

Design and Implementation of Hardware Based Entropy Analysis

S. Saravanan, K. Chakrapani and R. Silambamuthan

School of Computing (SOC), Sastra University, Thanjavur-613401, India

Abstract: The aim of this study is hardware implementation of the Entropy analysis. Designing and verifying entropy analysis is the major finding of this study. Entropy tells how much amount of data can be compressed. Entropy analysis plays a major role in scan based SoC testing. Size and complexity have been the major issues for current scenario of System-on-a-Chip (SoC) testing. Test data compression is a must for such cases. Entropy analysis is taken for both specified and unspecified bits (don't care bits). Unspecified bits are specified using Zero and One fill algorithms. The X-filling technique is applied for fixed to fixed codes. The proposed method is successfully tested on ISCAS89 benchmark circuits.

Key words: Don't care bits, entropy analysis, test data compression, zero and one filling techniques

INTRODUCTION

Integrated Circuits (IC's) have increased size and complexity since in the year 1950's. As per the Moore's law, scale of IC's double every 18 months. So it increases the size and complexity of the IC's. As size and complexity increases then it leads to challenge in the testing. Testing is a system which analyses the circuit response whether it behaves correctly after manufacturing. Size of the IC's is directly proportional to the test data volume.

Test data compression technique is used to reduce the test data volume. So test data compression is mandatory for such cases. The test data volume to be stored in Automatic Test Equipment (ATE) is reduced leading to less processing time. Different type of test data compression schemes are used in the scan based SoC testing.

The measure of amount of information contained in a set of test data is defined as Entropy. Entropy (Cover and Thomas, 1992) is used to measure the disorder in a system. Entropy analysis is a basic for all test data compression. It tells how much amount data can be compressed. And also it gives maximum theoretical limit for test data compression. The unspecified bits are specified by using X-filling techniques and its entropy analysis is taken for different symbol length.

EXISTING METHODOLOGY

Several research study focus on test data compression. Entropy tells that how minimum no of data reproduces the sending information. The theoretical limit for maximum data compression is showed in different

articles (Cover and Thomas, 1992). Convolution LFSR reseeding deals with fixed to fixed code (Koenemann, 1991) having fixed size of test vector and encoded as smaller fixed size of LFSR seed.

In the LFSR reseeding (Koenemann, 1991) combination expanders are used that have more outputs than inputs. So test channels requirement is reduced. Two types of combinational expanders were used for test data compression. They are Linear and Non-Linear combinational expanders. Linear combinational expanders are described in (Hamzaoglu and Patel, 1999; Bayraktaroglu and Orailoglu, 2001). Non-linear combinational expanders are described in (Reddy *et al.*, 2002; Li and Chakrabarty, 2003).

In the linear combinational expanders (Hamzaoglu and Patel, 1999; Bayraktaroglu and Orailoglu, 2001) encoded block combination is less than original block combination and then there is no need of encoding all possible combination in fixed to fixed code. If $E_{\text{dictionary}}$ have set of symbols which have been encoded then E_{data} have set of symbols which occur in message data. If E_{data} is a subset of $E_{\text{dictionary}}$, then it is complete encoding otherwise it is partial coding. The non linear combinational expanders implement the complete encoding. If the technique (Reddy *et al.*, 2002) does not have complete encoding, then there are two alternatives. The first one is to constrain the ATPG process to generate test data which is present in dictionary (Bayraktaroglu and Orailoglu, 2001). Second one is to bypass the dictionary which is not present (Li and Chakrabarty, 2003).

There are four test data compression codes which has been proposed, they are fixed to fixed code, fixed to variable code, variable to fixed code and variable to variable code. The study concentrates on fixed to fixed

Table 1: Test data partitioned into six bit

Vector 1	111101	111101	111100	110111
Vector 2	111111	101110	111101	111101
Vector 3	110101	111111	111111	011111
Vector 4	111101	111111	111011	101110

Table 2: Probability table for symbol length of six

i	Symbol (X _i)	Frequency	Probability (P _i)
1	111101	5	0.3125
2	111100	1	0.0625
3	110111	1	0.0625
4	111111	4	0.25
5	101110	2	0.125
6	110101	1	0.0625
7	011111	1	0.0625
8	111011	1	0.0625

Entropy: 2.65; Maximum Compression Ratio: 55.84%

code, with both source data and encoded data being in fixed size. The LFSR reseeding (Koenemann, 1991) is the best example for the fixed to fixed code.

In Fixed to variable code, the source data is fixed but the encoded data is variable. Huffman coding (Huffman, 1952) is the best example for fixed to variable code. In variable to fixed code, the source data is variable but encoded data is in fixed size. LZ77 (Wolff and Papachristou, 2002) is the best example for the variable to fixed code.

In variable to variable code, both source data and encoded data are in variable form. Variable Input Huffman Coding, (Gonciari *et al.*, 2002) Golomb Coding (Chandra and Chakrabarty, 2001a, b) and Frequency Directed Coding (Chandra and Chakrabarty, 2001a, b) are the best examples for variable to variable code.

Entropy analysis of test data: Entropy is used to measure how maximally test data can be compressed. Entropy tells the disorder of the system. It tells the minimum amount of data that reproduce the original test vector. Entropy gives theoretical limit of test data compression. For the entropy analysis test vector is partitioned and named as symbol (X_i). Next partition the test vector for fixed size. The fixed to fixed code only is discussed.

The given test vectors are partitioned and formed as fixed symbol. For different symbol length, entropy is also different. In the Table 1, test data is partitioned to six. Totally the test vector contains 16 test data. It is divided

into four groups each having six bit sizes. Probability, Entropy and Maximum Compression Ratio (MCR) are calculated. In Table 2, probability occurrence of six bit symbol is shown. The probability is nothing but chance of occurrence of actual symbol to total no of symbol in the test vector. The probability is calculated by dividing actual frequency with the total no of frequency in the test vector.

The probability value is represented in the last column. Entropy is calculated by using the formula H is shown in Eq. (1):

$$H = -\sum P_i \log_2 P_i \tag{1}$$

where, \sum from I = 1 to i = n, where 'P_i' is probability occurrence of symbol, 'X_i' is the symbol and 'n' is the total number of symbols in the test vector. The entropy and MCR are shown in Table 2. MCR is calculated by using the formula M as shown in Eq. (2). For the above test vector, MCR value is 55.84%:

$$M = (\text{symbol length}-H)/\text{symbol length} \tag{2}$$

Proposed system: In this section, the proposed method shows how entropy analysis is implemented into the hardware. Input test vector is given from the bench mark circuit ISCAS89 (Hamzaoglu and Patel, 1998). The test vector is partitioned into fixed symbol length like 4, 6 and etc. Test Vector partition is nothing but splitting the Test Vector by fixed symbol length. Test vector containing more no of test data. Reputation of test data is calculated and that is called as Test data frequency calculation. Test data frequency is nothing but finding the how much of time the particular Test data is occur in Test Vector. Then the probability of each Test data is calculated for Test Vector. Probability deals with the rate of occurrence of the particular test data to the total no of test data in Test Vector. Entropy value depends on the probability value of test data. Probability value of each test data is calculated and then Entropy of Test Vector has to be calculated. Entropy depends on probability of each test data and its logarithm base 2 value. For maximum compression, Entropy Value should be minimized. Finally the MCR value is calculated. MCR value tells that the theoretical compression limit for particular Test file. The Fig. 1 shows the block diagram for the hardware implementation of Entropy Analysis.

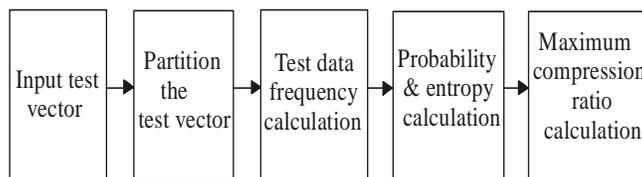


Fig. 1: Block diagram for the hardware implementation of entropy analysis

EXPERIMENTAL RESULTS

In this Section, the experimental performance for hardware implementation of entropy analysis is described.

The VHDL code is written for the entropy analysis and is shown in Fig. 3. Test file is given as input and it is partitioned as fixed size like 4, 6, etc.

In Zero filling, the unspecified bits are replaced by zeros (0's). Similarly in One filling, the unspecified bits are replaced by ones (1's). Both X filling algorithms are successfully tested on the ISCAS89 (Hamzaoglu and Patel, 1998) benchmark circuits. Table 3, 4 (Saravanan *et al.*, 2011) shows the Entropy analysis for the Zero filling and One filling in ISCAS89 benchmark circuit. Figure 4, 5 shows the Entropy analysis for the Zero filling and one filling.

Then the test vector is divided into test data. Redundancy of test data is counted and it is named as frequency. Frequency is calculated for each test data. Probability of each test data is calculated. Probability is nothing but the ratio of actual frequency of the test data into total no of frequency of test vector.

Probability value is indicated as 'prob' and is shown in Fig. 3. Entropy value is calculated and it is shown in

the Fig. 3 as 'ent'. Finally value of MCR is calculated and it is shown in the Fig. 3 as 'mct'. Final 'mct' is denoted as percentage, so multiply the 'mct' value by 100. Then the percentage of MCR is taken.

CONCLUSION

Requirement of test data volume is one of the major concerns for SoC testing. Entropy analysis is more useful for reducing the test data volume. In this study the study of entropy analysis in hardware implementation is proposed. Entropy tells the theoretical limitation of test data compression. The Entropy analysis is taken for different fixed symbol length and its theoretical limit of test data compression are also calculated. The unspecified bits are specified using Zero and one filling technique. Experimental results for the ISCAS89 benchmark circuits are tested by using VHDL coding.

REFERENCES

Bayraktaroglu, I. and A. Orailoglu, 2001. Test volume and application time reduction through scan chain concealment Proceedings of the Design Automation Conference, pp: 151-155.

Chandra, A. and K. Chakrabarty, 2001a. System-on-a-chip test data compression and decompression architectures based on Golomb Codes. IEEE T. Comput. Aided Design, 20(3): 355-368.

Chandra, A. and K. Chakrabarty, 2001b. Frequency-directed run-length codes with application to system-on-a-chip test data compression. Proceedings of the VLSI Test Symposium, pp: 42-47.

Cover, T. and J.A. Thomas, 1992. Elements of Information Theory. 2nd Edn., Wiley, Hoboken.

Gonciari, P.T., B.M. Al-Hashimi and N. Nicolici, 2002. Improving compression ratio, area overhead and test application time for systems-on-a-chip test data compression/decompression. Proceedings of the Design Automation and Test in Europe, pp: 604-611.

Hamzaoglu, I. and J.H. Patel, 1999. Reducing test application time for full scan embedded cores. Proceedings of the International Symposium on Fault-Tolerant Computing, pp: 260-267.

Hamzaoglu, L. and J.H. Patel, 1998. Test set compaction algorithms for combinational circuits. Proceedings of the CAD, pp: 283-289.

Huffman, D.A., 1952. A method for the construction of minimum redundancy codes. Proceeding IRE, 40: 1098-1101.

Koenemann, B., 1991. LFSR-Coded Test Patterns for Scan Designs. Proceedings of the European Test Conference, pp: 237-242.

Li, L. and K. Chakrabarty, 2003. Test data compression using dictionaries with fixed-length indices. Proceedings of the VLSI Test Symposium, pp: 219-224.

Table 4: Entropy analysis using one filling

Circuit	Test size	Symbol length		
		4	6	8
S5378	23754	47.63%	48.38%	51.29%
S9234	39723	46.84%	48.26%	48.58%
S13207	165200	57.04%	81.72%	82.20%
S15850	76986	65.49%	66.08%	66.77%

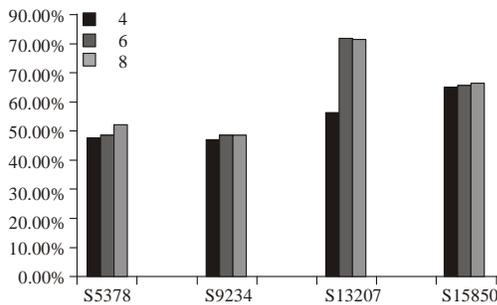


Fig. 4: Entropy analysis for zero filing

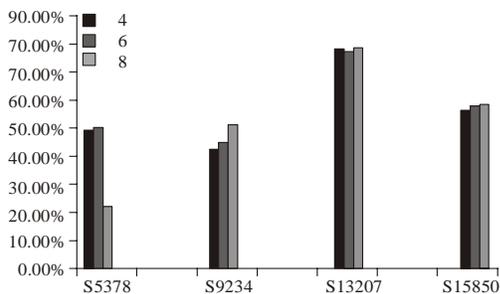


Fig. 5: Entropy analysis for one filling

- Reddy, S.M., K. Miyase, S. Kajihara and I. Pomeranz, 2002. On test data volume reduction for multiple scan chain designs. Proceedings of the VLSI Test Symposium, pp: 103-108.
- Saravanan, S., R. VijaySai, R. Silambamuthan, E. Deepa, G. Elakkiya and H.N. Upadhyay, 2011. Entropy Analysis for achieving high compression ratio. J. Theoret. Appl. Informat. Technol., 34(1): 96-99.
- Wolff, F.G. and C. Papachristou, 2002. Multiscan based test compression and hardware decomposition using LZ77. Proceedings of the International Test Conference, pp: 331-338.