Trouble With Scan

David M. Wu, Intel

Since I have been a Scan DFT advocate for years, I thought it may be fun to take a different position just to make this panel discussion more interesting, my statement here is certainly neither representing my company’s position nor reflecting my passion for scan DFT.

Knowing benefits of scan well, my intent here is to point out what kind of trouble we have with scan and BIST. And then I would argue that functional testing methods might give us a better quality if we can afford resource for manual test generation.

First of all, scan DFT based on only two major fault models: stuck-at-fault and transition fault. Beyond these two fault models, the effectiveness of scan DFT is debatable. To achieve the kind of quality level we are looking for, more fault models will need to be applied in order to capture all types of defects. But even if we know how to create additional fault models (other than stuck-at and transition faults), how will we automatically generate test patterns for these additional models? And, how can any test generator or tester handle such a huge test data volume? How are we going to fault isolate these additional defect types?

Someone may argue that one can use BIST on the top of the stuck-at and transition faults and apply millions of test patterns to detect 'non-modeled' defects. If you are simple minded, it sounds pretty convincing, but BIST does not come for free. In order to do a good BIST, a very strict set of design-for-test rules need to be enforced and it is generally not so easy to comply with all of the BIST rules. In addition, BIST really takes longer test time than the traditional scan test, simply because BIST apply much more patterns than the scan tests. Some one may argue that BIST can create many times of scan chains comparing to scan test and therefore it could save overall test time. But, please do not forget that too many scan chains will complicate routing and sometimes it becomes unacceptable.

Scan DFT does not come for free. The average full-scan overhead could range from 4%-8% depending on the quality level requirement of the target product. Sometimes, it may create performance impact if designers did not take loading into design consideration from the very beginning. Implementing scan DFT is like paying income taxes, the more money you make (the more test coverage you need), the more taxes you have to pay (the more silicon area you need to spend). To ‘sell’ the scan DFT to designers and enforce the DRC rules is a hair-twisting task. A DFT engineer is like an IRS auditor who always trying to make designers to believe that without DFT, TTM will suffer (just like if nobody paid taxes, there won’t be good education, or good highways).

How about speed binning? Can scan do an adequate job with scan only? Even if scan test can launch and capture test patterns at the system speed, the test patterns applied may not cover all critical paths or it may cause yield loss. Many companies tried to do speed binning with scan and/or BIST, but the success only limited to ASIC or lower frequency products. It will be a difficult task if one will need to do speed binning for a high-speed, high-density microprocessor using scan DFT only.

I guess one more thing I can think of is all of those analog circuits and high-speed IOs. They can certainly not be tested by scan DFT. In this case, functional test methods are more effective than scan DFT. I would argue that functional test really can do a very good job in detecting defects if one can generate all required functional test patterns within a short period of time. Functional test patterns exercise chip functionalities at the system speed. If can do better job in speed binning and detection of small delay defects in the critical paths.

In conclusion, I want to remind the readers again, my statement here is certainly neither representing my company’s position nor reflecting my passion for scan DFT. But we do still have some trouble with scan and I would like to hear from some test experts' opinion about 'can scan give us the kind of quality level we are looking for?'