International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:268
Vol.2, No.6, June 2013 DOI:10.15693/ijaist/2013.v2i6.73-77

Weight Pattern Generation Usingsequencialcircuits

J.Sahaya Rosy
PG Scholar,
SIT Engg. College,
Kariyapatti.

D.JohnPragasam
Asst.Prof/ECE,
SIT Engg. College,
Kariyapatti.

Dr.R.Ganesan
Prof/HOD,
SIT Engg College,
Kariyapatti.

*Abstract_* **Weighted pseudorandom built-in self-test (BIST) schemes are utilized to drive down the number ofvectors to achieve complete fault coverage in BIST used application. Set called weighted set comprising three weights, namely 0, 1, and 0.5 are utilized so for test pattern generation since they result in both low testing time and low consumed power. An accumulator-based 3-weight test pattern generation generates set of patterns with weights 0, 0.5, 1.Theaim of the proposed method uses pre-computation logic to improve the speed of weight pattern generation. Proposed system uses precomputation logic to overcome the delay of existing work.Precomputationlogic uses multiple pairs of FA to generate partial sum and carry by considering carry inputs then the final sum and carry are selected by the multiplexers.The implementation results which have been done by using Spartan-3 processor. The proposed algorithm significantly reduces area consumption when compared to the existing system. Modelsim Xilinx Edition (MXE) will be used for functional simulation and Xilinx ISE tools will be used for synthesis and performance analysis.**

Index Terms-Built-in self test (BIST), ,weighted test pattern generation,pre_computation logic.

I INTRODUCTION

***Weighted pattern generation*:**

Testing by random patterns has many advantages compared to other testing methods, for instance the self test capability, less computing time and the high coverage of parametric faults. Weighted patterns are used to test sequential logic, regardless of the number of patterns generated, the next step was to assign weights to the PI's in proportion to their relative importance. Off-line good machine simulation is performed by software using a set of random patterns as input. Recalling that each pattern from the generator activates only one PI at a time, it is possible to count how many gates inside the chip change for the first time from a logic 1 to 0, and vice versa, as the result of switching one of the PI's. The switching activity count is then accumulated over the complete set of patterns. By comparing the activity created by all PI's, one can determine not only the relative importance of each PI, but also the most effective way to bunch the decoder outputs for the next pass. Through the use of this weight vector to generate a new pattern set of equal length, iteration continues until the activity counter no longer increases. In random test pattern generation, a pattern may be repeated several times in the process. However, usingpseudorandom yields random patterns without repetition. This is equivalent to selection without replacement.

The length of the test generated in such a manner depends on the seed of the random number generator. The basic idea of Built-in self-test (BIST) is to generate test patterns on-chip and also compact the test responses of the circuit under test (CUT) on-chip. IJsually test registers are inserted that are based on linear feedback shift registers (e.g. BILBOs**)** or linear cellular automata. These test registers can generate pseudo-random (pseudo-)exhaustive patterns and perform signature analysis. The datapathsofprocessors and in particular circuits fordigital signal processing oftencontain accumulatorscomposed of binary adders or arithmetic logic and registers. Recently it has been shown that accumulators can be reemployedtocompact test responses.

Figure I shows the basic configuration. The input v isadded to the contents of the register, s(t). The sum givesthe next state, s(t+l) = s(t)+v. So with each clock pulse, anew pattern s is generated and applied to the circuit undertest. s(t) denotes the decimal encoding of the k-bit pattern at time t, (sk.l(t),sk.2(t), ..., so(t)) is the correspondingbinary vector. The input v is kept constant. In thefollowing we assume v#O since otherwise only pattern could be generated. The numberofstates of the accumulator is2k so after 2ksteps at the latest, states are repeated and the pattern sequence gets periodic

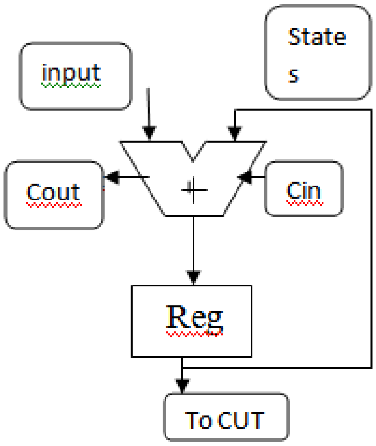


Figure.1 Accumulator as Pattern generator

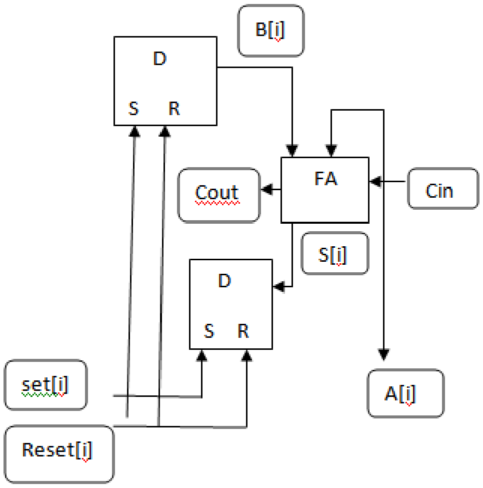


Figure.2Accumulator cell

The results have shown that accumulator generated patterns can achieve about the same fault coverage as pseudo-random patternsand require about the same test lengths .Since these accumulators can also be used as test response compactors, a complete self-test approach based on accumulators is feasible moreover, using accumulators does not introduce additional delays and does not lead to performance degradation.[13]

II WEIGHT PATTERN GENERATION USING ACCUMULATOR

The implementation of the weight pattern generation using accumulator scheme is based on the accumulator cell presented in Figure 2 which consists of a Full Adder (FA) cell and a D-type flip-flop with asynchronous set and reset inputs whose output is also driven to one of the full adder inputs. without loss of generality, that the set and reset are active for high signals. For this accumulator cell, three configurations can be utilized.

For circuits with hard-to-detect faults, a number of random patterns have to be generated before a high fault coverage is achieved. Therefore,aweighted pseudorandom techniques have been proposed where inputs are biased by changing the probability of a “0” or a “1” on a given input from 0.5 value (for pure pseudorandom tests) to some other value. In order to reduce the hardware implementation cost, other methods based on assignments of multiple weights utilized weights 0, 1, and0.5. The scope of the project intimates thatmost of the VLSI circuits, for e.g., data path architectures, digital signalprocessing chips ,they commonly contain arithmetic modules [accumulators or arithmetic logic units (ALUs)].[3] This has fired the idea of arithmetic BIST (ABIST).The basic idea of ABIST is to utilize accumulators for built-in testing by (compression of the CUT responses, or generation of test patterns) and it has been shown that it results in low hardware overhead and low impact on the circuit operating speed.[4]

This accumulator weight generation scheme copes with the inherent drawbacks of the previous paper work. More precisely: 1) it does not impose any requirements about the design of the adder (i.e., it can be implemented using any adder design); 2) it does not require any modification on the adder; and so that, 3) it does not affect the operating speed of the adder.[1]

Table 1 Truth Table of full Adder

#	Cin	A[i]	B[i]	S[i]	Cout	Comment
1	0	0	0	0	0	
2	0	0	1	1	0	Cout=cin
3	0	1	0	1	0	Cout=cin
4	0	1	1	0	1	
5	1	0	0	1	0	
6	1	0	1	0	1	Cout=cin
7	1	1	0	0	1	Cout=cin
8	1	1	1	1	1	

The implementation of the weighted-pattern generation scheme isbased on the full adder truth table,shown in table 1.

From the truth table, we observe that Cout=Cin,that is to transfer the carry input to the output of carry, it is enough to setA[i]=NOT(B[i]).This scheme is based onthis observation.From the accumulator cell shown in fig 2 three configurations can be made based on this set and reset values.

IIIPROPOSED METHOD

Proposed system uses pre_computation logic to reduce the delay of existing work.Pre_computation is a techniquewhereby a critical signal is computed earlier in the computation. This has the effect of shortening the critical pathby changing when the critical signal is computedPre_computation logic uses multiple pairs of FA to generate partial sum and carry by considering carry input cin=0and cin=1 then the final sum and carry are selected by the multiplexers.

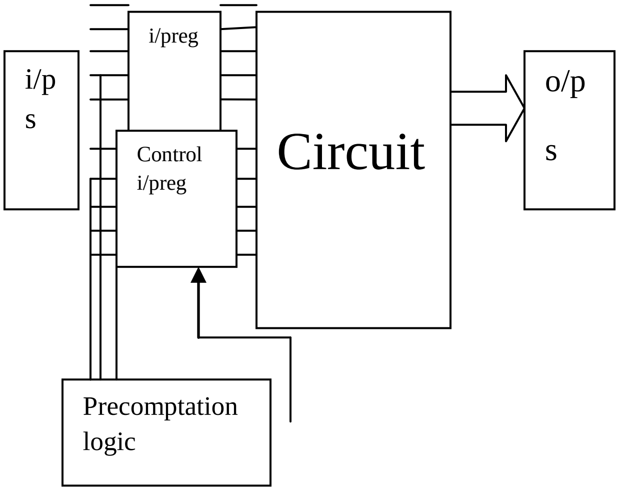


Figure.3Basic pre_computation Architecture

Pre-computation logic is significantly less complex than the combinational logic. The basic pre_computation architecture is shown in figure3. Adding of extra full adder to accumulator shows the modification on the accumulator as shown in figure 4.

We present a powerful sequential logic optimization method that is based on selectively precomputing the output logic values of the circuitin one clock cycle before they are required in operation, and we are allowed to use the precomputed values to reduce internal switching activity in succeeding clock

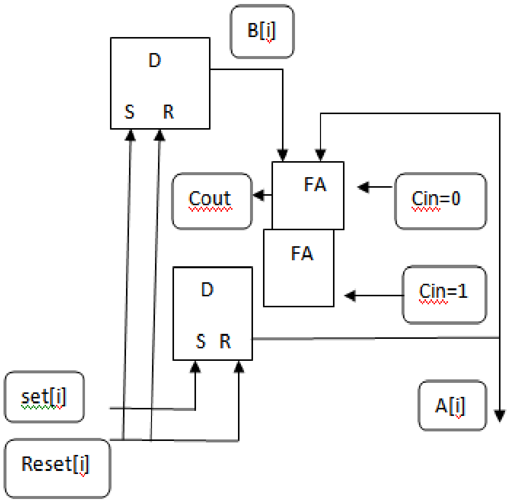


Figure.4 Modification on Accumulator

The primary optimization step of precomputation logic, involves in computing the output values for a subset of input conditions. If the output values are precomputed, the original circuit can be "turned off" in the next clock cycle and will have substantially switching activity is reduced. The result achieved byprecomputation logic determines the reduction in power dissipation ,increase in area and delay,in relative to the original circuit.By a logic-level sequential circuit, a method of precomputation logic will achieve maximal reduction in power dissipation.

.

Pre_computation can be done only if there is sufficient information in the circuit to allow precomputing the valueone cycle ahead of time. Precomputation is possible if none of the signals in the combinational transitive fan-in set are primary inputs.

In the accumulator-based 3-weight system we use the asynchronous method for feedback path and time consumption which causes the race condition. Due to the race condition we add one full adder to increase the throughput as shown in figure 3

1V RESULT AND DISCUSSION

The Fig.4 given below is shown that there is a considerable reduction in time and area based on the implementation results which have been done by using Spartan-3 processor. The proposed algorithm significantly reduces area consumption when compared to accumulator based weight pattern generation.

Here we are comparing the results of the time and area of existing and our proposed work using Spartan _3 processor.

Table 1 Existing result

V CONCLUSION

An accumulator-based 3-weight system uses precomputation logic to reduce the delay of experimental results. Precomputationlogic uses multiple pairs of full adder in the circuit to generate partial sum and carry by considering carry input cin=0and cin=1 then the final sum and carry are selected by the multiplexers.

An accumulator-based 3-weight testpattern generation scheme is presented; the scheme generates setof patterns with weights 0, 0.5, and 1.[1] [6]

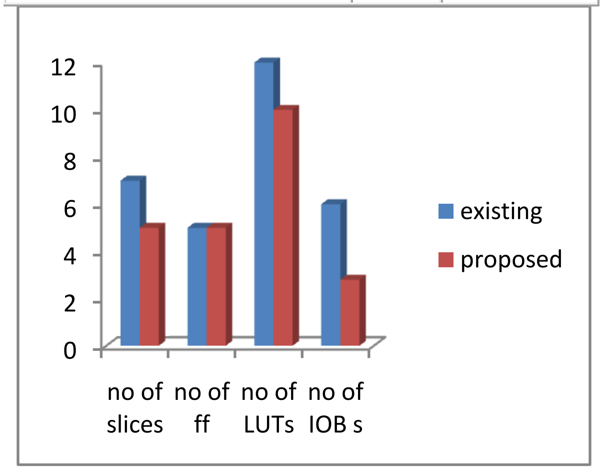
Device Utilization Summary (estimated values)			[-]
Logic level of Utilization	**Used**	**Available**	**Utilized**
Number of Slices	7	960	0%
Number of Slice Flip Flops	5	1920	0%
Number of 4 input LUTs	12	1920	0%
Number of bonded IOBs	6	66	9%

In the accumulator-based 3-weight system we use the asynchronous method for feedback path and time consumption which causes the race condition. Due to the race condition we add one full adder to increase the throughput. The aim of the proposed method is satisfied by using pre-computation logic to improve the speed of weight pattern generation. Proposed system uses precomputation logic to overcome the delay of existing work

Table 1 Proposed result

Comparison graph

Device Utilization Summary (estimated values)		
Logic level of Utilization	**Used**	**Available**
Number of Slices	5	960
Number of Slice Flip Flops	5	1920
Number of 4 input LUTs	10	1920
Number of bonded IOBs	7	66



REFERENCES

[1]“AccumulatorBased 3-Weight Pattern Generation”Antonis Paschalis, IoannisVoyiatzis, and Dimitrisgizopoulos IEEE transactions on very large scale integration (vlsi) systems, vol. 20, no. 2, february 2012

[2]Voyiatzis, “an accumulator—based compaction scheme with reduced aliasing for on-line bist of rams”IEEE Trans. Very Large Scale Integr. Syst., vol. 16, no. 9, pp. 1248–1251, Sep. 2008.

[3 I. Voyiatzis, "An ALU based BIST scheme for word-oganizedrams,"IEEE Trans. Comput., vol. 57, no. 8, pp. 1012–1022, Aug. 2008.

[4]S. Manich, L. Garcia-Deiros, and J. Figueras, " Minimizing test time in arithmetic test-pattern generators with constrained memory resources," IEEE Trans. Comput.-Aided Des.Integr. Circuits Syst., vol. 26, no. 11 pp. 2046–2058, Nov. 2007.

[5]S.Manich,L.Garcia,and J.Figueras,"Arithmetictest Pattern generation: a bitlevel formulation of the optimization problem," presented at the Des. CircuitsIntegr. Syst. (DCIS), Lisbon, Portugal, 2005.

[6] Voyiatzis, D. Gizopoulos, and A. Paschalis, "Accumulator-basedweighted pattern generation," presented at the IEEELineTestSymp., Saint Raphael, French Riviera, France, Jul. 2005

[7] S.Zhang,S.C.Seth,andB.B.Bhattacharya,"Efficient test compaction for pseudo-random testing," in Proc 14th Asian Test Symp., 2005.

[8]Voyiatzis, N. Kranitis, D. Gizopoulos, APaschalis an"Accumulator-based built-in self-test generator for robustly detectable sequential fault testing"U.S. Patent 6 886 124, Apr. 26, 2005.

[9]S. Wang, "Low hardware overhead scan based 3-weight weightedrandom BIST architectures," U.S. Patent 6 886 124, Apr. 26, 2005.

[10]S. Manich, L. Garcia, L. Balado, J. Rius, R. Rodrøguez, and J. Figueras, "Improving the efficiency of arithmetic bist by combining targeted andgeneral purpose patterns," presented at the Des. Circuits Integr. Syst.(DCIS), Bordeaux, France, 2004.

[11]A. D. Singh, M. Seuring, M. Gossel, and E. S. Sogomonyan, "Multimode scan: Test per clock BIST for IP cores," *ACM Trans. Design*Autom. Electr. Syst., vol. 8, no. 4, pp. 491–505, Oct. 2003.

[12]S. Manich, L. Garcia, L. Balado, E. Lupon, J. Rius, R. Rodriguez, and J.Figueras, "On the selection of efficient arithmetic additive test pattern generators*,"* in Proc. Eur.Test Workshop, 2003.

[13]A. Stroele, "A self testapproach using accumulators as test patterngenerators," in Proc. Int. Symp. Circuits Syst., 1995.

Authors Profile

J.Sahaya Rosy received the **B.E.** degree in electronics and communication engineering from the Francis Xavier college of Engineering, Tirunelveli, Anna University, Chennai, India, in 2010.Currently doing **M.E.** in VLSI DESIGN in SETHU COLLEGE OF ENGINEERING,VIRUDHUNAGAR, AnnaUniversity, Chennai.

D.John pragasam received his the **B.E.** degree in electronics and communication engineering from the Govt. College of Engineering, Salem, Anna University, Chennai, India, in 2007.Currently doing **M.E.** in electronics and communication engineering (Communication Systems) in Anna University of technology, Coimbatore, India. His research interest includes wireless communication (**WiFi,WiMax**), Mobile Ad hoc networks ,Sensor Networks ,Neural Networks and fuzzy logic, Communication networks.

Dr.R.Ganesanreceived his B.E. Instrumentation & Control Engineering from ArulmiguKalasalingam College of Engineering and ME (Instrumentation) from Madras Institute of Technology in the year 1991 and 1999 respectively. He has completed his PhD from Anna University, Chennai, India in 2010. He is presently working as Professor and head in the department of M.E-VLSI Design at Sethu Institute of Technology, Madurai, India. He has published more than 25 research papers in the National & International Journals/ Conferences.His research interests are VLSI design, Image Processing, Neural Networks and Genetic algorithms.