Experiments Name: Combinational Logic Design (Canonical Form)

Objective:

- Become familiarized with the analysis of combinational logic networks.
- Learn the implementation of networks using the two canonical forms.

Apparatus:

- 1x IC 7411 Triple 3-input AND gates
- 2x IC 4075 Triple 3-input OR gates
- 1x IC 7404 Hex Inverters (NOT gates)
- Trainer Board
- Wires

Theory:

Minterms and Maxterms:

A minterm is a product term that includes all the variables in the function, where each variable appears either in its direct or complemented form. For example, in a function of two variables, the minterms are:

- A'B'
- A'B
- AB'
- AB

A maxterm, on the other hand, is a sum term that includes all the variables in the function, where each variable appears either in its direct or complemented form. For example, in a function of two variables, the maxterms are:

- (A+B)
- (A'+B)
- (A+B')
- (A'+B')

It is important to note that minterms and maxterms are complements of each other. That is, the minterm expression for a given Boolean function can be obtained by taking the complement of its maxterm expression, and vice versa.

The four minterms and maxterms for 2 variables, together with symbolic designations, are listed in Table 1.

		Minterms		Maxterms	
x	**y**	**Term**	**Designation**	**Term**	**Designation**
0	0	x'y'	m_0	x + y	M_0
0	1	x'y	m_1	x + y'	M_1
1	0	xy'	m_2	x' + y	M_2
1	1	xy	m_3	x' + y'	M_3

Table 1

Replace x with A and y with B.

Canonical Forms:
In digital logic, a canonical form is a unique and standard representation of a Boolean function using either minterms or maxterms. There are two types of canonical forms: Sum of Products (SOP) and Product of Sums (POS).

A Boolean function expressed in SOP canonical form is a sum of minterms, where each minterm represents a unique combination of input variables that produces a logic high output. For example, the Boolean function F(A, B, C) = A'B'C + A'BC' + ABC' can be expressed in SOP canonical form as F(A, B, C) = m(1, 2, 6), where m(i) denotes the i-th minterm.

A Boolean function expressed in POS canonical form is a product of maxterms, where each maxterm represents a unique combination of input variables that produces a logic low output. For example, the Boolean function F(A, B, C) = (A+B+C) (A+B'+C') (A'+B+C) (A'+B+C') (A'+B'+C') can be expressed in POS canonical form as F(A, B, C) = M(0, 3, 4, 5, 7), where M(i) denotes the i-th maxterm.

Circuit Diagram:

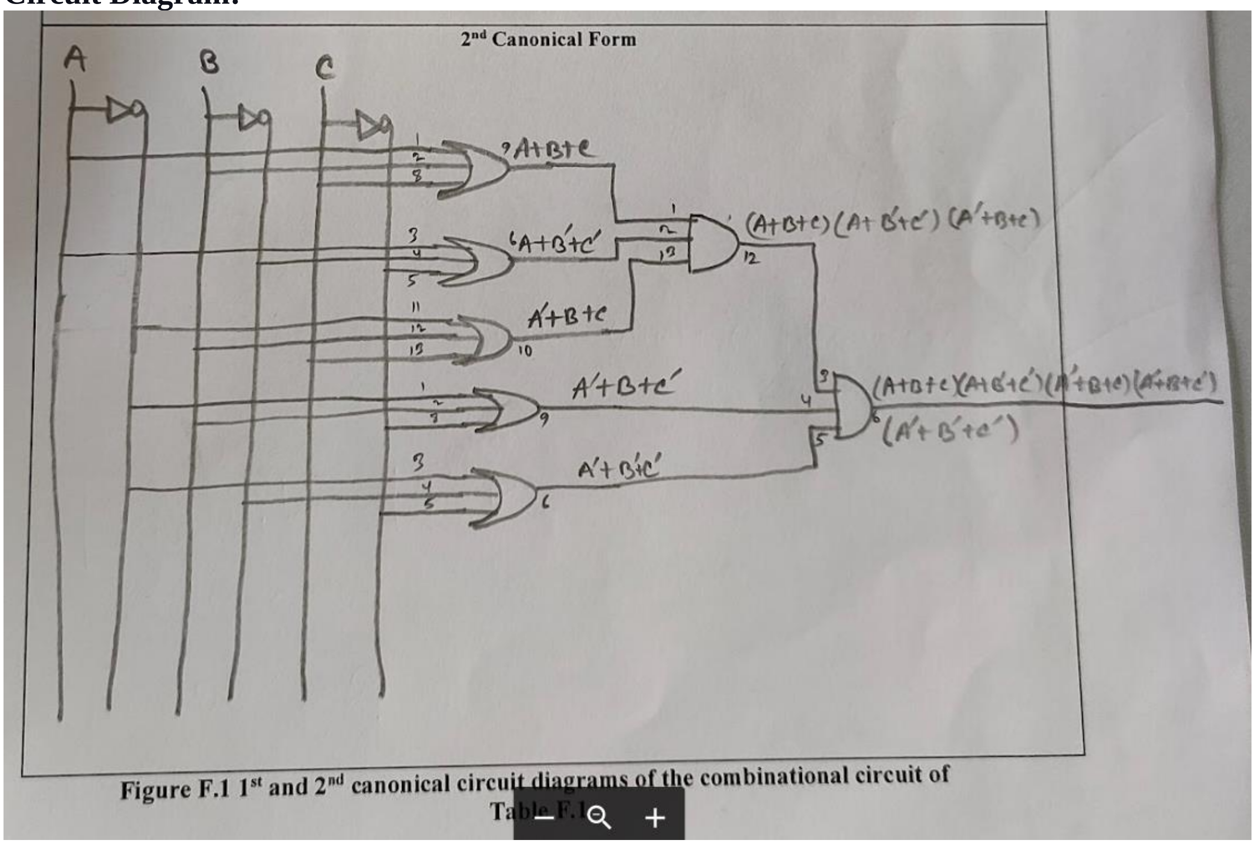


Figure F.1 1st and 2nd canonical circuit diagrams of the combinational circuit of Table F.1

Experimental Procedure:

01. First, we write down all min terms and max terms of three inputs ABC in Table F.1.

02. Then, we express the function in 1st and 2nd Canonical Forms and write it in Table F.2.

03. After that, we draw two circuit diagram for 1st and 2nd Canonical Forms with PIN labels in Figure F.1.

04. Then, we start to construct the 1st Canonical Form Circuit in the trainer board. First, we connect one min term and verify its output. Then, we connect 2nd min term and check for output. One by one we connect all min term and check output for each. Then we connect all min term in an OR gate and check the final output.

05. After that, we start to construct the 2nd Canonical Form Circuit in the trainer board. First, we connect one max term and verify its output. Then, we connect 2nd max term and check for output. One by one we connect all max term and check output for each. Then we connect all max term in an AND gate and check the final output.

Simulation:

Attached.

Experimental Data Table:

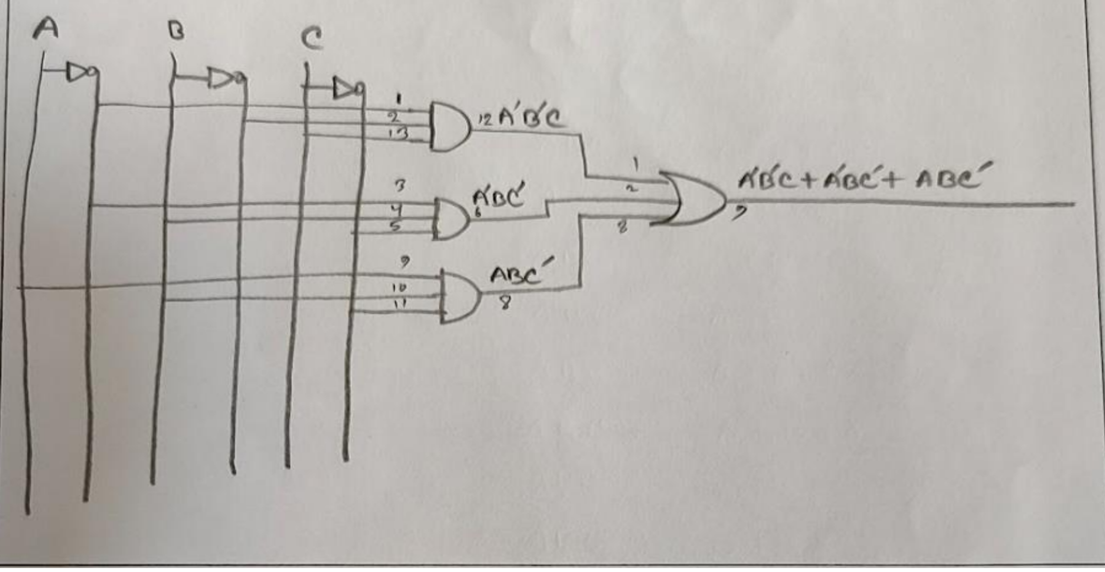
Input Reference	*A B C*	*F*	Min term	Max term
0	0 0 0	0	A'B'C'	A+B+C
1	0 0 1	1	A'B'C	A+B+C'
2	0 1 0	1	A'BC'	A+B'+C
3	0 1 1	0	A'BC	A+B'+C'
4	1 0 0	0	AB'C'	A'+B+C
5	1 0 1	0	AB'C	A'+B+C'
6	1 1 0	1	ABC'	A'+B'+C
7	1 1 1	0	ABC	A'+B'+C'

Table F.1 Truth table to a combinational circuit

	Shorthand Notation	Function
1st Canonical Form	$F = \Sigma(1,2,6)$	$F = A'B'C + A'BC' + ABC'$
2nd Canonical Form	$F = \Pi\ (0,3,4,5,7)$	$F = (A+B+C)(A+B'+C')(A'+B+C)(A'+B+C')(A'+B'+C')$

Table F.2 1st and 2nd canonical forms of the combinational circuit of Table F.1

1st Canonical Form



2nd Canonical Form

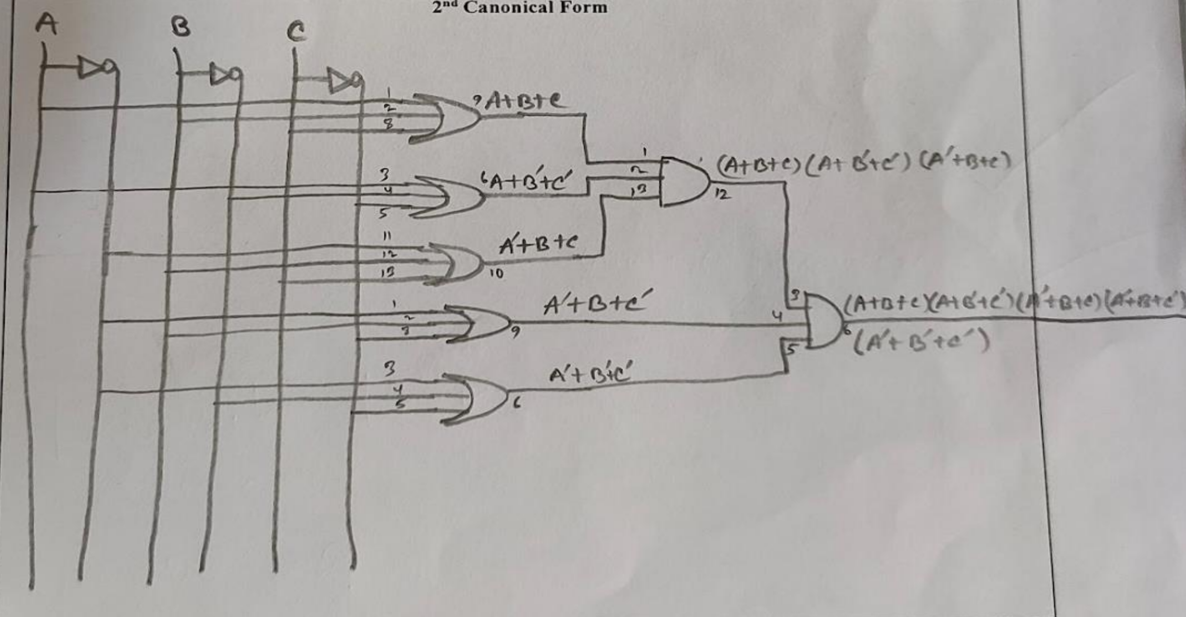


Figure F.1 1st and 2nd canonical circuit diagrams of the combinational circuit of Table F.1

Results:

After constructing two circuits of the 1st and 2nd Canonical Forms, we found that both outputs are the same. That means we can express a function in either the 1st Canonical Form or the 2nd Canonical Form; both outputs will be the same. We can check which Form takes fewer gates than others and then reduce the cost of building the circuit.

Questions and Answers (Q/A):

01. A minterm is a product term that includes all the variables in the function, where each variable appears either in its direct or complemented form. For example, in a function of two variables, the minterms are:
    - A'B'
    - A'B
    - AB'
    - AB

A Boolean function expressed in SOP canonical form is a sum of minterms, where each minterm represents a unique combination of input variables that produces a logic high output. For example, the Boolean function F(A, B, C) = A'B'C + A'BC' + ABC' can be expressed in SOP canonical form as F(A, B, C) = m(1, 2, 6), where m(i) denotes the i-th minterm.

No, the following expression is not the first canonical form. Here, AB' is a minterm and ABC' is also a minterm. And two minterm are connected by OR gate. So, it's a Sum of Products. But its not a standard SOP. So, we can't call it by 1st canonical form.

02. No, just because a Boolean expression is in canonical form does not necessarily mean that it is in its minimal form. Canonical form simply represents a unique and standard representation of a Boolean function using either minterms or maxterms. On the other hand, the minimal form of a Boolean expression represents the simplest possible expression of that function in terms of its variables and operators.

    It is possible for a Boolean expression to be in canonical form but not be in its minimal form, meaning that there may be a simpler expression that represents the same function. For example, consider the Boolean function F(A,B,C) = A'B + AB' + ABC in SOP canonical form. We can simplify this expression using Boolean algebra as:

    F(A, B, C) = A'B + AB' + ABC

    = AB' + A'B + ABC (commutativity)

    = B(A+A'C) (factorization)

    = B (simplification)

    Thus, the minimal expression of F is simply F = B, which is simpler than its canonical form.

    In summary, while canonical form provides a standardized way to represent a Boolean function, it does not necessarily ensure that the expression is in its minimal form. To find the minimal form of a Boolean expression, additional simplification using Boolean algebra or other optimization techniques may be necessary.
03. Draw a Table
04. Draw the 7411 IC

05. Draw IC Diagram for 1st Canonical Form
06. Simulation Attached.

Discussion:

This experiment teaches us about two canonical forms, 1st and 2nd. We learn how to build a combinational circuit using the canonical form. We also verify that two canonical forms give us the same output. While constructing the 1st canonical form in this experiment, we face some problems with the NOT IC. After changing the NOT IC, we get the accurate result. Finally, we built two circuits successfully.