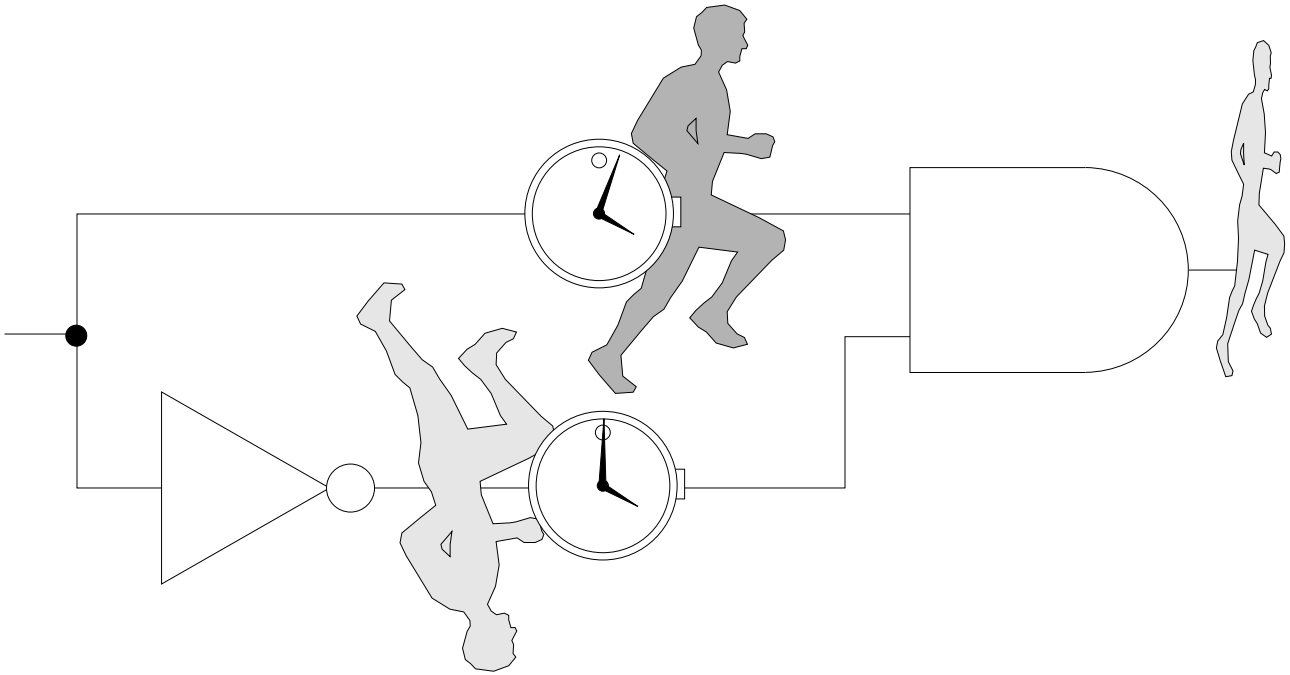
Glitches and Hazards in Digital Circuits



"*After a moment you change your mind*"

Hazards

Glitches and a Hazards

A *glitch* is a fast "spike" usually unwanted.

A *hazard* in a circuit may produce a glitch.
if the propagation delays are unbalanced.

The Classification of Hazards by the Glitch They May Produce

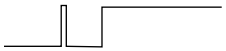
***static-zero hazard*;**

signal is static at zero, glitch rises. 

***static-one hazard*;**

signal is one, glitch falls. 

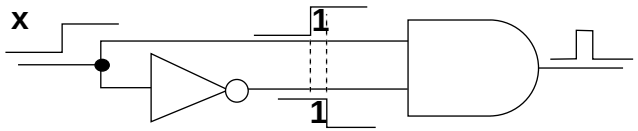
***dynamic hazard*;**

signal is changing, up  **or down** 

The Two Basic Static-Hazard Circuits

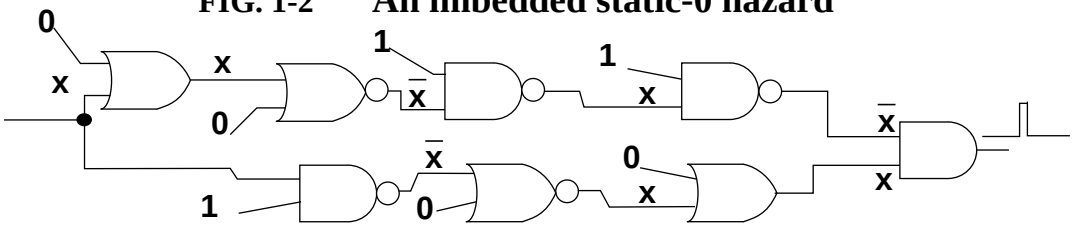
Basic Static-Zero Hazard Circuit

FIG. 1-1 **Basic static-0**



Any circuit with a static-0 hazard must reduce to the equivalent circuit of FIG. 1-1, if other variables are set to appropriate constants.

FIG. 1-2 **An imbedded static-0 hazard**

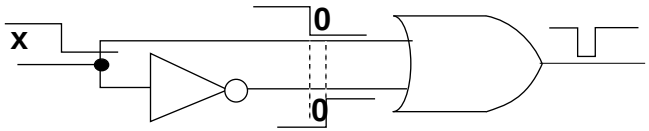


Static-zero Hazard's Characteristics

- **Two parallel paths for x.**
- **One inverted.**
- **Reconverge at an AND gate.**

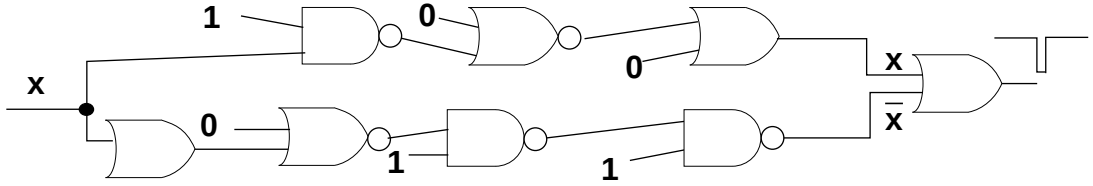
Basic Static-One Hazard Circuit

FIG. 1-3 **Basic static-1 hazard circuit**



Any circuit with a static-1 hazard must reduce to the equivalent circuit of FIG. 1-3

t FIG. 1-4 **An imbedded static-1 hazard**



Static-One Hazard's Characteristics

- **Two parallel paths for x.**
- **One inverted.**
- **Reconverge at an OR gate.**

The Two Basic Dynamic-Hazard Circuits

Basic Dynamic Hazard Circuits

A static hazard with an extra gate for the static level change.

Three parallel paths, one containing a static hazard.

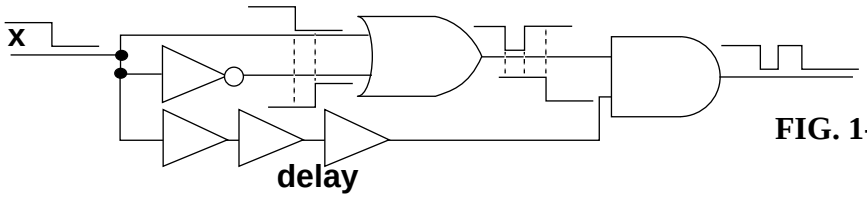


FIG. 1-5 **The basic dynamic hazard circuit with an imbedded static-1 hazard.**

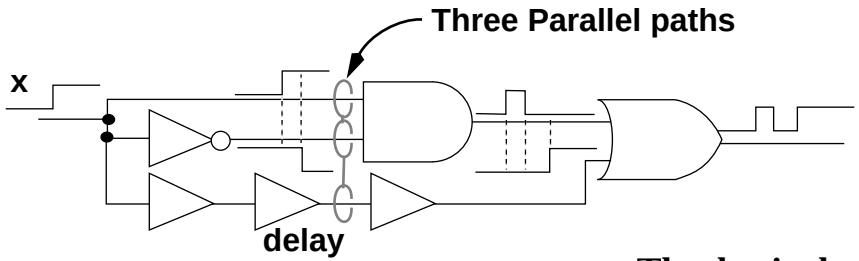


FIG. 1-6 **The basic dynamic hazard circuit with an imbedded static-0 hazard.**

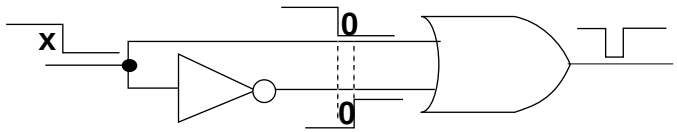
Note that a dynamic hazard always has three parallel paths.

Adding Delay to Hazards

Adding delay can remove hazards, if one has good control of propagation delays.

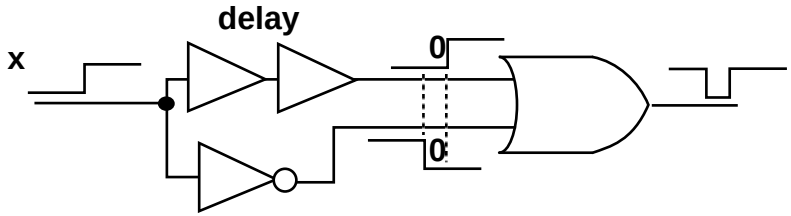
The original circuit with the delay in the inverter.

FIG. 1-7 Basic static-1 hazard circuit from FIG. 1-3.
Note the hazard appears on the falling edge of x.



- **Adding an equal delay in the other path removes the falling-edge glitch.**
- **Adding too much delay will make the glitch appear on the rising edge.**

FIG. 1-8 Adding delay, moves the glitch from $\overline{x}$ to x.
To kill the glitch balance the delays exactly, if you can!

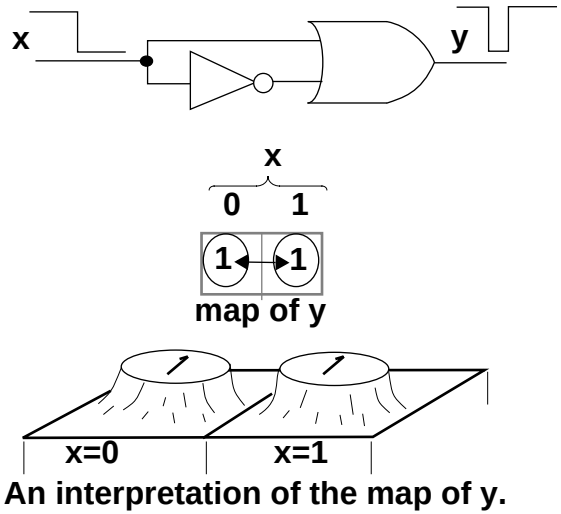


At the silicon layout level, one might balance delays closely enough to suppress the glitch.
With standard cells and field-programmable arrays, balancing is harder.

But see "Absorption of Glitches by Gates" on page 53.

Hazards on a Karnaugh Map

Adjacent but nonoverlapping circles on the map are hazards.



An interpretation of the map of y.

FIG. 1-9 **Map of a static-1 hazard.**
On the Σ of Π map, each OR gate input is a separate circle.

Standing on top of a hill gives a "1." Changing hills causes a "0" glitch as one crosses the valley.

The ⟷ shows the hazard.

A Static-1 Hazards on a Map

Σ of Π maps can only show static-1 hazards, not static-0 or dynamic hazard.

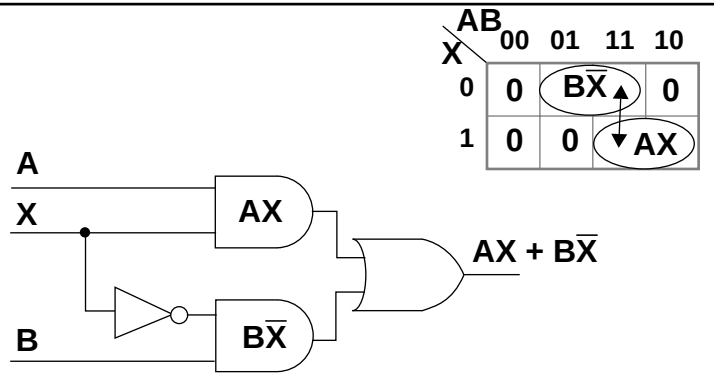


FIG. 1-10

AND gates have been added to the hazard.

The hazard is still the inverter and the OR gate.

The hazard appears only when A = 1, B = 1.
Then signal X travels right through the ANDs.

Masking a Hazard.

To mask static-1 hazards add a gate that stays high across the ↔ transition.
This gate is logically redundant.

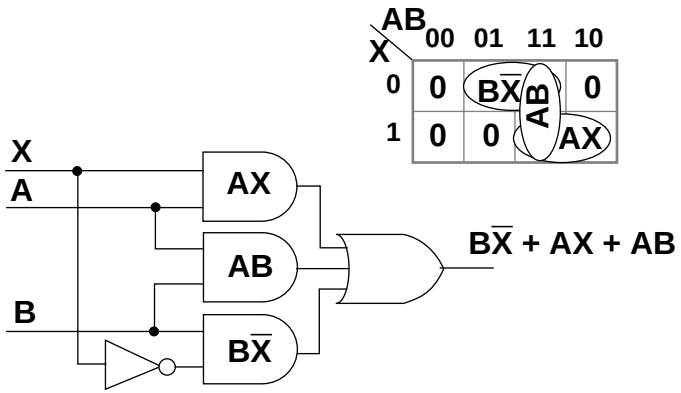


FIG. 1-11

The equation
F = BX̄ + AX
has redundant term AB added
F = BX̄ + AX + AB

This fills the valley between terms BX̄ and AX.

DeMorgan's General Theorem (Review)

Simple form of DeMorgan's Theorems

$$A \cdot B = \overline{\overline{A} + \overline{B}} \qquad \overline{A \cdot B} = \overline{A} + \overline{B} \qquad \overline{D + E} = \overline{D} \cdot \overline{E} \qquad D + E = \overline{\overline{D} \cdot \overline{E}}$$

The general form

$$\overline{F(A,B,C,\ldots +,\cdot,)} = F(\overline{A},\overline{B},\overline{C},\ldots,\cdot,+,)$$

$$F = [\overline{A} \cdot B \cdot C + D \cdot (A \cdot B + C)] \cdot \overline{A}$$

a) Bracket all groups of ANDs

$$F = \{[\{\overline{A} \cdot B \cdot C\} + \{D \cdot (\{A \cdot B\} + C)\}] \cdot \overline{A}\}$$

b) Change AND to OR and OR to AND
Clean brackets

$$\{[\{\overline{A} + B + C\} \cdot \{D + (\{A + B\} \cdot C)\}] + \overline{A}\}$$

$$\{\overline{A} + B + C\} \cdot \{D + \{A + B\} \cdot C\} + \overline{A}$$

c) Invert all variables

$$\overline{F} = \{A + \overline{B} + \overline{C}\} \cdot \{\overline{D} + \{\overline{A} + \overline{B}\} \cdot \overline{C}\} + A$$

Examples

$$F = \overline{A} \cdot B \cdot C \Rightarrow \{\overline{A} \cdot B \cdot C\} \Rightarrow \overline{F} = \{A + \overline{B} + \overline{C}\}$$

$$F = \overline{A} \cdot B \cdot C + A \cdot \overline{B} \Rightarrow \{\overline{A} \cdot B \cdot C\} + \{A \cdot \overline{B}\} \Rightarrow \overline{F} = \{A + \overline{B} + \overline{C}\} \cdot \{\overline{A} + B\}$$

$$F = \overline{A} \cdot B \cdot (C + \overline{A \cdot B}) \Rightarrow \{\overline{A} \cdot B\} \cdot (C + \{\overline{A \cdot B}\}) \Rightarrow \overline{F} = \{A + \overline{B}\} + (\overline{C} \cdot \{\overline{A + B}\})$$

Product of Sum (Π of Σ) Maps (Review)

Take the usual Σ of Π map for F.

$F = \overline{X}B + XA$

X \ AB	00	01	11	10
0	0	1	1	0
1	0	0	1	1

Circling the "1s" on the map gives an implementation of F.
Circling the "0s" gives an implementation of $\overline{F}$.
Make a map of $\overline{F}$ by circling the zeros

Extract the formula for $\overline{F}$.

$\overline{F} = \overline{X}\cdot\overline{B} + X\cdot\overline{A}$

X \ AB	00	01	11	10
0	0	1	1	0
1	0	0	1	1

Apply generalized DeMorgan to the formula for $\overline{F}$
This gives a formula for F.

$\overline{F} = \overline{X}\cdot\overline{B} + X\cdot\overline{A}$

$\overline{F} = \{\overline{X}\cdot\overline{B}\} + \{X\cdot\overline{A}\}$

$\{\overline{X} + \overline{B}\}\cdot\{X + \overline{A}\}$

$F = \{X + B\}\cdot\{\overline{X} + A\}$

The result is the Π of Σ expression for F.
It was obtained by circling "0s", not "1s."

$F = \{X + B\}\cdot\{\overline{X} + A\}$

X \ AB	00	01	11	10
0	0	1	1	0
1	0	0	1	1

Showing a Static-0 Hazard on a Π of Σ Map

Π of Σ maps can only show static-0 hazards, not static-1 or dynamic hazards

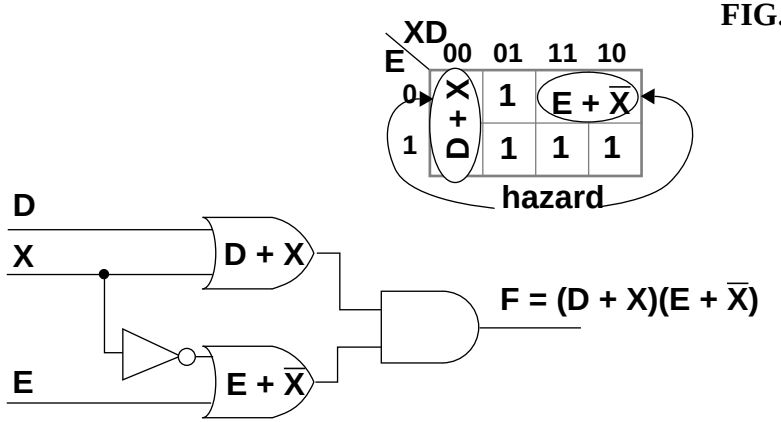


FIG. 1-12

Get Π of Σ implementation.

Plot "0s" not "1s."

Connect letters with OR, not AND.

$$D + X,\ E + \overline{X}$$

Gaps between adjacent circles show static-0 hazards.

Consider wrap around

Masking a Static-0 Hazard on a Π of Σ Map

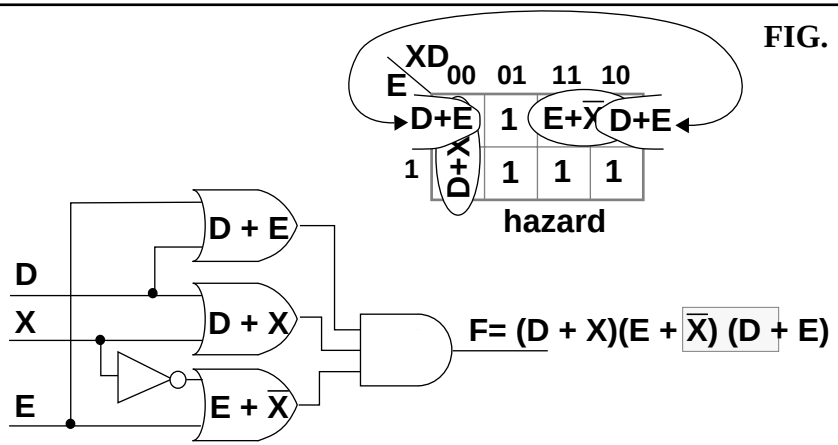


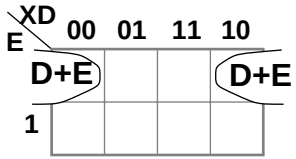
FIG. 1-13

To mask the static-0 hazard:

AND F with a term which stays 0 across the hazard.

The hazard is X,D,E = 1,0,0 to 0,0,0.

The term which stays 0 across the gap is X,D,E = -,0,0, or (D + E).



Algebra and Hazards.

In hazards, delays temporarily make $x = \bar{x}$.
In algebra with hazards, treat x and $\bar{x}$ as separate variables.

For work with hazards, do not use:

Complementing	Simplification	Multiplying Out	Consensus
$x\bar{x} = 0$	$x + \bar{x}y = x + y$	$(x + y)(\bar{x} + z) = xz + \bar{x}y$	$xy + yz + \bar{x}z = xy + \bar{x}z$
$x + \bar{x} = 1$	$(\bar{x} + y) = xy$	$xy + \bar{x}z = (x + z)(\bar{x} + y)$	$(x + y)(y + z)(\bar{x} + z) = (x+y)(\bar{x} + z)$
	$\bar{x}y + xy = y$		
	$(\bar{x} + y)(x + y) = y$		

For work with dynamic hazards, avoid the distributive law. (Factoring)

The distributive laws can create dynamic hazards from static hazards, even a masked one.

They will not remove or create *static* hazards.

The Distributive Laws

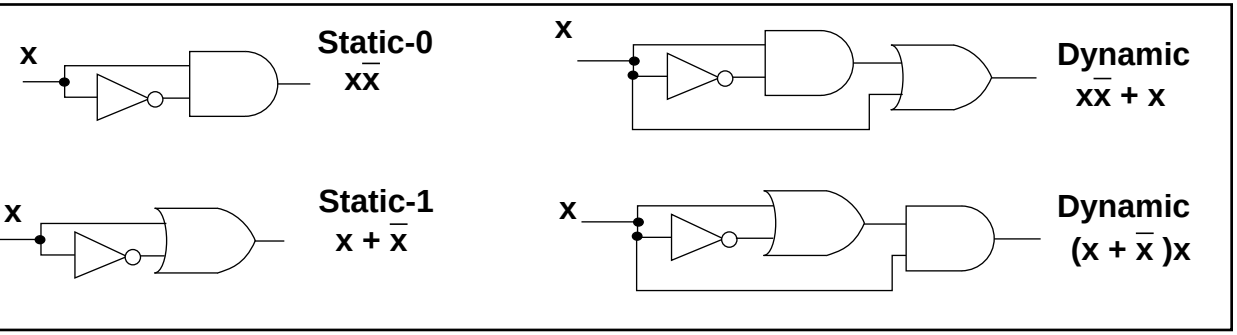
$$x(y + z) = xy + xz$$

$$x + yz = (x + y)(x + z)$$

Algebra of Hazards

The basic forms for hazards and their equations.
x and $\bar{x}$ are treated as separate variables.
If a circuit has a hazard, the equation of the circuit will reduce to one of these forms.

FIG. 1-14



An Example

Below, a hazard in x must reduce to a basic hazard circuit when c=1 or when c=0.

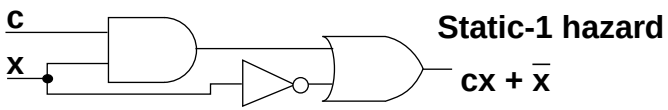


FIG. 1-15 **Circuit equation is**

$cx + \bar{x}$

when c = 1 get

$1x + \bar{x} = x + \bar{x}$

The hazard is "exposed"

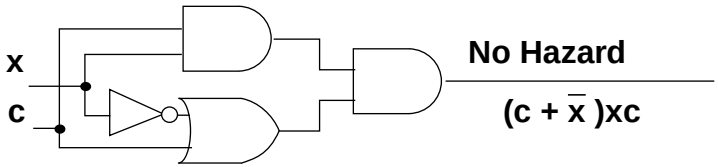


FIG. 1-16 **Circuit equation is**

$(c + \bar{x})xc$

When c = 1, get $(1 + \bar{x})x1 = x$

When c = 0, get $(0 + \bar{x})x0 = 0$

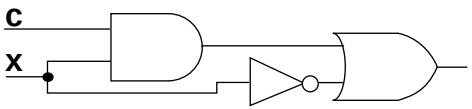
There are no hazard

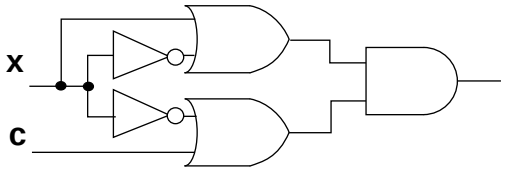
The Distributive Law and Hazards

**The distributive laws can change 2 parallel paths into 3,
and thus may create a dynamic hazard from a static one.
They can create a dynamic hazard from a masked hazard (FIG. 1-17 bottom).**

FIG. 1-17 The distributive law changing static hazards to dynamic hazards.

ORIGINAL CIRCUIT **CIRCUIT AFTER APPLYING DISTRIBUTIVE LAW**





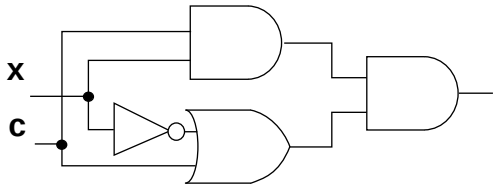
Static-1 hazard

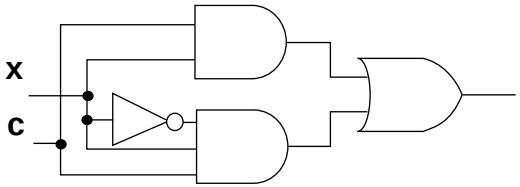
$xc + \bar{x} = (c + \bar{x})(x + \bar{x})$

When c=1 $x + \bar{x}$

$(1 + \bar{x})(x + \bar{x}) = x + \bar{x}$ Static-1 hazard when c = 1;

$(0 + \bar{x})(x + \bar{x}) = \bar{x}(x + \bar{x})$ Dynamic hazard when c = 0





Masked Hazard

$xc(c + \bar{x}) = xc + xc\bar{x}$

When c=1 $x1(1 + \bar{x}) = x$

$x1 + x1\bar{x} = x + x\bar{x}$ Dynamic hazard when c = 1

Locating and Repairing Hazards Algebraically

- This method will find all hazards
  static-1, static-0, and dynamic.
- The circuits do not need to be Σ of Π or Π of Σ.
  $F = (a + b + c\bar{b})de + (\bar{e}a + \bar{d}b)\bar{c}$
- Much faster than maps;
  It will find all types of hazards on one pass.
- It can also find how to mask them.

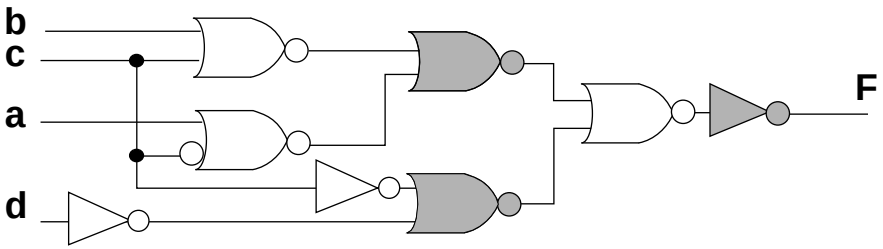
Method

1. Remove confusing extended overbars.
2. Find which variables cannot have hazards.
3. Check for hazards in each variable.
   Select one variable for checking
   make other variables 1 or 0 to bring out hazard
4. Find masking terms if needed.

1. $\overline{(A + B)} + \overline{A \cdot C} \Rightarrow \bar{A} \cdot \bar{B} + (\bar{A} + \bar{C})$
2. Need both X and $\bar{X}$
3. $X \cdot \bar{X}$, $X + \bar{X}$, $X + X \cdot \bar{X}$
   $AX + (B\bar{X} + C)$
   $1X + (1\bar{X} + 0)$
   $\bar{X} + \bar{X}$

Example

Find All The Hazards In F.



DeMorgan's Laws in Graphical Form (Review)

FIG. 1-18 Equivalent graphical forms for AND, OR, NAND and NOR, using DeMorgan's theorem.

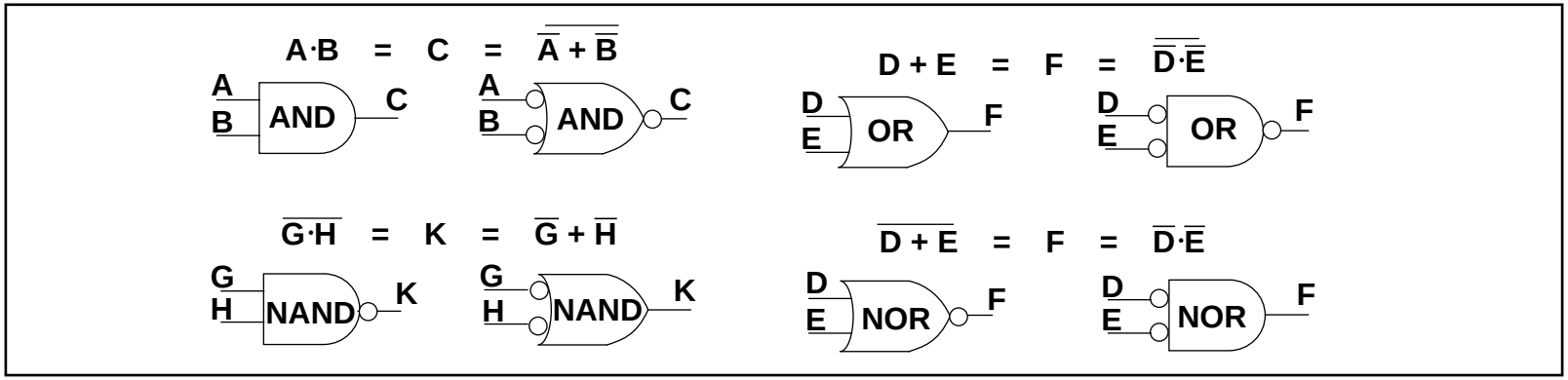
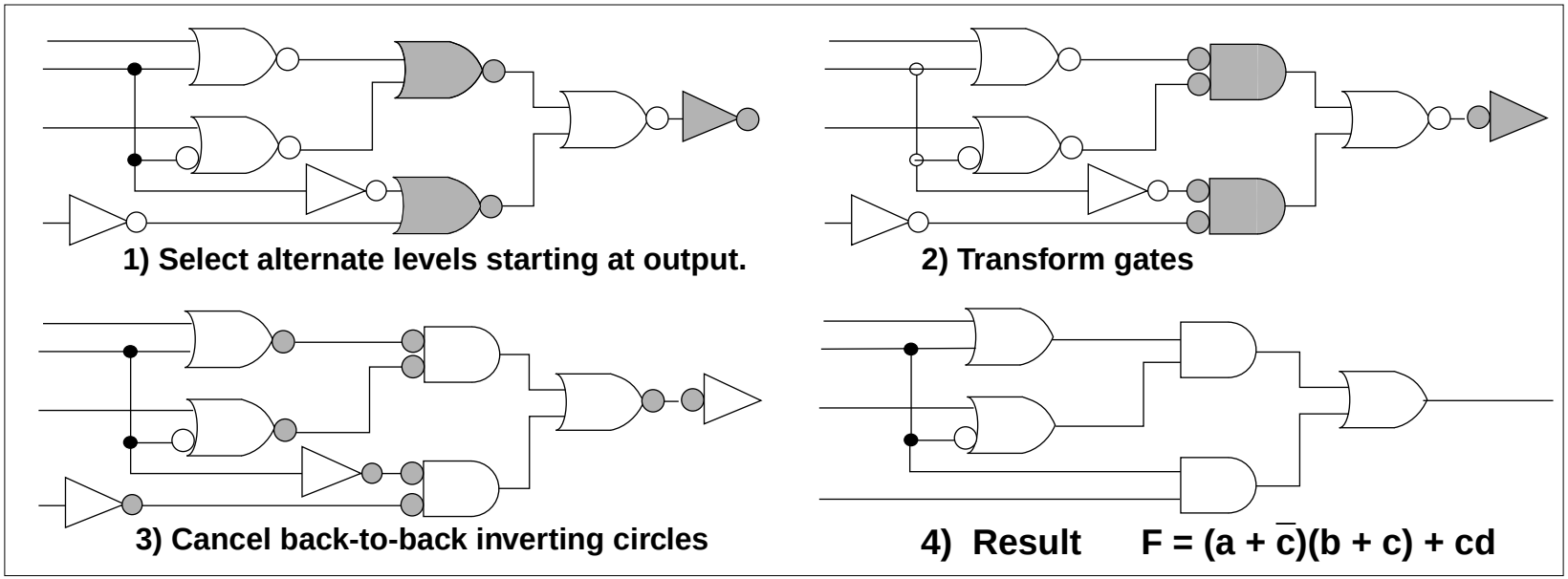


FIG. 1-19 Removing confusing inversions.



Estimating which variables might have hazards.

A hazard, has two paths which reconverge in an AND or OR gate.
One path must have an even number of inversions,
and the other path must have an odd number.
One need only check for hazards in variables which have such paths.

Checking a circuit for potentially hazardous paths.

FIG. 1-20

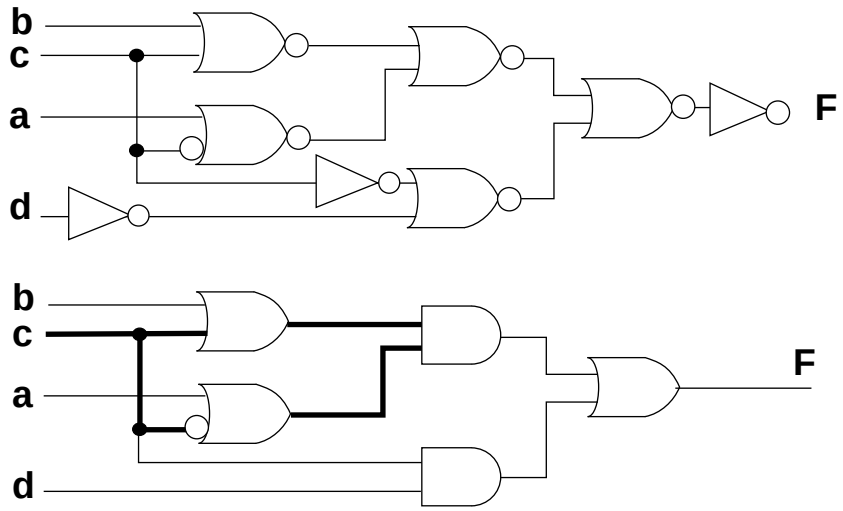
To tell what variable need to be checked for hazards:

Remove most inverting circles using DeMorgan's laws.

Check for reconvergent paths one of which is inverting.

Only variable c has such a path; hence only c can have a hazard.

Can also tell from expression. Only c has both c and $\bar{c}$ type



$$F = (a + \bar{c})(b + c) + cd$$

Locating Hazards From the Circuit Equation

1. **Take the circuit equation.**
   $$F = (a + \bar{c})(b + c) + cd$$
2. **Note which variables do not have both $\bar{x}$ and x.**
   In this case a, b and d. => only c needs to be checked.
3. **Substitute 0s and 1s for the other variables. Try to get forms like:**
   $$c\bar{c}, \quad c + \bar{c}, \quad c\bar{c} + c, \quad (c + \bar{c})c.$$

a	b	c	d	$(a + \bar{c})$	$(b + c)$	$+ cd$	f	Type of hazard.
0	0	c	0	$(0 + \bar{c})$	$(0 + c)$	$+c0$	$c\bar{c}$	Static-0
0	0	c	1	$0 + \bar{c}$	$0 + c$	$c1$	$\bar{c}c+c$	Dynamic
0	1	c	1	$0 + \bar{c}$	$1 + c$	$c1$	$\bar{c}+c$	Static-1
0	1	c	0	$0 + \bar{c}$	$1 + c$	$c0$	$\bar{c}$	
1	0	c	0	$1 + \bar{c}$	$0 + c$	$c0$	c	
1	0	c	1	$1 + \bar{c}$	$0 + c$	$c1$	$c+c$	
1	1	c	1	$1 + \bar{c}$	$1 + c$	$c1$	$1+c$	
1	1	c	0	$1 + \bar{c}$	$1 + c$	$c0$	1	

Static-0 hazard when **a,b,d = 0,0,0,**
Dynamic hazard when **a,b,d = 0,0,1,**
Static-1 hazard when **a,b,d = 0,1,1.**

Same Example With More Organization and Less Writing

Equation.

$$F = (a + \bar{c})(b + c) + cd$$

Note only c can have a hazard.

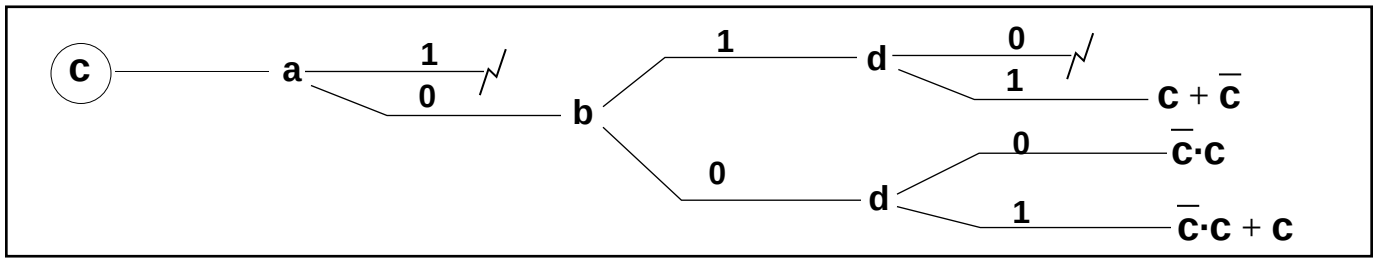
Select c to to be the variable that changes.

Sequentially substitute 1 or 0 for the other letters.

A little thought shows a must be 0, else $a + \bar{c} = 1$ => no $\bar{c}$ => no hazard

Set a = 0 first.

a b c d	**$(a + \bar{c})(b + c) + cd$**	a,b,c,d.
a b **c** d	$(\mathbf{a} + \bar{\mathbf{c}})(b + \mathbf{c}) + \mathbf{c}d$	*a must be 0, or no $\bar{c}$.*
0 b **c** d	$(\mathbf{0} + \bar{\mathbf{c}})(b + \mathbf{c}) + \mathbf{c}d = \bar{\mathbf{c}}(b + \mathbf{c}) + \mathbf{c}d$	
0 1 **c** d	*try b = 1* $= \bar{\mathbf{c}}(1 + \mathbf{c}) + \mathbf{c}d = \bar{\mathbf{c}} + \mathbf{c}d$	
0 1 **c** 1	*d must be 1* $= \bar{\mathbf{c}} + \mathbf{c}1 = \mathbf{c} + \bar{\mathbf{c}}$	*Static-1 for* 0 1 **c** 1
0 0 **c** d	*try b = 0* $= \bar{\mathbf{c}}(0 + \mathbf{c}) + \mathbf{c}d = \bar{\mathbf{c}}\cdot\mathbf{c} + \mathbf{c}d$	
0 0 **c** 0	*d may be 0* $= \bar{\mathbf{c}}\cdot\mathbf{c} + \mathbf{c}0 = \bar{\mathbf{c}}\cdot\mathbf{c}$	*Static-0 for* 0 0 **c** 0
0 0 **c** 1	*or d may be 1* $= \bar{\mathbf{c}}\cdot\mathbf{c} + \mathbf{c}1 = \bar{\mathbf{c}}\cdot\mathbf{c} + \mathbf{c}$	*Dynamic for* 0 0 **c** 1



Locating Hazards; More Complex Exmple

Equation. $F = [(a + bc)d + (\bar{b} + a\bar{c})\bar{d}\,]\bar{a}b$

Note which variables do not have both $\bar{x}$ and x.
Here all variables need further checking.

Select one letter to to be the variable that changes.

Sequentially (one at a time) substitute 1 or 0 for the other letters.
A little thought helps select which letter to make 1 (or 0) first.

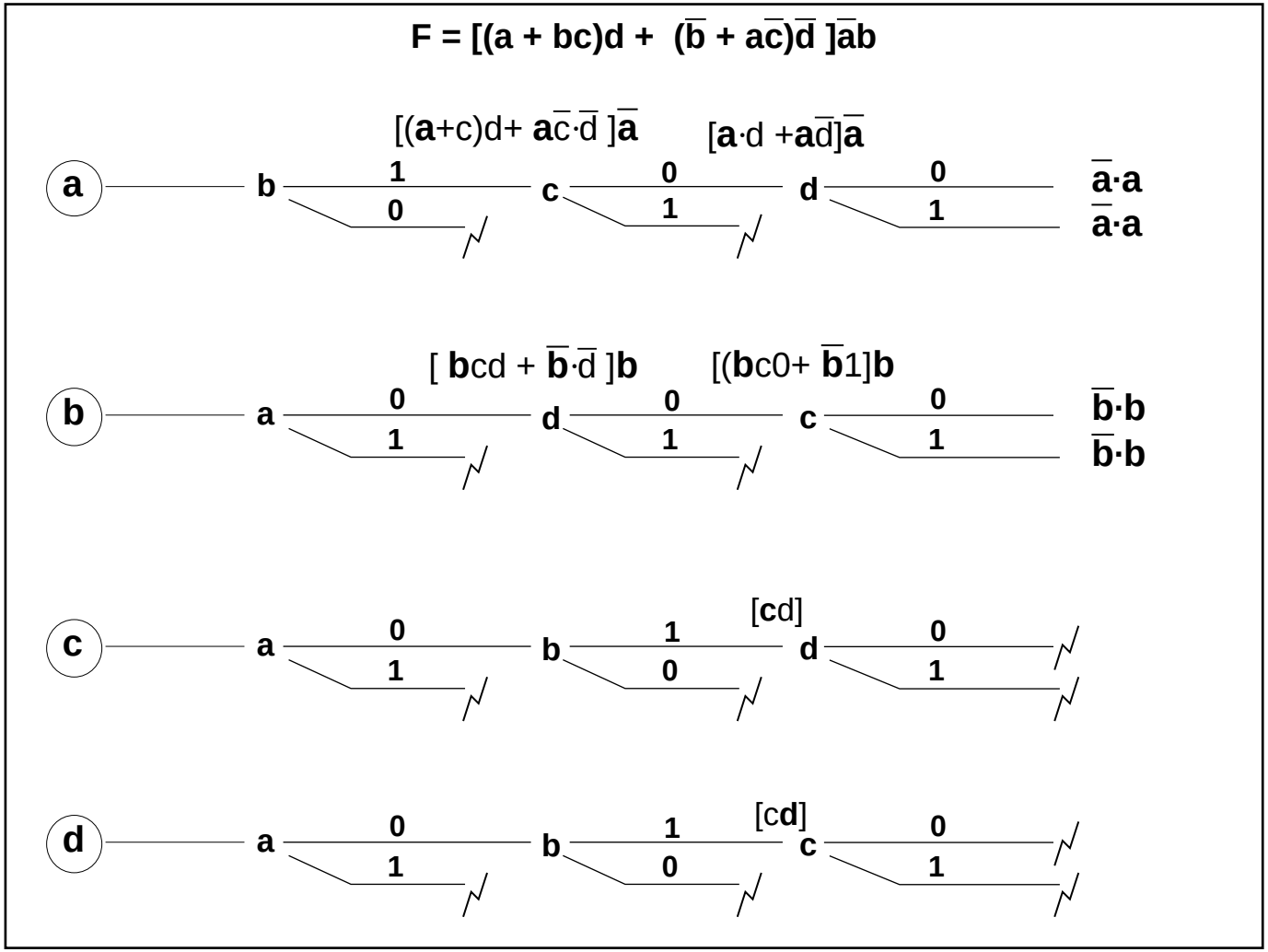
abcd $[(a + bc)d + (\bar{b} + a\bar{c})\bar{d}\,]\bar{a}b$

---

abcd	$[\,(\mathbf{a} + bc)d + (\bar{b} + \mathbf{a}\bar{c})\bar{d}]\bar{\mathbf{a}}b$	*b must be 1, or* $F \equiv 0$
a1cd	$[\,(\mathbf{a} + 1c)d + (0 + \mathbf{a}\bar{c})\bar{d}]\bar{\mathbf{a}}1 = [(\mathbf{a}+c)d + \mathbf{a}\bar{c}\cdot\bar{d}\,]\bar{\mathbf{a}}$	*c, must be 0, or no* ***a***
a1 0d	*set c = 0* $= [(\mathbf{a}+0)d + \mathbf{a}1\bar{d}\,]\bar{\mathbf{a}} = [\mathbf{a}\cdot d + \mathbf{a}\bar{d}]\bar{\mathbf{a}}$	
a100	*d may be 0* $= [\mathbf{a}\cdot 0 + \mathbf{a}1]\bar{\mathbf{a}} = \mathbf{a}\cdot\bar{\mathbf{a}}$	*Static-0 for* **a**100
a101	*or d may be 1* $= [\mathbf{a}\cdot 1 + a0]\bar{\mathbf{a}} = \mathbf{a}\cdot\bar{\mathbf{a}}$	*Static-0 for* **a**101

a**b**cd	$[\,(a + \mathbf{b}c)d + (\bar{\mathbf{b}} + a\bar{c})\bar{d}]\bar{a}\mathbf{b}$	$\bar{a}$ *must be 1, or* $F \equiv 0$
0**b**cd	$[\,(0 + \mathbf{b}c)d + (\bar{\mathbf{b}} + 0\bar{c})\bar{d}]1\mathbf{b} = [\,\mathbf{b}cd + \bar{\mathbf{b}}\cdot\bar{d}\,]\mathbf{b}$	*d must be 0 or no* $\bar{b}$
0**b**c0	$[(\mathbf{b}c0 + \bar{\mathbf{b}}1]\mathbf{b} = [\bar{\mathbf{b}}]\mathbf{b}$	*Static-0 for* 0 **b** - 0
		This hazard is independent of c.
ab**c**d	$[\,(a + b\mathbf{c})d + (\bar{b} + a\bar{\mathbf{c}})\bar{d}]\bar{a}b =$	$\bar{a}$*,b must be 1,1, or* $F \equiv 0$
01**c**d	$[\,(0 + 1\mathbf{c})d + (0 + 0\bar{\mathbf{c}})\bar{d}]11 = [\,\mathbf{c}d\,]$	*There is no c, hence no hazard*
abc**d**	$[\,(a + bc)\mathbf{d} + (\bar{b} + a\bar{c})\mathbf{d}]\bar{a}b =$	$\bar{a}$*,b must be 1,1, or* $F \equiv 0$
01c**d**	$[\,(0 + 1c)\mathbf{d} + (0 + 0c)\mathbf{d}]11 = [\,c\mathbf{d}\,]$	*There is no d, hence no hazard*

Graph of the previous hazard search



Locating Hazards; Example three

Equation. $F = \bar{y}(\bar{e} + \bar{b}\cdot\bar{c}) + b(c\cdot e + \bar{a}\cdot\bar{c}\cdot\bar{e}) + a\cdot c\cdot e\cdot y$

Select one letter, call it X, to to be the variable that changes.
Variables which do not have both forms, $\bar{X}$ and X, have no haxards.

If only one X, set all symbols ANDing X to 1. $+ a\cdot c\cdot e\cdot \mathbf{X}$ *set* a,c,e *to1,1,1 or no* **X**
If only one X, set symbols ANDing X at 1, and ORing X at 0. $\bar{y}(\bar{e} + \mathbf{X}\cdot\bar{c})$ *set* $\bar{c},\bar{e},\bar{y}$ *to 1,0,1 or no* ***X****.*
If all Xs have a common factor, fix factor at 1. $b(c\cdot\mathbf{X} + \bar{a}\cdot\bar{c}\cdot\bar{\mathbf{X}}) + a\cdot c\cdot\mathbf{X}\cdot y$ *c must be 1 or no* ***X***

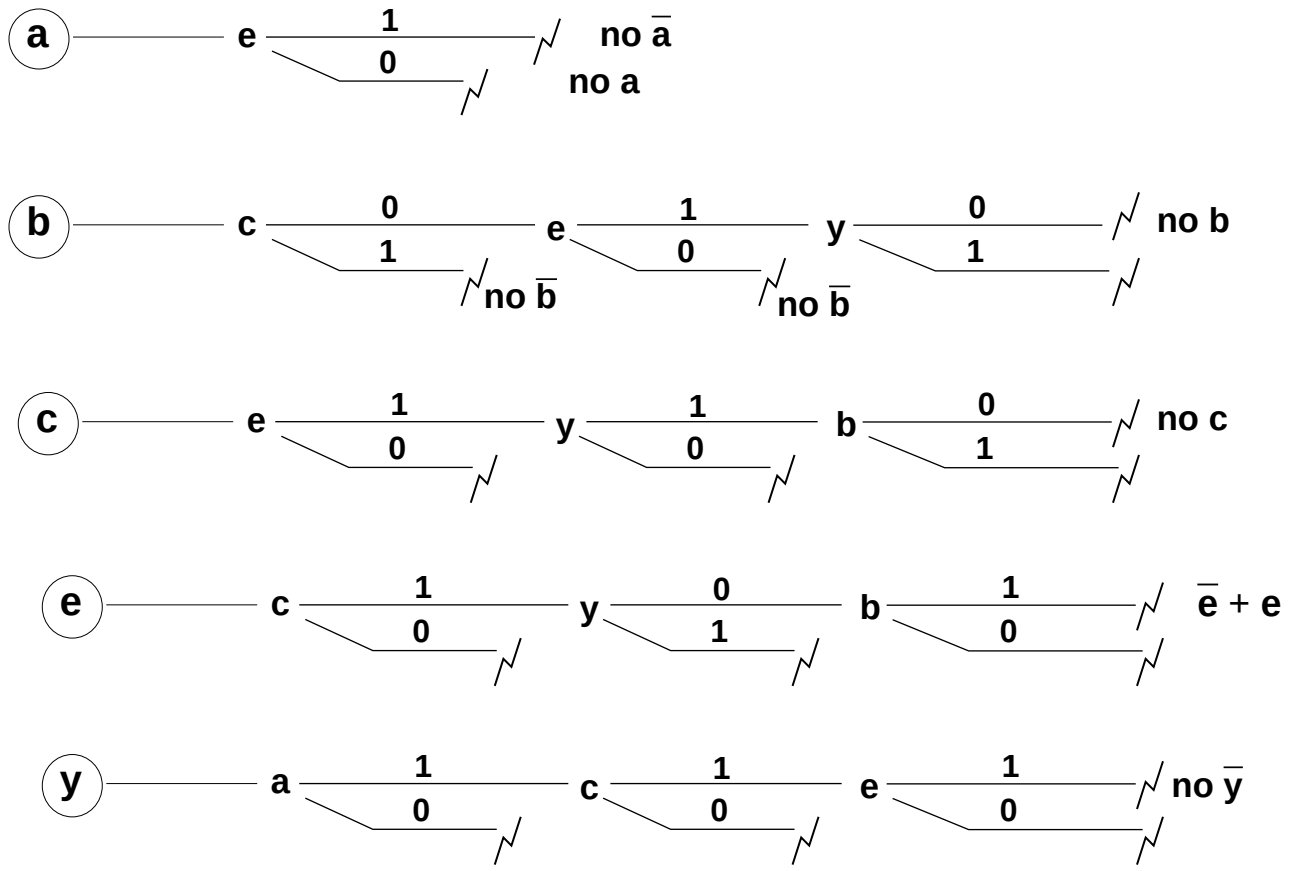
abce y $\bar{y}(\bar{e} + \bar{b}\cdot\bar{c}) + b(c\cdot e + \bar{a}\cdot\bar{c}\cdot\bar{e}) + a\cdot c\cdot e\cdot y$

---

abce y	$\bar{y}(\bar{e} + \bar{b}\cdot\bar{c}) + b(c\cdot e + \mathbf{\bar{a}}\cdot\bar{c}\cdot\bar{e}) + \mathbf{a}\cdot c\cdot e\cdot y$	*c must be 1, or no a*
ab1ey	$\bar{y}(\bar{e} + \bar{b}\cdot 0) + b(1\cdot e + \mathbf{\bar{a}}0\bar{e}) + \mathbf{a}1e\cdot y = y\cdot\bar{e} + b\cdot e + \mathbf{a}\cdot e\cdot y$	*no* $\bar{a}$ *=> no hazards in a*
a**b**ce y	$\bar{y}(\bar{e} + \mathbf{\bar{b}}\cdot\bar{c}) + \mathbf{b}(c\cdot e + \bar{a}\cdot\bar{c}\cdot\bar{e}) + a\cdot c\cdot e\cdot y$	$\bar{c},\bar{e},\bar{y}$ *must be 1,0,1 or no* $\bar{b}$*.*
a**b**010	$1(0 + \mathbf{\bar{b}}\cdot 1) + \mathbf{b}(0\cdot 1 + \bar{a}\cdot 1\cdot 0) + a\cdot 0\cdot 1\cdot 0 = \mathbf{\bar{b}} + 0$	*no b => no hazards in b.*
ab**c**e y	$\bar{y}(\bar{e} + \bar{b}\cdot\mathbf{\bar{c}}) + b(\mathbf{c}\cdot e + \bar{a}\cdot\mathbf{\bar{c}}\cdot\bar{e}) + a\cdot\mathbf{c}\cdot e\cdot y$	*e must be 1 or no c.*
ab **c**1y	$\bar{y}(0 + \bar{b}\cdot\mathbf{\bar{c}}) + b(\mathbf{c}\cdot 1 + \bar{a}\cdot\mathbf{\bar{c}}\cdot 0) + a\cdot\mathbf{c}\cdot 1\cdot y = \bar{y}\cdot\bar{b}\cdot\mathbf{\bar{c}} + b\cdot\mathbf{c} + a\cdot\mathbf{c}\cdot y$	$\bar{y}\cdot\bar{b}$ *must be 1,1 or no* $\bar{c}$
a 0**c**01	$= 1\cdot 1\cdot\mathbf{\bar{c}} + 0\cdot\mathbf{c} + a\cdot\mathbf{c}\cdot 0$	*no c => No hazards.*
abc**e** y	$\bar{y}(\mathbf{\bar{e}} + \bar{b}\cdot\bar{c}) + b(c\cdot\mathbf{e} + \bar{a}\cdot\bar{c}\cdot\mathbf{\bar{e}}) + a\cdot c\cdot\mathbf{e}\cdot y$	*c must be 1 or no e*
ab1**e** y	$\bar{y}(\mathbf{\bar{e}} + \bar{b}\cdot 0) + b(1\cdot\mathbf{e} + \bar{a}\cdot 0\cdot\mathbf{\bar{e}}) + a\cdot 1\cdot\mathbf{e}\cdot y = \bar{y}\cdot\mathbf{\bar{e}} + b\mathbf{e} + a\cdot\mathbf{e}\cdot y$	$\bar{y}$ *must be 1 or no* $\bar{e}$
ab1**e**0	$= 1\mathbf{\bar{e}} + b\mathbf{e} + a\cdot\mathbf{e}\cdot 0 = \mathbf{\bar{e}} + b\mathbf{e}$	*b must be 1*
a11**e**0	$= \mathbf{\bar{e}} + \mathbf{e}$	*Static-1 for* a11**e**0
abce **y**	$\mathbf{\bar{y}}(\bar{e} + \bar{b}\cdot\bar{c}) + b(c\cdot e + \bar{a}\cdot\bar{c}\cdot\bar{e}) + a\cdot c\cdot e\cdot\mathbf{y}$	a,c,e *must be 1,1,1 or no* y.
1b11 **y**	$\mathbf{\bar{y}}(0 + \bar{b}\cdot 0) + b(1\cdot 1 + 0\cdot 0\cdot 0) + 1\cdot 1\cdot 1\cdot\mathbf{y} = \bar{b} + \mathbf{y}$	*no* $\bar{y}$ *=> No hazards in* ***y****.*

Graph of the previous hazard search

$$F = \bar{y}(\bar{e} + \bar{b}\cdot\bar{c}) + b(c\cdot e + \bar{a}\cdot\bar{c}\cdot\bar{e}) + a\cdot c\cdot e\cdot y$$



Masking Hazards

Mask a static-1 with an AND gate.

A hazard is between two squares

Since hazard is static-1, the function is 1 in those two squares

Mask it with a function which is guaranteed

- 1 in those two squares
- 0 elsewhere

That function is G = AB

.

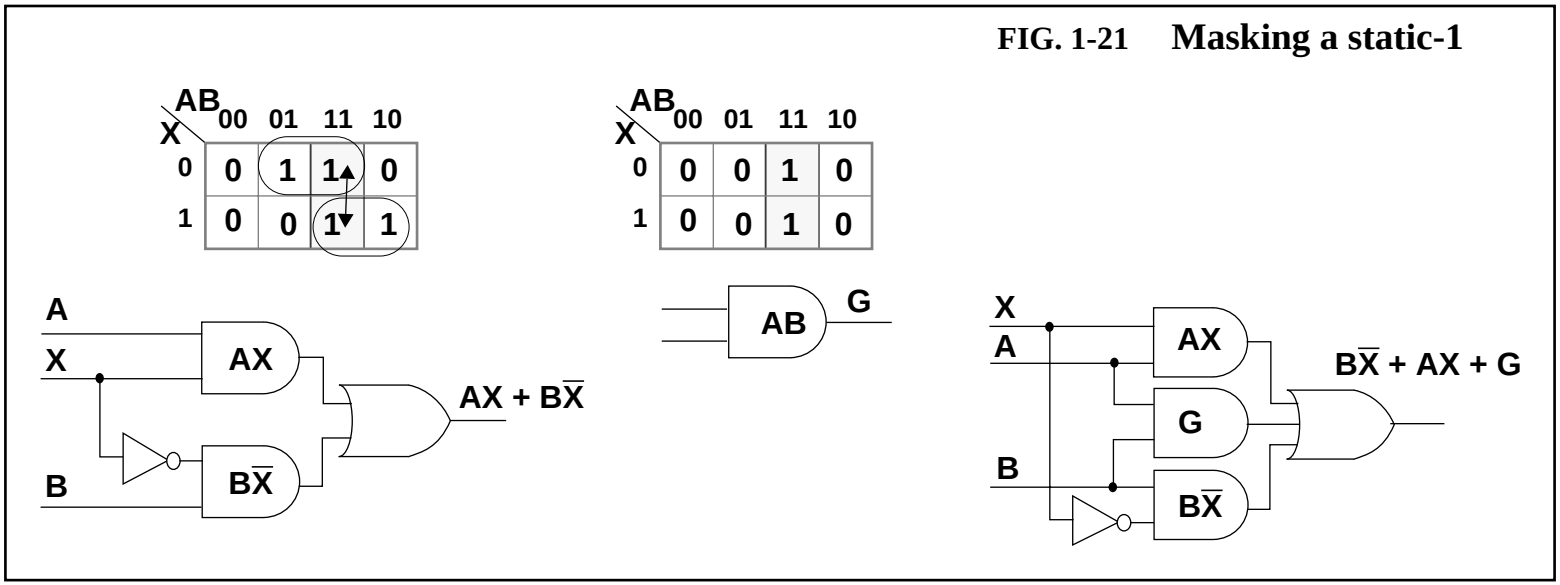


FIG. 1-21 Masking a static-1

Masking Hazards Algebraically

Expression $F = (a + \bar{c})(b + c) + cd$

$c + \bar{c}$ *Static-1 for* 0 1 **c** 1
$\bar{c} \cdot c$ *Static-0 for* 0 0 **c** 0
$\bar{c} \cdot c + c$ *Dynamic for* 0 0 **c** 1

Masking the static-1 hazard, $c+\bar{c}$

The hazard appears when a,b,d = 0,1,1 and $c \longleftrightarrow \bar{c}$.

F was designed to be "1" there.

but F actually has a hazard ⊓ there.

Define G:

G=1 when a,b,d = 0,1,1
G=0 anywhere else.

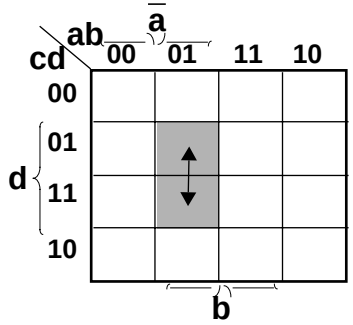
F + G = F Since when G is 1, F is 1.
elsewhere F + 0 = F

But F + G is solidly 1 over the hazard.

Desired G = $\bar{a}bd$

F with $c+\bar{c}$ masked is
$F + G = F = (a + \bar{c})(b + c) + cd + \bar{a}bd$

Expression $F = (a + \bar{c})(b + c) + cd$



G

cd \ ab	00	01	11	10
00	0	0	0	0
01	0	1	0	0
11	0	1	0	0
10	0	0	0	0

F+G

cd \ ab	00	01	11	10
00	F	F	F	F
01	F	F=1	F	F
11	F	F=1	F	F
10	F	F	F	F

Expression $F = (a + \bar{c})(b + c) + cd$

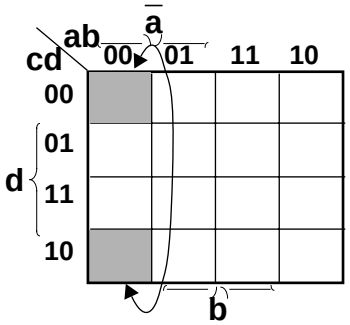
$\bar{c}\cdot c$ *Static-0 for* 0 0 **c** 0

Masking the static-0 hazard, $\bar{c}\cdot c$

The hazard appears when a,b,d = 0,0,0 and $c \leftrightarrow \bar{c}$.

F was designed to be "0" there.

but F actually has a hazard ⎍ **there.**



Define H:

H=0 when a,b,d = 0,0,0

H=1 anywhere else.

H

cd \ ab	00	01	11	10
00	0	1	1	1
01	1	1	1	1
11	1	1	1	1
10	0	1	1	1

F·H = F Since when H is 0, F is 0.

elsewhere F·1 = F

But F·H is solidly 0 over the hazard.

F·H

cd \ ab	00	01	11	10
00	F=0	F	F	F
01	F	F	F	F
11	F	F	F	F
10	F=0	F	F	F

Desired H = (a+b+d)

F with $c\bar{c}$ **masked is**

$F\cdot H = F = (a+b+d)((a + \bar{c})(b + c) + cd)$

Note it could have been masked as (a+b)(a + c)(b + c) + cd

Masking Both Hazards at Once

Combining the masks for static-1 and static-0

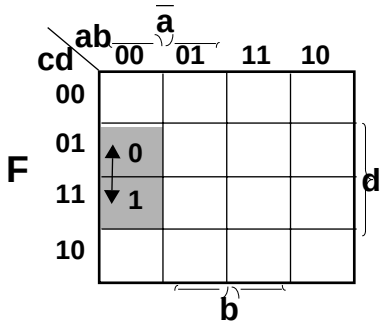
F with both $c\bar{c}$ and $c+\bar{c}$ masked is

$$F = (a+b+d)\big[(a + \bar{c})(b + c) + cd\big] + \bar{a}bd$$

Prove that the static hazards are both masked.

a	b	c	d	(a+b+d)((a + $\bar{c}$)(b + c)	+ cd) + $\bar{a}$bd	F	Type of hazard.
0	0	c	0	0+0+0 (0 + c	0 + c c0) 100	0c$\bar{c}$	No more hazard
0	0	c	1	0+0+1 (0 + $\bar{c}$	0 + c c1) 101	$\bar{c}$c+c	Dynamic
0	1	c	1	0+1+1 (0 + $\bar{c}$	1 + c c1) 111	1	No more hazard
0	1	c	0	0+1+0 (0 + $\bar{c}$	1 + c c0) 110	$\bar{c}$	

Masking Dynamic Hazards, $c\bar{c} + c$



Extract the Static Hazard

Cannot hold F constant over a,b,d = 0,0,1 because F changes with c.

However-
A static hazard resides inside every dynamic hazard.

$F=(a + \bar{c})(b + c)+ cd$ ← *this makes static hazard dynamic*

$f_1=(a + \bar{c})(b + c)$ ← *static-0 hazard, $\bar{c}c$, for a,b=0,0*

Make f_1 the part of F with the embedded static-0 hazard.

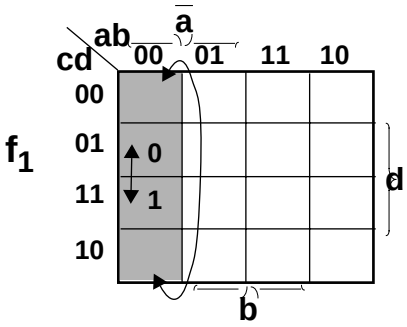
$$F = f_1 + cd$$

f_1 has a static hazard in c when a,b = 0,0,1

Define g_1 = 0, when a,b=0,0
= 1, otherwise

$$f_1 = f_1 g_1 = (a + b)\{(a + \bar{c})(b + c)$$

has masked the hazard.



$$F = f_1 + cd$$
$$= f_1 g_1 + cd$$
$$= (a + b)\{(a + \bar{c})(b + c)\} + cd$$

This F has masked the embedded static hazard and thus the dynamic hazard.

Masking does not introduce new hazards?

Were new hazards introduced because a and $\bar{a}$ now appear in F?

This will not happened.

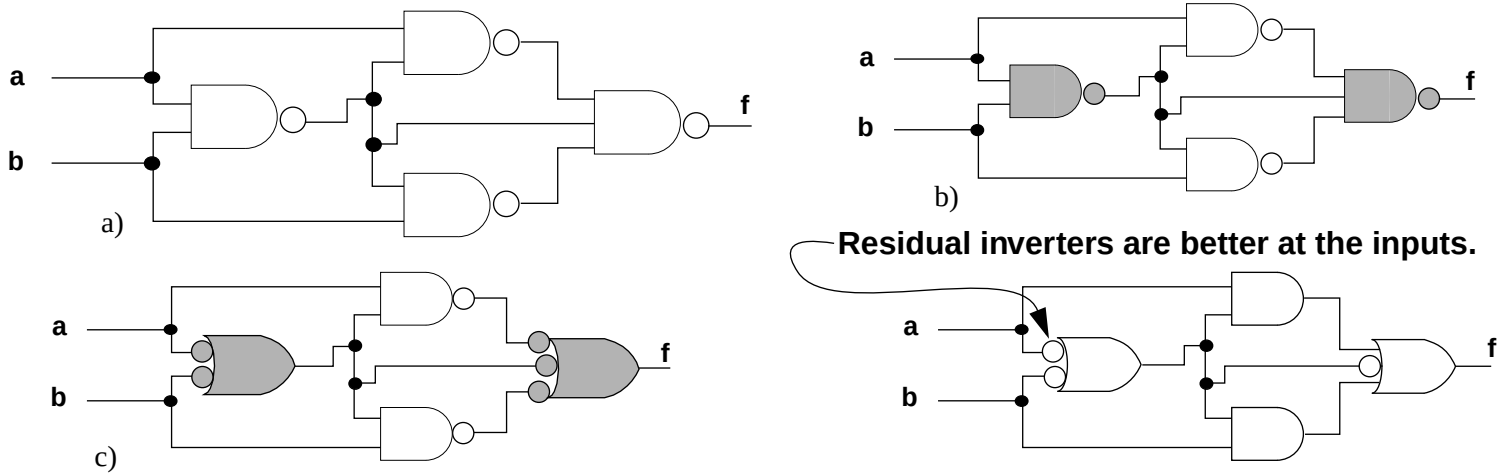
As a check, consider a.

b c d	(a + b)	{(a + c)(b + c)} + cd + $\bar{a}$bd	F	Type of hazard.
1 0 1	(a +1)	{(a + 1)(1 + 0)} + 01 + $\bar{a}$11	1	No hazard
1 1 1	(a +1)	{(a + 0)(1 + 1)} + 11 + $\bar{a}$11	1	No hazard

An Example of Locating Hazards

Find all the hazards in FIG. 1-22.

FIG. 1-22 Circuit with a lot of paths for potential hazards.



Cancel inverters using DeMorgan's law graphically.
Write circuit equation.

$$F = a(\bar{a} + \bar{b}) + b(\bar{a} + \bar{b}) + \overline{(\bar{a} + \bar{b})}$$

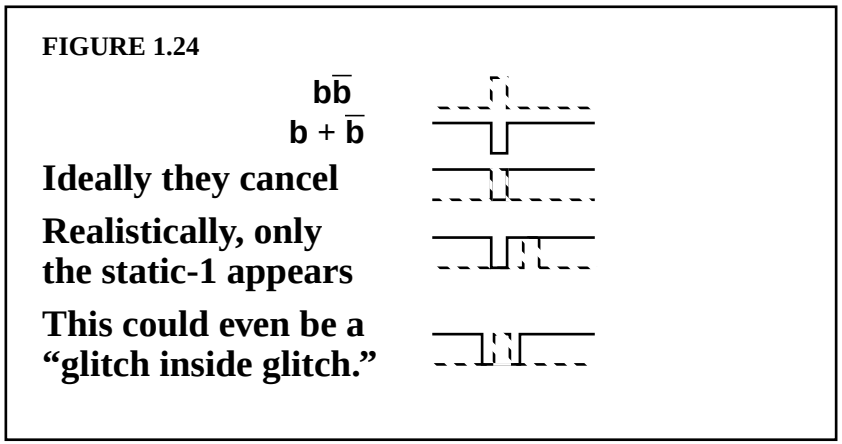
Potential hazards in both a and b.

Table to expose hazards in b with a fixed.

a	$a(\bar{a} + \bar{b}) + b(\bar{a} + \bar{b}) + ab$	F	Type of hazard.
0	$0(1 + \bar{b}) + b(1 + \bar{b}) + 0b$	b	No hazard
1	$1(0 + \bar{b}) + b(0 + \bar{b}) + 1b$	$\bar{b} + b\bar{b} + b$	Static-1 hazard

Explanation of $\bar{b} + b\bar{b} + b$

FIG. 1-23 Explanation the weird hazard



A rising glitch $b+\bar{b}$ is ORed with a falling glitch $b\bar{b}$.

With very good luck the two glitches to come exactly at the same time and cancel.
More likely the rising glitch will be lost in the static 1 signal.
Only the falling glitch will appear.

From symmetry, changes in "a" have the same behavior.

Implementing Hazard Free Circuits

Sum-of-Product Circuits Have No Static-0 Hazards

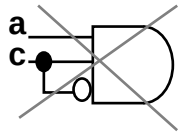
Sum of products circuits always have an equation of the form

$F = abc + ab\bar{d} + ab\bar{c}d + \ldots\ldots + a\bar{b}c\bar{d}$

Static-0 hazards are like $c\bar{c}$. { $c + \bar{c}$ is static-1}
To get $c\bar{c}$ in F as above on must place c and $\bar{c}$ as inputs to the same AND gate.
This is ignorant.

Rule I:

Except for the gross carelessness of including terms like $ac\bar{c}$,
Σ of Π implementations have no static-0 hazards.



Sum-of-Product Circuits Have No Dynamic Hazards

Σ of Π circuit have equations of the form

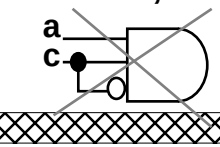
$F = abc + ab\bar{d} + ab\bar{c}d + \ldots\ldots + ab\bar{c}d + abc\bar{c}d$

Dynamic hazards are of the form $c\bar{c} + c$ or $(c+\bar{c})c$.
In F, try fixing a, b and d at any combination of 0 or 1.

A dynamic hazard in c, must have a term containing $c\bar{c}$.
In F above, one can only get a dynamic hazard by using the "ignorant" term $abc\bar{c}d$.

Thus Rule II is:

Except for the gross carelessness of including terms like $ac\bar{c}$,
Σ of Π implementations have no dynamic hazards.



Sum-of-Product Circuits Have Only Easily Eliminated Static-1 Hazards

Σ of Π circuits can still have static-1 hazards
They are easily found and removed using:
a Karnaugh map,
or algebraically.
.

FIG. 1-25 Map of function

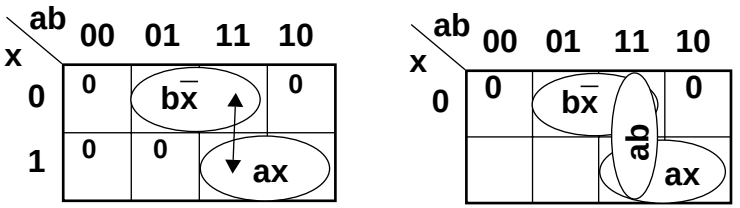
$$F = b\bar{x} + ax$$

It is Σ of Π
The hazards must all be static-1.

Hazard when a,b = 1,1.
Add term ab to mask the hazard.

$$F = b\bar{x} + ax + ab$$

Is shown on the right.



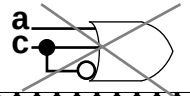
Product-of Sum Circuits Have No Static-1 Hazards

Π of Σ circuit equations are of the form

$F = (a+b+c)(a+b+\overline{d})(a+b+\overline{c}+d)(\ldots\ldots)(a+b+\overline{c}+d)$

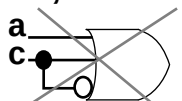
Static-1 hazards are of the form $c + \overline{c}$.
To get $c+\overline{c}$ in F one must place c and $\overline{c}$ as inputs to the same OR gate.
This is ignorant.

Except for the gross carelessness of including terms like $a+c+\overline{c}$,
Π of Σ implementations have <u>no</u> static-1 hazards.



Product-of Sum Circuits Have No Dynamic Hazards

Except for the gross carelessness of including terms like $ac\overline{c}$,
Π of Σ implementations have <u>no</u> dynamic hazards.



Product-of Sum Circuits Have Only Easily Eliminated Static-0 Hazards

Π of Σ circuits can still have static-0 hazards
They are easily found and removed using a Π of Σ Karnaugh map

Example: Single-Variable-Change Hazard-Free Circuit From a Map

A digital function defined by a map; FIG. 1-26(left).

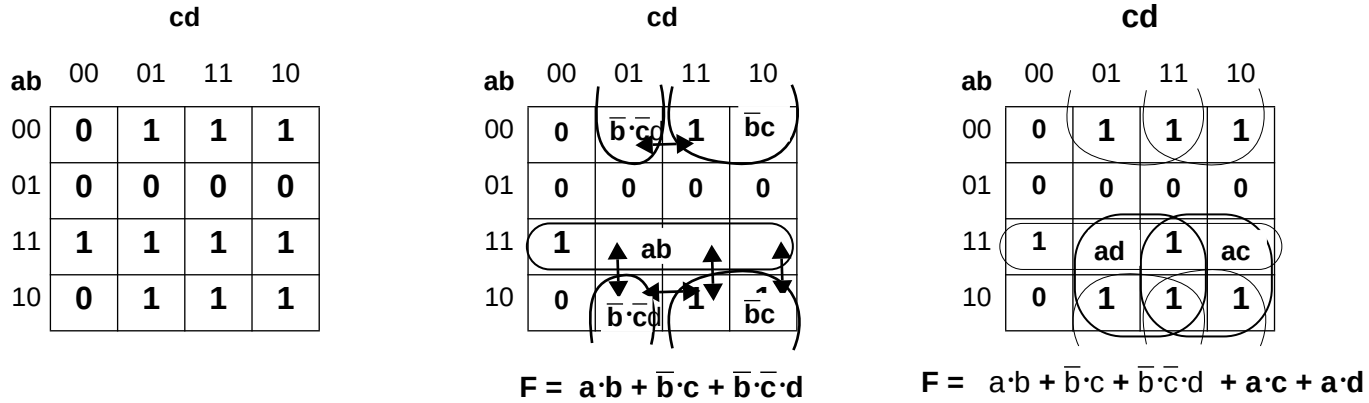
Choose a circling for the map; see FIG. 1-26 (middle),
⟷ indicate the hazards.

$$F = a\cdot b + \bar{b}\cdot c + \bar{b}\cdot\bar{c}\cdot d$$

Then add circles which cover the arrows; FIG. 1-26(right).
The hazard free equation, on this final map, is -

$$F = a\cdot b + \bar{b}\cdot c + \bar{b}\cdot d + a\cdot c + a\cdot d$$

FIG. 1-26 **Left) Example to be implemented as a hazard free circuit.**
Centre) A possible Σ of Π encirclement showing hazards.
Right) The map with the hazards covered.



Factoring and Hazards For Σ of Π / Π of Σ

1. **Algebraic operations do not create hazards unless they make x interact with $\bar{x}$.**

See "Algebra and Hazards.," p.12

2. **Except the distributive laws can convert static ⟷ dynamic hazards. They can even convert masked hazards ⟷ dynamic hazards.)**

3. **The distributive laws cannot create or destroy static hazards.**

4. **DeMorgan's law has no effect on hazards.**

FIG. 1-27 Distributive Laws

Normal Law

$$(a + b)x = b\bar{x} + ax$$

Special For Boolean Only

$$ab + x = (a + x)(b + x)$$

Example: hazard free expression from last page.

$$F = a\cdot b + \bar{b}\cdot c + \bar{b}\cdot d + a\cdot c + a\cdot d$$

Factoring (uses the distributive law) reduces parallel paths.

Use $xc + xd = x(c + d)$

$$F = a\cdot b + \bar{b}\cdot(c + d) + a\cdot(c + d)$$
$$= a\cdot b + \underbrace{(a + \bar{b})(c + d)}$$
$$= a\cdot b + u$$

No dynamic hazards (or static)

Multiplying out (also uses distributive law) creates parallel paths

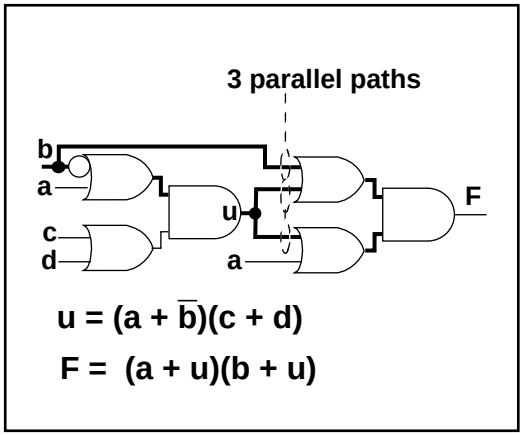
Use $ab + x = (a + x)(b + x)$

$$F = a\cdot b + u$$
$$F = (a + u)(b + u); \qquad u = (a + \bar{b})(c + d)$$

Dynamic hazard when a=0, c or d =1

u => $\bar{b}$ $\qquad F$ => $(0 + \bar{b})(b + \bar{b})$

We can still guarantee no static hazards



Using Forbidden Algebra That May Insert Hazards

$$F = a\cdot b + b\cdot c + \bar{b}\cdot d + a\cdot c + a\cdot d$$
$$= a(b + c + d) + \bar{b}(c + d)$$
$$= a(b + c + d) + \bar{b}(b + c + d)$$
$$= (a + \bar{b})(b + c + d)$$

Static-0 when a,c,d= 0,0,0

Using bb = 0. *(don't do that!)*

Hazards With Multiple Input Changes

Two-variable-change hazards

Two-variables changes, move two squares on the Karnaugh map.

Some 2-change hazards are maskable. (upper arrow in FIG. 1-28)
Many 2-variable hazards are not maskable. (lower arrow)

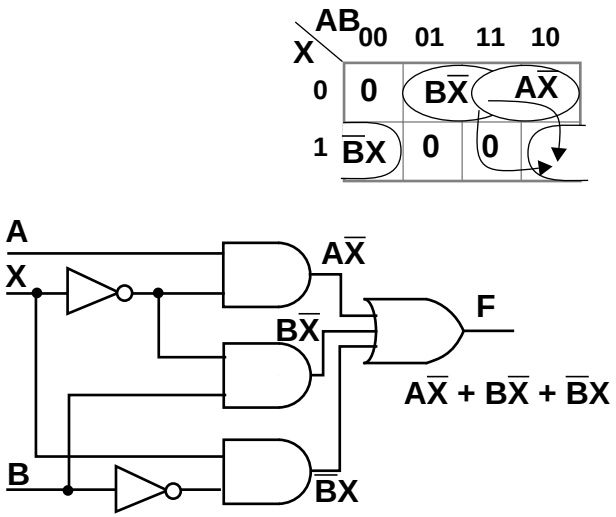


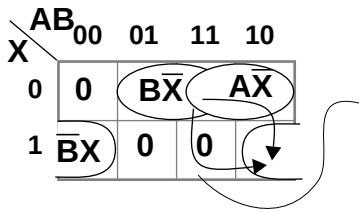
FIG. 1-28

Start at square A,B,X =1,1,0 (the tail of the arrows)
Change both B and X to move to square A,B,X =1,0,1 (the head of the arrows).

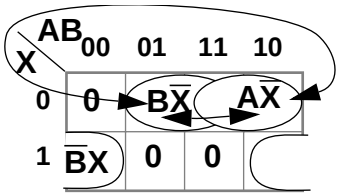
If B changes slightly before X,
one travels the upper route ↘.
The valley between $A\overline{X}$ and $\overline{B}X$ may glitch.
A masking term $A\overline{B}$ can cover the valley.
It only removes the glitch on the upper path.

If X changes slightly before B,
one takes the lower path ↳.
This will always glitch.
It cannot be covered.
Covering the offending "0" changes the function.

Nonmaskable or *function* hazards

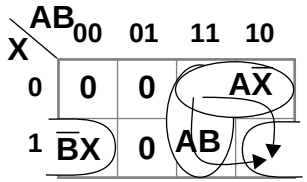


The lower arrow goes through a "0."
This "0" is part of the function.
F is supposed to be low for input A,B,X=1,1,0.
One cannot fill in the "0" to mask the hazard.
It is a *nonmaskable* or *function* hazard.



Another 2-variable function hazard.
If B changes first (short path) there is no glitch.
If A changes first (long path) there is a glitch.

Associated Maskable, Single-Variable Hazards



$F = A\bar{X} + AB + \bar{B}X$

When AB=10 => $F = \bar{X} + X$

When AX=11 => $F = B + \bar{B}$

Maskable, single-variable, static-1, hazards.

Maskable, Double-Variable Hazards

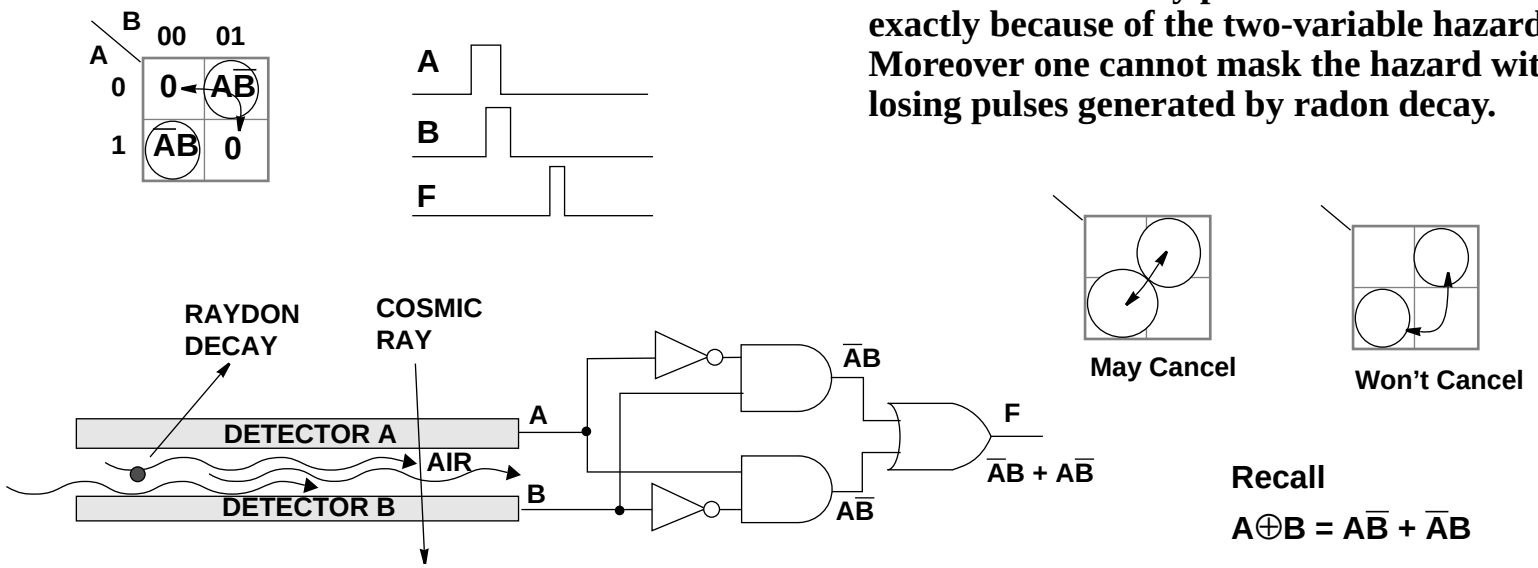
See a little later

Example of a 2-Variable Change Hazard

Detect the decay of radon passing through a long thin air duct.
There is a detector on each side of the duct.
A radon decay inside the duct will activate one detector.
Cosmic rays from outside will activate both detectors.
If both detectors respond at the same time, the result is ignored.

The rejection circuit uses an XOR which has a 2-variable hazard.
One cannot reliably exclude the double pulses with XOR.

FIG. 1-29 **Suppose one AND gate is slower than the other. Then the cosmic ray pulses will not cancel exactly because of the two-variable hazard. Moreover one cannot mask the hazard without losing pulses generated by radon decay.**



Recall

$A \oplus B = A\overline{B} + \overline{A}B$

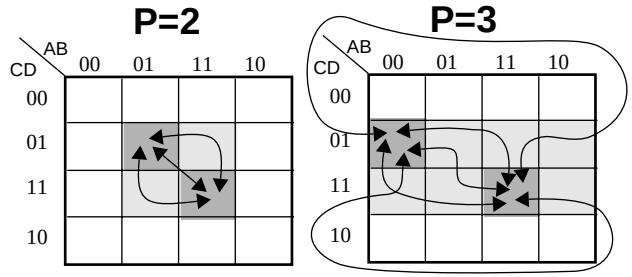
General Multiple Variable Changes

Definition of Static Multivariable Hazards

Consider a function F(a,b,c, ...).

Let P of its variables change at the same time.

There are 2^P squares through which this P variable transition may travel.



For static hazards F has the same value before and after the changes.

Conditions to be Hazard Free

F is free of multiple-variable static hazards for the P changes, if and only if:

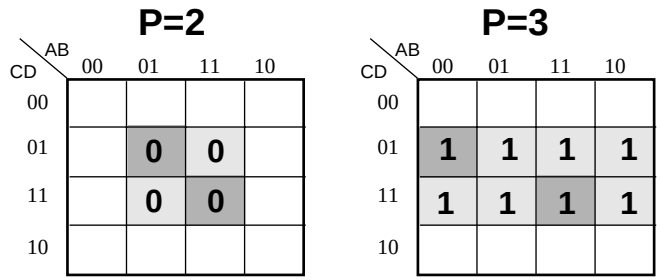
1) Condition for no function hazards
All 2^P squares, that may be travelled through, have the same value.

2) Condition for no maskable hazards
Fixing the stationary variables cannot reduce F to one of the forms

$$a + \bar{a}, \quad c + \bar{c}, \quad a\bar{a}, \quad c\bar{c},$$

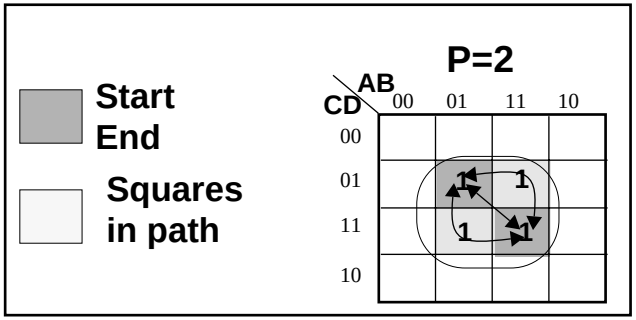
or combinations of these like

$$a+\bar{a} + c+\bar{c}, \quad \bar{a}a\bar{c}c, \quad c+\bar{c} +a\bar{a}.$$



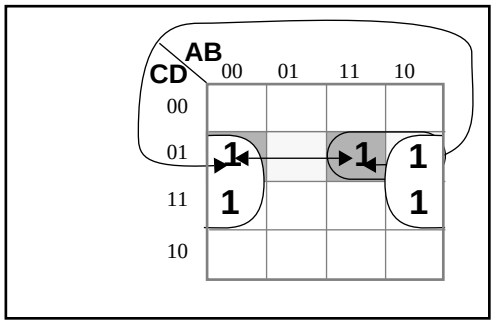
Example: A Double Change With No Hazard

Two variables changing A,C,=,0,0 ->1,1.
P = 2 = numper of variables changing at once.
The possible transitions cover $2^P = 4$ squares.
There are no hazards for any of the transitions.



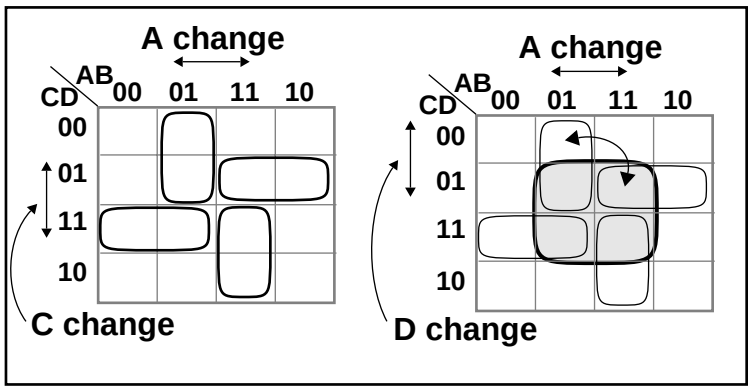
Example: A Function Hazard

Two variables changing A,B = 0,0 -> 1,1
Transitions can move over $2^P = 4$ squares.
If the transition via the "wrap around," has no hazard.
The direct path will cross the gap and give a glitch.



Example: Maskable Hazards

Any transitions between the centre variables, A and C, is a hazard.
The hazards can be masked by covering the centre four squares as shown on the right.
If other pairs, such as A and D change, a function hazard results, as shown.



Example; A Real Two-Variable Maskable Hazard

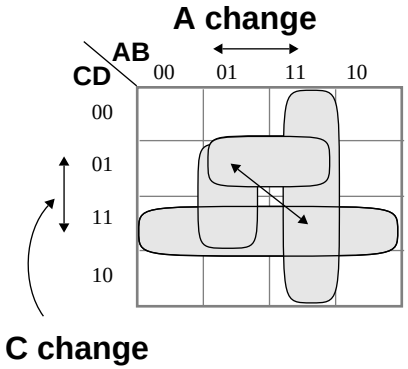
There is a "hole" in the centre.
It may give a glitch when A and C both change at once.
The equation for the function is

$$F = AB + \overline{A}BD + CD + B\overline{C}D$$

Fix BD = 11 to expose the hazard.

$$F = A + \overline{A} + C + \overline{C}$$

This is a static-1, two-variable hazard.



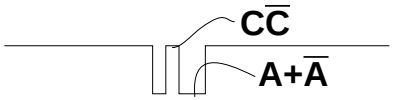
Example: A Nonmapable Two-Variable Maskable Hazard

This hazard cannot be mapped onto Σ of Π

$$F=BA + \overline{A}D + \overline{C}(C + \overline{B})$$

Set BD=11

$$F= A + \overline{A} + \overline{C}C$$

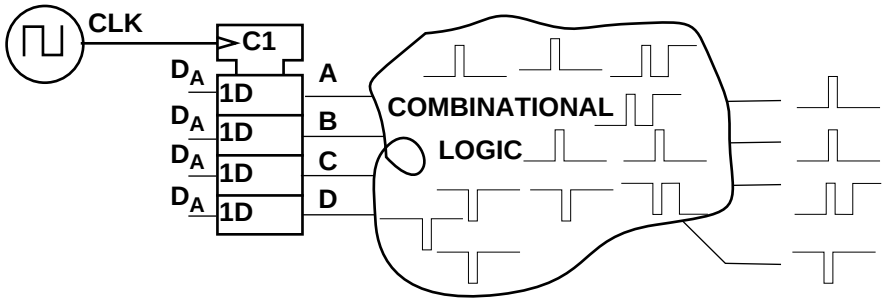


When Are Hazards Important?

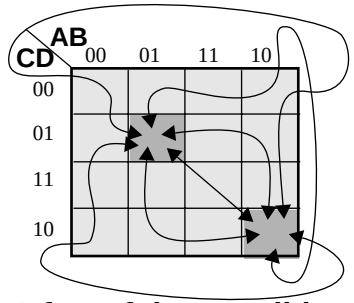
Multiple Variable Change Hazards are Plentiful

Take a synchronous circuit
Let 4 flip-flops change at once.
Then P=4, 2^P=16 possible map squares.
Most paths will have function hazards

FIG. 1-30 The vast number of glitches generated by multiple variable changes



GLITCH HEAVEN

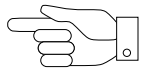


P=4

2^P=16

A few of the possible paths for 4-variable changing

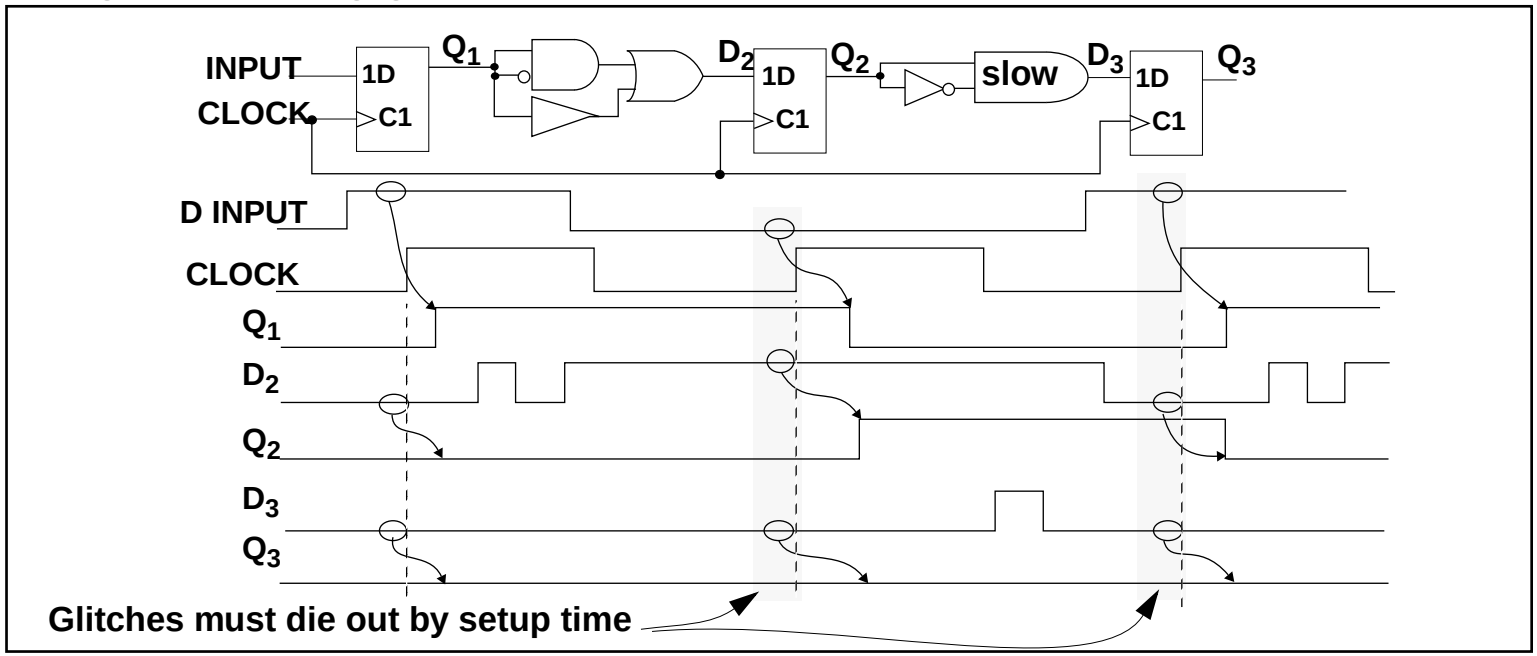
With 2 variables changing it is almost impossible not to have hazards.
With more variables changing they are like waves in the ocean.



Hazards do not hurt synchronous circuits

In clocked logic, flip-flops only respond to the inputs slightly before the clock edge.
See the circles on the waveforms below.
All variables change shortly after the clock edge.
The clock cycle is made long enough so the glitches die out long before the clock edge.

FIG. 1-31 **The flip-flops only respond in the circled region on the waveforms below.**
A glitch at any other time will not influence state of the machine.
The glitches die out long before the clock edge.
The glitches have negligible influence.

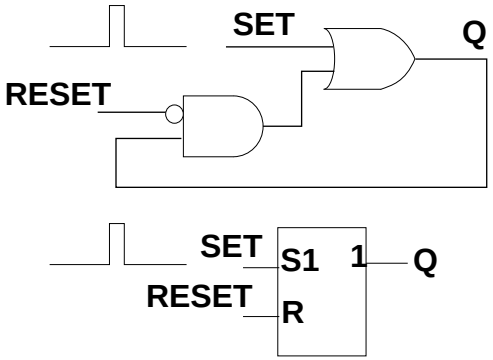


Hazards Kill Asynchronous Circuits

By asynchronous circuits, we mean ones with feedback that can latch signals.

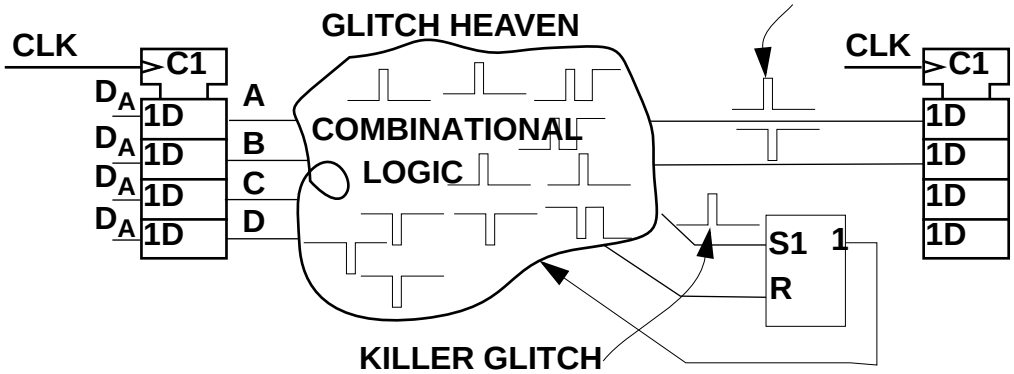
A glitch may causes a wrong value to be latched. All hazards must be eliminated, or proven harmless.

Analog simulation can prove it harmless.



Example: Placing an R-S Latch in a Synchronous Circuit

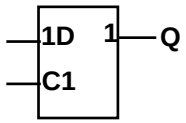
FIG. 1-32 The Russian Roulette of digital design

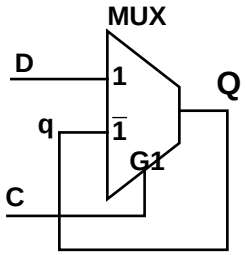


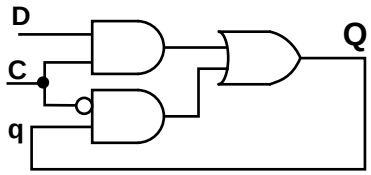
Example: The Hazard in the Transparent D-Latch

The simple form of D-latch (transparent latch) is just a MUX.
The latch has a hazard when D,q =1,1.
This glitch can feed back and latch itself.

FIG. 1-33





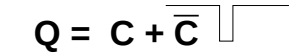


$Q = DC + \overline{C}q$

SET D=1, q=1

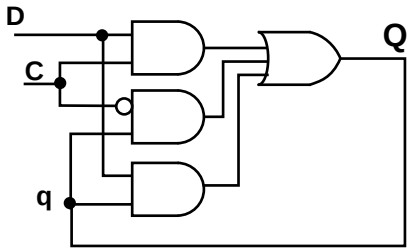
$Q = 1C + \overline{C}1$

$Q = C + \overline{C}$



To mask the hazard
Keep Dq =11 across the change in C

$Q = DC + \overline{C}q + Dq$



Hazard-free latch

$Q = DC + \overline{C}q + Dq$

SET D=1, q=1

$Q = 1C + \overline{C}1 + 1$

$Q = 1$

Outputs where hazards are of concern

Some displays are very sensitive to glitches.

Light emitting-diode displays will show slight “ghosts” in dim light.

Cathode-ray tube displays will usually show any glitches on their input signals.

Memories

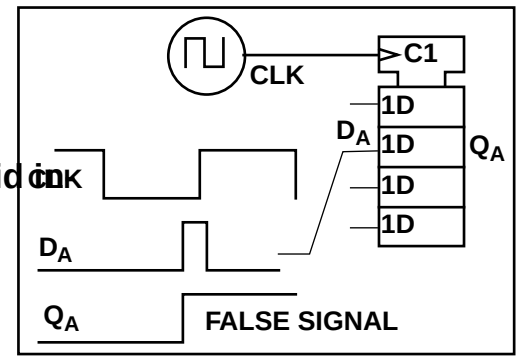
Memory chips are asynchronous latches, and are sensitive to glitches.

Memory control leads must be glitch free.

Glitches in asynchronous inputs to synchronous circuits

Asynchronous inputs to synchronous circuits must be hazard free.

An input glitch on the clock edge, may be captured a valid input.



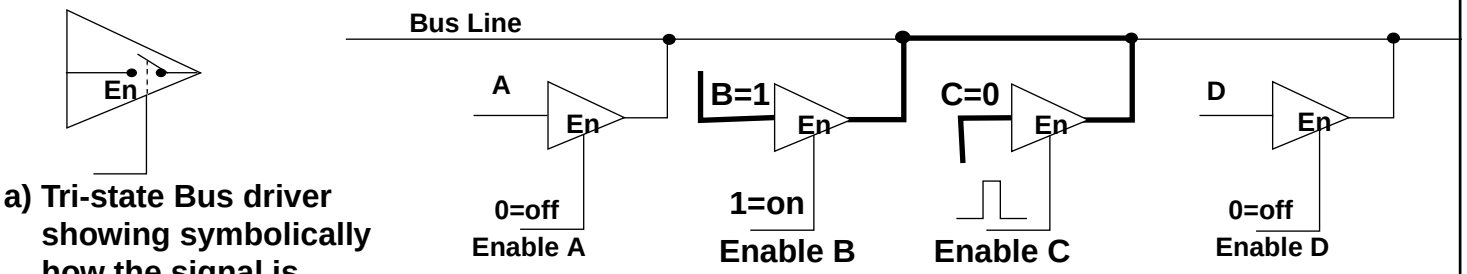
Hazards may hurt bus drivers

In control circuits for tri-state bus drivers.
Hazards on the driver-enable lines may turn on two drivers at once.
If one driver is "1" and one "0", a high current will flow.

FIG. 1-34 A bus with several drivers.

The "En" (Enable) signal connects and disconnects the driver.

Normally only one driver is connected at a time.
Only one "En" signals is high at a time.
A hazard may turn on a second driver.
Below, B is high and C is low.
The glitch shorts the power to ground.



a) Tri-state Bus driver showing symbolically how the signal is disconnected.

A short high-current glitch on a bus -
Normally does not damage the gates (at least in the short term).
It causes a dip in the power supply voltage.
This may cause flip-flops or memory cells to change state.

Turn-Off Glitches Don't Bother Bus Drivers

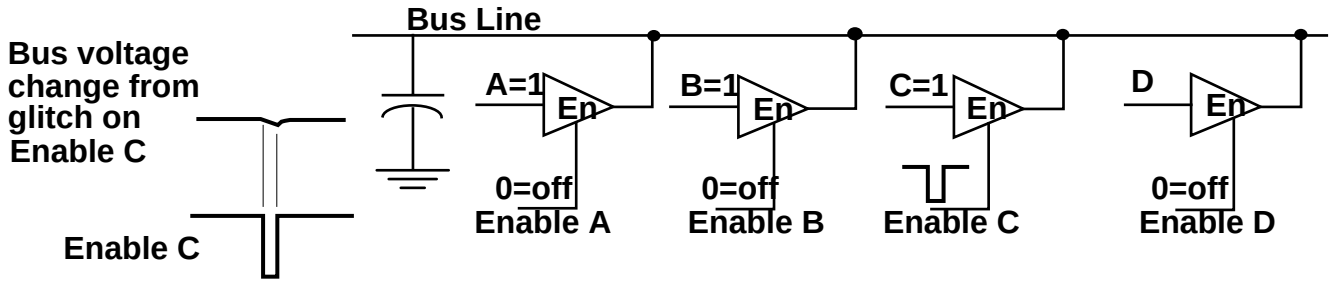
Case: Drivers that connect for En = 1, and disconnect (tri-state) for En = 0.
These drivers are sensitive only to static-0 hazards, not static-1.

A static-1 glitch may puts opposing signals on the bus.

A static-0 glitch only turns off the "on driver" momentarily.
There is no other bus driver trying to pull the bus the other way.
The bus stray capacitance will maintain the bus voltage during the glitch.

FIG. 1-35 A glitch that turns off a driver momentarily does no damage.

With no connected driver for a short time.
The bus capacitance will not allow significant change in the bus voltage.



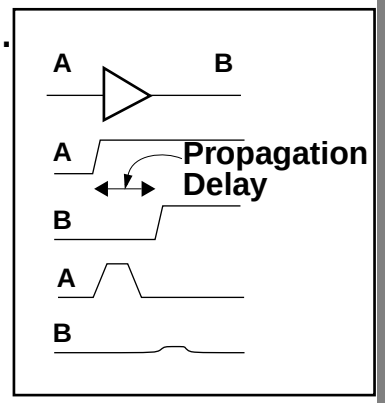
Σ of Π logic (NAND followed by NAND) has static-1, but no static-0 hazards.
Preferred for bus control which connects on En=1.

Π of Σ logic (NOR followed by NOR logic) has static-0, but no static-1 hazards
Preferred for bus drivers which connects on En=0.

Absorption of Glitches by Gates

Gate Inertia (Inertial Delay)

For a fast pulse to propagate through a gate,
it must be long enough to charge the gate internal capacitance.
Otherwise its energy is too small to change the output.



A rule of thumb
A pulse shorter than the propagation delay of the gate
will not pass through it.
This is called *gate inertia* or *inertial delay.*

Synchronous circuits would have <u>many many</u> glitches except for inertial delay.

Digital simulator software uses inertial delay to keep simulation waveform displays from being a wasteland of glitches.
Simulators normally suppress glitches shorter that a certain duration.
The default duration is the propagation delay of the gate passing the glitch.

Glitches Increase Power Consumption

Power consumption is very important in:
- **circuits run from batteries.**
- **circuits that have cooling problems.**

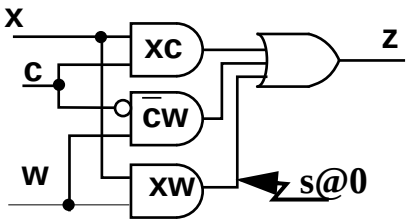
CMOS uses power only during variable changes.
Power is used charging and discharging gate capacitances.
One glitch uses little power,
but mega-glitches may happen in a large IC.

Glitches are known to waste significant power.

Masking is not a viable for power saving

- **Masking gates add extra capacitance.**
  This uses more power than the saving by hazard elimination.
- **Most circuit glitches are multi-variable.**
  Most of these are unmaskable.
- **Masking glitches creates untestable redundant gates .**

FIG. 1-36 **A circuit with a masked hazard, and an**



$z = xc + \bar{c}w + xw$

Adding the masking gate xw increased circuit capacitance, and hence power consumption from 6 inputs to 9. (50%)

Glitch happens only with x,w,c = 1, 1, 1->0

Map for z

$z = xc + \bar{c}w + xw$

c \ xw	00	01	11	10
0	0	0	$\bar{c}w$	
1	0	xc	xw	0

Map for z with xw stuck-at-zero

$z = xc + \bar{c}w + 0$

c \ xw	00	01	11	10
0	0	0	$\bar{c}w$	
1	0	xc		0

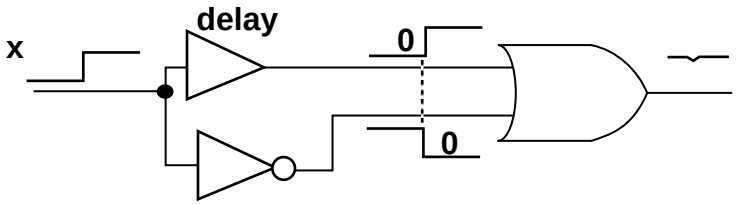
FIG. 1-37 **No normal test can find gate xw output stuck-at-0.**

Avoiding Glitches by Balancing Delays

Review of Adding Delay to Hazards

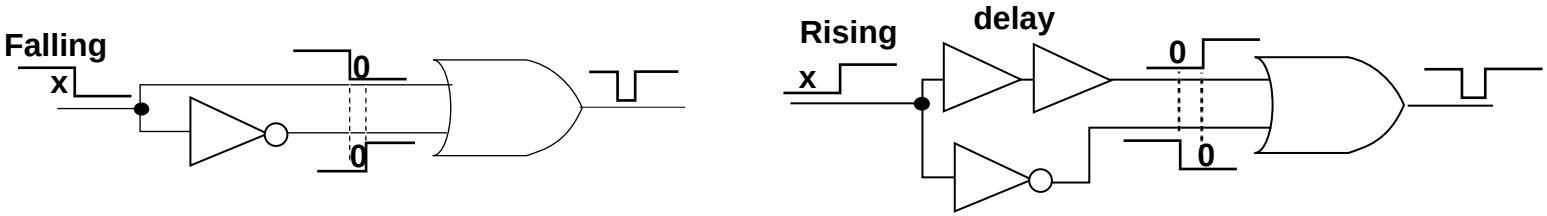
Most glitches can be removed by balancing the two paths at the hazard inputs, but it requires good control of propagation delays.

FIG. 1-38 **Basic static-1 hazard circuit from FIG. 1-3 (FIG. 1-39 left). With delays balanced, glitch is eliminated.**



- **Adding too much delay will make the glitch appear on the opposite input edge.**

FIG. 1-39 **Adding delay, moves the glitch from $\overline{x}$ to x.**

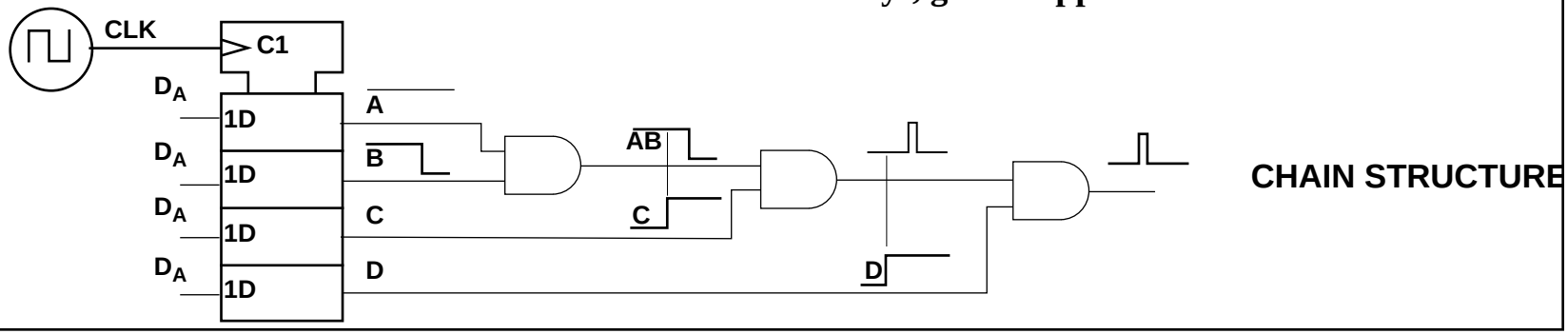


At the silicon layout level, balanced delays usually suppress glitchs. With standard cells and field-programmable arrays, balancing is hard.

Avoiding Glitches by Balancing Delays

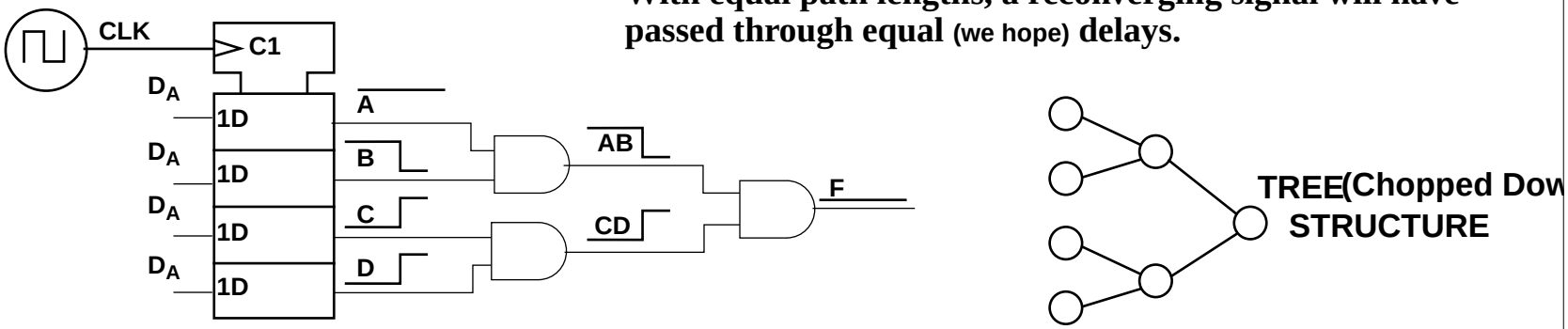
Take synchronous logic.
Register outputs all change at once.
Unbalanced delays can cause many glitches.

FIG. 1-40 Register clocking gives multiple variable changes. With unbalanced delays, glitchs appear.



Balancing the path delays can remove most glitches.
A tree structure balances delays.

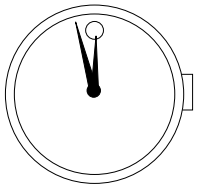
FIG. 1-41 Clocking the register changes several variables at once. With equal path lengths, a reconverging signal will have passed through equal (we hope) delays.



Balancing May Not Be Too Hard

- **Inertial delay allows balance if the two paths are within a gate delay. (more or less)**
- **A few glitches getting through is not fatal if the objective is power saving.**

Summary Of Hazards



Single variable change hazards

Can be found and cured.

Multiple variable change hazards

Can be found
Are very plentiful
Cannot be cured in general, they are part of the logic.
May be reducable to single variable change.

See race-free state assignment in the Section on races and cycles.

Hazards are not important in truly synchronous circuits

Except for power consumption.

Don't mention false-paths.

Hazards are important in

Asynchronous circuits.

Latches and flip-flops
Pulse catchers
Debouncers

Memory interface signals
High speed displays
Bus Control