Dynamic Hazards

The static hazards were caused by $x$ and $x'$ temporarily having the same value.

When we considered a gate in the network we assumed that no earlier static error occurred. Hence, any input to a gate would change exactly once to its steady-state value and multiple inputs to a gate could change at different times due to delays due to differing fan-out path lengths from the single network input that is changing to the gate.

Once a static error occurs in the network, however, it means that a gate is sending wrong information to its fan-outs possibly every time one of its inputs change. This misinformation can then spawn other static errors.

As a result, gates downstream of the static error could see a one or more inputs change multiple times before achieving its steady state. Such behavior is part of what causes dynamic hazards.
For a dynamic hazard to occur (assuming none occurred earlier in the network) on the output of a gate we must have:

- a set of inputs changing so that the output changes state (transitional inputs)

- at least one input that has a static hazard and a delay or advance appropriately related to the tranistional inputs.

Given transitional inputs we must have the following to get a dynamic hazard:

- For AND 0 → 1: static 1-hazard after all transitional inputs;

- For AND 1 → 0: static 1-hazard before all transitional inputs;

- For OR 0 → 1: static 0-hazard before all transitional inputs;

- For OR 1 → 0: static 0-hazard after all transitional inputs;

The static hazard input line and the transitional input line must be distinct, So we must track delay and static hazards on three distinct paths.
\[ A = x'_2 + y'_3 \]
\[ B = (x (x' + y'_3))' \]
\[ C = (y (x' + y'_3))' \]
\[ D = x (x' + y'_3) + y (x' + y'_3) \]

Relabel 25 to 7 26 to 8 35 to 9 36 to 0

\[ f = x_1 x'_7 + x_1 y'_9 + x'_8 y_4 + y'_0 \]

P-sets \{ x_1 x'_7 \} \{ x_1 y'_9 \} \{ x'_8 y_4 \} \{ y'_0 y_4 \}
Locating Dynamic Hazards

We next discuss how we can use our intuitive characterization of dynamic hazards and the prime implicants of $\mathcal{F}$, which we call P-sets to create an algorithm to detect dynamic hazards.

As with static hazards the presentation is based on the fact that $\mathcal{F}$ represents the steady state and transient behavior of the given network. This can be viewed as a two-level network which is equivalent with respect to hazards to the original multilevel network we analyzed to produce $\mathcal{F}$.

It is best to understand the algorithm from the point of view that we are manipulating the terms in $\mathcal{F}$ to cause the dynamic hazard (or show that one cannot exist). In this way you can reproduce the algorithm as needed rather than memorize a cookbook recipe.
Locating Dynamic Hazards

\[ f = xy' + x'y \]
\[ \mathcal{F} = x_1x_7' + y_0y_4 + x_1y_9' + x_8y_4 \]

The PI’s of the path-labelled version of \( \mathcal{F} \) are called the P-sets. If a dynamic hazard exists it is always possible to generate it using P-sets and a static 0-hazard.

Step 1: Choose an unstable P-set. \( U = [x_1, x_7'] \). (\( x \) is the unstable variable that will transition.)

Step 2: Find a stable P-set with a literal of the unstable variable which has a path independent of \( U \), indicated by subscript on the literal of the unstable variable that is not in \( U \).

For our example, since 8 does not appear in \( U \):

\[ S = [x_8', y_4] \]
**Comments:** Note that, in general, \( S \) can also contain a literal of the unstable variable with a path that is in \( U \). But there must be at least one which has a distinct path, i.e., not in \( U \). For our example, we have identified gate B as the potential static 0-hazard and an independent path through gates A, C, and D by which a transition in \( x \) can affect the output of the network.
Step 3: Set the steady state variables that appear in $U$ and $S$ so that all literals evaluate to 1 in both sets.

For our example, there is only one steady state variable $y$ which we can set to 1 to satisfy the requirement. Also since $x$ is transitioning and $y$ steady state (its subscripts can be ignored)

$$F = x_1 x_7' + x_8' y + y' y + x_1 y'$$

$$F = x_1 x_7' + x_8' \cdot 1 + y' y + x_1 y'$$
Comments: This activates $U$ for the transition of the unstable variable and sets $S$ so that its value is a function of the unstable variable only ($x'$ in this case) and therefore can be toggled via the transition.
Step 4: Assign any remaining steady state variables so that all remaining stable P-sets are inactive or satisfy the following:

- has a literal of the unstable variable with a subscript different from those appearing in $U$.
- the literals of the unstable variable are the same as that in $S$ (possibly with a different subscript), i.e., if $S$ contains $x'$ then the active stable P-set must also have $x'$ and not $x$.

Note: Setting the remaining steady state variables, if any, also makes inactive all other unstable P-sets with unstable variables different from that in $U$. Unstable P-sets with the same unstable variable as $U$ do not matter.

For our example, there is only one steady state variable which has already been set so we need only check consistency with the conditions above. The only other stable P-set is $[x_1, y'_9]$. Since the subscript 1 appears in $U$ this set must be made inactive. But, $y = 1 \rightarrow y'_9 = 0$ and the set is inactive.

$y = 1$ is steady state, $x$ is unstable

\[
\mathcal{F} = x_1x'_7 + x'_8 \cdot 1 + y'y + x_1y' \\
\mathcal{F} = x_1x'_7 + x'_8 \cdot 1 + 0 \cdot 1 + x_1 \cdot 0 \\
\mathcal{F} = x_1x'_7 + x'_8
\]
\[ \mathcal{F} = x_1 x'_7 + x'_8 \cdot 1 + y'_0 y_4 + x_1 y'_9 \]
\[ \mathcal{F} = x_1 x'_7 + x'_8 \cdot 1 + 0 \cdot 1 + x_1 \cdot 0 \]
\[ \mathcal{F} = x_1 x'_7 + x'_8 \]

\( y = 1 \) is steady state, \( x \) is unstable

**Comments:** This step is the heart of the algorithm. The two conditions address two different aspects of the problem of determining the timing to combine an independent toggle of the output with a static 0-hazard, i.e., a transient 1 output.

Clearly, the inactive P-sets are removed completely from consideration since they are 0 for the steady state assignments before and after \( x \) transitions.

The condition that an active P-set other than \( S \) have a path independent of \( U \) is necessary to allow the hazard generated by \( U \) to appear on the output of the network or to guarantee that we have a dynamic hazard.
For example, suppose we had \( U = [x_1, x'_7] \), \( S = [x'_8, y_4] \) and another stable active P-set \( Z = [x_1, y_4] \). The expression for \( \mathcal{F} \) given the steady state assignment \( y = 1 \) would be:

\[
\mathcal{F} = x_1 x'_7 + x_1 + x'_8 \\
\mathcal{F} = x_1 + x'_8 \\
f = x + x'
\]

by absorption and the hazard would be a static hazard not dynamic.
On the other hand, suppose we had $U = [x_1, x'_7]$, $S = [x'_8, y_4]$ and another stable active P-set $Z = [x'_7, y_4]$. The expression for $\mathcal{F}$ given the steady state assignment $y = 1$ would be:

\[
\mathcal{F} = \begin{align*}
&= x_1 x'_7 + x'_7 + x'_8 \\
&= x'_7 + x'_8 \\
of &= x'
\end{align*}
\]

by absorption. So we have dynamic behavior but no way to cause a hazard with $U$.  


The second condition also guarantees that the output of the network is in fact changing for the two steady state assignments before and after $x$ changes. For example, suppose we had $U = [x_1, x'_7]$, $S = [x'_8, y_4]$ and another stable active P-set $Z = [x_2, y_4]$. Subscript 2 is not in $U$ so the first condition is satisfied. However, the expression for $\mathcal{F}$ given the steady state assignment $y = 1$ would be:

$$\mathcal{F} = x_1 x'_7 + x_2 + x'_8$$

$$f = x + x' = 1.$$ 

For our example, since there are no other steady state variables to manipulate this could not be fixed and there would be no dynamic hazard generated by $U$ and $S$. We would have to consider another pair.
So for our example we have found that this realization of the exclusive or, $xy' + x'y$, has a dynamic hazard when $y = 1$ and $x$ is in transition.

U indicates a static 0-hazard on the output of gate B (where paths 1 and 7 meet). We have two possibilities:

$$
y = 1 \quad x : 0 \to 1 \quad D : 1 \to 0
$$

$$
y = 1 \quad x : 1 \to 0 \quad D : 0 \to 1
$$

The first will have a dynamic error if the timing of the network is such that the static error propagates from B to D after the new steady-state value of 0 has been reached by D, i.e., if the path through A to C to D settles faster than that through B to D. If this timing is not the case then a dynamic hazard will not result.

The second will have a dynamic error if the static hazard propagates through B to D before the signal through A to C to D settles to 1.
For the second scenario we have the following bit sequences. (static 0-hazard first then transition.)

\[ F = x_1x'_7 + x'_8 \]

<table>
<thead>
<tr>
<th></th>
<th>( x_1 )</th>
<th>( x'_7 )</th>
<th>( x'_8 )</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
The network is symmetric so we would expect a dynamic hazard in the network with \( y \) unstable.

Step 1: \( U = [y'_0, y_4] \)

Step 2: \( S = [x_1, y'_9] \)

Step 3: Setting \( x = 1 \) makes \( U \) and \( S \) active for transitions of \( y \).

Step 4: The remaining stable P-set \([x'_8, y_4]\) contains \( y_4 \) and \( S \) contains \( y'_9 \). Also the subscript of the literal of \( y \) that appears in this P-set also has a subscript that appears in \( U \). For both reasons we must make it inactive in order for a dynamic hazard to be detected. But we have already set \( x = 1 \) so \( x'_8 = 0 \) and the P-set is inactive.

A dynamic hazard with \( x = 1 \) and \( y \) in transition exists and

\[
\mathcal{F} = y'_0y_4 + x_1y'_9 \\
\mathcal{F} = y'_0y_4 + y'_9
\]

determines the paths and timing needed.
We have only used unstable 1-sets or P-sets to generate the static 0-hazards that induce dynamic hazards. What about static 1-hazards? Recall that static 1-hazards are associated with unstable 0-sets or S-sets.

The S-sets can be computed as follows for this network:

\[
\begin{align*}
\mathcal{F} &= x_1 x'_7 + y'_0 y_4 + x_1 y'_9 + x'_8 y_4 \\
\mathcal{F}^D &= (x_1 + x'_7) (y'_0 + y_4) (x_1 + y'_9) (x'_8 + y_4) \\
\mathcal{F}^P &= (x_1 + y'_9 x'_7) (x'_8 y'_0 + y_4) \\
\mathcal{F}^P &= x_1 y_4 + x_1 y'_0 x'_8 + y_4 x'_7 y'_9 + x'_7 x'_8 y'_0 y'_9 \\
\mathcal{F} &= (x_1 + y_4) \cdot (x_1 + y'_0 + x'_8) \cdot (y_4 + x'_7 + y'_9) \\
&\quad \cdot (x'_7 + x'_8 + y'_0 + y'_9)
\end{align*}
\]

Stable S-sets:

\[[x_1, y_4] \quad [x'_7, x'_8, y'_0, y'_9] \]

Unstable S-sets:

\[[x_1, y'_0, x'_8] \quad [y_4, x'_7, y'_9] \]
An algorithm similar to the P-set one can be used to manipulate this form to generate a dynamic hazard in a Product of Sums form.

\[
F = (x_1 + y_4) \cdot (x_1 + y'_0 + x'_8) \cdot (y_4 + x'_7 + y'_9) \\
\cdot (x'_7 + x'_8 + y'_0 + y'_9) \\
F = (x_1 + y_4) \cdot (x_1 + 0 + x'_8) \cdot (y_4 + x'_7 + y'_9) \\
\cdot (x'_7 + x'_8 + 0 + 0) \\
F = (x_1 + 1)(x_1 + 0 + x'_8)(1 + x'_7 + 0)(x'_7 + x'_8 + 0 + 0) \\
F = (x_1 + x'_8)(x'_7 + x'_8)
\]

Step 1: \(U = [x_1, y'_0, x'_8]\)

Step 2: \(S = [x'_7, x'_8, y'_0, y'_9]\)

Step 3: \(y = 1\) makes \(S\) and \(U\) active for transitions of \(x\).

Step 4: \(Z = [x_1, y_4]\) must be made inactive since subscript 1 appears in \(U\). \(y = 1\) does make it inactive. The other unstable S-set is also inactive.

So we have completed the procedure successfully for the sets \(U = [x_1, y'_0, x'_8]\) and \(S = [x'_7, x'_8, y'_0, y'_9]\). So by the theorem there is a dynamic hazard with \(x\) changing and \(y = 1\) as expected. (The hazard with \(y\) changing can also be deduced via S-sets.)

Note using P-sets we had \(F = (x_1x'_7 + x'_8)\) which is just the S-set expression multiplied out.
What does it mean in terms of delay paths? We are using a static 1-hazard instead of the static 0-hazard used with P-sets.

\[
\mathcal{F} = (x_1 + x'_8)(x'_7 + x'_8) \\
\mathcal{F} = x'_1 x'_7 + x'_8
\]

Before we had a static 0-hazard first then transition yielding the same bit sequences. But now we interpret it as a static 1-hazard on gate D (meeting point of paths 1 and 8) after the transition! To see this note that \(x'_7\) changes first then the hazard is caused by \(x_1\) and \(x'_8\).

<table>
<thead>
<tr>
<th>(x_1)</th>
<th>(x'_7)</th>
<th>(x'_8)</th>
<th>((x_1 + x'_8)(x'_7 + x'_8))</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>((1+0)(0+0))</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>((1+0)(1+0))</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>((0+0)(1+0))</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>((0+1)(1+0))</td>
<td>1</td>
</tr>
</tbody>
</table>
We need only consider P-sets or S-sets not both for the purpose of detecting dynamic hazards.

This can be summarized in the following:

**Theorem** If a network contains a dynamic hazard, it must contain at least one pair of unstable sets, one of which is a P-set and the other of which is an S-set. The sets must share the same unstable variable and any other variable that appears in both must be complemented in one and uncomplemented in the other.

The complemented/uncomplemented statement follows from the different definitions of active for P-sets and S-sets.
**Theorem:** A two-level network in which no first level gate has a variable and its complement as input contains no dynamic hazards.

**Proof:** Recall the proof that an AND/OR network contains no static 0-hazards since it can have no unstable 1-sets. Similar reasoning shows it can have no unstable P-sets and therefore no dynamic hazards by the corollary above.

Similar reasoning yields the dual result for the OR/AND network to complete the proof.

□.
Summary of hazards:

<table>
<thead>
<tr>
<th>network</th>
<th>static 0</th>
<th>static 1</th>
<th>dynamic</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND/OR</td>
<td>no</td>
<td>possible</td>
<td>no</td>
</tr>
<tr>
<td>OR/AND</td>
<td>possible</td>
<td>no</td>
<td>no</td>
</tr>
<tr>
<td>multilevel</td>
<td>possible</td>
<td>possible</td>
<td>possible</td>
</tr>
</tbody>
</table>

**Fundamental Hazard Theorem for Two-level Networks**: An AND/OR (OR/AND) network is hazard-free if it has no static 1-hazards (static 0-hazards).