Hazards of Switching Circuits

08.11.2020

—

Datta Sai Jaideep-19BCN7035| Hara Teja Yadam-19BCE7433

Vangala Bhanu Prakash-19BCD7117 | V.Charan Sri Harsha-19BCE7229

Sai Goutham Kumar-19BCE7651

Slot-D+TD

Guided By – Dr. Anil Negi

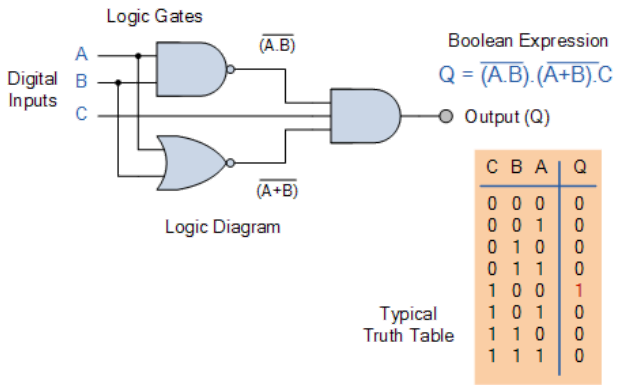
Abstract

This article is concerned with the detection and elimination of hazards from switching circuits, especially combinational logic circuits.It has explained different types of hazards possible and a method for their detection and elimination. The method involves simple manipulation of Boolean expression or K-Maps corresponding to logic networks.It not only detects the hazards in the circuit but also specifies the type of hazards in the circuit.Further, it can determine the terms the absence of which from the expression cause the hazards. Thus, the hazard can be eliminated by including these missing terms in the expression.The procedure has been given first before solving and problems. There are three methods(Spikes method,Hoffman's method,and FORTRAN's method) for detection and elimination of hazards but we named only two methods(Hoffman's method,FORTRAN'S method) and used only one method i.e. Hoffman's method.Two related results have been included.These results have been analyzed.

Objective

All the circuits designed using logic gates are considered as **Digital Circuits**. These circuits deal with binary inputs and outputs such as 0's and 1's. These Digital Circuits are also called **Switching Circuits** because in these circuits the voltage levels are assumed to switch from one value to another instantaneously. So, it is commonly assumed that, if the voltage value lies between 0 and 5, the binary value is 0. If voltage is greater than 5 the binary value is considered as 1.

Combinational Logic Circuits are memoryless digital logic circuits whose output at any instant in time depends only on the combination of its inputs.These are made up of any of the 7 logic gates(AND,OR,NOT,NOR,NAND,XOR,XNOR).To design a circuit we need an boolean expression, truth table and design of the circuit.The following is a logic circuit complete representation example.



C	B	A	Q
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

The common combinational logic circuits are MUX'S, DE-MUX'S, Encoders, Decoders, Adder………

Sequential Logic Circuits are the digital logic circuits whose output depends on past inputs and present inputs.The Sequential Logic Circuits are also designed using logic gates.These logic circuits are Flip-Flops and Latches.They have feedback-loops.

[D]

Hazards and types of hazards

Hazards in any *s*ystem are obviously an undesirable effect caused by either a deficiency in the system or external.influences. Hazards/glitches occur due to unequal propagation delays along different paths in a combinational circuit.

Spurious pulses occurring on the output of combinational switching circuits during an input transition are called combinational hazards.

In digital logic hazards are usually referred to in one of three ways:
1.Static Hazards:When one input variable changes, the output changes momentarily when it shouldn't.This particular type of hazard is usually due to a NOT gate within the logic.
There are 2 types of static hazards:

1)Static 1 hazard: The value goes from 1 to 0 instead of staying 1.
2)Static 0 Hazard: The value goes from 0 to 1 instead of staying 0.

2.Dynamic Hazards:A dynamic hazard is the possibility of an output changing more than once as a result of a single input change.

3.Function Hazards:Function hazards are non-solvable hazards which occur when more than one input variable changes at the same time.

This article's basic objective is to help the designer in detecting such hazards and to show a hazard elimination method.

[F]

Methodology

This article shows two hazard detection-elimination methods, they are FORTRAN and Hoffman's method.Both of these methods eliminate Static hazards only.

- Hoffman's method
  - This is a direct and simple method.

- It uses K-Map for detection and elimination of hazards.
- As the input variables increase the difficulty of detecting hazards also increases.
- We can only use this method to eliminate static hazards
- A static-one-hazard can be detected by observing the products used for the function on a K-map.
- If any two logically adjacent cells with a '1' output are not covered by a common product or implicant, a static-one-hazard can occur when a single input change moves from one cell to the other.
- A static-one-hazard can also be detected algebraically.
- If only the distributive law is used to convert the SOP form to a POS form and the resulting POS form contains the sum of a variable and its complement, then this is an indication of a static 1 hazard.
- A static 1 hazard can be prevented by adding a product term so that all pairs of logically adjacent cells with a '1' output have at least one common product covering them.
- A static zero hazard can be detected by observing the sums used for the function on a k-map.
- If any two logically adjacent cells with a '0' output are not covered by a common sum, a static hazard can occur when a single input change moves from one cell to the other.
- A static 0 hazard can also be detected algebraically.
- If only the distributive law is used to convert the POS form to a SOP form and the resulting SOP form contains the product of a variable and its complement, then this is an indication of a static 0 hazard.
- A static 0 hazard can be prevented by adding sum terms so that all pairs of logically adjacent cells with a '0' output have at least one common sum covering them.
- This can be accomplished by using all prime implicants in the POS form rather than using a minimized POS form.

[A,B]

- FORTRAN’S method
  - This method involves only simple manipulation of the Boolean expression corresponding to the logic network.
  - It not only detects the existence of hazards and determines the variable sets with respect to which the hazards exist, but also, for each of these sets, pin-points the subcubes within which each input transition involving

exactly the changes of all variables in the set produces the hazardous output.

- ➢ Further, it can determine the terms the absence of which from the expression cause the hazards
- ➢ Thus, the hazard can be eliminated by including these missing terms in the expression.
- ➢ For a better understanding of this method visit this reference article [C].
- ➢ Note: This is a detailed method with theorems and proofs.
- ➢ This method is a very powerful method but it is lengthy to explain and state in this article.

Dynamic hazards often occur in larger logic circuits where there are different routes to the output (from the input). If each route has a different delay, then it quickly becomes clear that there is the potential for changing output values that differ from the required expected output.
It should be noted that if all static hazards have been eliminated from a circuit, then dynamic hazards cannot occur.

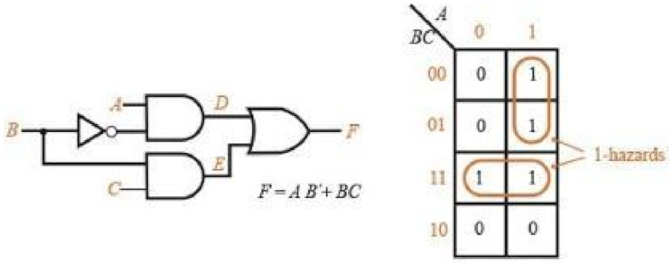
Hazards such as Function hazards can not be logically eliminated as the problem lies with actual specification of the circuit. The only real way to avoid such problems is to restrict the changing of input variables so that only one input should change at any given time.

Modeling of the problem

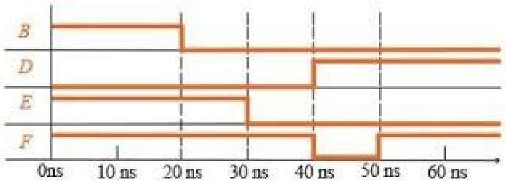
As we know all types of hazards, let’s know each hazard in detail.

- Static hazard
  - This particular type of hazard is usually due to a NOT gate within the logic.There are 2 types of static hazards:
    - Static 1 hazard: The value goes from 1 to 0 instead of staying 1.
    - Static 0 Hazard: The value goes from 0 to 1 instead of staying 0.

For example:
Let us consider an imperfect circuit that suffers from a delay in the physical logic elements.



(a) Circuit with a static 1-hazard



(b) Timing chart

If we first look at the starting diagram, it is clear that if no delays were to occur, then the circuit would function normally. However since this isn't a perfect circuit, an error occurs when the input changes from 111 to 101.When B changes from 1 to 0.During this change, there is a 10ns delay between both the inputs from B.

Due a small delay in the inputs, the output F shall have a glitch. F should always be 1 but from 40 ns to 50 ns the value of F is 0.
This is Static 1 hazard.The opposite happens in a static 0 hazard.

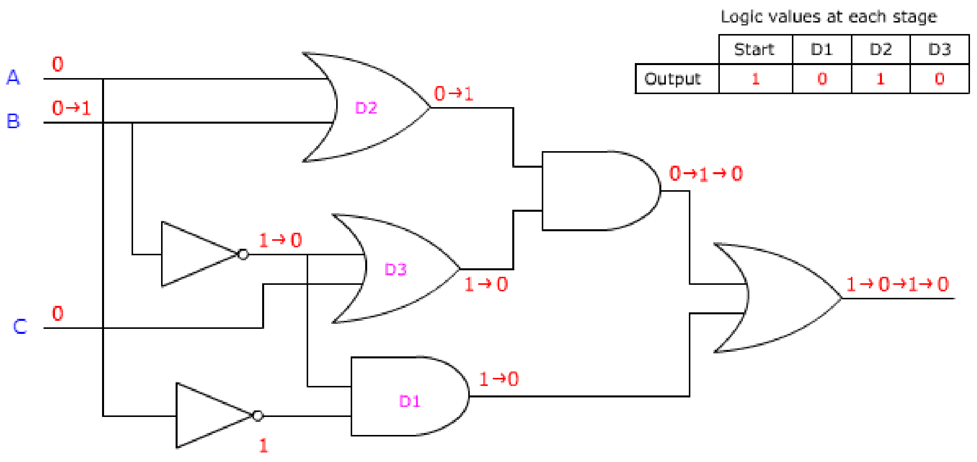
Now, let’s take the algebraic expression for a better understanding.
F= AB'+BC = (B'A+B)(B'A+C)=(B'+B)(A+B)(B'+C)(A+C)

After expanding into such a form,F ,has a sum (B'+B). Due to variable delays in different branches of the circuit, this sum may be effectively zero for very short time intervals. If all other sums are '1', then the output should stay '1', but since they are all ANDed with (B '+B), its momentary zero would cause an output glitch which would be a static 1 hazard.

A SOP form with AND-OR implementation can never have a static '0' hazard.
A POS form with OR-AND implementation can never have a static '1' hazard.

- Dynamic hazard
    - Dynamic hazards take a more complex method to resolve. There is no dynamic hazard iff there are no static hazards
    - From the below circuit, the delay in D1 is less than the delay in D2, and the delay in D3 is greater than the delay in D2.

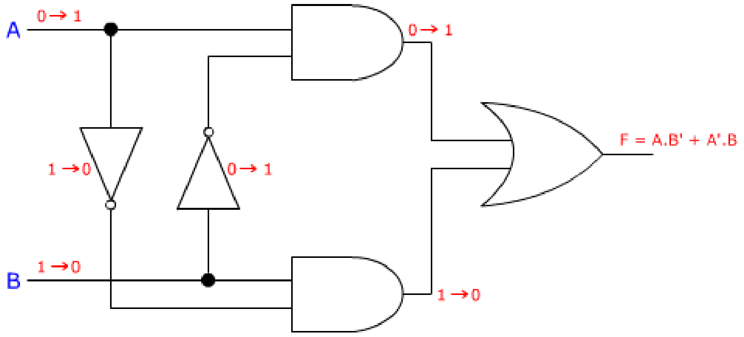
- Lets say that input B changes from 0 to 1.
- The NOT gate shown changes from 1 to 0. Remembering that D1 is the shortest of the three delays, the AND gate shown will also change from 1 to 0.
- Because the other delays (D2, D3) are longer than D1, and because the other gates have no delay (or negligible delay), our output changes from 1 to 0. (The first effect of the hazard).
- The next delay to occur is D2.
- So the OR gate shown implements the change (0 OR 1 = 1).
- Now the AND gate has only seen the change in one of its inputs due to the delay D3 being longer than the other delays. So momentarily, both inputs are logic 1 which means the AND gates output goes to logic 1 hence changing the output of the entire circuit.
- The next delay to occur is D3.
- This output rests at 0.
- So output changes from 1 to 0 to 1 to 0 instead of being static value 1.



Logic values at each stage

	Start	D1	D2	D3
Output	1	0	1	0

- Function hazard



  -
  - Let us set up the initial state of our circuit. A = 1, B = 0. Now let's say there is a delay in the NOT gate marked (X). The inputs now change

simultaneously so that A = 0 and B = 1 (remember in an equally delayed circuit or a perfect circuit, the circuit output would match the specification).

- If we observe what the circuit should do, and do not change the output of the NOT gate X (this simulates a delay in gate X), it should be clear that the output of the circuit changes. Now we change the output of NOT gate X and the circuit goes back to the proper state

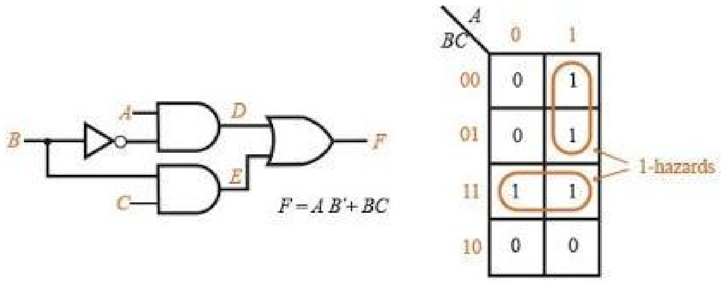
[E,G]

Solution of the problem

Hoffman's method(For Static Hazards and Dynamic Hazards)

Note:The following problems are two-level implementation circuits.

Let's take the previous example from “modeling of the problem” and make a hazard free circuit.



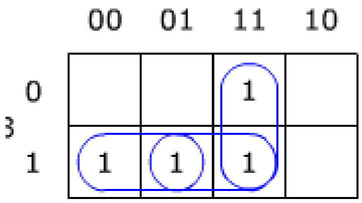
(a) Circuit with a static 1-hazard

Intial F=AB'+BC

Based on Hoffman's method, This is a Static one hazard since the boolean expression is SOP and it's a AND-OR implementation.

[B]

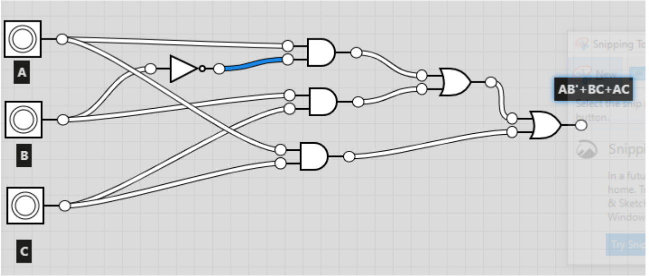
So, the missing term in the boolean expression is A.C



Hazard Eliminated

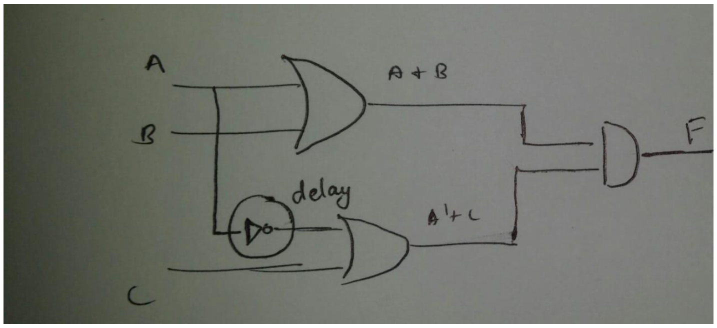
So F=B.C + B'.A + A.C

Finally,This is a Hazard free circuit of F

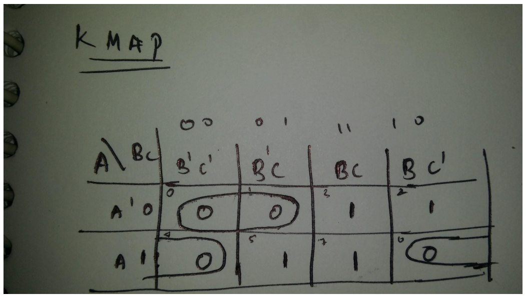


Let’s take another example,

Let the given circuit be,



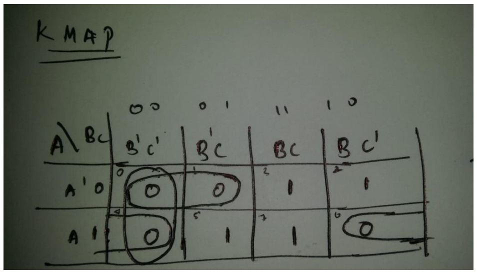
So, F=(A+B)(A’+C) and it’s K-Map is



KMAP

A\BC	00 B'C'	01 B'C	11 BC	10 BC'
A' 0	0	0	1	1
A 1	0	1	1	0

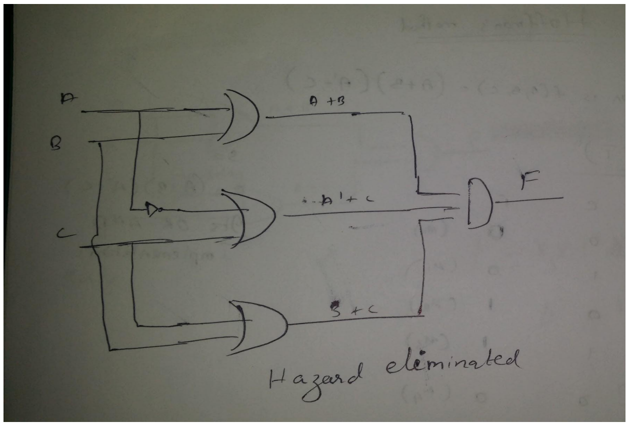
Based on Hoffman’s method, This is a Static zero hazard since the boolean expression is POS and it’s a OR-AND implementation.The cells M0 and M4 are adjacent to each other and are not encircled together.



KMAP

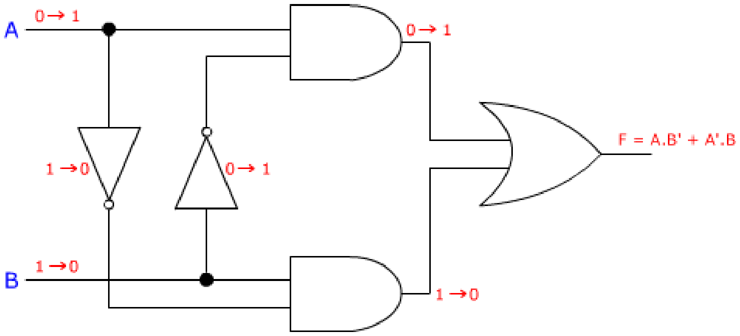
A\BC	00 B'C'	01 B'C	11 BC	10 BC'
A' 0	0	0	1	1
A 1	0	1	1	0

So, the missing term in the boolean expression is B+C



So F=(A+B)(A’+C)(B+C)
Finally,This is a Hazard free circuit of F

Function Hazards
The simplest example is the exclusive-or function(from “modeling of the problem”).



The most effective way to solve this hazard would be to carefully design the PCB so that delays are all equal, or at least match the delays on each path. i.e. Delay of A's path = Delay of B's path. Yet adding more gates to the circuit by the same methods as described in dynamic and static hazards will not work.

Analysis of the result

In this article we used Hoffman’s Method for detecting a hazard and eliminating it. Hoffman’s method is short to process and easy to understand.

From the above results, we can see that any static hazard can be detected and eliminated easily.We add a new gate to the circuit in order to eliminate the hazard.Similarly,we add an additional term to the boolean expression based on the uncovered adjacent cells of 1’s or 0’s.

If you carefully look at both the examples, the functions taken are the same in both the examples.Both K-Maps are the same.When a function is converted from SOP to POS then, static one hazard gets converted to static zero hazard and vice versa. Similarly, if an AND-OR implemented circuit is converted to OR-AND circuit, then static one hazard changes to static zero hazard and vice versa.

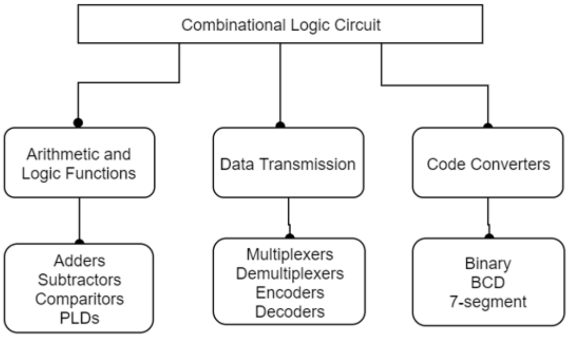
But for a dynamic hazard, we need to clear the static hazard in the two-level circuit part and the other two level circuit part till we cover the whole circuit.It's a time taking process in case of dynamic hazard.

As a final note on dynamic hazards, it should be noted that if all static hazards have been eliminated from a circuit, then dynamic hazards cannot occur.

Applications

Combinational logic circuits are the basic building blocks of digital systems. They have an extensive variety of applications. They are used in

- Digital measuring techniques
- Digital processing
- Industrial processing
- Computers
- Calculators



- Digital Communication

Combinational logic circuits as arithmetic and logic circuits have applications as

- Programmable logic devices
- Adders & Subtractors

Combinational logic circuits as data transmission circuits have applications as

- Parallel to serial convertors
- Data routing
- Serial to parallel converters
- Bit Compression
- ADC and DAC
- Encoders & Decoders

Combinational logic circuits as code converter circuits have applications as

- Binary to gray code conversion
- Excess-3 to BCD
- Seven segment code

A comparison with synchronous circuits suggests four opportunities for the application of asynchronous circuits: high performance, low power, improved noise and electromagnetic compatibility (EMC) properties, and a natural match with heterogeneous system timing.

Conclusion

This article successfully dealt with hazards in combinational circuits and a method for detection and elimination of such hazards and also their application in real-time. Hazards occurring in combinational circuits may cause a temporary false-output value. When this condition occurs in asynchronous sequential circuits, it may result in a transition to a wrong stable state.These hazards consume a lot of energy and they may decrease the performance of a circuit.

So, by avoiding these hazards we increase the efficiency of a circuit.

We expect that the method stated and explained for detection and elimination of hazards to be helpful to a circuit designer.

References

We referred to many articles, papers, E-books and websites before concluding this article.

A. EICHELBERGER, E. (1965), Hazard detection in combinational and sequential switching circuits, IBM J. Res. Develop. 9, 90-99. **(Article)**

B. HUFFMAN, D. (1957), The design and use of hazard-free switching networks, J. Assoc. Comput. Mach. 4, 47-62. **(Research paper)**

C. A Logic Hazard Detection and Elimination Method by LEVENTE L., SANTANU DAS and HENRY Y. H. CHUANG.**(Article)**

D. Digital Electronics volume 1_Combinational logic circuits-wiley-ISTE(2016).**(Text book)**

E. E. Eichelberger, "Hazard detection in combinational and sequential switching circuits," in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, null, 1964 pp. 111-120.**(Text Book)**

F. http://www.ee.surrey.ac.uk/Projects/Labview/Sequential/Course/02-Hazards/hazards.htm

G. https://www.allaboutcircuits.com/technical-articles/hazards-in-combinational-logic/

H. https://www.watelectronics.com/what-is-a-combinational-logic-circuit-types-and-applications/

I. Supplemental Material for ELEN 021, Logic Design**(c) Copyright 1997 Sally L, Wood (Article)**.

J. Applications of Asynchronous Circuits C. H. (KEES) VAN BERKEL, MEMBER, IEEE, MARK B. JOSEPHS, AND STEVEN M. NOWICK.**(Article)**