Introduction to Automata Theory, Formal Languages and Computation

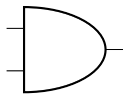
Shyamalendu Kandar

Latex by Ahmad Armaghan
StudentId : 909735631
Professor: Mr. Ali Razavi Ebrahimi
Course: Languages Theory 1115157-01

December 21, 2020

device. As gates are digital components, all these voltage ranges are restricted to two values 'high' and 'low'. In digital electronics, 'high' is represented as '1' and low is represented as '0'. The electronic gate acts as a switching device which either permits fl ow of current or blocks it. The basic gates are AND, OR, NOT, and XOR.

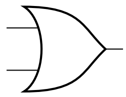
The AND gate has two or more inputs and a single output (Fig. 1.9). If all of the inputs are '1' the output is '1'; otherwise, the output is '0'. For a twoinput AND gate, with input labels 'A' and 'B', the output function is written as T = AB. A two-input AND gate is represented by the following symbol and the truth table is as follows.



A	B	O/P
0	0	0
0	1	0
1	1	1
1	0	0

Figure 1: *The AND Gate with Truth Table*

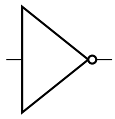
The OR gate also has two or more inputs and a single output (Fig. 1.10). If any of the inputs is '1', the output is '1'; otherwise, it is '0'. For a two-input OR gate, with input labels 'A' and 'B', the output function is written as T = A + B. A two-input OR gate is represented by the following symbol and the truth table is as follows.



A	B	O/P
0	0	0
0	1	1
1	1	1
1	0	1

Figure 2: *The OR Gate with Truth Table*

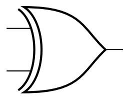
The NOT gate has a single input and a single output (Fig. 1.11). The function of the NOT gate is to reverse the input. The output function of the NOT gate is represented as T = A, where 'A' is the input. The symbol and truth table of the NOT gate is given as follows.



A	O/P
0	1
1	0

Figure 3: *The NOT Gate with Truth Table*

The exclusive-OR, in short XOR, gate is a complex gate (Fig. 1.12). A two-input XOR gate gives the output '1' when its two inputs are different and the output '0' when the inputs are the same. For two inputs 'A' and 'B', the output functions are represented by $T = A \oplus B = AB + AB$



A	B	O/P
0	0	0
0	1	1
1	1	0
1	0	1

Figure 4: *The XOR Gate with Truth Table*

1.6 Digital Circuit

A digital circuit is a circuit using logic gates where the signal must be one of two discrete levels. Each level is interpreted as one of two different states depending on the voltage level (on/off, 0/1 or true/ false). The digital circuit is operated by the logic of the Boolean algebra. This logic is the foundation of digital electronics and computer processing.

Depending on the output function, digital circuits are divided into two groups:

1 **Combinational circuits:** The circuits where the output depends only on the present input, i.e., output is the function of only the present input, are called combinational circuits.

2 **Sequential circuit:** The circuits where the output depends on the external input and the stored information at that time, i.e., output is the function of external input and the present stored information, are called sequential circuits. O/P = Func.(External I/P and Present stored information)

The difference between the sequential and combinational circuit is that sequential circuit has memory in the form of fl ip fl op, whereas combinational circuit does not have the memory element. A general block diagram of sequential circuit is shown in Fig. 1.13.

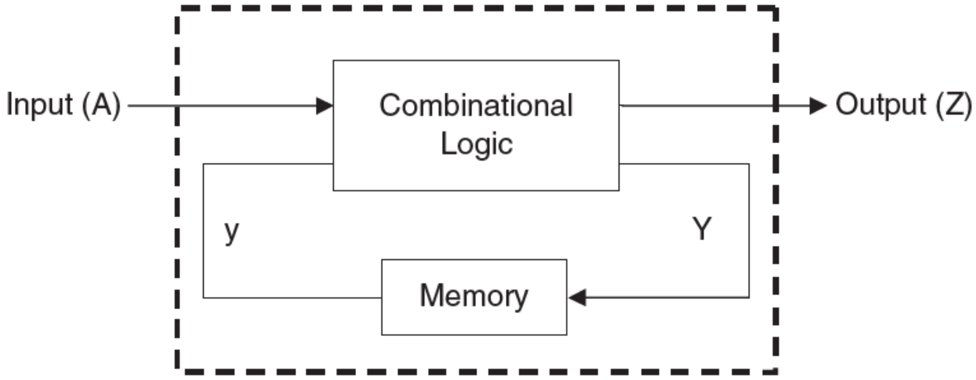


Figure 5: *Block Diagram of Sequential Circuit*

Sequential circuits fall into two classes: synchronous and asynchronous. Synchronization is usually achieved by some timing device, such as clock. A clock produces equally spaced pulses. These pulses are fed into a circuit in such a way that various operations of the circuit take place with the arrival of appropriate clock pulses. Generally, the circuits, whose operations are controlled by clock pulses, are called synchronous circuit.

The operation of an asynchronous circuit does not depend on clock pulses. The operations in an asynchronous circuit are controlled by a number of completion and initialization signals. Here, the completion of one operation is the initialization of the execution of the next consecutive operation. The following (Fig. 1.14) block diagram is that of a synchronous sequential circuit.

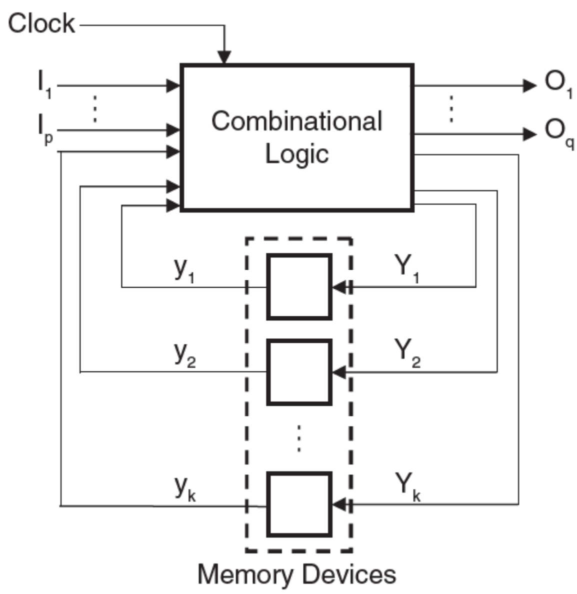


Figure 6: *Synchronous Sequential Circuit*

A synchronous sequential machine has fi nite number of inputs. If a machine has n number of input variables, the input set consists of 2n distinct inputs called input alphabet I. In the fi gure, the input alphabet is I = $\{I1, I2, \ldots\ldots, Ip\}$. The number of outputs of a synchronous sequential machine is also fi nite.

A synchronous sequential circuit can be designed by the following process:

•From the problem description, design a state table and a state diagram (whichever is fi rst applicable).

•Make the state table redundant by machine minimization. This removes some states and makes the table redundant.

•Perform state assignment by assigning the states to binary numbers. n binary numbers can assign 2^n states.

•After performing the state assignment, derive a transition table and an output table.

•Derive the transitional function and the output function from the transitional table and the output table.

•Draw the circuit diagram.

The following examples design some synchronous sequential circuit.

Example 1.6 **Design a sequential circuit which performs the following:**

A	B	O/P	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

The carry is added with the I/P's in the next clock pulse.
Solution: Let us take two input strings $X_1 = 0111$ and $X_2 = 0101$.

Here, the output at time t_i is a function of the inputs X_1 and X_2 at the time t_i and of the carry generated for the input at t_{i-1}.
O/P = func.(I/P at t_i and the carry generated for the input at t_{i-1})
Therefore, this is a sequential circuit (Fig. 1.15).

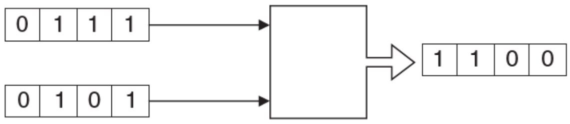


Figure 7:

If we look into the previous table, we will see two types of cases arisen there. These are

1 The producing carry '0'

2 The producing carry '1'.

We have to consider this as the O/P depends of the carry also.

Let us take the cases as states. So, we can consider two states, A for (1) and B for (2). If we construct a table for the inputs X_1 and X_2 by considering the states, it will become

	Next State, O/P(Z)			
Present State	$X_1X_2 = 00$	=01	=11	=10
A	A,0	A,1	B,0	A,1
B	A,1	B,0	B,1	B,0

This type of table is called the state table. This type of graph is called state graph or state diagram.

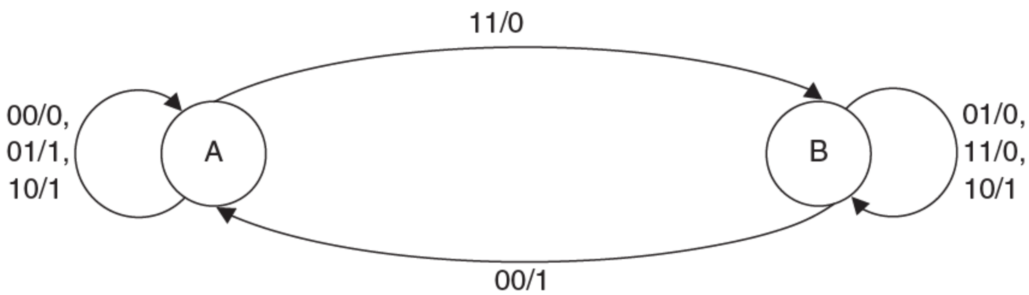


Figure 8: *State Diagram for the Sequential Circuit*

For designing a circuit, we need only '0' and '1', i.e., Boolean values. So, the states A and B must be assigned to some Boolean number. As there are only two states A and B, only one-digit Boolean value is suffi cient. Let us represent A as '0' and B as '1'.

By assigning these Boolean values to A and B, the modifi ed table becomes

Present State(y)	Next State,(Y) $X_1X_2 = 00$	=01	=11	=10	O/P(Z) =00	=01	=11	=10
0	0	0	1	0	0	1	0	1
1	1	1	1	1	1	0	1	0

The function for next state $Y = X_1X_2 + X_{1y} + X_{2y}$

The function for output $Z' = X_1'X_{2y}' + X_1'X_{2y'} + X_1X_{2y}' + X_1X_{2y} = X_1 \oplus X_2 \oplus y.$

From this function, the digital circuit is designed as denoted in Fig.1.17.

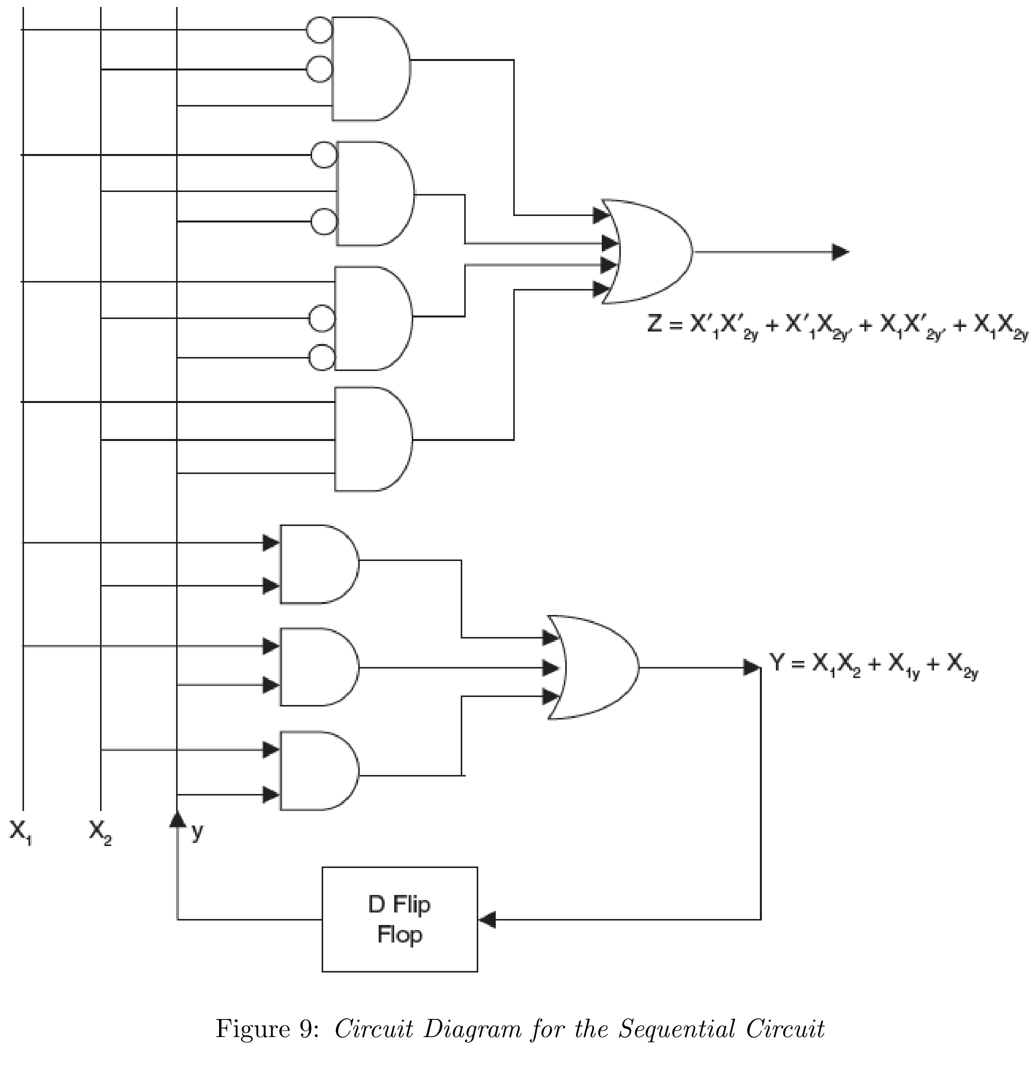


Figure 9: *Circuit Diagram for the Sequential Circuit*

This is the circuit for full binary adder.