Functional Illinois Scan Design at RTL

Ho Fai Ko and Nicola Nicolici
Department of Electrical and Computer Engineering
McMaster University, Hamilton, ON, L8S 4K1, Canada
Email: henryko@grads.ece.mcmaster.ca, nicola@ece.mcmaster.ca

Abstract

This paper shows that by creating functional scan chains at the register-transfer level (RTL), not only the timing of the circuit can be improved, but also the test data compression provided from the Illinois scan architecture is similar or even better than the gate level counterpart.

1 Introduction

The Illinois Scan Architecture was proposed by Hamzaoglu and Patel [3] and it is aimed at reducing test data volume with minimum area overhead. Illinois Scan operates in two modes: the *broadcast scan mode* and the *serial scan mode*. In the broadcast scan mode, a single scan input (SI) is connected to multiple scan chains, thus allowing the same set of test patterns to be shifted into multiple chains simultaneously. Scan flip-flops (SFFs) in different scan chains (SCs), but at the same scan depth, will receive identical set of test data from the common SI in the broadcast mode, thus concurrently reducing the volume of test data and test application time. However, due to the strong correlation of the test data that is fed in different SCs in the broadcast mode, the resulting fault coverage of the design may decrease [4]. To improve fault coverage, a serial mode would then be used to reconnect the multiple SCs into a single long SC, thus allowing SFFs that drive the same logic cone to receive any arbitrary set of test data. However, the large volume of serial test patterns will reduce the effectiveness of this technique to compress the overall test data set.

Hamzaoglu and Patel proposed a technique to rearrange scan cells in order to reduce the number of serial patterns for Illinois Scan [3]. Although the greedy heuristic could improve the effectiveness of Illinois Scan for test data volume reduction, this approach is test set dependent. An incremental test generation algorithm for finding the optimal scan chain length k for Illinois Scan was proposed by Pandey and Patel [7]. Despite the fact that by incrementally reducing the fault list can reduce automatic test pattern generation (ATPG) time in a single run for multiple Illinois Scan configurations, the recursive nature of the algorithm may become computational expensive for large designs. In addition to the serial mode and broadcast mode with scan chain length k, Pandey and Patel introduced a reconfiguration technique for Illinois Scan such that a single additional broadcast mode is inserted [8]. Another technique for reconfiguring the Illinois Scan architecture was proposed by Samaranayake et al.[10]. This technique can improve test data compression, however, the analysis of compatibility of SCs in different configurations by applying ATPG recursively may significantly impact the processing time.

Unlike the gate-level scan methodology where scan logic insertion gives no consideration to the structure of the circuit in the normal (functional) mode of operation, register-transfer level (RTL) scan tries to share the existing functional paths between flip-flops (FFs) as scan paths in the test mode, in order to connect SFFs in *functional scan chains*. There are various methods proposed in the literature to construct scan chains at the RTL (or above), in order to reduce the unnecessary area overhead and guarantee high fault coverage (e.g., [2, 5, 9] only to mention a few relevant ones), however none of the existing solutions considers reducing the volume of test data and test application time when constructing the functional scan chains.

Due to its effectiveness to compress test data with very low area overhead, Illinois scan is emerging as a promising low-cost test method [4]. However, to the best of authors' knowledge, the reported studies for improving the performance of Illinois scan assume that scan chain reorganization occurs at the logic level of the design abstraction. Given the fact that this may affect the timing of the circuit, as well as conflict with design synthesis methodologies aimed at achieving timing closure early in the implementation flow, in this paper we investigate the suitability of building functional Illinois scan chains at the RTL.

2 Reconfigurable Illinois Scan at RTL

In this section we introduce a scan chain construction and reconfiguration technique that can be planned prior to logic/physical synthesis and test pattern generation. We describe the basic principle of the new reconfigurable architecture and we summarize a new functional Illinois scan synthesis algorithm at RTL.

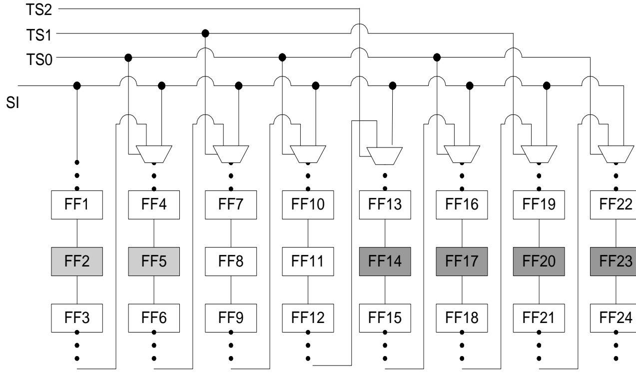

Figure 1. Reconfigurable Illinois scan

2.1 The Proposed Reconfigurable Architecture

The purpose of the proposed reconfigurable Illinois scan architecture (RISA) is to eliminate correlations between patterns in multiple SCs by employing multiple broadcast modes. Figure 1 illustrates an example of the proposed architecture. In this example, there are 24 FFs equally distributed in 8 SCs. It is assumed that FF2 and FF5 drive the same logic cone, while FF14, FF17, FF20 and FF23 drive another logic cone. In mode 0 when the test select signals $\{TS2, TS1, TS0\}$ have the values of $\{0, 0, 0\}$, the architecture functions in the broadcast mode, where a single SI drives 8 SCs. However, when in mode 1, when the TS signals change to $\{0, 0, 1\}$, the SCs are reconfigured such that the adjacent SCs are combined into a single SC, resulting in 4 SCs with length of 6 to be driven by the same SI. In this mode, the conflict between FF2 and FF5 will be eliminated since they will be at different scan depths in the same SC. When the TS signals change to $\{0, 1, 1\}$ (mode 2), the SCs will be reconfigured again such that only 2 SCs with length of 12 will be driven by the same SI. In this mode, all the conflicts between FF14, FF17, FF20 and FF23 are eliminated. Finally, in mode 3 with $\{TS2, TS1, TS0\}$ equal to $\{1, 1, 1\}$ we will end up with a single scan chain where as long as a test pattern exist for a fault it will be generated. Note, although the test data volume is increased every time the SCs are reconfigured, the rate of increase of the RISA is lower when compared to that of the serial broadcast mode in the original Illinois scan architecture.

If there are 2^n SCs in a design, then the total number of broadcast modes in the RISA is $n+1$ (including the serial scan mode). Whenever the RISA is reconfigured, the adjacent SCs will be chained together as one long SC. There will be n TS signals, whose values can be stored in n state elements, which is negligible from the added area standpoint. The update of the TS signals can be done in between test sessions, with different broadcast modes, using a small on-chip controller programmable through the boundary scan interface. The decimal equivalents of the $n+1$ values for the n TS signals are $2^i - 1$, with i from 0 to n.

2.2 Functional Illinois Scan at RTL

The main advantages of building the Illinois scan chains at RTL can be summarized as follows: (i) scan chain reconfiguration analysis is based only on the conflicts between the FFs driving the same logic cone and, therefore, it is ATPG-independent (and hence computationally efficient); (ii) the complexity of the control and data flow graph (CDFG) extracted from the RTL specification is significantly lower than the one of its synthesized logic network; this is of great importance for functional scan chains (that share test and functional logic) because the design structure must be analyzed to identify the functional paths that can be reused as scan paths and the potential conflicts between different FFs; (iii) the scan chain order, which is suitable for test data compression using Illinois scan (regardless of the manufacturing or the diagnostic test sets that will be applied), is determined prior to design synthesis; this will, firstly, avoid any potential timing problems caused by reconfiguring the synthesized scan network and, secondly, by concurrently synthesizing the test and functional logic the circuit speed can be improved, when compared to the gate level scan design. The above advantages are substantiated in the next section.

Due to space limitations, the reader is referred to [6] for further details on functional scan synthesis at RTL. In this section we only explain the constraints that must be accounted for during the functional scan synthesis process in order to generate scan structures that have reduced volume of test data and test application time when combined with RISA. There are three constraints that need to be discussed. The first constraint aims to ensure that FFs that drive the same logic cone will not be placed at the same scan depth in multiple SCs. This helps reduce the correlations between test patterns in multiple SCs. For example, if $FF2$ and $FF5$ in Figure 1 are placed at different scan depths, the conflict between these FFs will be eliminated without the need to reconfigure the architecture. To aid the satisfaction of the first constraint, the second constraint tries to place FFs that drive the same logic cone in the same SC. This is because as soon as the FFs that drive the same logic cone are placed in the same SC, conflicts in test patterns between SCs will be removed. By accounting for these two constraints in the functional scan synthesis process, the generated scan structure will already have a reduced number of correlations between test patterns in multiple SCs, and thus, allowing more faults to be detected in earlier broadcast modes of RISA. Moreover, when both first and second constraints cannot be satisfied, a third constraint will be applied to place the conflicting FFs in adjacent SCs. For instance, in Figure 1, $FF2$ and $FF5$ are conflicting with each other. By putting them adjacent to each other in different SCs, the correlation in test patterns for $FF2$ and $FF5$ can be removed as soon as the architecture is reconfigured. As a result, the effectiveness in compressing test data with RISA can be improved.

3 Experimental Results

To demonstrate the effectiveness of the proposed method, we have implemented the new functional Illinois scan synthesis algorithm in C and we have performed experiments on several circuits described at RTL using Verilog. For synthesis and ATPG we have used a commercial tool flow [12]. The algorithm was applied to a DSP Core from the SCU benchmark set [1] and the ITC99 benchmark *B14* [11].

Tables 1 and 2 show the experimental results for area overhead and circuit performance for the DSP Core and B14 respectively, when comparing the non-scan circuit with the gate level full scan and the proposed RTL Illinois scan. In both tables, column 1 shows the timing constraints used for logic synthesis. Column 2 provides the total number of scan chains and Columns 3, 5 and 7 indicate whether the timing constraints were met during synthesis for the non-scan circuit (i.e., no DFT included), the circuit with gate level scan and the RTL scan circuit. Columns 4 and 6 show the area overhead when compared to the non-scan circuit for gate level scan and RTL scan accordingly. Column 8 shows the difference in area overhead between gate level scan and RTL scan. One thing to note is that there are 639 FFs in the original circuit and gate level scan, while RTL scan has 664 FFs after synthesis. This is because without RTL scan, the synthesis tool can optimize the circuit by removing the redundant FFs. Despite the presence of these redundant FFs, the area overhead of RTL scan increases on average by only 0.05% when compared to the gate level scan. This shows that our approach inserts insignificant hardware overhead to the scan design in order to build the reconfigurable Illinois scan architecture at RTL. Instead, the proposed method tries to reduce area overhead of the scan design by reusing existing functional paths in the circuit as scan paths. In the case of B14, the average increase in area overhead for RTL scan when compared to that of gate level scan is only 0.34%. In addition, by constructing scan chains at the RTL, the synthesis tool can better optimize test logic and functional logic concurrently, thus improving the timing of the design. Although the timing of B14 was not improved, RTL scan of the DSP core helps improve the timing to 10.35 ns, an improvement of 2.42% when compared to the original circuit, which fails to meet timing constraints beyond 10.6 ns.

Table 3 shows the detailed testability results generated by a commercial ATPG tool [12] for B14, in order to illustrate that the proposed reconfigurable Illinois scan architecture reduces test data volume by having multiple broadcast modes. The timing constraint is listed in column 1. Column 2 shows the number of SCs that are driven by a single scan input in the broadcast mode. The columns labeled FC give the fault coverage for stuck-at faults. CTP corresponds to the number of compressed test patterns generated by ATPG and VTD denotes the volume of test data for the particular Illinois scan configuration. CPU represents the test generation time in seconds. The last two rows of the table show the cumulative VTD and CPU, as well as the compression ratio for the gate level and the RTL scan design. By consciously constructing the Illinois scan architecture to reduce the number of conflicts between FFs at the same scan depth located in different SCs, while placing conflicting SCs adjacent to each other, most of the faults become detectable when a single SI drives multiple scan chains. This leads to savings in both test data volume and test application time. This is supported by the experimental results for B14, where a fault coverage of 99.51% can already be obtained for RTL scan compared with only 98.81% for gate level scan (when the common SI drives four SCs).

Table 1. Area/Performance for the DSP Core

Period	# of	Original (No DFT)	Area OH				Δ
			Gate Full		RTL Full		
(ns)	SC	Timing Met?	%	Met?	%	Met?	%
10.80	32	Yes	4.84	Yes	5.52	Yes	-0.68
10.75	32	Yes	4.89	Yes	5.86	Yes	-0.97
10.70	32	Yes	6.01	Yes	5.78	Yes	0.23
10.65	32	No	4.80	No	4.94	Yes	-0.15
10.60	32	Yes	5.94	No	5.52	Yes	0.41
10.55	32	No	5.39	No	5.76	No	-0.37
10.50	32	No	6.41	No	5.99	Yes	0.42
10.45	32	No	5.90	No	5.52	No	0.37
10.40	32	No	5.61	No	5.56	Yes	0.05
10.35	32	No	5.94	No	5.73	Yes	0.20

Table 2. Area/Performance for B14

Period	# of	Original (No DFT)	Area OH				Δ
			Gate Full		RTL Full		
(ns)	SC	Timing Met?	%	Met?	%	Met?	%
5.90	16	Yes	6.73	Yes	7.96	Yes	-1.23
5.85	16	Yes	9.32	Yes	11.73	Yes	-2.41
5.80	16	Yes	1.32	Yes	4.93	Yes	-3.61
5.75	16	Yes	13.85	Yes	8.70	Yes	5.15
5.70	16	Yes	13.29	Yes	10.52	No	2.77
5.65	16	No	4.97	No	9.14	No	-4.17
5.60	16	Yes	5.19	Yes	6.08	No	-0.90
5.55	16	Yes	10.29	Yes	8.65	Yes	1.64

Tables 4 and 5 show the simplified testability results for the DSP core and B14, only for the cases where the timing of RTL scan is satisfied. Column 1 gives the timing constraints used for logic synthesis. Columns labeled FC represent fault coverage for stuck-at faults. VTD denotes the cumulative volume of test data. CR corresponds to the compression ratio, which is calculated by $\frac{VTD\ of\ circuit\ when\ FFs\ are\ connected\ in\ a\ single\ chain}{VTD\ of\ circuit\ with\ Illinois\ scan\ architecture}$. Lastly, columns labeled CPU represent the cumulative test generation time for the ATPG flow in seconds. Despite the different logic networks that were obtained after logic synthesis, as well as the different test sets generated for RTL scan of the DSP core and B14, the experimental results show that the reconfigurable Illinois scan architecture reduces the volume of test data for both gate level scan and RTL scan. However, by utilizing the control and data flow information extracted from the RTL description to build the scan structure, the reduction in volume of test data is further enhanced when compared with gate level scan insertion. For B14, the average compression ratio for the RTL scan is 9.56X, an improvement of 1.17X compared to gate level scan, which

Table 3. Detailed testability results for B14 for a single clock period constraint

Period (ns)	SC #	Gate Full FC (%)	Gate Full CTP	Gate Full VTD	Gate Full CPU (s)	RTL Full FC (%)	RTL Full CTP	RTL Full VTD	RTL Full CPU (s)	Δ FC (%)	Δ CTP	Δ VTD	Δ CPU (s)
5.90	16	95.03	487	7792	70.16	98.02	521	8336	24.41	2.99	-34	-544	45.75
	8	97.52	44	1408	55.57	98.89	40	1280	9.05	1.37	4	128	46.52
	4	98.81	31	1984	40.47	99.51	44	2816	7.50	0.70	-13	-832	32.97
	2	99.26	33	4224	36.34	99.54	1	128	6.40	0.28	32	4096	29.94
	1	99.27	2	512	37.17	99.54	1	256	6.39	0.27	1	256	30.78
	Cumulative			15920	239.71			12816	53.75			3104	185.96
	Compression			8.39				10.43				1.24	

Table 4. Testability results for the DSP Core for multiple clock period constraints

Period (ns)	Gate Full FC (%)	Gate Full VTD	Gate Full CR_Gate	Gate Full CPU (s)	RTL Full FC (%)	RTL Full VTD	RTL Full CR_RTL	RTL Full CPU (s)	Δ FC (%)	Δ VTD	Δ CR_Gate/CR_RTL	Δ CPU (s)
10.80	98.99	27468	5.04	230.51	99.66	25347	5.46	81.95	0.67	2121	1.08	148.56
10.75	99.16	36771	3.78	236.35	99.63	28161	4.94	79.13	0.48	8610	1.31	157.22
10.70	99.07	34209	4.26	303.59	99.50	31584	4.62	106.05	0.43	2625	1.08	197.54
10.65	99.02	31857	4.75	262.67	99.58	31185	4.85	94.57	0.56	672	1.02	168.10
10.60	98.82	37275	4.25	2008.28	99.56	28560	5.55	87.61	0.74	8715	1.31	1920.67
10.50	98.98	34251	4.47	307.02	99.56	29484	5.20	92.18	0.58	4767	1.16	214.84
10.40	98.98	36372	3.84	1919.32	99.57	28392	4.92	88.41	0.59	7980	1.28	1830.91
10.35	99.09	30534	4.93	1924.50	99.62	29757	5.06	94.95	0.52	777	1.03	1829.55

Table 5. Testability results for B14 for multiple clock period constraints

Period (ns)	Gate Full FC (%)	Gate Full VTD	Gate Full CR_Gate	Gate Full CPU (s)	RTL Full FC (%)	RTL Full VTD	RTL Full CR_RTL	RTL Full CPU (s)	Δ FC (%)	Δ VTD	Δ CR_Gate/CR_RTL	Δ CPU (s)
5.90	99.27	15920	8.39	239.71	99.54	12816	10.43	53.75	0.27	3104	1.24	185.96
5.85	99.33	17696	7.99	308.45	99.21	14944	9.46	61.54	-0.11	2752	1.18	246.91
5.80	99.21	16672	8.40	164.59	99.41	14976	9.35	51.52	0.19	1696	1.11	113.07
5.75	99.39	17872	7.79	149.79	99.47	14720	9.46	57.00	0.09	3152	1.21	92.79
5.55	99.33	17088	8.03	363.61	99.45	15056	9.11	54.11	0.12	2032	1.13	309.50

only has an average compression ratio of 8.12X. In addition, the test generation time for RTL scan has also improved for both of the circuits compared with gate level scan. The average improvements are 847.82 secs for the DSP core and 176.93 seconds for B14. It is also important to note that the computational time for scan insertion in the RTL code is in the range of tens of seconds on a Pentium-M at 1.3 GHz, which justifies the advantage of analyzing the design and developing the DFT infrastructure at the RTL.

4 Conclusion

In this paper we have explored the effectiveness of creating scan chains at a higher-level of design abstraction for test data compression using the Reconfigurable Illinois Scan Architecture. It was found that the DFT infrastructure built using only the control and data flow information available at the RTL can lead to similar improvements in test data compression (for a fault coverage target over 99%), regardless of the final implementation of the logic network or the manufacturing test set. In addition, it was shown that, although functional scan chain design at RTL may introduce redundant FFs used only in the test mode, the area overhead is similar to the one given by gate level scan insertion. This is due to the simultaneous synthesis of functional and test logic, which can also lead to improved circuit speeds.

References

[1] *SCU-RTL Benchmarks*. Santa Clara University, http://www.engr.scu.edu/mourad/benchmark/.

[2] T. Asaka, S. Bhattacharya, S. Dey, and M. Yoshida. H-SCAN+: A practical low-overhead RTL design-for-testability technique for industrial designs. In *Proc. International Test Conference*, pages 265–274, 1997.

[3] I. Hamzaoglu and J. H. Patel. Reducing test application time for full scan embedded cores. In *Proc. IEEE VLSI Test Symposium*, pages 369–375, April 2000.

[4] F. F. Hsu, K. M. Butler, and J. H. Patel. A case study on the implementation of the illinois scan architecture. In *Proc. International Test Conference*, pages 538–547, 2001.

[5] Y. Huang, C. C. Tsai, N. Mukherjee, O. Samman, D. D. W. T. Cheng, and S. M. Reddy. On RTL scan design. In *Proc. International Test Conference*, pages 728–737, 2001.

[6] H. F. Ko. Functional scan design at RTL, *M.A.Sc. Thesis*. http://www.ece.mcmaster.ca/ñicola/thesis/henry_masc_2004.pdf.

[7] A. R. Pandey and J. H. Patel. An incremental algorithm for test generation in illinois scan architecture based designs. In *Proc. Design Automation and Test in Europe*, pages 368–375, March 2002.

[8] A. R. Pandey and J. H. Patel. Reconfiguration technique for reducing test time and test data volume in illinois scan architecture based designs. In *Proc. 20th IEEE VLSI Test Symposium*, pages 9–15, April 2002.

[9] S. Roy. RTL based scan BIST. In *Proc. VHDL International Users' Forum*, pages 117–121, October 1997.

[10] S. Samaranayake, E. Gizdarski, N. Sitchinava, F. Neuveux, R. Kapur, and T. W. Williams. A reconfigurable shared scan-in architecture. In *Proc. 21th IEEE VLSI Test Symposium*, pages 9–14, April 2003.

[11] M. Sonza-Reorda, F. Corno, and G. Squillero. ITC'99 Test Benchmarks. http://www.cad.polito.it/tools/itc99.html.

[12] Synopsys Test Tools. TetraMAX ATPG. http://www.synopsys.com/products/test/tetramax_dsA4.pdf.