

ANALYSIS PROCEDURE

The analysis for a combinational circuit consists of determining the function that the circuit implements. The analysis starts with a given logic circuit diagram and culminates with a set of Boolean functions or a truth table, together with a possible explanation of the operation of the circuit. If the logic diagram to be analyzed is accompanied by a recognizable function name or a statement of what the diagram is assumed to accomplish, then the analysis problem reduces to a verification of the stated function. We can perform the analysis by manually finding the Boolean equations or truth table or by using logic simulation to apply inputs to the circuit, as mentioned in the last section. Both methods will be covered here.

The first step in the analysis is to make sure that the given circuit is combinational and not sequential. The diagram of a combinational circuit has logic gates with no feedback or storage elements. A feedback path exists if, from the output of a gate, an input of the same gate can be reached via a path of interconnections or gates. Feedback paths or storage elements in a digital circuit may result in a sequential circuit and must be analyzed by using different procedures.

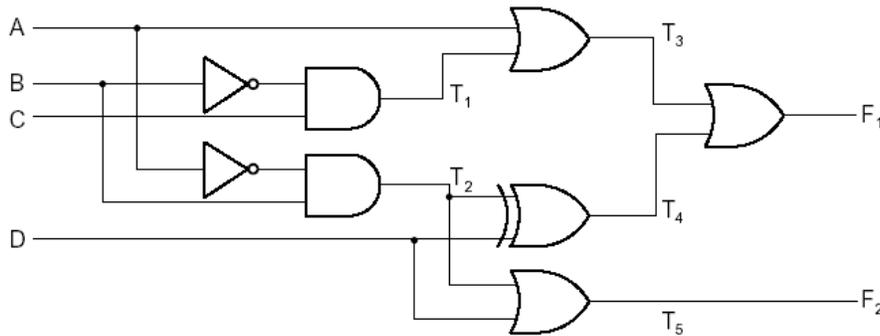
Once the logic diagram is verified to be that of a combinational circuit, one can proceed to obtain the output Boolean functions or the truth table. If the function of the circuit is to be investigated, then it is necessary to interpret how the circuit operates from the derived Boolean functions or truth table. The success of such an investigation is enhanced if one has previous experience and familiarity with a wide variety of digital circuits.

Derivation of Boolean Functions

To obtain the output Boolean functions from a logic diagram, proceed as follows:

1. Label all gate outputs that are a function only of input variables or their complements with arbitrary symbols. Determine the Boolean functions for each gate output.
2. Label the gates that are a function of input variables and previously labeled gates with different arbitrary symbols. Find the Boolean functions for the outputs of these gates.
3. Repeat the process outlined in step 2 until the outputs of the circuit are obtained in terms of the input variables.

The analysis of the combinational circuit below illustrates this procedure.



Note that the circuit has four binary input variables A , B , C , and D and two binary output variables F_1 and F_2 . The outputs of gates that are a function of input variables only are T_1 and T_2 . The Boolean functions for these two outputs are

$$T_1 = \overline{BC}$$

$$T_2 = \overline{AB}$$

Next, we consider the outputs of gates that are functions of gates with symbols already defined:

$$T_3 = A + T_1 = A + \overline{BC}$$

$$T_4 = T_2 \oplus D = (\overline{AB}) \oplus D = \overline{AB}\overline{D} + AD + \overline{BD}$$

$$T_5 = T_2 + D = \overline{AB} + D$$

The Boolean functions for the outputs are thus

$$F_2 = T_5 = \overline{AB} + D$$

$$F_1 = T_3 + T_4 = A + \overline{BC} + \overline{AB}\overline{D} + AD + \overline{BD} = A + \overline{BC} + \overline{BD} + \overline{BD}$$

The last simplification for F_1 can be performed by algebraic manipulation or by means of a map.

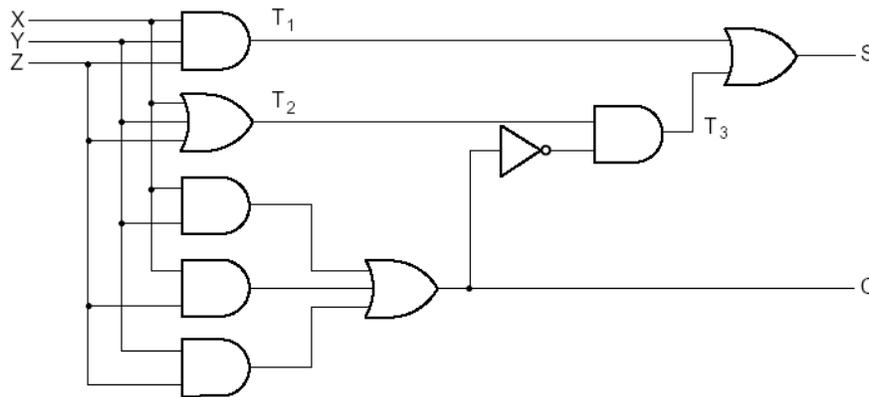
If the circuit to be analyzed is designed hierarchically, the diagrams for the lowest level block instances are analyzed first; those for block instances in the next level up are analyzed next, and so on, until Boolean equations are obtained for all outputs.

Derivation of the Truth Table

The derivation of the truth table for a combinational circuit is a straightforward process once the output Boolean functions are known. Alternatively, to obtain the truth table from the logic diagram without going through the derivation of the Boolean functions, proceed as follows:

1. Determine the number of input variables in the circuit. For n inputs, list the binary numbers from 0 to $2^n - 1$ in a table.
2. Break the circuit into small single-output blocks by labeling each block output with an arbitrary symbol.
3. Obtain the truth table for the blocks with functions that depend on input variables only.
4. Proceed to obtain the truth table for blocks with functions that depend on previously defined inputs and block outputs, until the columns for all circuit outputs are determined.

The foregoing process is illustrated by the combinational circuit for a binary adder shown below



The problem here is to verify that the circuit forms the arithmetic sum of the three bits at inputs X , Y , and Z . The output pair (C, S) ranges in value from binary 00 to 11 (decimal 3), depending on the number of 1's in the inputs. For example, when $(X, Y, Z) = 101$, (C, S) must be equal to binary 10 to indicate that there are two 1's on the inputs.

The table below shows the procedure for deriving the truth table of the circuit. First, we split up the circuit into blocks by labeling selected gate outputs. Next, we form the eight binary combinations for the three input variables and then find and logically combine truth tables. The truth table for C is determined from the values of the inputs X , Y , and Z . C is equal to 1 when $(X, Y) = 11$, $(X, Z) = 11$, or $(Y, Z) = 11$. Otherwise, C is equal to 0. The truth table for \bar{C} is the complement of C . The truth tables for T_1 and T_2 are the AND and OR functions of the input variables, respectively. The binary values for T_3 are derived from ANDing T_2 and \bar{C} . Thus, T_3 is equal to 1 when both T_2 is equal to 1 and \bar{C} is equal to 1. Otherwise, T_3 is equal to 0. Finally, S is equal to 1 for those combinations in which either T_1 or T_3 or both are equal to 1.

Truth Table for Binary Adder

X	Y	Z	C	\bar{C}	T_1	T_2	T_3	S
0	0	0	0	1	0	0	0	0
0	0	1	0	1	0	1	1	1
0	1	0	0	1	0	1	1	1
0	1	1	1	0	0	1	0	0
1	0	0	0	1	0	1	1	1
1	0	1	1	0	0	1	0	0
1	1	0	1	0	0	1	0	0
1	1	1	1	0	1	1	0	1

Inspection of the truth table reveals that $(C, S) = 00, 01, 10, \text{ or } 11$ when the total number of 1's on the three inputs X , Y , and Z is either zero, one, two, or three, respectively. This verifies the operation of the circuit as a binary adder.

If the circuit to be analyzed is hierarchically designed with blocks containing more than a few gates, it is wise to find the Boolean equations for the circuit outputs and evaluate these equations to get the truth table. The alternative, combining truth tables of inputs to complex logical structures to get the output truth table, is difficult to do manually.

The above material has been excerpted and adapted from *Logic and Computer Design Fundamentals*, Second Edition, Updated, M. Morris Mano and Charles R. Kime, Prentice Hall, 2001.