Complex reset architecture of multiple reset sources splits SoC chip into several reset domains.

Data signal metastability captured on a Reset Domain Crossing (RDC) path can cause chip to fail.

Dependency of reset domain of a sequential on concurrent assertion/de-assertion of several reset sources complicates RDC analysis.

RDC verification of SoC, in addition to critical bugs, may also catch false crossings having overlapping reset domains.

Complexity Of Reset Domain Verification Analysis

- Rise in functional modes leads to rise in combinational resets resulting in complex reset architecture.
- Two categories of crossings having multiple dependent reset assertions, reported as bugs by traditional methodologies:
  1. All resets in source domain impact destination domain
  2. Some resets in source domain do not impact destination domain

Reset Detection Optimization Methodology

- The proposed method utilizes a combination of structural reset analysis, expression analysis, and functional analysis to prune noisy and inefficient RDC paths in the design during static verification.

  Stage 1: Comprehensive expression analysis of the resets at source and destination to identify and prune safe candidates.

  Stage 2: Propagation of user defined reset sequencing across combinational resets in addition to reset expression analysis to identify and prune safe candidates out of second category crossings.

CASE STUDY

- The proposed methodology was benchmarked on a highly complex real SoC with more than 1.8 million registers, and 5 RAMs.

- Out of Identified 287 reset domains including asynchronous and synchronous resets, 90k RDC crossings detected during RDC analysis.

- Comparison of RDC results on the SoC with and without proposed methodology, with reset grouping and reset ordering applicable:

<table>
<thead>
<tr>
<th>Reset Domain Crossings</th>
<th>Number of crossings</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDC crossings with source and destination registers in different asynchronous reset domains without proposed methodology</td>
<td>34562</td>
</tr>
<tr>
<td>RDC crossings with source and destination registers in different asynchronous reset domains with proposed methodology</td>
<td>23929</td>
</tr>
<tr>
<td>Ordered RDC paths based on sequencing information without proposed methodology</td>
<td>23792</td>
</tr>
<tr>
<td>Ordered RDC paths based on sequencing information with proposed methodology</td>
<td>27893</td>
</tr>
</tbody>
</table>

- Around ~34% crossings pruned as false paths, ~20% increase in ordered crossings.

CONCLUSIONS

- Proposed automatic technique improves quality of results for static RDC analysis, and reduces closure time required for real RDC issues.
- Improved tool performance.
- Methodology ensures no critical path is missed and false crossings are pruned.
- Advanced techniques utilize reset ordering information as well to simplify the reset architecture and enhances the tool capabilities.

REFERENCES

