Test-Quality Comparison between Full-Scan, Partial-Scan and On-Line Techniques for a Periodic Synchronizer

Octavian Petre, Hans G. Kerkhoff
MESA+ Research Institute
Testable Design and Testing of Microsystems Group
P.O. Box 217, 7500 AE Enschede, The Netherlands
E-Mails: O.Petre@el.utwente.nl, H.G.Kerkhoff@el.utwente.nl

Abstract—Because of shrinking minimum feature sizes, IC clock frequencies are increasing. It is no longer possible, nor desired, to distribute a single clock over the entire IC. Multiple-clock domains design will no longer be an isolated design style. This new trend in the industry, referred to as future standard by some companies, poses many test problems due to special modules utilized at the interface between clock domains. These modules are called synchronizers. This paper will present and compare different test strategies used for testing multiple-clock domain systems in general and synchronizers in particular.

Keywords--full-scan testing, partial-scan testing, on-line testing, periodic synchronizer

I. INTRODUCTION

The new industry trend in the field of IC digital design is to use more than one clock domain for Systems-on-Chip (SoC). This design approach has proven to be successful in the following cases:

- The target implementation is comprised of high-speed parts where the clock cannot be distributed over the entire IC without speed penalty
- The designer has to deal with clock-less data out of which the clock must be derived
- There is a need for different clocks domains (e.g. required in some communicating systems).
- There are power constrains on the design and clock gating must be used.

In all of the above-mentioned cases, a different test strategy must be employed since the well-known full-scan strategy cannot handle the diversity in digital design techniques without an increase of untested faults. However, the trend is to move to high-speed system designs, which will lead inevitably to multiple-clock domains systems and ever more complicated synchronization and test strategies.

Our paper will concentrate on testing multiple-clock domains systems, which have uncorrelated clocks and use advanced synchronizers schemes to transmit data between them. This article will present a comparison between the scan techniques available for synchronous design style and an enhanced on-line technique.

In section II, a brief overview of the testing issues in multiple-clock domains systems will be presented. Section III will describe the implementation of the full-scan technique for the periodic synchronizer and the limitations introduced by this method. Section IV will present the partial-scan technique used to overcome some limitations of the full-scan technique. In section V, the enhanced on-line test strategy applied to the periodic synchronizer will be introduced. In section VI a comparison between all the presented methods is given. Finally, in section VII the conclusions will be given.

II. MULTIPLE-CLOCK DOMAINS AND RELATED TESTING ISSUES

In order to ensure reliable data transfer between different clock domains, many interesting synchronizer designs have been suggested over the years. Synchronizer designs can be classified based on the relation between clock domains [1]; their main purpose is to decrease the probability of metastability phenomena [1, 4, 5]. These phenomena may occur whenever a flip-flop samples data that is changing during the flip-flop's aperture time. In this special situation, the flip-flop may exhibit a metastable state and the output signal will not be defined as a binary value for an arbitrary duration.
This metastable probability must be sufficiently small so the data, that is to be fed in the subsequent combinational logic circuit (CLC), is not jeopardizing the normal functionality of the IC. This probability can be as low as $3.86 \times 10^{-24}$ as discussed in reference [1].

Testing problems of synchronizers arise due to the usage of analogue modules like phase comparators, controllable delay lines, etc. Since synchronizers are surrounded by digital logic, they are assumed to be tested using digital techniques.

The most common methodology used today for testing digital IC’s is the full-scan technique. This technique is widespread and during the last years, it has been enhanced for tackling multiple-clock domains designs.

The main idea and requirement is that clock domains must have controllable clocks. This assumption is fatal for synchronizers since some of them incorporate delay lines, which create very small clock domains in the test-mode (i.e. clock domains, which only control very few flip-flops). This issue is discussed in more detail in the following section.

III. THE FULL-SCAN TECHNIQUE APPLIED TO THE PERIODIC SYNCHRONIZER

In order to illustrate the problem discussed above, the periodic synchronizer (PS) shown in Figure 1 (without the multiplexers) has been chosen.

An explanation of the behaviour of the design is given in [7, 8]. As one can notice, delay lines are used to synchronize the data between the clock domains.

The testing tool recognizes these delayed versions of the clock as different clock domains that must be controllable in test mode. The added multiplexers in Figure 1 are not fully tested since in test mode the select input of the multiplexers is set to a fixed value.

Using the straightforward full-scan technique to test the PS, 106 stuck-at faults remain undetectable, mainly due to the untested multiplexers (MUX); the faults within the Predictor module have not been introduced in the fault list since it is mainly an analogue module. The area overhead of the Design-for-Test structures, specified in NAND gates, is 90.

IV. PARTIAL-SCAN TECHNIQUE APPLIED TO THE PERIODIC SYNCHRONIZER

As shown in the previous section, the main disadvantage of the full-scan technique is the insertion of the MUX’s in all the delay lines. One may think of a different strategy in order to avoid the MUX’s insertion. A partial-scan technique can be used, so that the flip-flops that are clocked using the $dclk$ clock will not be replaced with their scan counterparts. Of course, one wants to have a fault coverage as high as possible. Therefore, those non-scanable flip-flops will be modelled, for the sake of the ATPG tool, as purely combinational buffers. The proposed modelled structure is shown in Figure 2. As one may notice, only one of the two flip-flops has been replaced with a scan version since the subsequent one is already a scan flip-flop by default.

This strategy works only perfectly for the data-path, which contains all digital modules below the keepout signal. For the control part, all the flip-flops must be made scanable just like in the full-scan technique. Using the partial-scan technique, the number of stuck-at-faults that are not tested drops to 26. This is a major improvement compared with the full-scan version. The area overhead in this case is also less, since fewer flip-flops are transformed in scan flip-flops; it is 18 NAND gates equivalent.
V. ADDITIONAL HARDWARE FOR ON-LINE TESTING

The general on-line solution for testing the synchronizers is presented in Figure 3.

Apart from the synchronizer itself, one recognizes an additional synchronization cell (shaded area), two code generators (CG) and a comparator module (Comp). All of the above mentioned circuits add redundancy to the data path. The CG module generates the \( x_{\text{redundant}} \) signal, which after the synchronization becomes \( x_{s_{\text{redundant}}} \). At the same time, the data signals \((x_0 \ldots x_7)\) are also synchronized and the resulting values are \( x_{s0} \ldots x_{s7} \). A second CG module calculates another redundant signal; the result is compared with the synchronized version of \( x_{\text{redundant}} \). If there is a difference between these two signals, then the synchronizer may not work properly due to a fault.

In our case, these two code generators are in fact parity-code generators (PCG). This strategy has been applied to the synchronizer in Figure 1.

The presented solution detects any stuck-at fault in the data path whenever they are justified. The probability that a fault, which is present in the data-path, will be detected after some cycles is calculated in [7].

The straightforward application of the on-line test strategy is only useful for the synchronizers without a control path (e.g. brute-force synchronizer). The above methodology leave all the faults presented in the control path untested. These faults decrease the MTBF (Mean-Time Between Failure)[1,3-5].

To solve this testing issue, the following solution has been implemented.

In order to detect any stuck-at fault in the control path on-line, one must understand the behaviour of the keepout signal.

If a stuck-at fault is present at the input D or output Q lines of any of the four flip-flops (the flip-flops above the keepout signal), then the keepout signal will remain high for more than one clk period, if the clk period is at least 4 times higher than the xclk period. With the previously mentioned restriction, the correct operation of the signal keepout is to change for one clk period from time to time, or not change at all, depending on the xclk frequency.

A design that is able to monitor these conditions is shown in Figure 4.

Since a branch in the keepout signal is introduced, the stuck-at-faults of the keepout stem from the data-path MUX's will remain undetected. In order to avoid this situation, the keepout signal was encoded using a dual-rail representation; all the 2:1 MUX's from the data-path were replaced by 4:1 MUX's. The two extra inputs of the new MUX's were tied to the complement values of signals to be synchronized, as can be seen in Figure 5.
One may also notice the insertion of eight test points [9]. These test points behave as wires in the functional mode. Many commercial tools use them to increase the fault coverage when random test pattern generation is performed. In our case these test points are used in scan mode to test all the faults that may appear at the branch of the synchronized signals, $x_{s0} \ldots x_{s7}$, feeding C2.

Before proceeding, two comments about the on-line structure for testing the synchronizers are required:
1. The on-line structure tests only the synchronizer module and not the surrounding logic.
2. The surrounding logic is fully synchronous with a particular clock and is therefore tested using full/partial-scan strategy. The 8-test points insertion are also included in this strategy.

It will be shown later on that all the faults that may appear are either detected by the structure in Figure 4, the PCG blocks or the surrounding scan methodology.

A. Faults in the data-path

All the justified stuck-at faults in the data path are detected using the PCG’s. The PCG’s are fully testable. Any fault within those two modules is detected by means of the Pass/Fail signal.

The $x_{s0} \ldots x_{s7}$ branches towards C2 are tested by the scan technique using the inserted test points.

B. Faults in the control-path following the keepout splitting

For the control signals $\text{keepout}$ and $\overline{\text{keepout}}$ an interesting situation appears. The faults that may emerge in these signals (stem and branches) are ‘converted’ to faults in the data-path by choosing the complement values of the signal to be synchronized rather than the correct value. The PCG blocks can subsequently detect these faults. However, by using this detection strategy an even number of bits must be synchronized. Otherwise, the PCG’s will mask the faults on the $\text{keepout}$ and $\overline{\text{keepout}}$ stems.

C. The other faults in the control-path

The structure in Figure 4 is used to reveal the faults that are in front of the splitting of the $\text{keepout}$ signal as explained earlier. The logic in Figure 4 is tested using the standard full-scan technique with the signal TE tied to “1”.

The fault coverage is not reduced since the case under the condition $\text{TE} = 0$ is tested on-line

The final scheme of the PS with all the on-line structures is shown in Figure 5.

Using the on-line test technique, there are only four stuck-at-faults that cannot be tested, which is the best result so far. These undetected stuck-at-faults occur at:
- The stem of the $\text{pxclk}$ signal s-a-0/1,
- The right branch of the $\text{keepout}$ s-a-0
- The right branch of the $\overline{\text{keepout}}$ signal s-a-1.

The area overhead for the on-line structures, including the test points insertion, is 267.5 equivalent NAND gates.

VI. COMPARISON BETWEEN DIFFERENT TEST STRATEGIES

In Table 1 the comparison is made between the three methods presented in the previous sections, in terms of area-overhead, number of un-testable stuck-at-faults, etc.

As can be seen, the best method to assure a high quality testing for the PS modules, is the on-line technique even if the area overhead is high comparable to the other methods.

<table>
<thead>
<tr>
<th></th>
<th>Full Scan</th>
<th>Partial Scan</th>
<th>On-line</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Area Overhead (AO)</strong> in NAND gates</td>
<td>90</td>
<td>18</td>
<td>267.5</td>
</tr>
<tr>
<td><strong>Untested stuck-at faults (0/1)</strong> (USaF)</td>
<td>106</td>
<td>26</td>
<td>4</td>
</tr>
<tr>
<td><strong>AO*USaF</strong></td>
<td>9540</td>
<td>468</td>
<td>1070</td>
</tr>
<tr>
<td>Suitable for production test</td>
<td>YES</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>NO Delay line redesign</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>Suitable for safety critical applications</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>At-Speed-Test</td>
<td>NO</td>
<td>NO</td>
<td>YES</td>
</tr>
</tbody>
</table>

VII. CONCLUSIONS

In this paper, different test strategies are presented for testing a periodic synchronizer. The approaches are mainly characterized from the point of view of area-overhead and stuck-at fault coverage.

The proposed on-line test method provides the highest fault-coverage among the three techniques. Another benefit of the on-line method is that during the life cycle of the component any s-a fault that might
appear is immediately detected, suggesting the use of this technique in safety critical applications. Another advantage is that the on-line DfT hardware is an add-on component, which does not interfere with the design of the delay-lines. The automatic design process benefits from this feature.

The next step in our research will be to further investigate the faults that may occur in the control parts of the synchronizers and to study their effect on the MTBF parameter. In addition, the introduction of delay faults in our model and the reduction of DfT area overhead are envisioned.

ACKNOWLEDGEMENTS

The Technology Foundation STW, applied science division of NWO, has financially supported this research.

REFERENCES


