2-level logic minimisation while handling hazards

Motivation

- Simply said, hazards are unwanted transitions that happen on changes in input
- Of course, if a particular input bit is altered repeatedly, it is expected that some state/output bits will also alter accordingly
- How to distinguish between such expected transitions and unwanted transitions?
- For that we only consider monotonic transtions in the input, i.e. bits changing as 0⟶1 or 1⟶0
- Under such circumstances, we would like to ensure that observable changes are also monotonic (hazard free)
- As it will be seen, even for monotonic changes in inputs, the response may be non-monotonic
- We shall see how that may happen
- We shall also explore ways to stop that from happening

Basic notions

Monotonic transitions
: When the input changes from minterm m_1 to minterm m_2, the corresponding bits change either from 0 to 1 or from 1 to 0, but not back and forth

Fundamental mode of operation
: After an input transition, no new inputs may arrive until the circuit has stabilised

SIC fundamental mode
: Fundamental mode of operation where only a single input bit may change

MIC fundamental mode
: Fundamental mode of operation where multiple input bits may change in a narrow interval of time

Static hazard
: Situation where it is possible for an output to undergo a momentary transition (glitch) when it is expected to remain unchanged for monotonic transitions of the input

Static 1-hazard
: Static hazard where output momentarily goes to 0 when it should remain at 1 for monotonic transitions of the input

Static 0-hazard
: Static hazard where output momentarily goes to 1 when it should remain at 0 for monotonic transitions of the input

Dynamic hazard
: Output changes multiple times instead of just once going from 0 to 1 or from 1 to 0 for monotonic transitions of the input

Transition cube (Multiple valued transition cube)
: For minterms m_1 as the start point and m_2 as the end point, the cube $T = [m_1, m_2]$ contains all the minterms that can be reached for monotonic transition from the start to the end

  If $m_1 = l_{11}l_{12}\ldots l_{1n}$ and $m_2 = l_{21}l_{22}\ldots l_{2n}$ then $T = \left(l_{11}+l_{21}\right)\left(l_{12}+l_{22}\right)\ldots\left(l_{1n}+l_{2n}\right)$

  [010, 100] contains the following minterms: 000, 010, 100, 110; symbolically, $[x'yz, xy'z'] = (x+x')(y+y')z' = z'$

Multiple valued open transition cube
: For minterms m_1 as the start point and m_2 as the end point, the cube $T = [m_1, m_2) = [m_1, m_2] - \left\{m_2\right\}$

Function hazards
: Consider the input changing from minterm m_1 to minterm m_2 monotonically for some function f so that $f(m_1) = f(m_2)$, when multiple inputs change, there could be an intermediate minterm m where $f(m) \neq f(m_1)$; such a situation is called a function hazard where a glitch may be inevitable because of the underlying function

AB

	f	00	01	11	10
	00	0	0	0	0
CD	01	0	1	0	0
	11	0	1	0	0
	10	0	1	1	0

- In this case the transition happens from $m_1 = A'B'C'D'$ to $m_2 = A'B'C'D$
- There is no function hazard for this transition; transition cube is shown as the dashed closed path

AB

	f	00	01	11	10
	00	0	0	0	0
CD	01	0	1	0	0
	11	0	1	0	0
	10	0	1	1	0

- In this case the transition happens from $m_1 = A'BC'D'$ to $m_2 = ABC'D$
- The transition can go through $m = A'BC'D$ where $f(m) = 1$, so there can be static-0 function hazard
- There is a function hazard for this transition; transition cube is shown as the dashed closed path

AB

	f	00	01	11	10
	00	0	0	0	0
CD	01	0	1	0	0
	11	0	1	0	0
	10	0	1	1	0

- In this case the transition happens from $m_1 = A'BCD$ to $m_2 = A'B'CD'$
- There is no function hazard for this transition; transition cube is shown as the dashed closed path

AB

	f	00	01	11	10
	00	0	0	0	0
CD	01	0	1	0	0
	11	0	1	0	0
	10	0	1	1	0

- In this case the transition happens from $m_1 = A'BCD$ to $m_2 = AB'CD'$
- The transition can go through cubes $ABCD$ followed by $ABCD'$ so that the value of the function can change several times there can be dynamic hazard function hazard
- There is a function hazard for this transition; transition cube is shown as the dashed closed path

Logic hazard
: For a combinational logic function f, circuit implementation C and an input transition t, if f is function hazard free for t but there is an output glitch for t, then f has a logic hazard for C on t

1⟶1 transition

AB

	f	00	01	11	10
	00	0	0	0	0
CD	01	0	1	0	0
	11	0	1	0	0
	10	0	1	1	0

- Consider the transition from $m_1 = A'BCD$ to $m_2 = A'BCD'$
- However, as the input changes, f makes a transition from 1 to 0 for $c_1 = A'BD$ and from 0 to 1 for $c_2 = BCD'$
- During this transition, there can be a glitch (static-1 hazard) — can it be avoided
- Consider the inclusion of $c = A'BC$ in the implementation of f; it's the transition cube shown as the dashed closed path in green
- Now, f continues to be 1 during the transtion from m_1 to m_2, thereby avoiding this hazard
- Cube c is *required* to avoid this hazard

1⟶0 transition

AB

	f	00	01	11	10
	00	0	0	0	0
CD	01	0	1	0	0
	11	0	1	0	0
	10	0	1	1	0

AB

	f	00	01	11	10
	00	0	0	0	0
CD	01	0	1	0	0
	11	0	1	0	0
	10	0	1	1	0

- Consider the transition from $m_1 = A'BCD'$ to $m_2 = ABCD$
- Note that partial transitions are hazard free for the indicated cubes
- Now, for the transition in the inputs $0110 \longrightarrow 0111 \longrightarrow 1111$, the cube $c_1 = A'BD$ has output transitions $0 \longrightarrow 1 \longrightarrow 0$ which is a glitch
- This glitch may travel to the output, as explained next
- For the same transition sequence, the cube $c_3 = A'BC$ has output transitions $1 \longrightarrow 1 \longrightarrow 0$
- There may be speed differences of the gates realising c_1 (slower) and c_3 (faster). With the c_3 gate switching faster because it has only one transition and the c_1 gate rising and then falling after some delay, the OR of the output transitions may go as $1 \longrightarrow$ weak $0 \longrightarrow$ weak $1 \longrightarrow 0$ and so have a glitch
- This could have been avoided if c_1 did not have the *illegal intersection* with $T = [m_1, m_2]$ where m_1 is absent
- Consider the intersection with $c_1' = A'BC'D$ instead of c_1 for this transition
- Now, f makes a monotonic $1 \longrightarrow 0$ transition for the transtion from m_1 to m_2, thereby avoiding this hazard

Required cube for a static 1⟶1 input transition
: $T = [m_1, m_2]$ is a required cube for a function hazard free static 1⟶1 input transition

Required cube for a dynamic 1⟶0 input transition
: Maximal $[m_1, m] \subset T = [m_1, m_2]$, $f(m_1) = 1$ and $f(m_2) = 0$ where $f = 1$ is a required cube, T is function hazard free

Prviledged cube
: The transition cube for a dynamic transition is called a priviledged cube

Illegal intersection
: A cube c not containing m_1 is said to have an illegal intersection with the priviledged cube $T = [m_1, m_2]$

2-Level hazard free logic minimisation

Conditions for hazard-free transition

Assume that $[m_1, m_2]$ is the transition cube corresponding to a **function hazard free transition** from m_1 to m_2 for some Boolean function f; also let C be an implementation (cube cover) for f.

- If f has a 0⟶0 transition in $[m_1, m_2]$, C is free of logic hazards for the input changing monotonically from m_1 to m_2
- If f has a 1⟶1 transition in $[m_1, m_2]$, C is free of logic hazards for the input changing monotonically from m_1 to m_2 if and only if $[m_1, m_2]$ in included in C
- If f has a 1⟶0 transition in the priviledged cube $[m_1, m_2]$, C is free of logic hazards for the input changing monotonically from m_1 to m_2 if and only if C is free of any illegal intersection with the priviledged cube

Equivalent goals

Find a 2-level circuit implementation, where:

- each required cube is completely contained in some product
- no privileged cube is illegally intersected by a product

Dynamic hazard free (DHF) prime implicants
: A DHF prime implicant is a maximal implicant which has no illegal intersections with any priviledged cube

Synthesis goal may be restated as: find a 2-level circuit implementation, where:

- instead of prime implicants, DHF prime implicants are identified and used
- each required cube is completely contained in some product

How to achieve these goals?

i. Generate all DHF prime implicants
   a. Generate all prime implicants
   b. Reduce to DHF prime implicants
   c. Discard any reduced implicant covered by some other implicant
ii. Identify all required cubes, note that all on-set minterms are always covered by the required cubes
   a. Use given function hazard free input transitions to determine required cubes
iii. Generate covering problem to cover the required cubes using the DHF prime implicants
iv. Covering may not be satisfiable (solution may not exist)

An example

Inputs

Given a function and four function hazard free input transitions

AB

	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

Required cubes

Required cubes are shown in green -- each required cube must be completely contained within a DHF prime implicant

AB

	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

Dynamic 1⟶0 transition

AB

	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

RCs: A'C'D', A'BC'

AB

	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

Static 1⟶1 transition

AB

	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

RC: AC'

AB

	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

Dynamic 1⟶0 transition

AB

	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

RCs: A'C, BCD

Priviledged cubes + start points

Priviledged cubes are shown in red; any product intersecing a priviledged cube must also include its start point; otherwise the intersection is illegal and must be pruned

AB

	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

Prime implicants to DHF prime implicants

The seven prime implicants are shown in blue along with the priviledged cubes to ascertain any illegal intersections

P1 AB

✓	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

P1: C'D'

P2 AB

✓	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

P2: A'B

P3 AB

✓	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

P3: BC'

P4 AB

✓	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

P4: AC'

P5 AB

✓	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

P5: A'C

P6 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

P6: Further processing needed

P7 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

P7: Further processing needed

Reduction of prime implicants with illegal intersection

Illegal intersections are highlighted in light pink

P6 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

Original

P6 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

After first reduction

P6 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

After retaining and removing BCD,
P6: BCD

P6 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

After second reduction, P6: BCD

P6 AB

✓	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

After discarding redundant cube,
P6: BCD

P7 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

Original

P7 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

After reduction

P7 AB

✗	f	00	01	11	10
	00	1	1	1	1
CD	01	0	1	1	1
	11	1	1	1	0
	10	1	1	0	0

Discarded as redundant

Covering required cubes with DHF prime implicants

	P1	P2	P3	P4	P5	P6
AC'				✗		
A'C'D'	✗					
A'BC'		✗	✗			
A'C					✗	
BCD						✗

Essential DHF prime implicants are highlighted

Hazard free 2-level minimised SOP

- *f=C'D'+AC'+A'C+BCD+A'B* on taking P2 or
- *f=C'D'+AC'+A'C+BCD+BC'* on taking P3

Questions

- What about 0⟶1 transitions?
- Do those required separate handling?
- Enough to consider the corresponding 1⟶0 transitions?
- Can there be a minterm *m* in the on-set of the function but not included in a required cube?
- If so, then what can be said about the behaviour of the function at *m*?
- What is *m* is not reachable from any other minterm?
- Or for that matter, what is no other minterm is reachable from *m*?