SCAN CELL DESIGN FOR ENHANCED DELAY FAULT TESTABILITY

Gerrit van Brakel, Yizi Xing and Hans G. Kerkhoff,
MESA Research Institute, University of Twente,
P.O.Box 217, 7500 AE Enschede, the Netherlands ·
tel. +31-53-892649; fax +31-53-341903.

Abstract
This paper addresses problems in testing scannable sequential circuits for delay faults. Modifications to improve circuit controllability and observability for the testing of delay faults are implemented efficiently in a scan cell design. A lay-out on a gate array has been designed and evaluated for this scan cell.

1. Introduction
As the timing of VLSI-circuit designs becomes more optimized, their sensitivity to timing failures due to delay faults increases. To prevent performance reduction in reliability and speed, circuits must be tested against delay faults. Testing a delay fault in a combinational circuit can be done by supplying two successive vectors of values, called the initial and the final vector, to the inputs of the circuit under test (CUT) and observing the behaviour at the outputs. The CUT contains a delay fault if any output does not stabilize to its correct value within one operational clock period after the supply of the final vector. The combinational part of a sequential circuit can be tested similarly. In this case, however, parts of the initial and the final vector must be supplied at the state register outputs, and part of the observation must take place at the state register inputs. For static fault testing these tasks are commonly eased by using a scan path.
This paper addresses the problems faced when a scan path is to be used for delay fault testing. After a discussion of problems in Section 2 and possible solutions in Section 3 we will present in Section 4 a scan cell design that provides enhanced delay fault testability with minimal overhead in area and control signal lines.

2. Delay fault testing using a scan path

2.1 circuit controllability
If a scan path is used to for delay fault testing, many intermediate states may occur at the inputs of the combinational part of the circuit before the desired initial vector changes to the final vector at these inputs. With respect to the test these intermediate states must be interpreted as hazards that may invalidate the test. To avoid test invalidation the appearance of these hazards must be prevented. Obtaining the final vector from the initial vector by shifting only one position guarantees a hazard free transition, but this imposes severe restrictions on possible combinations of initial and final vectors [1], which make a conventional scan path unsuitable for delay fault testing. In [1] a modified scan cell is proposed that circumvents the hazard problem by allowing to invert each bit in stead of shifting. This increases the number of testable delay faults, but still imposes restrictions on input vectors. In [2] an extra slave latch is added to a Level Sensitive Scan Design (LSSD) scan cell, which is operated by a separate clock. In Section 3 we'll propose to modify the scan cell design by adding a slave latch clocked by the system clock and controlling the clocking of the regular slave latch by a multiplexer.

2.2 circuit observability
Observation of the CUT is often done by sampling its output values at the end of the clock period. However, observation of a correct signal value does not guarantee absence of a delay fault, as the presence of a hazard after the sampling time invalidates the test result. Although test invalidation can sometimes be prevented by using robust tests, generally a majority of delay faults cannot be tested robustly, so that non-robust tests must be used [4,5]. One way to prevent test invalidation for non-robust tests is to monitor the CUT output values for stability after the end of the clock period. The absence of instability for a long period combined with a correct sampled value suggests the gates under test to be free of delay faults. Monitoring enables us to collect accurate test results using non-robust tests. In section 3 we'll propose to add instability sensing circuitry to scan cell designs to increase fault coverage and aid diagnosis.

0-7803-0768-2/92$03.00 ©IEEE

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on December 16, 2009 at 10:03 from IEEE Xplore. Restrictions apply.

2.3 test timing

To enable a sharp discrimination between the faulty and the fault free case the timing of delay fault tests must be precise. The timing of scan paths however is generally slow and uncertain, because the data paths between scan cells can be long and have low priority in optimization. Therefore the timing of the test should not rely on the timing of the scan path. In the design presented in section 4 the time critical events are controlled locally, independent from the scan path, to achieve optimal accuracy.

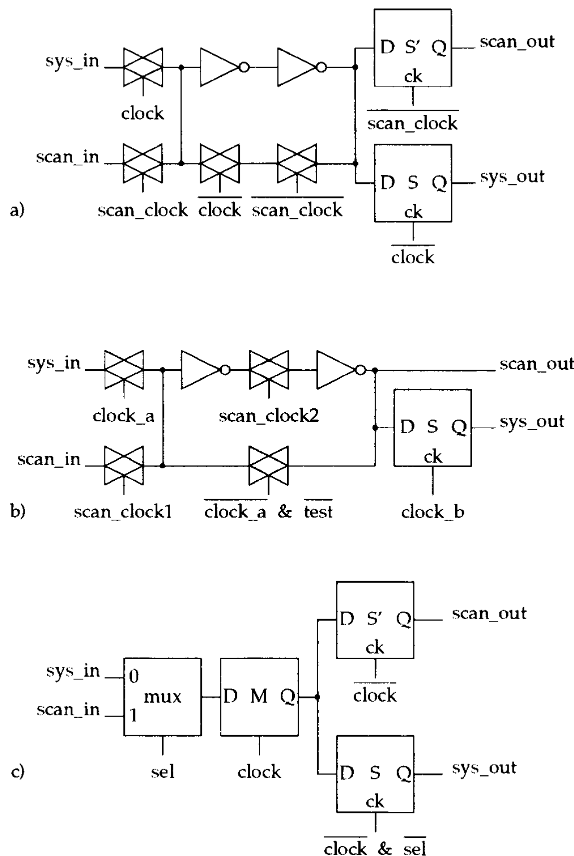

fig. 1. hazard free scan register cells

3. Scan cell building blocks

3.1 a hazard free scan cell

To provide a race free scan register cell (SRC) two memory elements (latches) are necessary, commonly called 'master' and 'slave'. Shifting data through the scan path is done by clocking master data into slaves alternated by clocking slave data into masters. If the SRC is to be used in an environment where transitions on the output to the system during shifting are considered to be harmful hazards, each output value is to be held by an additional memory element. This means at least three memory elements are necessary to provide a race and hazard free scan cell. Fig. 1 shows some hazard free SRCs. Fig. 1a shows the modified LSSD-cell from [2] with separate clocks for scanning and system operation. Fig. 1b shows a scan cell circuit that converts in scan mode the static master latch to two dynamic memory elements [3]. During normal operation *scan_clock1*=0, *scan_clock2*=1 and *test*=0. Data is clocked in using *clock_a* and *clock_b*. In scan mode *clock_a*=0, *clock_b*=0 and *test*=1. Shifting is done dynamically, applying a two phase clocking scheme to *scan_clock1* and *scan_clock2*. This scan cell implementation has only one additional transmission gate, driven by *scan_clock2*, when compared to a conventional scan cell. However there is a penalty of two additional signals necessary to control the cell. Our solution, shown in fig. 1c, has a moderate overhead in transistors but does not need any additional control signals. The circuit consists of a D-flip flop with an extra slave latch. Data is clocked both in normal and in scan mode by the *clock* signal. Selection between system data and scan data is done by a multiplexer controlled by the *sel* signal. Using a selection signal instead of an extra clock signal enables us to control a stability sensor, as we'll see in section 4.

3.2 instability sensing

Instability in a signal during a time interval can be detected either by detecting the transition that causes it, or by detecting that both a logic 0 as well as a logic 1 have been present during the interval. Fig. 2 presents circuits using either one of these principles. The circuit in fig. 2a relies on the short circuit current flowing in a CMOS circuit during a transition [6]. This current discharges the node that is charged when the circuit is reset. The discharging of the node can be sensed by a sense amplifier. The circuits in fig. 2b and 2c detect instability by identifying both signal levels to have been present during some interval. Both use two set-reset flip-flops to record the presence of logic levels during the interval. The flip-flops of fig. 2c are implemented dynamically around circuit nodes that can be charged and discharged. In each circuit one flip-flop is used to detect a logic 1 and the other to detect a logic 0 on the *sense* input. If both logic levels were present, an instability is detected. Although the circuit in fig. 2a is small compared to that of fig. 2b and 2c, it has the disadvantage of being critically dependent on parasitic effects. Simulations show that due to capacitive cross-talk the circuit does not operate properly. Moreover this circuit is only realizable in CMOS. In our design we'll use the circuit of fig 2c, that is compact and reliable for the short period of time that hazard data need be stored.

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on December 16, 2009 at 10:03 from IEEE Xplore. Restrictions apply.

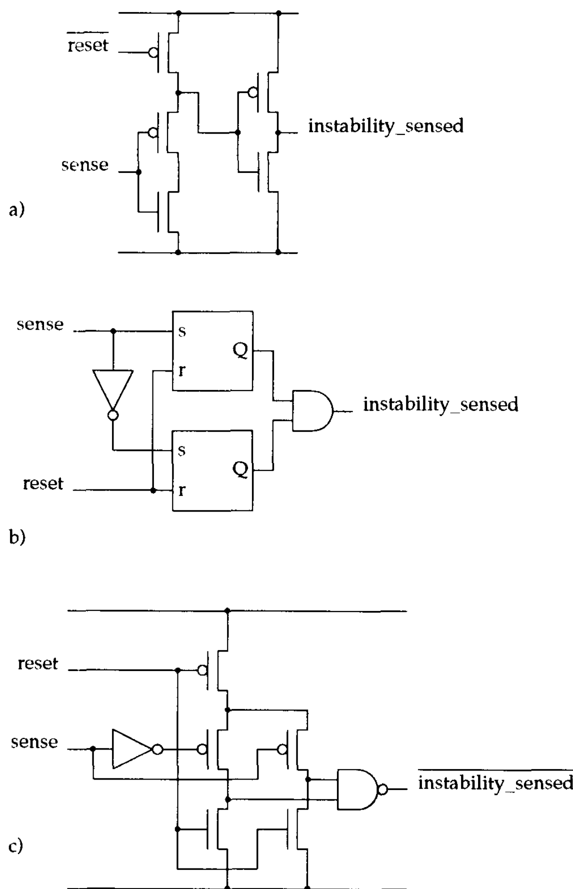

fig. 2. Instability sensing circuits.

4. An enhanced scan cell design

Fig. 3 shows a scan cell design that combines the hazard free scanning mechanism and the instability sensing circuitry in a way that minimizes overhead in area and signal lines. The scan cell is built around the shift register cell from fig. 1c and an instability sensor from fig. 2c. Both elements can be replaced by the alternative implementations of fig. 1 and fig. 2 when desired. The cell is controlled by a clock and a select line. The select line controls three multiplexers and the instability sensor. One of the multiplexers is used on the falling edge of the clock signal to select between system input (*sel*=0) and scan input (*sel*=1). Another multiplexer is used on the rising edge of clock, to select between scan data (*sel*=0), fetched in the previous half clock cycle from system input or scan input, and instability sensor data (*sel*=1). The last multiplexer guards the system slave latch from clocking in new values during the period that *sel*=1.

4.1 detailed description of operation

In general a test with instability detection is executed by:

- setting *sel*=1;
- clocking in the initial vector through the scan path;
- setting *sel*=0 for a short time while *clock*=0 to make the initial vector available to the system when each data bit has arrived in it's corresponding master latch.
- clocking in the final vector in the same way;
- making the final vector available to the system by setting *sel*=0. Now the transition runs on the path;
- setting *sel*=1 to reset the instability sensor;
- setting *sel*=0 to activate the instability sensor at the time the signal should have stabilized;
- rising *clock* at the end of the observation time to sample the output value of the instability sensor in the slave latch that drives the scan path;
- setting *sel*=1 and shifting out all instability sensor data through the scan path;
- sampling the value at *sys_in* into latch *M* by setting in order: *clock*=1, *sel*=0, *clock*=0;
- setting *sel*=1 and clocking out the steady state values of *sys_in*.

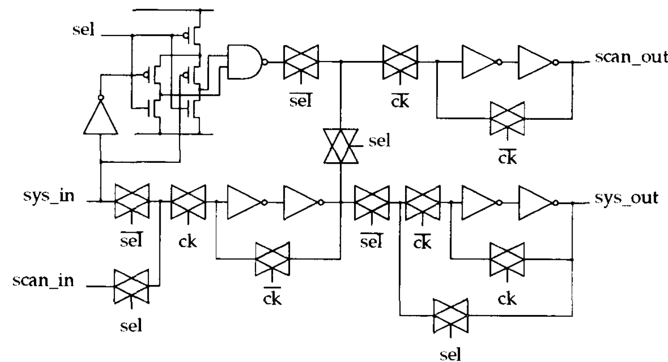

fig. 3. Implementation of enhanced scan cell

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on December 16, 2009 at 10:03 from IEEE Xplore. Restrictions apply.

A scan cell designed around the hazard free scan cell of fig. 1a can be controlled more easily because of the larger amount of control signal lines. The flow of data is however very similar to the one described above.
When a delay fault test is executed this way both timing sensitive events take place locally and are controlled by the same signal edge. One event is the release of the transition. This is controlled by the falling edge of *sel* and consist of transferring master data to the slave. The other event is activation of the instability sensor, also by the falling edge of *sel*. After activation the operation of the instability sensor is uncritical in timing. Therefore the quality of delay fault testing depends only on the quality of the *sel* signal compared to the *clock* signal.

4.2 implementation

The complete design of the scan cell as presented in fig. 3 consists of 47 transistors. Compared to a normal scan cell like in fig 1a (20 transistors) overhead is 135%. For a complete circuit the ratio of the number of transistors in the combinational part to the number of flip-flops varies from 30 to 650, and lies typically in the range 35..55 [6]. This means total overhead in number of transistors ranges from 4% to 55% and will be typically in the range 35%..50%.
A lay-out of the design presented in fig. 3 has been made for the UT-GA2000 gate array that runs on a 1.26μm BiCMOS process. It measures 636 x 205 μm^2. Overhead in area, compared to the standard scan flip flop in the library is 140%. Simulations suggest the correct operation of the cell: Test patterns can be delivered to circuit inputs without hazards and also the instability sensor functions well. Positive glitches of a width of at least 1.5 ns and negative glitches of a width of at least 1.3 ns are detected. These minimal glitches can be detected from 1.8 ns after the rising edge of the *sel* signal. Longer glitches may start before this time to be detected. To be identified as stable, a signal must have stabilized before 1.4 ns after the rising edge of the *sel* signal.

5. Conclusions

A scan cell suitable for executing high coverage delay fault tests has been presented. To be able to execute accurate tests using this cell it is not necessary to have optimized scan paths, a very well defined clock or robust tests. The area overhead to a conventional scan cell in number of transistors about 130 percent for the implementation of fig. 3, with no overhead in control signal lines. A lay-out for this design has been made on a gate array. The overhead of this scan cell when used in a circuit can vary from 4% to 50%.

References

[1] Z. Barzilai and B.K. Rosen, "Comparison of AC Self-Testing Procedures", *Proceedings 1983 International Test Conference*, pp. 89-94.

[2] Y.K. Malaiya and R. Narayanaswamy, "Testing for Timing Faults in Synchronous Sequential Circuits", *Proceedings 1983 International Test Conference*, pp. 560-571.

[3] C.T. Glover and M.R. Mercer, "A Method of Delay Fault Test Generation", *Proceedings 25th ACM/IEEE Design Automation Conference*, 1988, pp. 90-95.

[4] C.J. Lin and S.M. Reddy, "On Delay Fault Testing in Logic Circuits", *IEEE Transactions on Computer Aided Design*, vol CAD-6, no 5, Sept 1987, pp. 694-703.

[5] S.M. Reddy, C.J. Lin and S. Patil, "An Automatic Test Pattern Generator for the Detection of Path Delay Faults", *Proceedings International Conference on Computer Aided Design*, Nov. 1988, pp. 284-287.

[6] P. Franco and E.J. McCluskey, "Delay Testing of Digital Circuits by Output Waveform Analysis", *Proceedings 1991 International Test Conference*, pp. 798-807.

[6] F. Brglez, D. Bryan and K. Koźmiński, "Combinational Profiles of Sequential Benchmark Circuits", *Proceedings 1989 ISCAS*, 1989, pp.1929-1934.

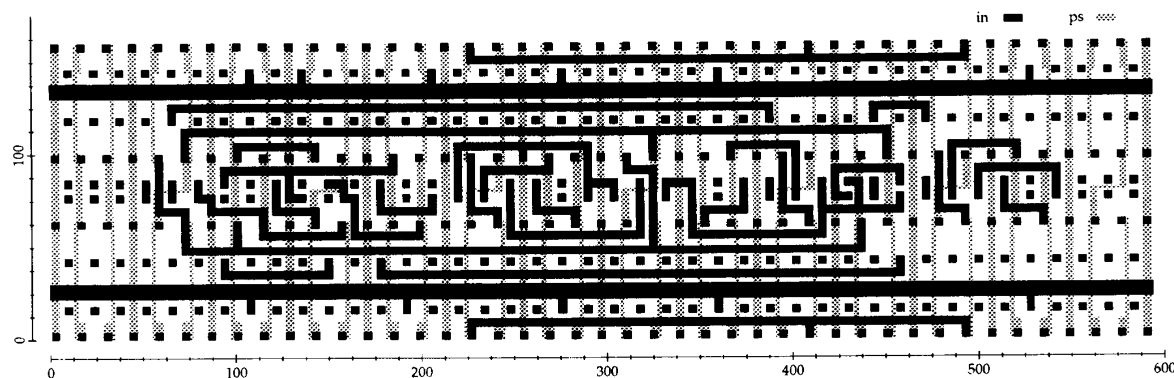

fig. 4. Lay-out on UT GA2000 Gate Array, scale in λ=1.26μm.

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on December 16, 2009 at 10:03 from IEEE Xplore. Restrictions apply.