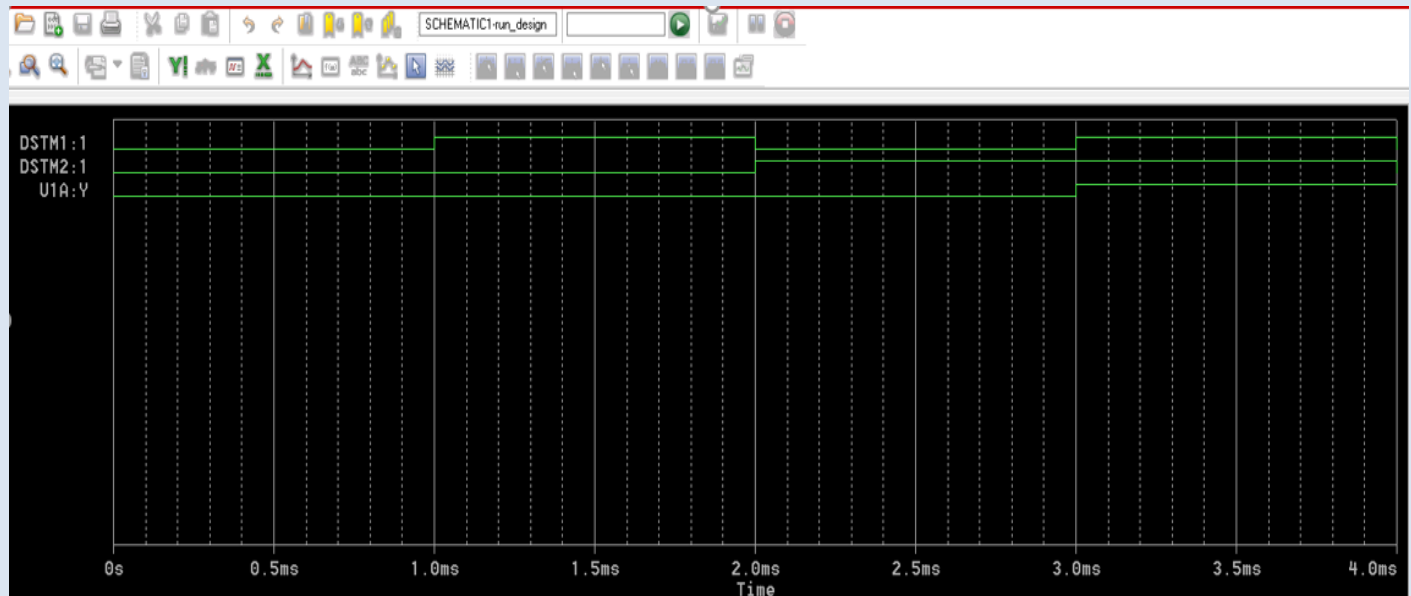
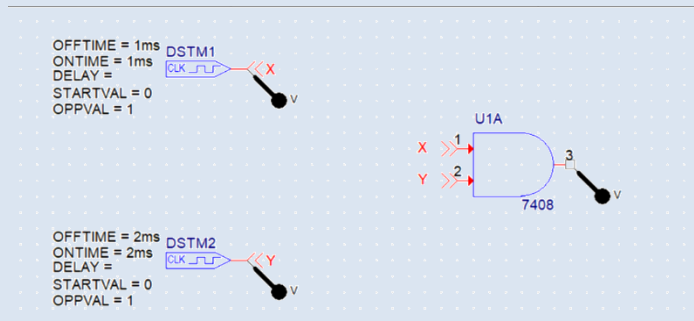


Fig.2 Two input AND gate



Observation:

Truth table of AND gate

Pin1(A)	Pin2 (B)	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

❖ OR gate:

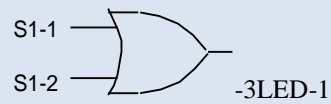
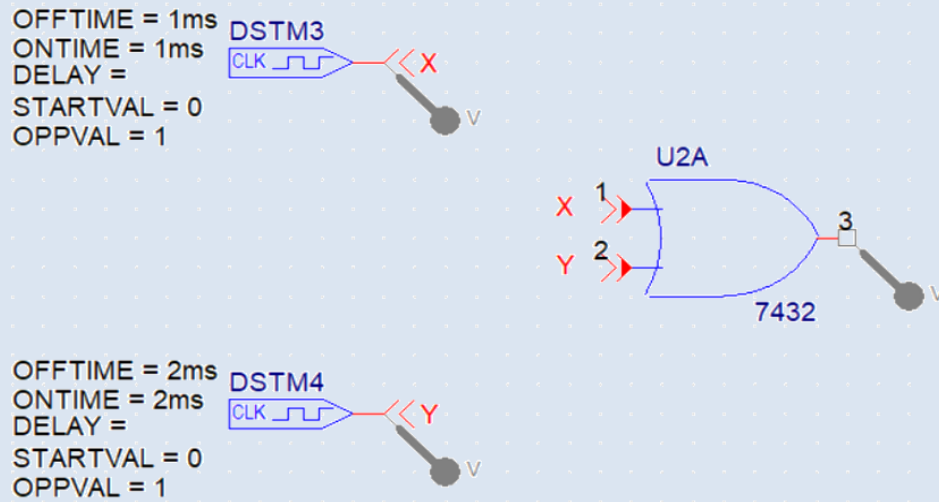


Fig.2 Two input OR gate

**Observation:**

Truth table of OR gate

Pin1(A)	Pin2 (B)	A OR B
0	0	0
0	1	1
1	0	1
1	1	1

❖ NAND gate:

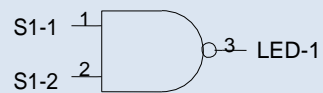
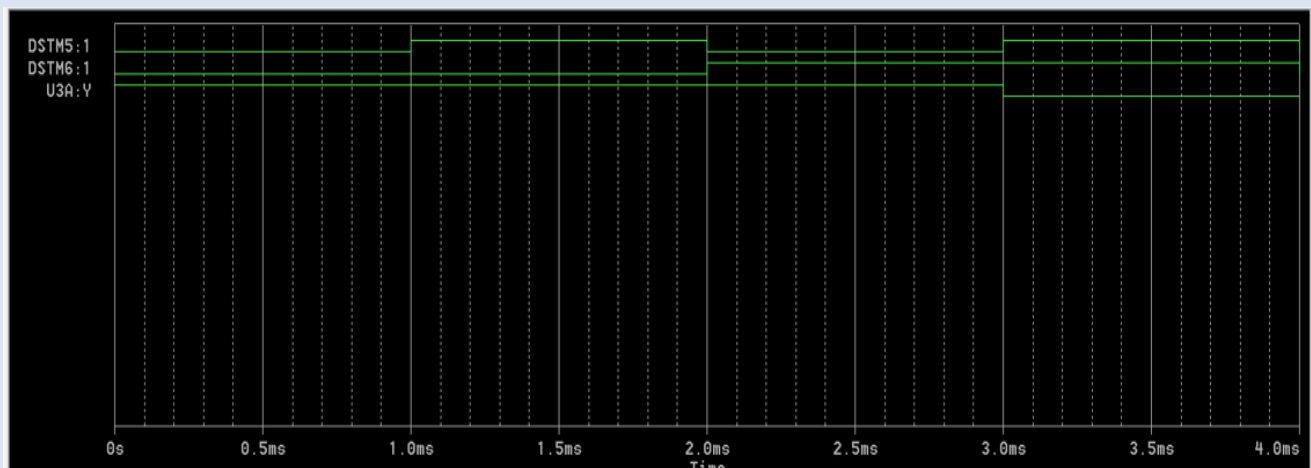
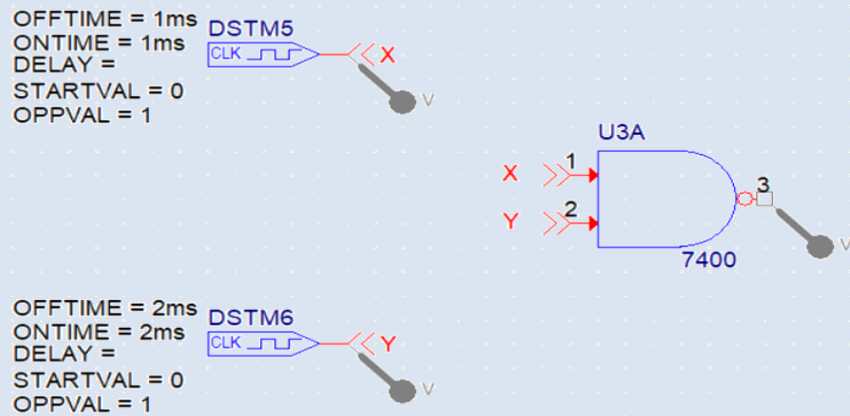


Fig.2 Two input NAND gate

**Observation:**

Truth table of NAND gate

Pin1(A)	Pin2 (B)	A NAND B
0	0	1
0	1	1
1	0	1
1	1	0

❖ NOR gate:

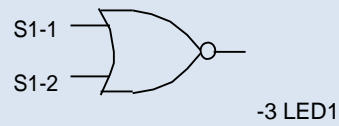
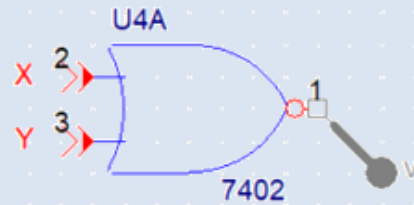
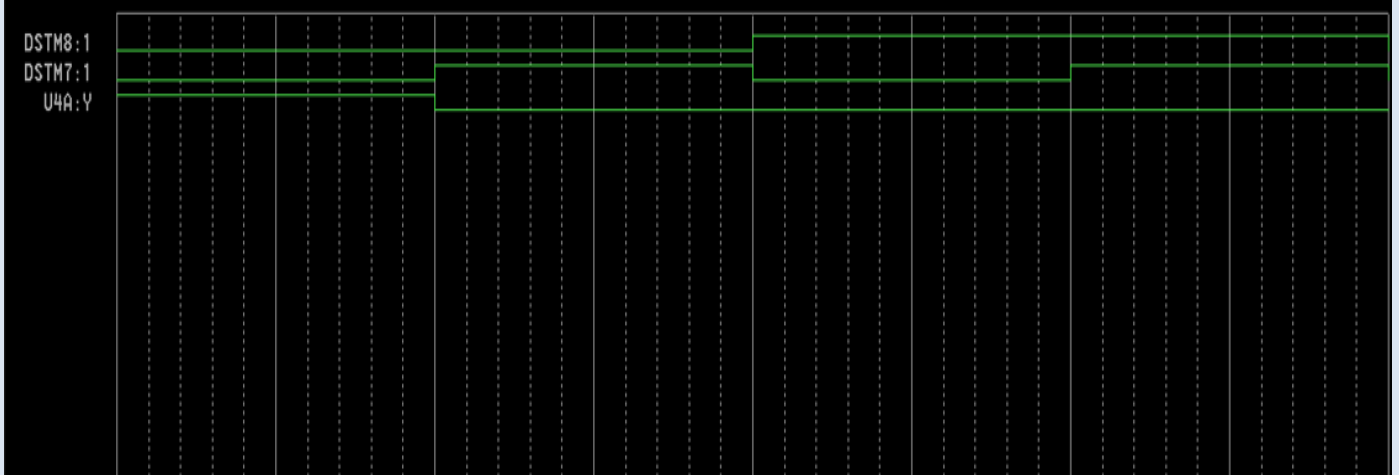


Fig.2 Two input NOR gate

OFFTIME = 1ms
 ONTIME = 1ms
 DELAY =
 STARTVAL = 0
 OPPVAL = 1



OFFTIME = 2ms
 ONTIME = 2ms
 DELAY =
 STARTVAL = 0
 OPPVAL = 1

**Observation:**

Truth table of NOR gate

Pin1 (A)	Pin2 (B)	A NOR B
0	0	1
0	1	0
1	0	0
1	1	0

❖ XOR gate:

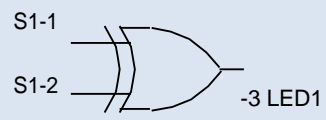
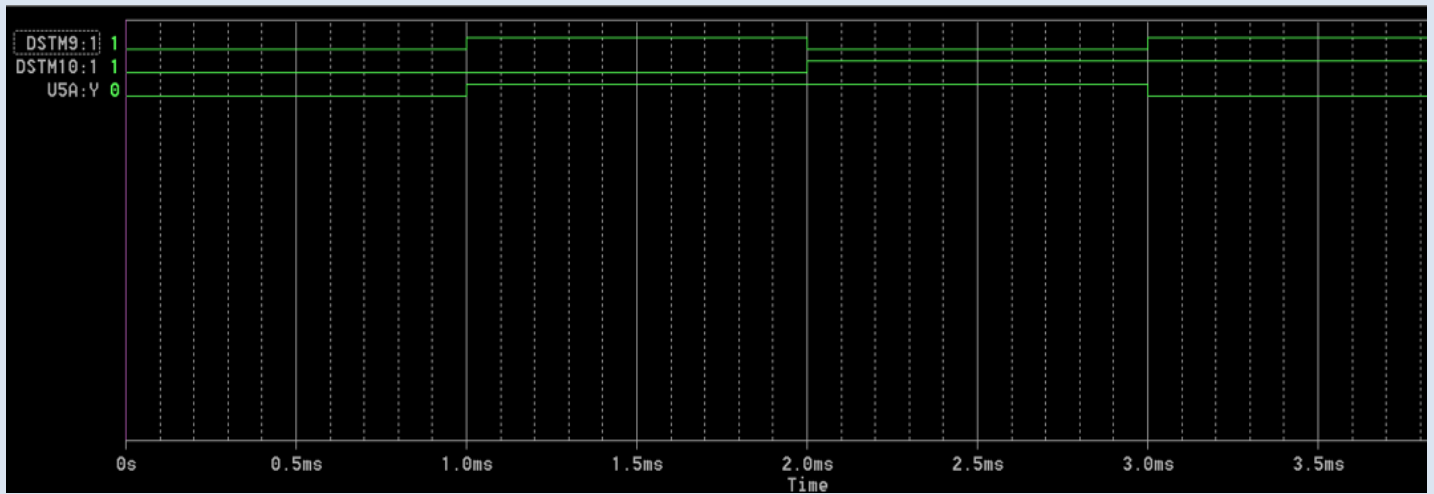
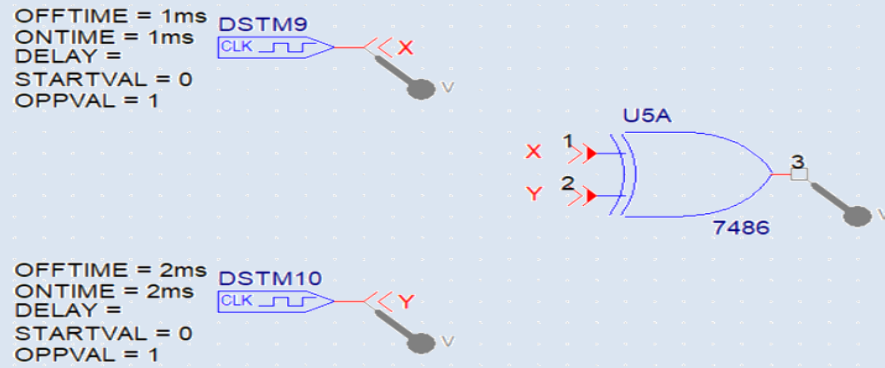


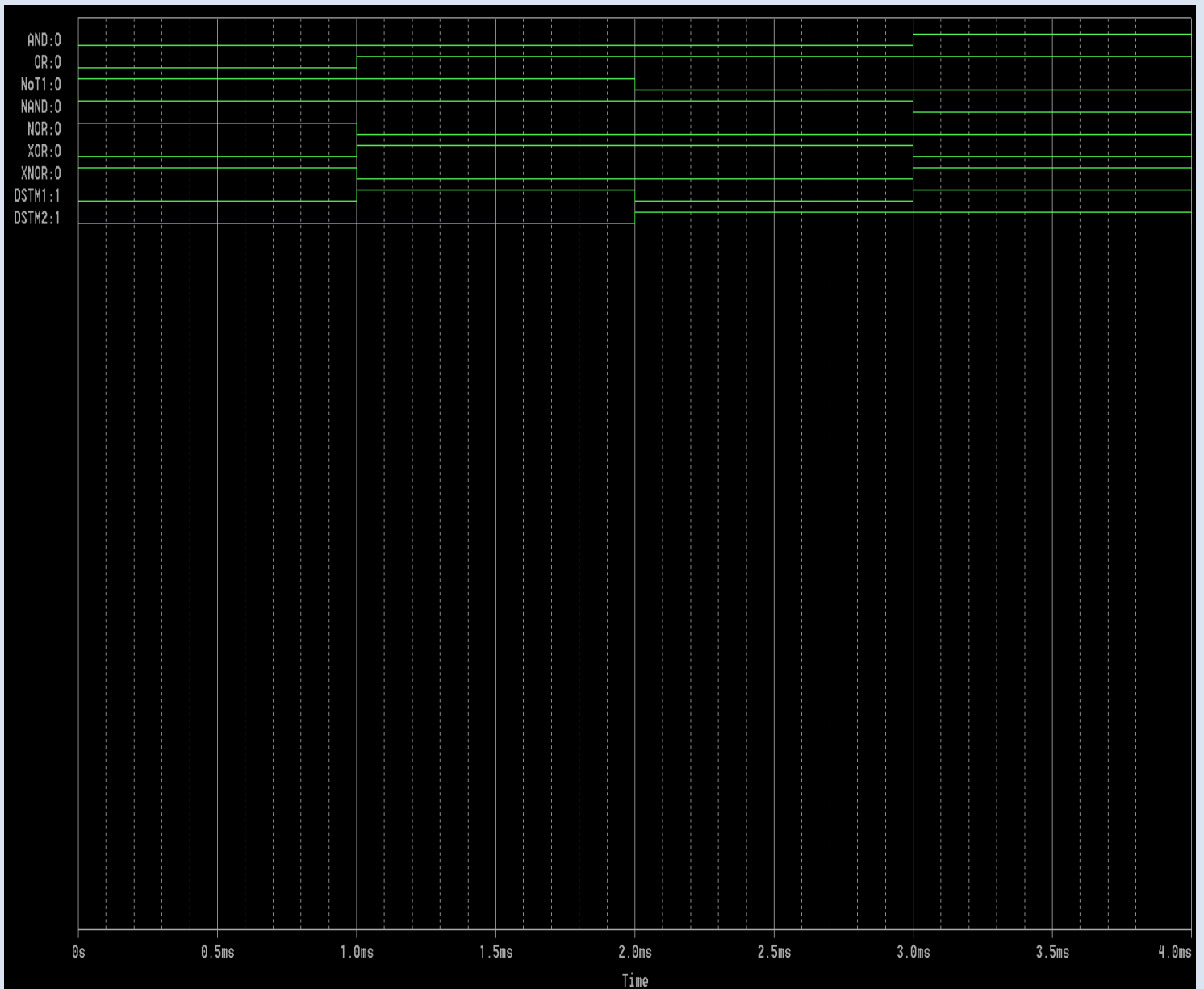
Fig.2 Two input XOR gate

**Observation:**

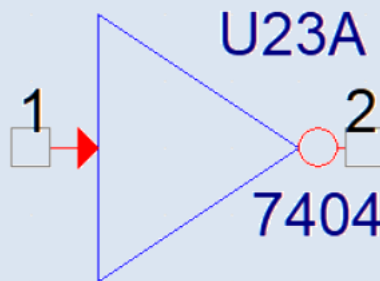
Truth table of XOR gate

Pin1 (A)	Pin2 (B)	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

SIMULATION OF ALL GATES:



4. Use an inverter gate from IC 7404 whose input pin is pin 1 and whose output pin is pin 2.



IC7404 (NOT gate)

Observation:

Truth Table of NOT gate

Pin 1	Pin 2
0	1
1	0

Part-2: Response of Logic Gates:

Connect the circuits of figures 4 and 5 and write the corresponding truth tables 3 and 4, respectively.

A.

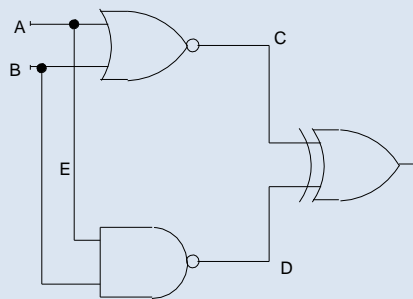
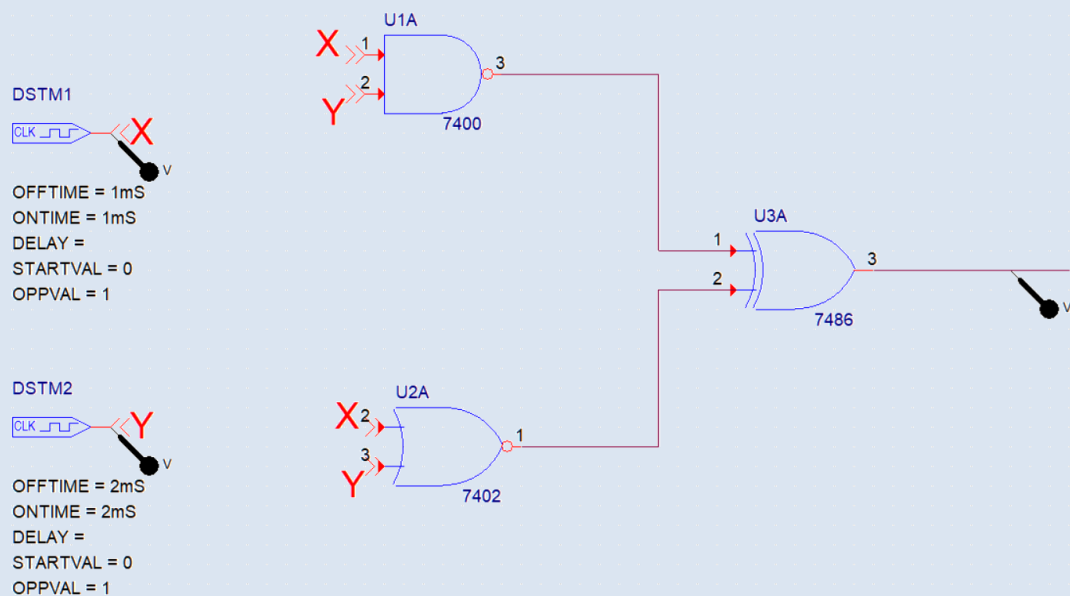
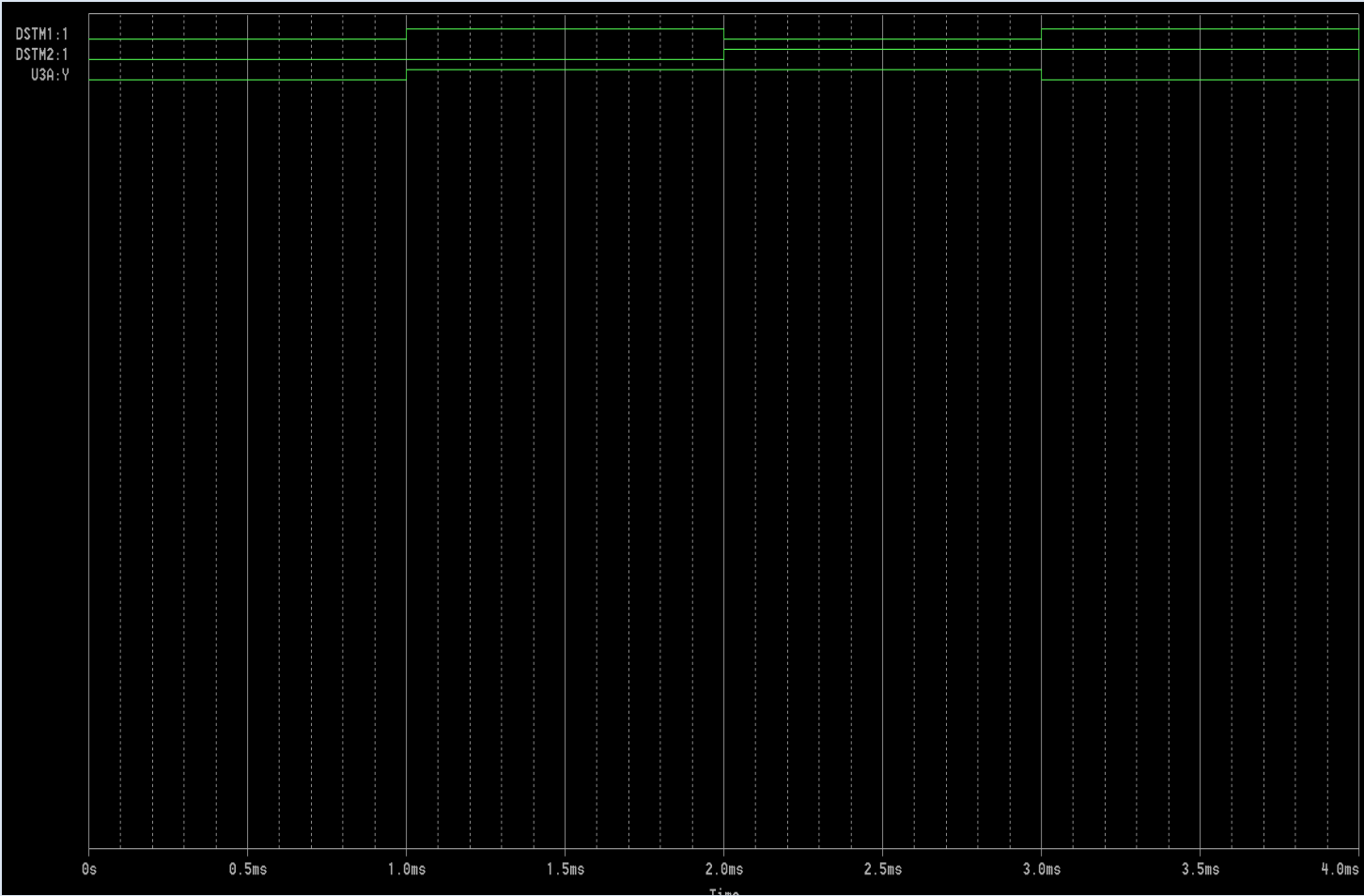


Fig. 4

Fig1:





A	B	C	D	E
0	0	1	1	0
0	1	0	1	1
1	0	0	1	1
1	1	0	0	0

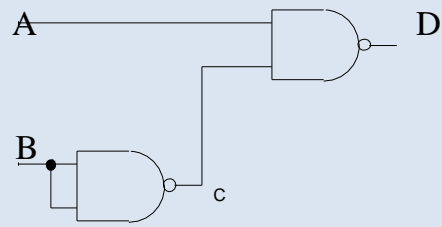
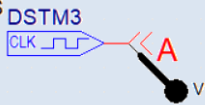
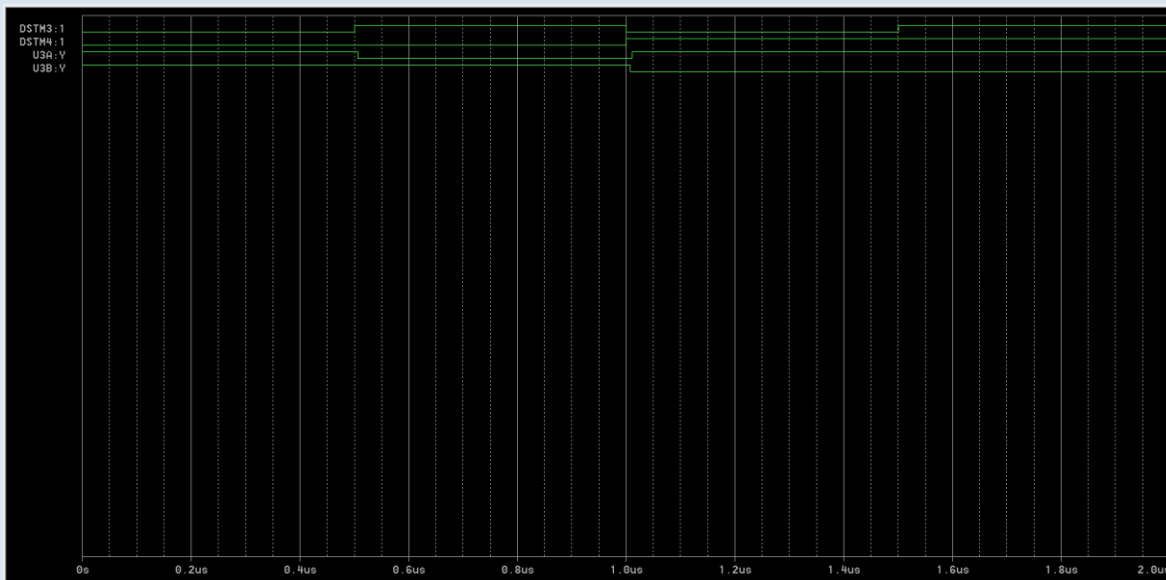
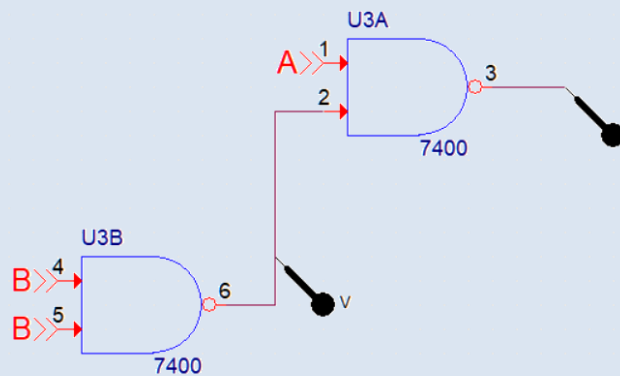
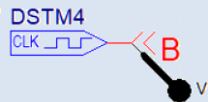
B.

Fig: 5

OFFTIME = .5uS
 ONTIME = .5uS
 DELAY =
 STARTVAL = 0
 OPPVAL = 1



OFFTIME = 1uS
 ONTIME = 1uS
 DELAY =
 STARTVAL = 0
 OPPVAL = 1



Truth Table:

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

Part-3:

Propagation Delay in Logic Gates:

Connect all inverters inside two 7404 ICs in cascade. The output will be the same as the input except that it will be delayed by the time it takes the signal to propagate through all six inverters. Set S2 to 100 kHz and apply clock pulses to the input of the first inverter (connect pin 1 to j14) record the wave forms and determine the time delay from the input to the sixth inverter. This is done with a dual trace oscilloscope by applying the input clock pulses to one of the channels and the output of the sixth inverter to the second channel and measuring the delay between the two signals as shown in Fig 6. By using measured delay between two signals calculate the propagation delay for each inverter gate.

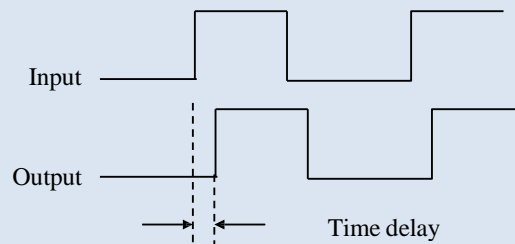
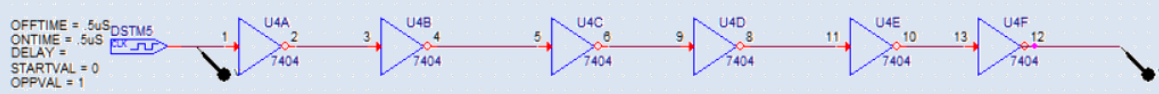
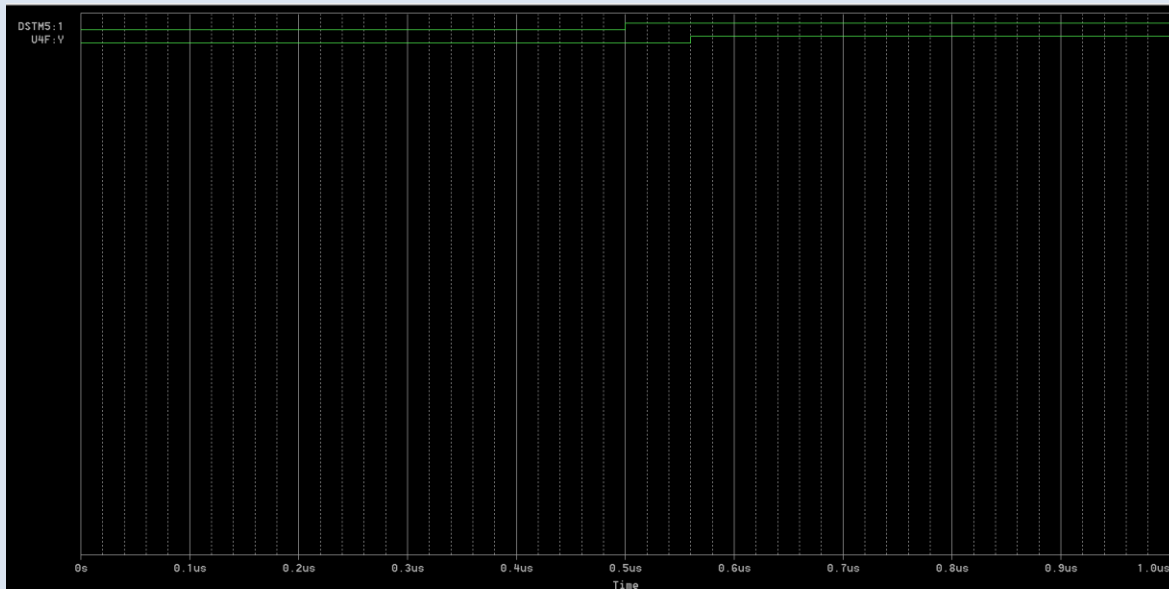


Fig. 6 Propagation delay



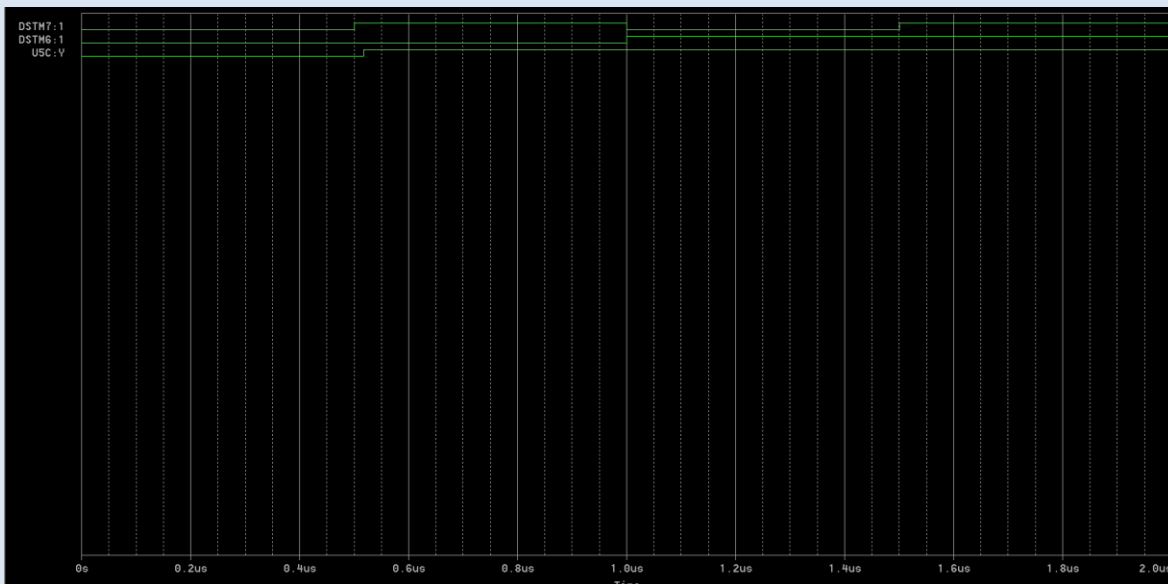
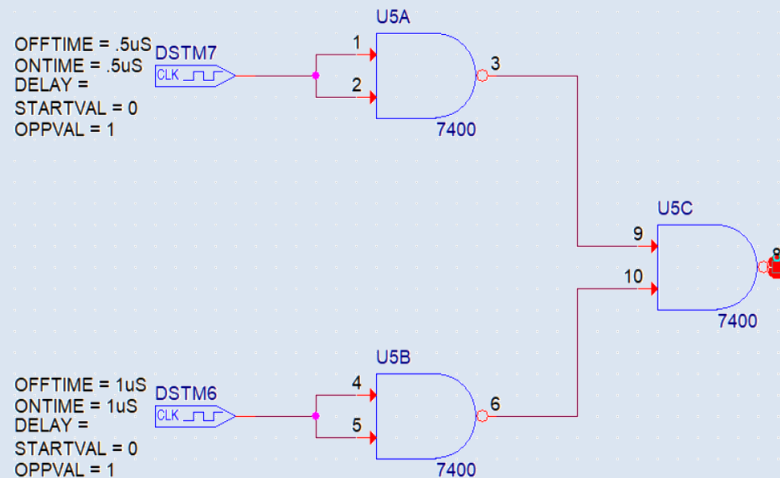
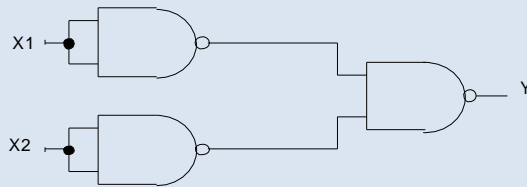


Inferences:

Here, we can see that the signal shifted causing a time delay of 0.06 microseconds between the input and the output signal through the series of NOT gates.

Part 4: Review Questions:

- Write a truth table for each circuit. Derive Boolean expressions for all outputs.

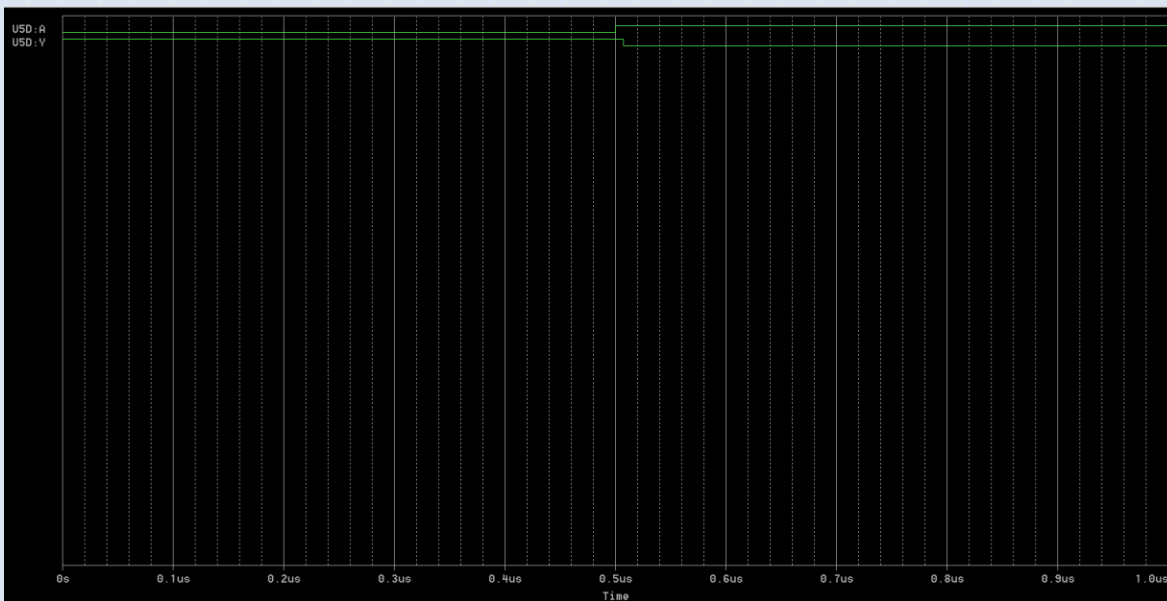
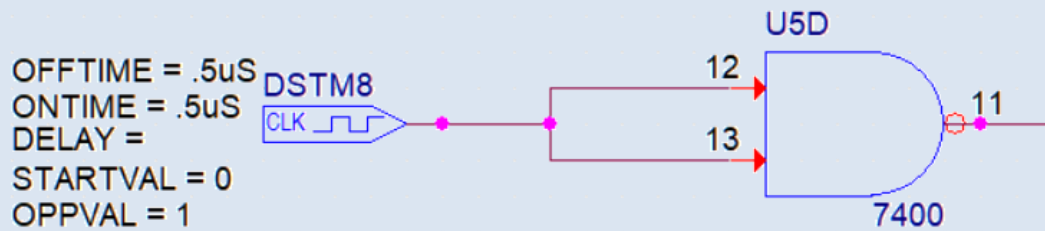
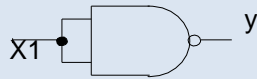
A.

The truth table for above circuit is:

X1	X2	Y
0	0	0
0	1	1
1	0	1
1	1	1

The Boolean expression in POS (product of sums) form would be: $Y = X1 + X2$

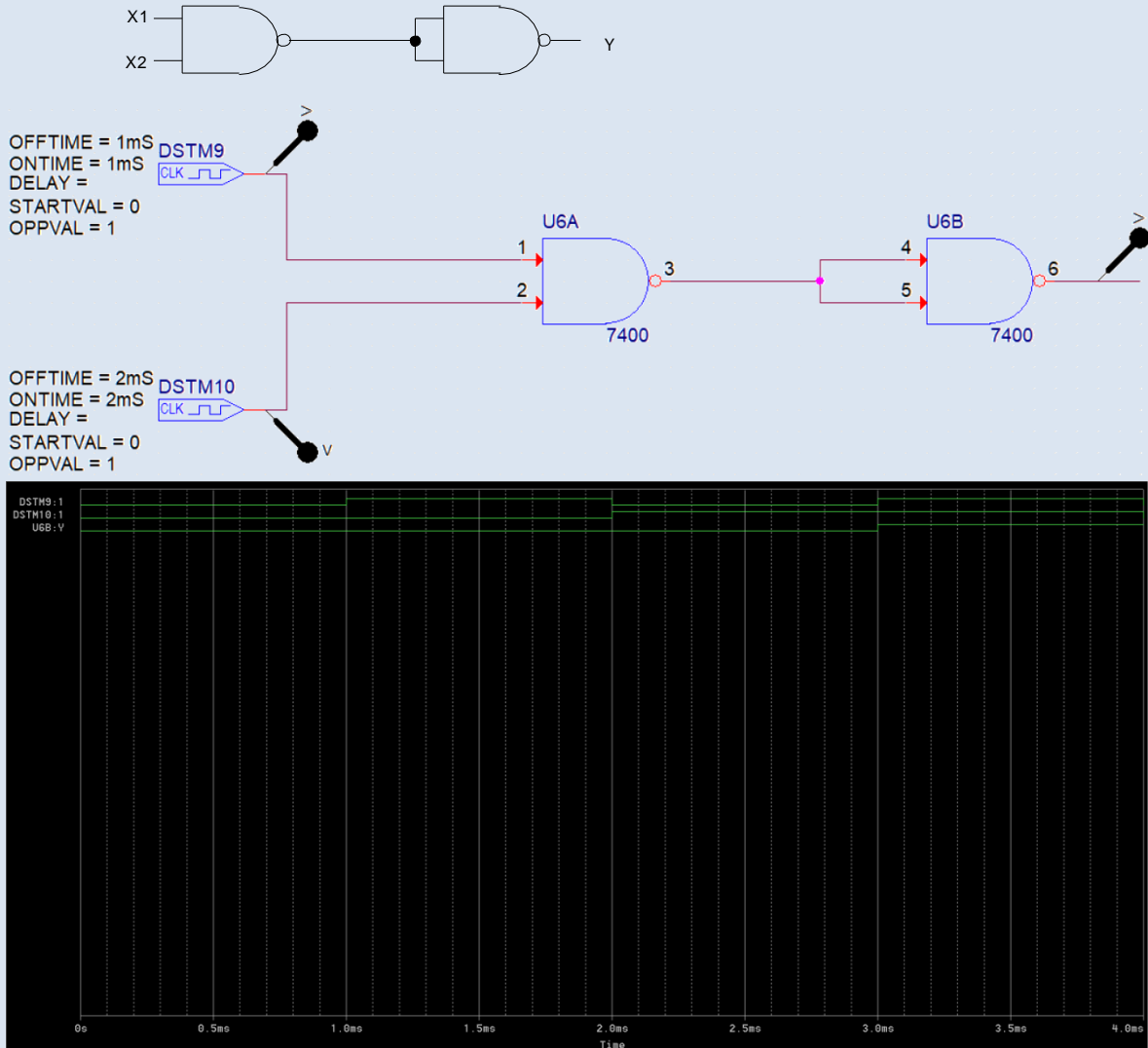
B.



The Boolean expression for the above circuit will be:

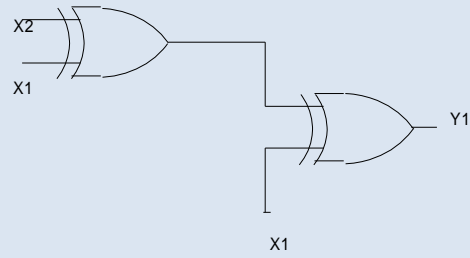
$$Y = X1'$$

C.



The Boolean expression of the given function is:
 $Y = X1 \text{ and } X2$

D.

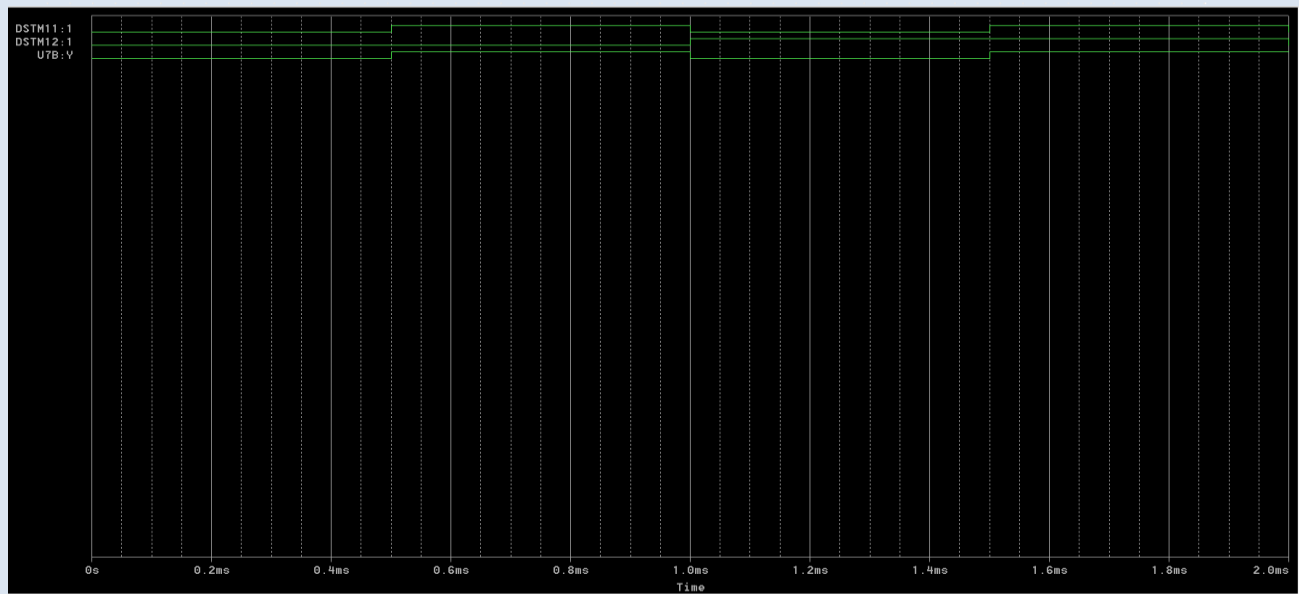
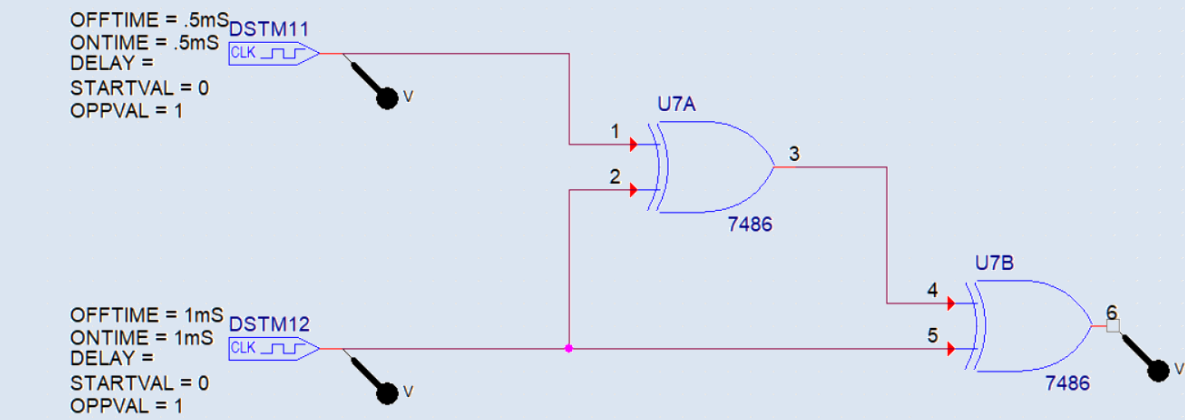


OFFTIME = .5mS
 ONTIME = .5mS
 DELAY =
 STARTVAL = 0
 OPPVAL = 1

DSTM11
 CLK

OFFTIME = 1mS
 ONTIME = 1mS
 DELAY =
 STARTVAL = 0
 OPPVAL = 1

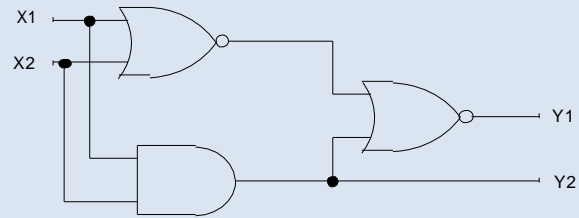
DSTM12
 CLK



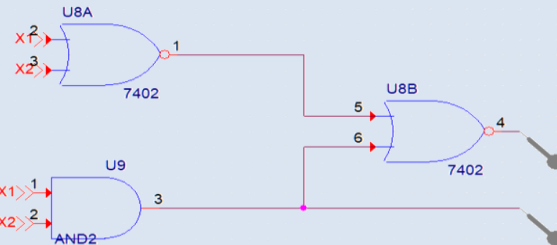
The Boolean expression is:

$$Y = X2$$

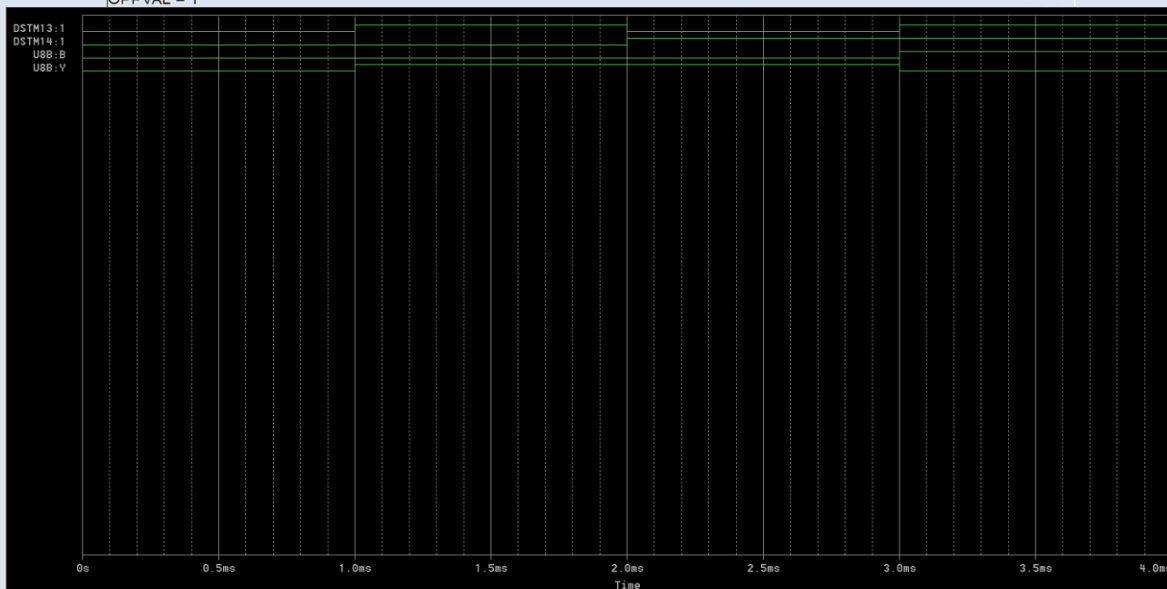
E.



OFFTIME = 2mS
 ONTIME = 1mS
 DELAY =
 STARTVAL = 0
 OPPVAL = 1



OFFTIME = 2mS
 ONTIME = 2mS
 DELAY =
 STARTVAL = 0
 OPPVAL = 1



The Boolean expressions of above circuit are:

$$Y1 = X1 \text{ AND } X2$$

$$Y2 = X1 \text{ XOR } X2$$

2. A burglar alarm for a car has a normally low switch on each of four doors. If any door is opened the output of that switch goes HIGH. The alarm is set off with an active-LOW output signal. What type of gate will provide this logic? Support your answer with an explanation.

Answer:

For the above stated purpose of burglar's alarm, **OR** gate would best serve the purpose.

It is because in OR gate, the output will be high even if anyone among multiple inputs is set high. Seeing this in context of Car's Burglar alarm, the alarm should go high if at least one or many among the four doors are closed.

The truth table of OR gate for 4 inputs is:

A	B	C	D	A +B+C+D
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Clearly, it can be seen that the output is high for all cases except when all inputs are low. Thus, the alarm will stay quiet when all doors are closed. For any other condition, it will make noise.

Hence, **OR** gate is best fit for the alarm.