12.5.3 Hazards

Hazard is an unwanted transient i.e. spike or glitch that occurs due to unequal path or unequal propagation delays through a combinational circuit. There are two types of hazards, viz. (i) Static hazard and (ii) Dynamic hazard as shown in Fig. 12.2.

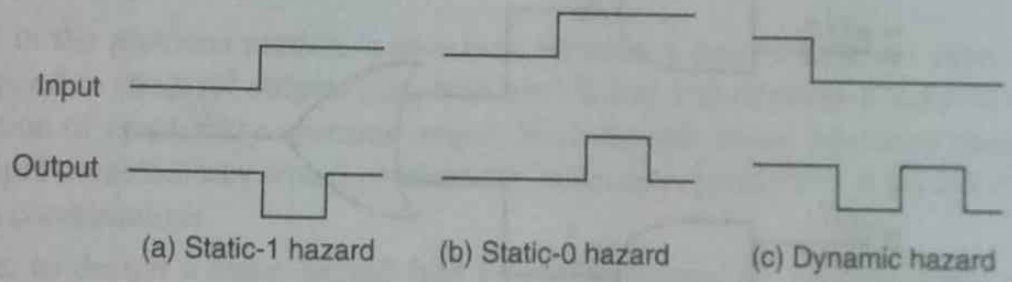


Fig. 12.2 Types of hazards

Similar to static and dynamic hazards caused by delay in combinational circuits, *essential hazards* occur in sequential circuits.

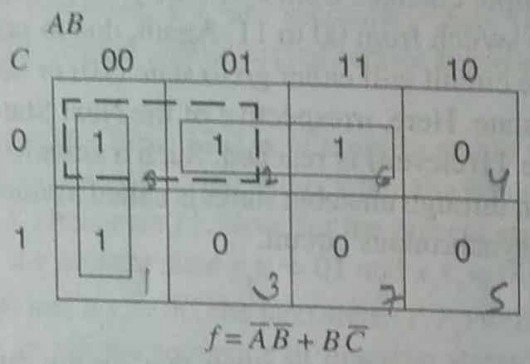
Static hazard Static hazard is a condition which results in a single momentary incorrect output due to change in a single input variable when the output is expected to remain in the same state. If the output momentarily goes to state '0' when the output is expected to remain in state '1' as per the steady state analysis, the hazard of this nature is known as *Static-1 hazard*. Similarly, if the output momentarily goes to state '1' when the output is expected to remain in state '0' as per the steady state analysis, it is known as *Static - 0 hazard*.

To understand the static hazard, consider the *K*-map for the function $f(A, B, C) = S(0, 1, 2, 6)$ and the corresponding circuit shown in Fig. 12.3 (a) and (b).

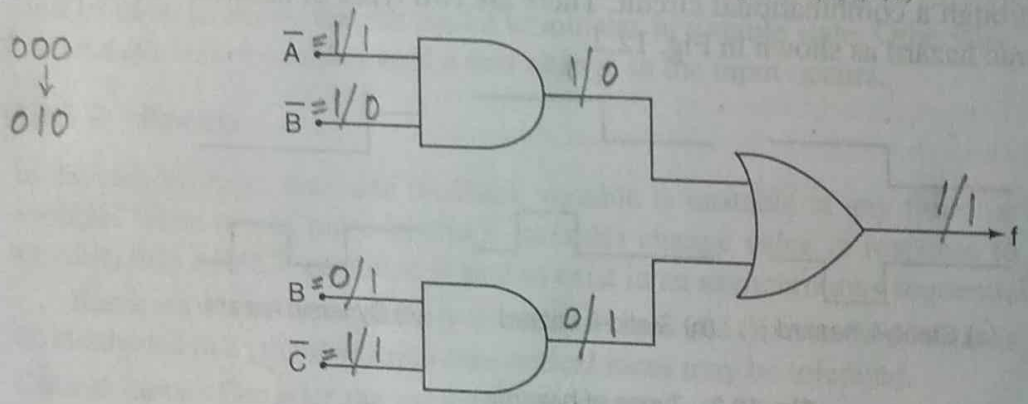
When the input $ABC = 000$, the output, $f = 1$ due to HIGH at the output of upper AND gate. Now, when the input ABC changes to 010, the output f should remain in the '1' state due to HIGH at the output of lower AND gate. In other words, due to the change in the value of input variable B, switching of high output from upper to lower AND gate takes place and hence, the output f is supposed to remain in HIGH state. But, due to unequal propagation delay, if the upper AND gate

output changes to 0 shortly before the lower AND gate output becomes '1', then, during this brief period, the output $f = 0$ momentarily. This situation is called *Static hazard.*

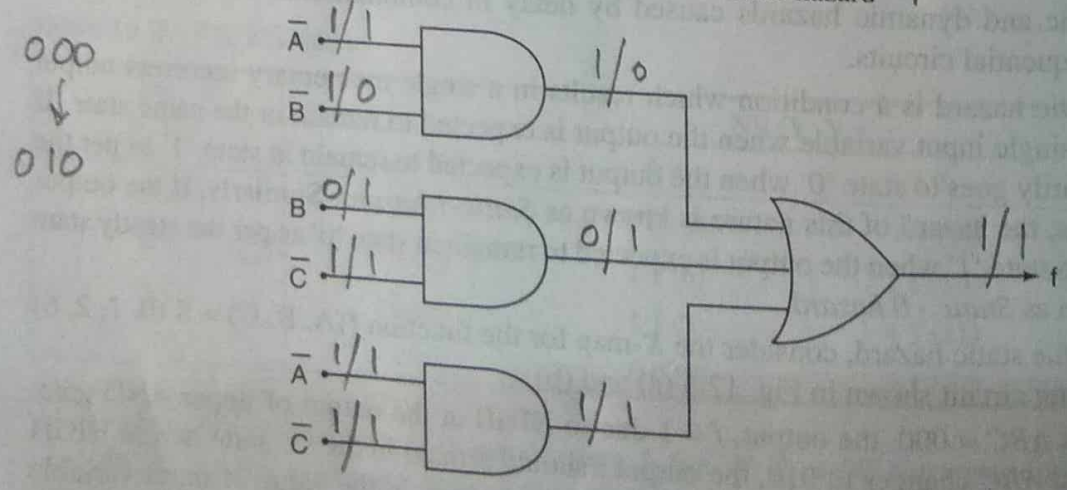
A static hazard can be removed by covering the adjacent cells with a redundant grouping that overlaps both groupings. In the *K*-map shown in Fig. 12.3(a), static hazard can be eliminated by covering adjacent cells corresponding to 000 and 010 as shown by dotted subcube. This leads to redundant grouping that overlaps both $\overline{A}B$ and $B\overline{C}$ groupings. The redundant term is $\overline{A}\,\overline{C}$ and the modified circuit is shown in Fig. 12.3(c). Now, when the input (ABC) changes from 000 to 010, the output will remain at '1' state (for both 010 and 000 inputs) because of high output at the newly added lower AND gate.



(a) K-map for the function $f = \Sigma\,(0,1,2,6)$



(b) Logic circuit with static hazard



(c) Logic circuit without static hazard

Fig. 12.3

Dynamic hazard When the output is supposed to change from 0 to 1 (or from 1 to 0), the circuit may go through three or more transients and produce more than one glitch. Such multiple glitch situations are known as *dynamic hazards*. Dynamic hazards can be eliminated by the same covering method, as explained earlier for static hazard elimination.

Essential hazard Essential hazard is a type of hazard that exists only in asynchronous sequential circuits with two or more feedbacks. Essential hazard occurs normally in toggling type circuits. It is an operational error generally caused by an excessive delay to a feedback variable in response to an input change, leading to a transition to an improper state. For example, an excessive delay through an inverter circuit in comparison to the delay associated with the feedback path may cause essential hazard. Such hazards cannot be eliminated by adding redundant gates as in static hazards.

On the other hand, the essential hazard can be eliminated by adjusting the amount of delay in the affected path. For this, each feedback loop must be designed with extra care to ensure that the delay in the feedback path is long enough compared to the delay of other signals that originate from the input terminals.

Hence, the essential hazard exists in an asynchronous sequential circuit due to a combination of delay and design specifications.

12.6 Design of Hazard Free Switching Circuits

Static, dynamic and essential hazards discussed in the previous sections can be eliminated by using different procedures discussed below.

12.6.1 Static Hazards Elimination

As discussed in the previous section, a transition between a pair of adjacent input combinations which correspond to identical outputs (i.e., both are 0's and 1's) contains a static hazard if it leads to the generation of momentary spurious output. Such hazards occur whenever there exists a pair of adjacent input combinations which produce the same output and there is no sub cube in K-map covering both combinations.

Therefore, to design a static hazard free switching circuit, all adjacent input combinations, having same output occur within some sub cube of the corresponding function. In other words, every pair of adjacent 1 cells and every pair of adjacent 0 cells in the K-map of a switching function should be covered by at least one sub cube. The expression derived from such a collection of sub cubes is called hazard free switching function and it leads to the implementation of hazard free switching circuit.

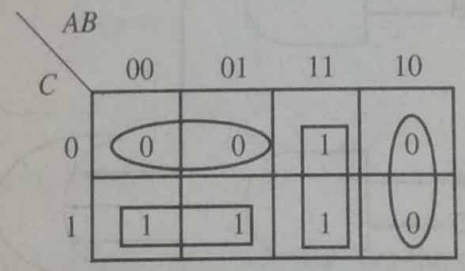


Fig. 12.4 K-map without hazard cover

Consider the switch in function $f(A, B, C) = \Sigma_m(1, 3, 6, 7)$ and the corresponding K-map in Fig. 12.4.

From the K-map shown in Fig. 12.4, the simplified sum of product (SOP) and product of sum expression can be written as:

$$f = AB + \overline{A}C \text{ and } f = (\overline{A} + B)(A + C)$$

The corresponding switching circuits are shown in Fig. 12.5(a) and (b) respectively.

In the circuit shown in Fig. 12.5(a), when the input (ABC) changes from 011 to 111 (or) 111 to 011, the output may momentarily go to '0' state due to the unequal propagation delay in the AND

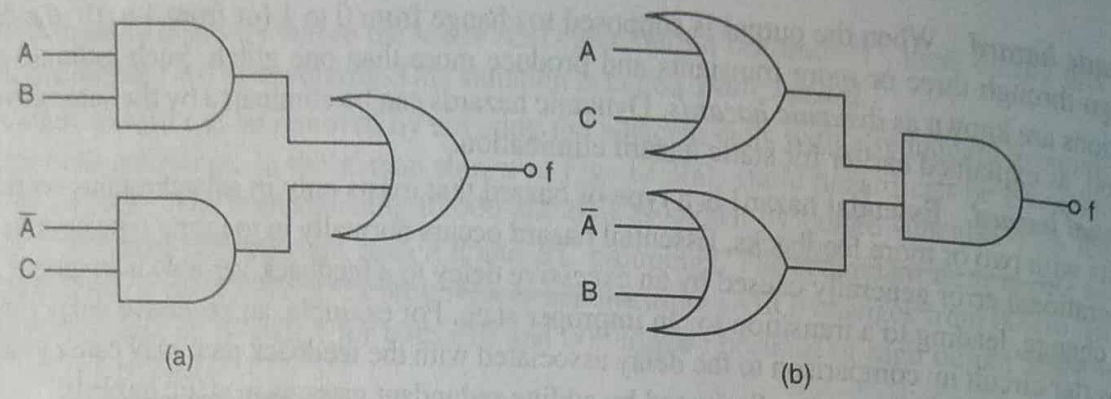


Fig. 12.5 Switching circuit with Hazards

gates, rather than remaining constant at '1' as per the function value. Similarly, in the circuit shown in Fig. 12.5(b), when the input changes from 000 to 100 or 100 to 000, the output may momentarily go to '1' state due to the unequal propagation delay in the OR gates, rather than remaining constant at '0' as per the function value. These momentarily wrong outputs are also called glitches.

Now, in order to make the switching circuits shown in Fig. 12.5 hazard free, every pair of adjacent 1 cells and every pair of adjacent '0' cells in the K-map shown in Fig. 12.4 should be covered by at least one sub cube. This results in two additional dotted sub cubes as shown in Fig. 12.6.

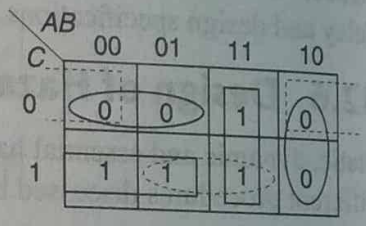


Fig. 12.6 K-map with hazard cover

The hazard free switching expression is given by

$$f = AB + \bar{A}C + BC; \quad f = (A + C)(\bar{A} + B)(B + C)$$

The corresponding hazard free switching circuits are shown in Fig.12.7(a) and (b) respectively.

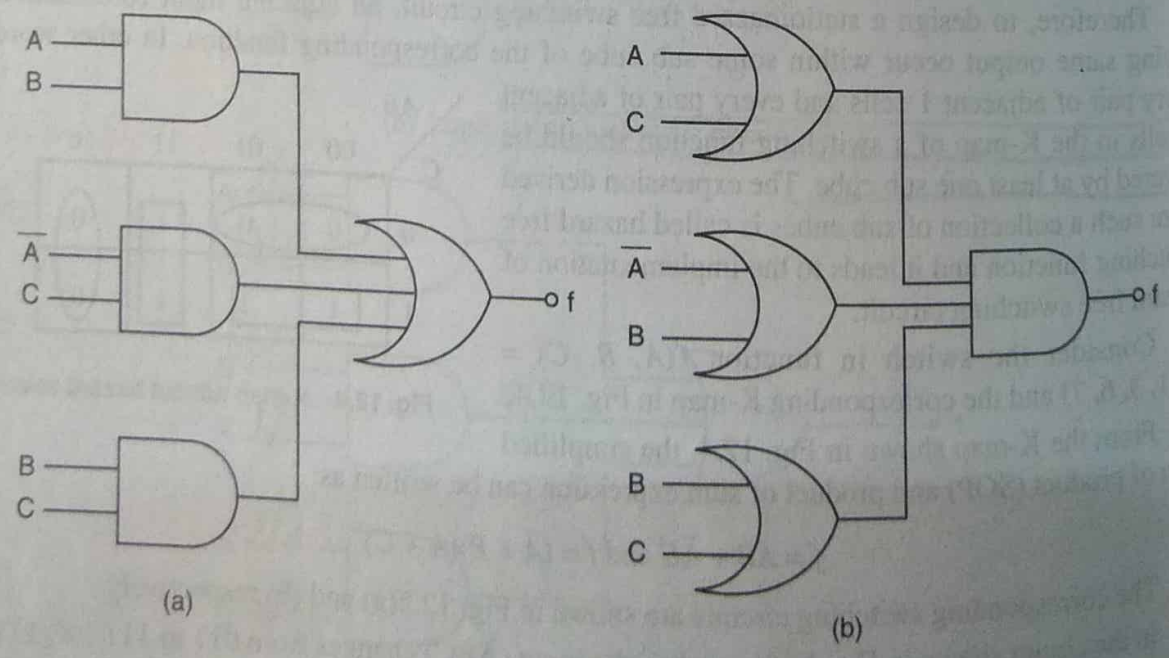


Fig. 12.7

12.6.2 Dynamic Hazards Elimination

Dynamic hazard can also be eliminated in a similar manner as that of static hazard elimination by covering every pair of 1 cells and every pair of '0' cells in the *K*-map by at least one sub cube.

12.6.3 Essential Hazards Elimination

Consider the flow table shown in Table 12.4 and the corresponding asynchronous sequential circuit shown in Fig. 12.8.

Table 12. 4 Flow table

Present State	Next State (Y_1Y_2)	
y_1y_2	$x = 0$	$x = 1$
00	00	01
01	11	01
11	11	11

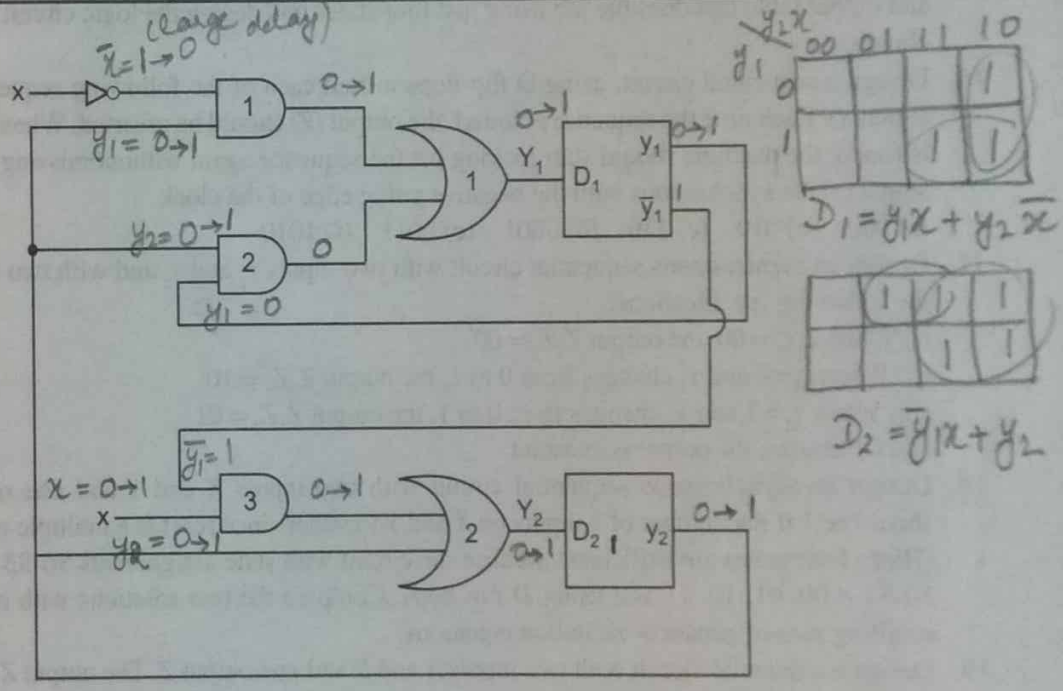


Fig. 12.8 Asynchronous sequential circuit

In the circuit shown in Fig. 12.8, y_1 depends on $\bar{x}$ while y_2 depends on x. Assume that the delay associated with the NOT gate is very large compared to the combined propagation delay of flip-flops and gates, and the present state (PS) of the circuit is 00 (i.e., $y_1y_2 = 00$). When the input (x) changes from 0 to 1, then Y_2 becomes 1 and thereby y_1 also becomes 1. This y_2 is fed back to the input of the first flip-flop through AND gate 1. If the change in input (x) is still not propagated through the NOT gate, then Y_1 becomes 1 and thereby y_1 also becomes 1. Therefore, the next state Y_1Y_2 is 11 instead of 01 as given in the flow table. This phenomenon, known as essential hazard, occurs due to the result of the input (x) change and y_2 changing faster than the propagation of the input through the NOT gate.

In the circuit shown in Fig. 12.8, the essential hazard can be eliminated by reducing the delay in the NOT gate or by increasing the delay in the feedback path of y_2 to AND gate 1.