Name : **LEONARD L. FORROSUELO** Date : **Oct 20, 2025**

Course-Year : **BSCpE - 3** Rating : _____

Subject-Section : **CPE361 – H3** Instructor: **Engr. Mervin John C. Tampus**

EXPERIMENT NO. 2
NAND and NOR GATES

OBJECTIVES:

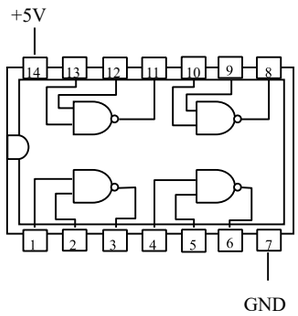
To demonstrate the operation and characteristics of a TTL logic gate and to show how it can be used to perform any of the three basic logic functions.

Materials

1 – SN74LS00N (7400) quad-two input TTL IC
1 – SN74LS02N (7402) quad-two input TTL IC
1 – 74LS20 TTL IC (4 input TTL IC)
1 – 4001 CMOS IC (quad 2 input NOR gate)
Multitester
1 – Logic Lab/ Digital Trainer
30 connectors/wire

Procedure

1. Mount the 7400 TTL IC on the breadboard socket. Be sure that it is seated firmly straddling the notch in the socket and that none of the pins are bent. Connect pin 14 to +5 volts and pin 7 to GND to supply power. The figure below shows the pin connections.



2. Connect one of the four gates in the IC as shown in Figure 2-1. The input will come from data switch SW1. You will monitor the input and output states with the L1 and L2 LED indications and your DC voltmeter.

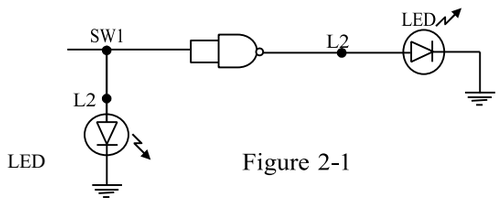


Figure 2-1

3. Set SW1 to the down position then the up position. Measure the DC input (pins 1 and 2) and output (pin 3) voltage for each position. Record your data in Table 1. Also note the LED indicator input/output states.

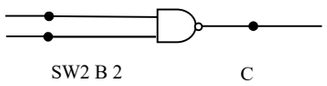
Table 1

INPUT	OUPUT
0	**1**
1	**0**

4. Assuming positive logic, the output logic levels are:
   Binary 0 = **0V** volts.
   Binary 1 = **5V** volts.

5. Study Table 1. What logic function is being performed? **NOT LOGIC**

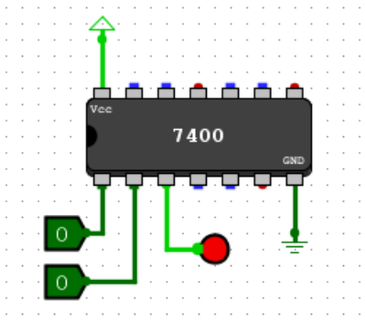
SW1 A 1 3



SW2 B 2 C

Figure 2-2

6. Wire the circuit shown in Figure 2-2. The inputs come from SW1 and SW2. You will measure the output voltage C at pin 3 of the 7400 IC.



7. With SW1 and SW2, apply the input voltage given in Table 2. Measure and record the output voltage for each set of inputs.

Table 2

INPUTS		OUTPUT
A	B	C
0V	**0V**	**5V**
0V	**5V**	**5V**
5V	**0V**	**5V**
5V	**5V**	**0V**

8. Using positive logic convert your electrical truth table in Table 2 into 1,s and 0,s in Table 3.

Table 3

INPUTS		OUTPUT
A	B	C
0	**0**	**1**
0	**1**	**1**
1	**0**	**1**
1	**1**	**0**

9. Study Table 3. What logic function is being performed? **NAND LOGIC**

10. Using negative logic, convert the data in Table 2 into 1,s and 0,s record in Table 4.

Table 4

INPUTS		OUTPUT
A	B	C
0	**0**	**0**
0	**1**	**1**
1	**0**	**1**
1	**1**	**1**

11. Study Table 4. What logic function is being performed? **OR GATE**

12. Remove the wires connecting pins 1 and 2 of the IC to SW1 and SW2. Let the gate inputs hang free. Note the output state. With open inputs, the TTL gate output is **0** volts or binary **0** for positive logic. This means that an open input acts like a binary **1**.

13. Wire the circuit shown in figure 2-3. With SW1 (A) and SW2 (B), apply the states shown in Table 5. Record the state for each set of inputs. Observe indicators L1, L2, and L3 to obtain your input and output data. Use positive logic (binary 1 = on, binary 0 = off).

A 1 3 4

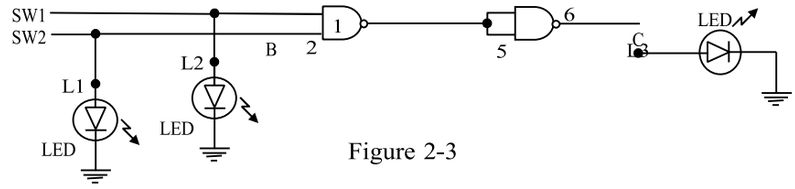


Figure 2-3

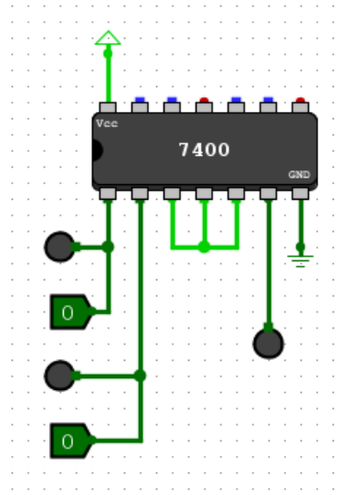


Table 5

INPUTS		OUTPUT
A	B	C
0	0	**0**
0	1	**0**
1	0	**0**
1	1	**1**

14. Study the circuit in Figure 2-3 and the data in Table 5. What logic function is being performed?
AND LOGIC

15. Connect the circuit shown in Figure 2-4. Monitor the inputs and output on LED indicators L1, L2, and L3. With SW1 (A) and SW2 (B), apply the input shown on Table 6. Record the output state corresponding to each set of inputs. Use positive logic.

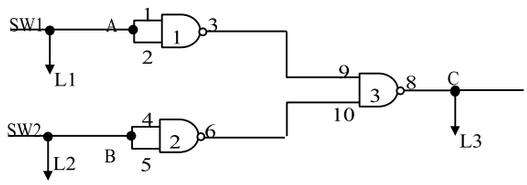


Figure 2-4

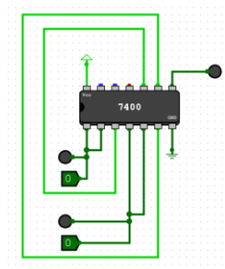


Table 6

INPUTS		OUTPUT
A	B	C
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**1**

16. Study Figure 2-4 and Table 6. What logic function is being performed? **OR LOGIC**

17. Modify your circuit in Figure 2-4 by adding the fourth gate in the 7400 to the output as shown in Figure 2-5. Only the output change is shown. The rest of the circuit stays as in Figure 2-4. 1

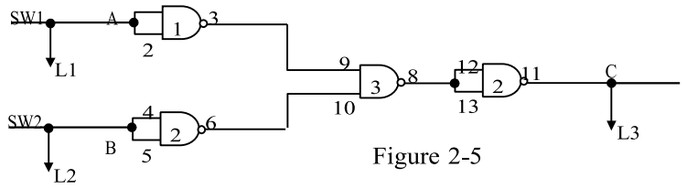
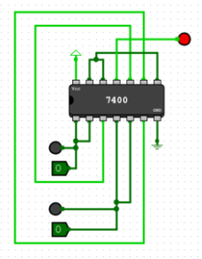


Figure 2-5



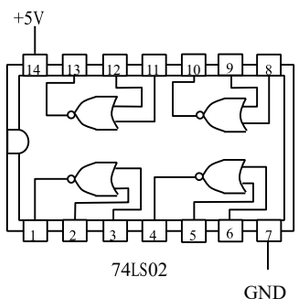
18. Using SW1 (A) and SW2 (B) data switches and monitoring LED indicators L1, L2, and L3, apply the states shown in Table 7. Record the output state for each set of inputs.

Table 7

INPUTS		OUTPUT
A	B	C
0	0	**1**
0	1	**0**
1	0	**0**
1	1	**0**

19. Study Table 7. What logic function is being performed? **NOR GATE**

20. Mount the 7402 TTL IC on the breadboard socket. Be sure that it is seated firmly straddling the notch in the socket and that none of the pins are bent. Connect pin 14 to +5 volts and pin 7 to GND to supply power. The figure below shows the pin connections.



21. Connect one of the four gates in the IC as shown in Figure 2-6. The input will come from data switch SW1. You will monitor the input and output states with the L1 and L2 LED indications and your DC voltmeter.

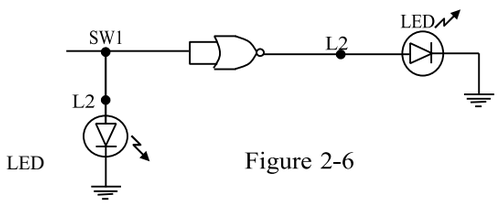


Figure 2-6

22. Set SW1 to the down position then the up position. Measure the DC input (pins 1 and 2) and output (pin 3) voltage for each position. Record your data in Table 8. Also note the LED indicator input/output states.

Table 8

INPUT	OUPUT
0V	**5V**
5V	**0V**

23. Assuming positive logic, the output logic levels are:
Binary 0 = __5__ volts.
Binary 1 = __0__ volts.

24. Study Table 8. What logic function is being performed? **NOT LOGIC**

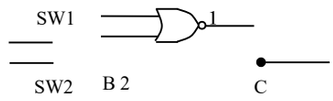
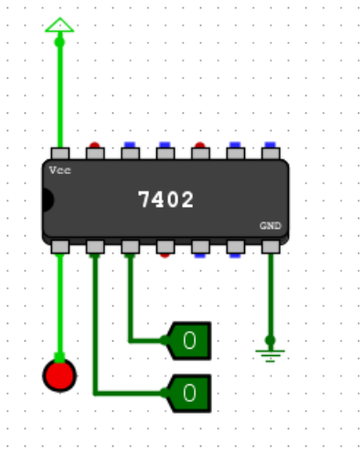


Figure 2-7

25. Wire the circuit shown in Figure 2-7. The inputs come from SW1 and SW2. You will measure the output voltage C at pin 1 of the 7402 IC.



26. With SW1 and SW2, apply the input voltage given in Table 9. Measure and record the output voltage for each set of inputs.

Table 9

INPUTS		OUTPUT
A	B	C
0	**0**	**5V**
0	**1**	**0V**
1	**0**	**0V**
1	**1**	**0V**

27. Using positive logic convert your electrical truth table in Table 9 into 1,s and 0,s in Table 10.

Table 10

INPUTS		OUTPUT
A	B	C
0	**0**	**1**
0	**1**	**0**
1	**0**	**0**
1	**1**	**0**

28. Study Table 10. What logic function is being performed? **NOR LOGIC**

29. Using negative logic, convert the data in Table 9 into 1,s and 0,s record in Table 11

Table 11

INPUTS		OUTPUT
A	B	C
0	**0**	**0**
0	**1**	**0**
1	**0**	**0**
1	**1**	**1**

30. Study Table 11. What logic function is being performed? **AND LOGIC**

31. Remove the wires connecting pins 1 and 2 of the IC to SW1 and SW2. Let the gate inputs hang free. Note the output state. With open inputs, the TTL gate output is **5** volts or binary **1** for positive logic. This means that an open input acts like a binary **0**.

32. Wire the circuit shown in figure 2-8. With SW1 (A) and SW2 (B), apply the states shown in Table 12. Record the state for each set of inputs. Observe indicators L1, L2, and L3 to obtain your input and output data. Use positive logic (binary 1 = on, binary 0 = off).

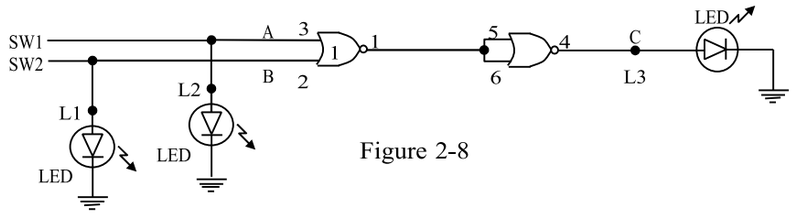


Figure 2-8

Table 12

INPUTS		OUTPUT
A	B	C
0	0	**0**
0	1	**1**
1	0	**1**
1	1	**1**

33. Study the circuit in Figure 2-8 and the data in Table 12. What logic function is being performed? **OR LOGIC**

34. Connect the circuit shown in Figure 2-9. Monitor the inputs and output on LED indicators L1, L2, and L3. With SW1 (A) and SW2 (B), apply the input shown on Table 13. Record the output state corresponding to each set of inputs. Use positive logic.

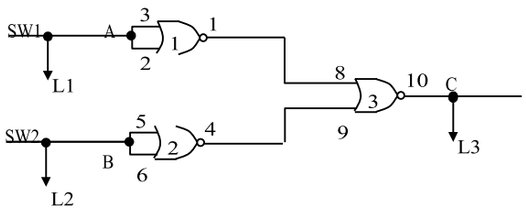


Figure 2-9

Table 13

INP JTS		OUTPUT
A	B	C
0	0	**0**
0	1	**0**
1	0	**0**
1	1	**1**

35. Study Figure 2-9 and Table 13. What logic function is being performed? **AND LOGIC**

36. Modify your circuit in Figure 2-9 by adding the fourth gate in the 7402 to the output as shown in Figure 2-10. Only the output change is shown. The rest of the circuit stays as in Figure 2-9.

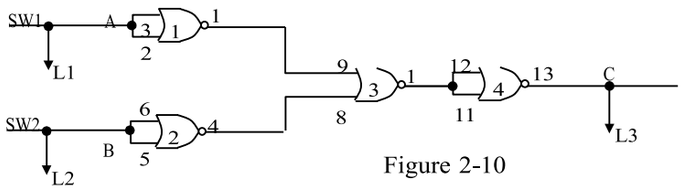


Figure 2-10

37. Using SW1 (A) and SW2 (B) data switches and monitoring LED indicators L1, L2, and L3, apply the states shown in Table 8. Record the output state for each set of inputs.

Table 14

INPUTS		OUTPUT
A	B	C
0	0	**1**
0	1	**1**
1	0	**1**
1	1	**0**

38. Study Table 14. What logic function is being performed? **NAND LOGIC**

39. Write the output expression of the circuit shown in Figure 2-11.
F = **AC + AD + BC + BD**

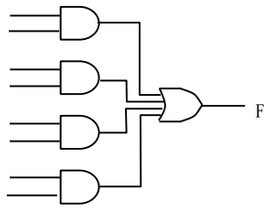


Figure 2-11

40. Figure 2-12 shows the NAND gate implementation of the circuit in figure 2-11. Wire the circuit shown in Figure 2-11. The pin connections for the 74LS00 and 74LS20 are given in Figure 2-13. Be sure to connect pin 14 to +5 volts and pin 7 to GND on each IC. Connect +5V to all unused inputs.

Figure 2-12

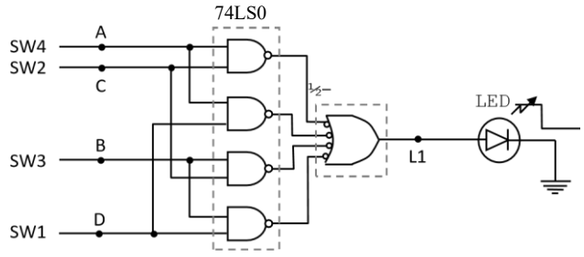
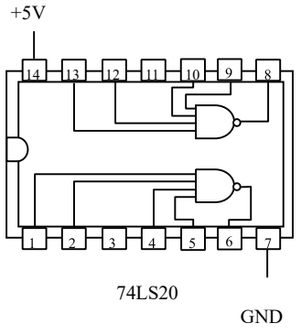


Figure 2-13



41. Apply the inputs A, B, C and D in Table 1 to the circuit with data switches SW1 through SW4. Monitor the ourtput on L1 and record the state for each set of inputs in the left hand F column in Table 15.

Table 15

INPUTS				OUTPUTS	
A (SW4)	B (SW3)	C (SW2)	D (SW1)	F (L1) Fig. 2-11	F (L1) Fig. 2-12
0	0	0	0	**0**	**0**
0	0	0	1	**0**	**0**
0	0	1	0	**0**	**0**
0	0	1	1	**0**	**0**
0	1	0	0	**0**	**0**
0	1	0	1	**1**	**1**
0	1	1	0	**1**	**1**
0	1	1	1	**1**	**1**
1	0	0	0	**0**	**0**
1	0	0	1	**1**	**1**
1	0	1	0	**1**	**1**
1	0	1	1	**1**	**1**
1	1	0	0	**0**	**0**
1	1	0	1	**1**	**1**

1	1	1	0	**1**	**1**
1	1	1	1	**1**	**1**

42. Using Boolean algebra, reduce the output equation obtained in step 39. The minimized expression is:
F = AC + AD + BC + BD
F=A(C+D)+B(C+D)
F=(A+B)(C+D)
F = (A+B) (C+D)

43. Construct the circuit shown in Figure 2-12. Remember to connect +5V to all unused inputs.

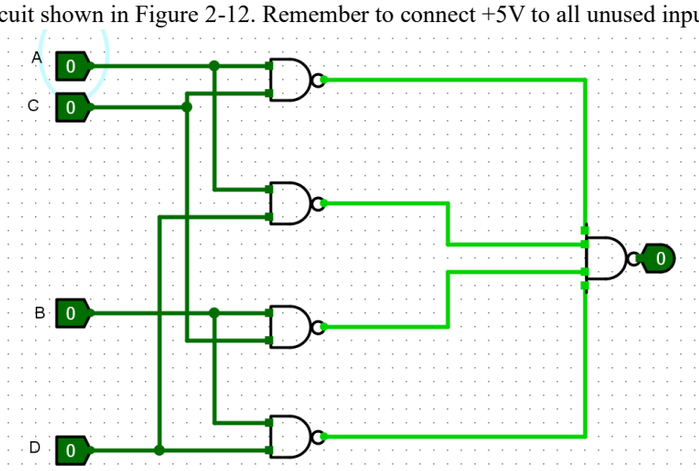


Figure 2-12

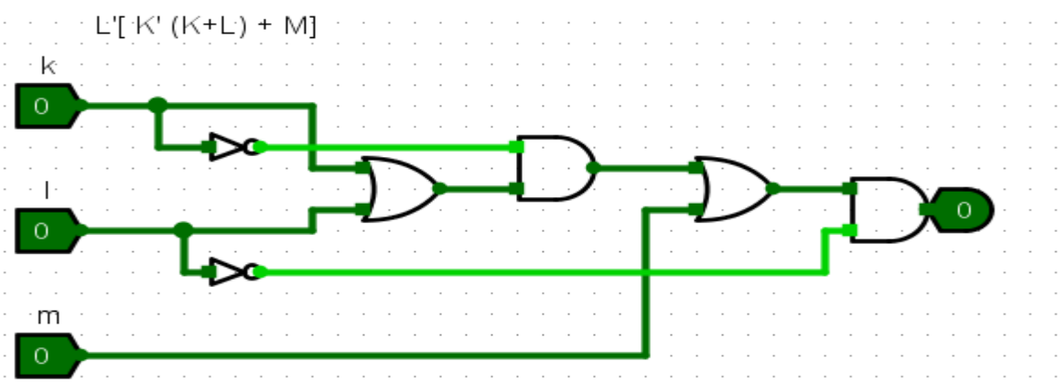
44. Write the output equation of the circuit in Figure 2-12. Compare it to the expression you derived in Step 42.
F = (AC)' + (AD)' + (BC)' + (BD)' **or**
F = A' + B' + C' + D'

45. Apply the inputs shown in Table 15 and record the output state in the right-hand column.

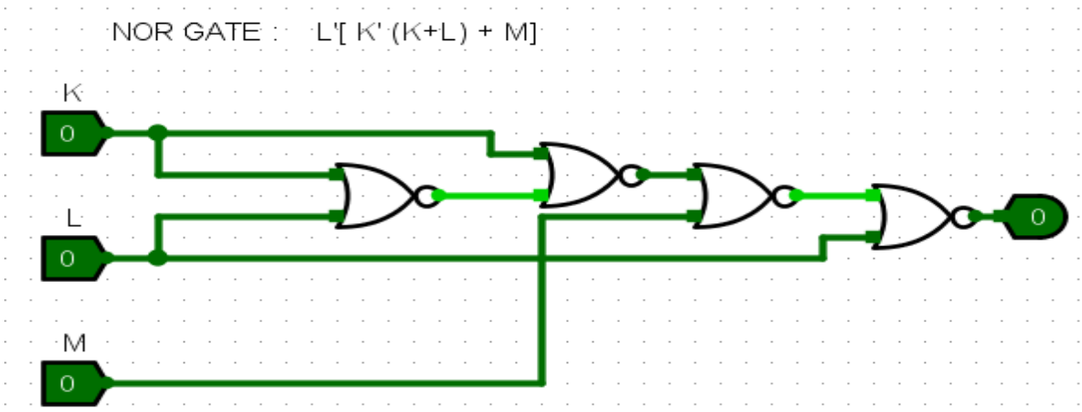
46. Compare the two F output columns in Table 15. What conclusion can you reach regarding the circuits in Figure 2-11 and 2-12?

By comparing the two F output columns in Table 15, it is evident that both circuits in Figures 2-11 and 2-12 produce identical outputs for every possible combination of inputs A, B, C, and D. This means that the two circuits are logically equivalent, performing the same Boolean function even though their configurations or arrangements of logic gates may differ.

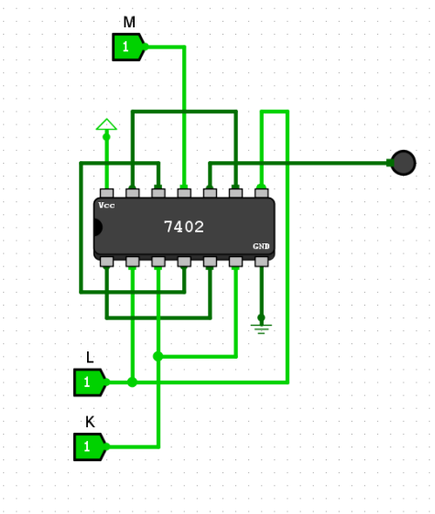
47. Draw the AND and OR gate logic diagram of the expression X= L' [K' (K+L) + M].



48. Redraw the circuit using positive NOR gates.



49. Implement your circuit in Step 47 with a 4001 CMOS quad two input NOR gate IC. The pin connections for the 4001 IC.. Connect +5 volts to pin 14 and ground to pin 7. Connect all unused inputs to +5 volts.



50. Develop a trith table for the circuit. Use SW2, SW3, and SW4 to apply the K, L and M inputs. Monitor your output on L1. Record your output in the left hand X column of Table 16.

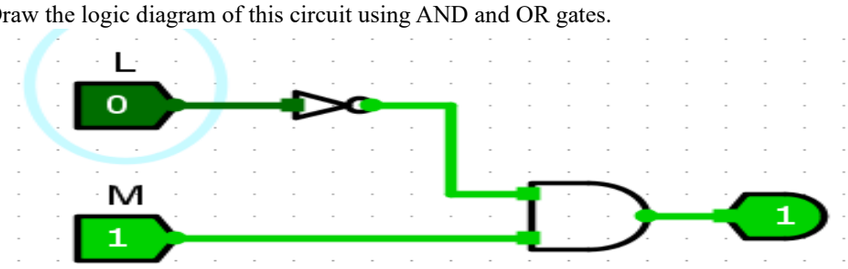
Table 16

INPUTS			OUTPUTS	
K (SW2)	L (SW3)	M (SW4)	X (Step 50)	X (Step 54)
0	0	0	**0**	**0**
0	0	1	**1**	**1**
0	1	0	**0**	**0**
0	1	1	**0**	**0**
1	0	0	**0**	**0**
1	0	1	**1**	**1**
1	1	0	**0**	**0**
1	1	1	**0**	**0**

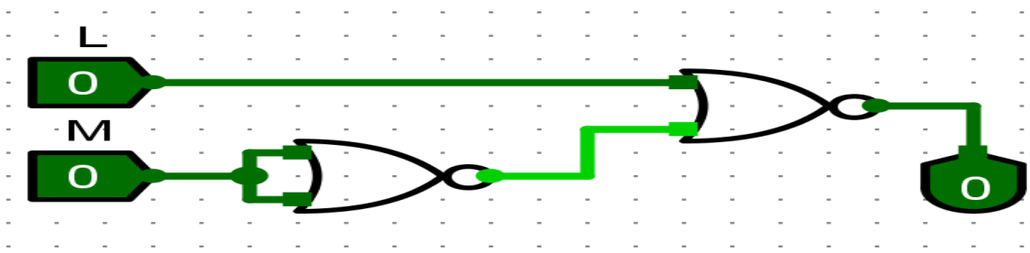
51. Reduce the expression in Step 47 using Boolean algebra. The minimized equation is;

X = L' [K' (K+L) + M]
X = L' [K'K + K'L + M]
X = L' [K'L + M]
X = L'KL + L'M
<u>X = L'M</u>

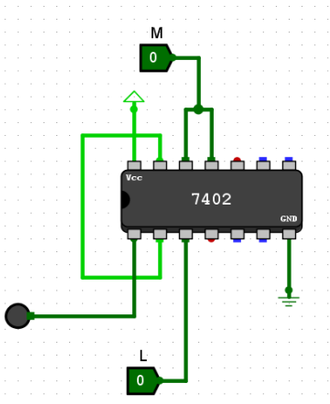
52. Draw the logic diagram of this circuit using AND and OR gates.



53. Implement the circuit developed in Step 52 with CMOS NOR gates.



54. Wire the minimized circuit and develop a truth table. Apply inputs K, L, and M with data switches SW2, SW3 and SW4. Monitor the output on L1. Use the right-hand X column in Table 16 to record your data.



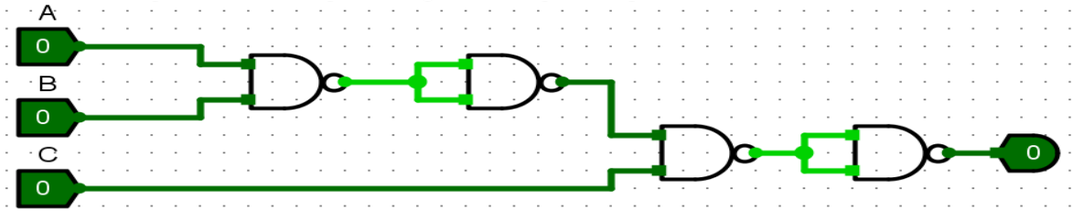
55. Compare the two X output columns in Table 16. What conclusions can you draw? What circuit minimization was really accomplished?

The two identical X output columns prove that the unsimplified and simplified logic circuits are logically equivalent, confirming that the circuit minimization process was functionally correct. The real achievement is the reduction in physical complexity, as the simplified design uses significantly fewer logic gates and inputs. This physical minimization makes the final circuit cheaper, smaller, faster (due to less propagation delay), and ultimately more power-efficient.

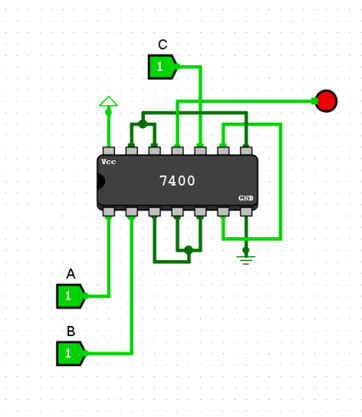
56. Write the truth table for a 3-input AND gate and sketch the appropriate logic symbol.

INPUTS			OUTPUT
A	B	C	F
0	**0**	**0**	**0**
0	**0**	**1**	**0**
0	**1**	**0**	**0**
0	**1**	**1**	**0**
1	**0**	**0**	**0**
1	**0**	**1**	**0**
1	**1**	**0**	**0**
1	**1**	**1**	**1**

57. Show how to implement a three input AND gate with a quad 2-input NAND.



58. Draw the circuit, implement it on your Trainer and verify its operation with a truth table.



OBSERVATION/CONCLUSION:

In this experiment, I gained a clear understanding of how NAND and NOR gates operate and how they can be combined to create other basic logic functions such as AND, OR, and NOT. I learned that these two are known as universal gates because any digital circuit can be built using only NAND or only NOR. Constructing the circuits on the breadboard allowed me to visualize how the theoretical logic concepts are applied in actual hardware. I also practiced simplifying logic expressions using Boolean algebra, which helps reduce the number of components needed and makes the circuit more efficient. Through this activity, I further strengthened my knowledge of how digital systems interpret and manipulate binary information. I also discovered that modern CPUs still rely on millions or even billions of these logic gates to execute every operation in computers and electronic devices today.