2015 28th International Conference on VLSI Design and 2015 14th International Conference on Embedded Systems

Thermal-Aware Test Data Compression Using Dictionary Based Coding

Rajit Karmakar and Santanu Chattopadhyay
Department of Electronics & Electrical Communication Engineering
Indian Institute of Technology Kharagpur, Kharagpur 721302, India
{rajit, santanu}@ece.iitkgp.ernet.in

***Abstract*—In this paper, we have proposed a new thermal-aware test data compression technique using dictionary based coding. Huge test data volume and chip temperature are two major challenges for test engineers. Temperature of a chip can be reduced to a large extent by minimizing transition count in scan chains using efficient don't-care filling. On the other hand, high compression ratio can be achieved by filling the don't-cares intelligently to get more similar sub-vectors from test vectors. Although, both of the problems rely on don't-care bit filling, most of the existing works have considered them as separate problems. In our work, we have combined both temperature reduction and compression into a single problem and solved it. We present an intermediate approach that performs a trade-off between temperature and compression ratio. Experimental results on ISCAS'89 and ITC'99 benchmarks show the flexibility of the proposed method to achieve a balance between temperature and compression ratio.**

***Index Terms*—Compression, temperature reduction, don't-care bits, dictionary based coding.**

I. Introduction

With increasing complexity of present day's integrated circuits (IC), the amount of test data needed to test the ICs has increased dramatically. More test patterns are required to improve fault coverage targeting delay faults, stuck-at-faults and some other subtle faults. Automatic Test Equipment (ATE) has to accommodate a large amount of test data, increasing memory size as well as memory cost. Increasing test data volume also results in longer test application time (TAT), which significantly increases total test cost. Built-in-self-test (BIST) is an alternative solution which avoids the usage of external storage to store test data. However, it suffers from random-resistant faults and bus contention during test application leading to inadequate fault coverage [1]. The other alternative is test data compression, which stores huge amount of test data (T_D) in ATE in a compressed form (T_E), helps to reduce the total memory requirement. The compressed data (T_E) have to pass through a decompressor to get back original uncompressed (T_D) test patterns before being applied to the circuit under test (CUT).

It is worth mentioning that, more than 95% bits of the test data are don't-cares [2]. To get better compression, most of the test compression techniques take the advantage of the flexibility of don't-care bits, which can be flipped either to '0' or '1' to get more numbers of matching sub-vectors from the test vectors without compromising fault coverage.

Another major concern during testing is the temperature of the IC. Excessive switching activity in scan chains during test pattern shifting increases overall circuit temperature and creates localized heating, called hotspots [3]. Although advanced cooling techniques can effectively solve the high-temperature problems, they substantially increase overall system cost and/or require a larger area. Several works of literature have tried to minimize the temperature of the IC during testing by efficiently filling the don't-care bits present in test patterns [3], [4]. 0-fill, 1-fill, minimum-transition-fill and random-fill are some of the popular don't-care bit filling techniques. Other thermal-aware don't-care bit filling works [3] attempt to reduce temperature using their own cost function to fill up the don't-care bits.

It is interesting to note that, both test data compression and thermal-aware don't-care filling concentrate on filling the don't-care bits efficiently to get better compression and low-temperature testing respectively. In the literature, these two problems have been solved separately. Test data compression works try to fill up the don't-care bits to get better compression ratio ignoring the thermal effect, while thermal-aware don't-care filling works try to minimize temperature by efficient don't-care bit filling without taking care of test data compression. It may so happen that, for a particular don't-care bit, better compression can be achieved if it is filled with '0' value, while a '1' value for that bit produces lower temperatures during testing. A trade-off between these two don't-care bit filling techniques is required to obtain a good compression ratio with reasonably low temperature.

In recent times, some works have addressed the problem of high power consumption in test data compression schemes [5], [6]. However, it may be noted that, power minimization may reduce overall temperature of a circuit, but, does not necessarily minimize peak temperature, which causes local hotspot due to non-uniform spatial power distribution in the circuit and hence permanent damage of it. The peak temperature of a block depends not only on power consumption, but also on heat exchange between adjacent blocks. So, the temperature of a circuit requires special attention.

In this paper, we have first shown the existence of variation in peak temperature of a circuit for a different selection of don't-care bit filling targeted towards compression. A thermal-

[0]This work is partially supported by the research project No. 9(5)/ 2010-MDD dt. 23/1/2011, sponsored by the DeitY, Govt. of India

1063-9667/15 $31.00 © 2015 IEEE
DOI 10.1109/VLSID.2015.14

aware test data compression technique, which can fill up the don't-care bits smartly taking care of compression and temperature simultaneously, has also been presented. This technique efficiently bridges the gap between test data compression and thermal-aware don't-care filling techniques. To the best of our knowledge, this is the first work, which addresses temperature reduction in test data compression schemes.

The rest of the paper is organised as follows. Section II describes the background of a dictionary based coding scheme, which we have used in our work. Motivation of the current work has been presented in Section III. Variation of temperature with a different clique selection for compression has been shown in Section IV. Section V proposes a thermal-aware test data compression scheme. Experimental results on different ISCAS'89 and ITC'99 benchmarks have been presented in Section VI. Section VII draws the conclusion.

II. Background of Dictionary Based Test Data Compression

In this paper we have used a dictionary-based coding technique for compression [1]. It is a popular test data compression technique in which the total test set is partitioned into several smaller equal-sized test slices. Suppose n test patterns, each of length L are required to test a circuit. Total test data can be calculated as $T_D = n \times L$. For a circuit with m number of scan chains, m bits of test data have to be transferred simultaneously from ATE to the circuit. Each such m bit data is called a test slice. If the length of a scan chain is l, total number of the slices formed is $n \times l$. The scan cells are divided into scan chains in a manner to keep the scan chains as balanced as possible. However, some don't-care bits may be required to pad to the shorter length test slices to make all of them equal sized. Two slices A and B can be treated as compatible if for any bit position i, either A_i and B_i ($1 \leq i \leq m, m$ is the size of the slice) are equal or at least one of them is a don't-care. A representative slice of the compatible slices may be stored in a dictionary instead of storing all of them. This helps to reduce the memory size required to store the test data. However, it may be noted that, dictionary size is kept relatively small to reduce hardware overhead. For a dictionary of size D, the representative slices from D largest compatible set of slices are stored in the dictionary. The rest of the slices are stored uncompressed in the memory. In dictionary-based coding, a single bit prefix is used to identify whether a slice belongs to the dictionary or not. The slices, whose representatives are stored in the dictionary, are denoted by a code of length $\lceil log_2|D| \rceil + 1$, while the code length for the rest of the slices is $m + 1$. Selection of D largest compatible set of slices is carried out via a clique partitioning heuristic [1]. The slices are represented by an undirected graph $G = (V, E)$, where each slice represents a vertex and the compatibility between two slices is denoted by an edge between them. The clique partitioning heuristic tries to find D largest possible cliques from the graph. The representative slices of these cliques are stored in the dictionary. A decompressor is used before CUT to get back the original patterns from the compressed codewords.

III. Motivation

Any clique partitioning technique uses the flexibility of don't-care bits to get the maximum number of matching slices. It may be noted that, most of the don't-care bits present in original patterns get filled up (either 0 or 1) at the time of generation of final patterns after clique partition, where all the compatible slices belonging to a single clique are replaced by the representative slice corresponding to that clique. For example, both of these two compatible slices (10XX01X1) and (X010X1XX) belonging to the same clique, are replaced by the representative slice (101001X1) in the final patterns. An observation from our experimentation is that around 72% of the total don't-care bits present in original test patterns get filled-up (either 0 or 1) in the clique partitioning process. However, as the clique partitioning is NP-Hard [1], different heuristics generate different clique sets. In that case, a slice may not belong to the same clique for different heuristics. The don't-care bits in the final test patterns also change as the don't-care bits present in a slice may fill up differently if the slice belongs to different cliques for different heuristics. It has a direct impact on the temperature profile of the circuit.

The temperature of a circuit largely depends on the dynamic power consumption, which can be estimated from the transition counts in the scan cells during shifting of test patterns through it. Different test pattern sets generate different thermal profiles of the circuit. As temperature is a major concern during testing, it is desirable to generate a test pattern set which produces a low - temperature profile of the circuit. Test designer has the flexibility to efficiently fill up the don't-care bits present in the original test patterns, to minimize the temperature of the circuit. Thermal-aware don't-care filling techniques fill up the don't-care bits efficiently to minimize temperature during testing, but may fail to produce a large number of compatible slices. This results in poor compression and hence increase the memory size and test cost. It is a challenging issue to fill up the don't-care bits efficiently to produce low temperature during testing, as well as a reasonably good compression ratio. Although, lots of research has been carried out to address test data compression and thermal-aware don't-care filling separately, none of the previous works have merged the problems into a single one and solved it. This has motivated us to develop a thermal-aware test data compression technique that brings balance between compression ratio and temperature of the IC.

IV. Variation of Temperature and Compression Ratio for Different Clique Partitions

The clique partitioning heuristic presented in [1] starts with the vertex with maximum number of neighbours and retrieves the largest clique associated with it from the graph and proceeds with searching for the next vertex with maximum neighbours from the remaining members of the graph until all the vertices have been retrieved. It may be noted that, instead

Slice index	Scan chain index							
	1	2	3	4	5	6	7	8
1	1	0	X	1	X	X	0	1
2	1	0	0	X	0	0	X	1
3	1	X	X	0	X	0	X	1
4	X	0	1	1	0	X	X	X
5	1	0	X	1	X	X	X	X
6	1	X	X	X	X	0	1	X
7	1	0	X	0	0	X	X	1
8	1	0	1	X	0	X	1	0
9	0	1	X	X	0	0	0	X
10	X	1	0	X	1	1	X	X
11	X	0	X	0	0	1	X	0
12	0	0	X	0	X	1	X	0
13	0	X	1	0	X	1	X	0
14	0	X	1	0	0	1	X	X
15	0	1	0	1	X	0	1	X
16	X	1	X	1	X	0	1	X
17	0	1	0	X	X	X	1	X
18	X	1	X	1	X	0	1	0

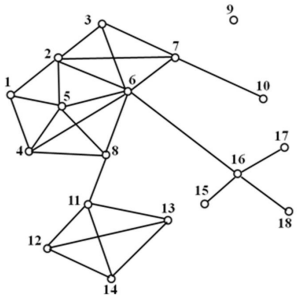

Figure 1. An example of test data for multiple scan chain set-up and corresponding compatibility graph.

of starting with vertex with highest neighbours, if we start with every vertex and then explore the total graph to find best possible clique sets from the graph, we get $n \times l$ number of clique sets for $n \times l$ number of vertices.

Figure 1 shows an example set of test data distribution in 8 scan chains and corresponding compatibility graph of the test data. There is a total of 18 slices, each of which has 8 bits.

For this particular example, we can have maximum 18 different clique sets. $Clique\_List\ CL_i$ $(1 \leq i \leq (n \times l))$ is a database that stores information regarding all generated cliques from the graph considering i as $start\_vertex$.

Some of the don't-care bits of the slices belonging to the same clique are filled up (either '0' or '1') to form a common representative slice for all of them.

All the slices from a particular clique are replaced by the corresponding representative slice of that clique. The modified details of all of the slices are stored in $Slice\_Info\ SI_i$ $(1 \leq i \leq (n \times l))$. These $Slice\_Info$ are used to generate modified test patterns (TP_i) $(1 \leq i \leq (n \times l))$ corresponding to CL_i. In the above mentioned example, the cliques of clique list CL_1 are {1, 2, 5}, {3, 6, 7}, {11, 12, 13, 14}, {4, 8}, {15, 16}, {10}, {17}, {18}, {9}, while the cliques of clique list CL_2 are {2, 3, 6, 7}, {1, 4, 5}, {11, 12, 13, 14}, {15, 16} {8}, {10}, {17}, {18}, {9}. It may be noted that, vertex 2 belongs to two different cliques for two different clique partitions. The don't-care bits present in slice 2 also fill up differently for two different cases. Original slice detail of slice 2 is (100X00X1), which is modified as (10010001) in SI_1, while the same slice is modified as (10000011) in SI_2. Final test pattern sets TP_1 and TP_2 also differ from each other. These two different test pattern sets may show different thermal behaviour.

We have shown a variation of the thermal profile for different clique selections for several ISCAS'89 and ITC'99 benchmarks. Circuit information regarding the test data volume and number of clique sets generated for different number of scan chains (N_{SC}) has been mentioned in Table I. Mintest [7] test sets for stuck-at fault are used for ISCAS'89 benchmarks, while test sets for stuck-at fault, generated from TetraMax [8] ATPG tool are used for ITC'99 benchmarks. Table II shows the variation of peak temperature and compression ratio (CR) for different clique selections.

Thermal profile of a circuit has been calculated using the thermal simulation steps mentioned below.

- Each Circuit description in Verilog format (.v) has been taken as input and mapped to Faraday 90nm standard cell library [9]. The circuits have been synthesized using Synopsys Design Vision Compiler [10] to generate a gate-level netlist from the Verilog design description.
- All the flip-flops have been replaced by scan flip-flops using Synopsys DFT Compiler for the testability purpose. Multiple scan chains have also been inserted.
- Encounter RTL-to-GDSII system from Cadence [11] has been used to generate floorplan of the given scan inserted design using standard cell library.
- The scan inserted netlist and the test patterns have been fed to a power estimator tool [3]. Power consumption of each of the logic elements (gate, flip-flop) with different types and inputs has been estimated using Synopsys Design Vision tool and stored in a database. This database has been used to convert the transitions in individual logic elements into their corresponding power values during circuit simulation for each scan shift and capture operation.
- The floorplan obtained from Cadence Encounter tool [11] has been divided into a number of blocks of same size for thermal simulation. The power trace has been computed for each of these blocks.
- Thermal simulations have been carried out using thermal simulation tool HotSpot [12] taking power trace and block-level floorplan as input.

However, accessibility of the power estimator tool [3] has been upgraded from single scan chain operation to multiple scan chain operation.

From Table II, a significant variation in peak temperature can be noticed for different clique selections while the corresponding variation of CR is reasonably very less. This shows the scope of obtaining good test data compression with low temperature profile, by efficient don't-care filling.

V. Thermal-Aware Compression

In the previous section, we have seen the variation in peak temperature and compression ratio with different clique selections. Although, selecting a solution with lower peak temperature and good compression ratio from all clique lists may be a choice to address both low temperature and good compression issues, however, this method relies on detailed enumeration to find clique set with low temperature and hence may not be applicable for the complex circuits with huge amount of test data. Moreover, this method does not fill the don't-care bits to minimize peak temperature. To get a significant reduction in the peak temperature of the chip, some thermal-aware don't-care filling technique has to be incorporated in the clique partitioning heuristic. However,

Table I
CLIQUE SET INFORMATION FOR DIFFERENT SCAN CHAINS

Circuit name	Test vector length	No of test vectors	T_D	Number of clique sets			
				N_{SC} = 32	N_{SC} = 48	N_{SC} = 64	N_{SC} = 128
$s13207$	700	236	165200	5192	3540	2596	1416
$s15850$	611	126	76986	2520	1638	1260	630
$s38417$	1664	99	164736	5148	3465	2574	1287
$s38584$	1464	136	199104	6256	4126	3128	1632
$b14$	277	762	211074	6858	4572	3810	2286
$b15$	485	457	222102	7312	5027	3656	1828

Table II
VARIATION OF TEMPERATURE AND COMPRESSION RATIO FOR DIFFERENT CLIQUE SELECTIONS (FOR D =128)

Circuit name	N_{SC} = 32				N_{SC} = 48				N_{SC} = 64				N_{SC} = 128			
	Temp(K)		CR		Temp(K)		CR		Temp(K)		CR		Temp(K)		CR	
	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min
$s13207$	495.8	452.5	73.4	73.2	554.5	487.5	81.3	81.0	632.0	592.0	85.2	84.8	801.4	742.1	91.9	91.4
$s15850$	486.6	454.8	67.9	67.0	534.3	504.8	73.8	72.7	583.2	544.6	76.8	76.0	685.5	643.8	82.3	81.1
$s38417$	475.1	467.4	45.5	43.7	529.6	514.9	54.2	51.8	537.4	529.1	43.1	37.1	721.1	707.7	34.0	32.2
$s38584$	555.2	538.7	63.0	62.3	662.8	638.6	68.0	67.0	733.2	706.3	68.3	67.3	967.4	941.9	70.6	68.9
$b14$	461.8	456.4	56.8	55.0	471.9	467.5	47.4	45.3	544.2	524.7	47.5	45.8	532.4	505.0	52.1	51.2
$b15$	438.8	423.3	73.3	73.0	480.0	454.4	79.1	78.4	491.4	468.8	81.2	80.2	541.2	527.9	75.4	73.4

compression ratio may have to be compromised to some extent to get a good thermal-aware test pattern set.

High power consumption during shifting of test vectors in the scan chains plays a major role in temperature increase of the circuit. Large number of intermediate patterns are generated during shifting in (out) of a particular test vector through the scan chains. Power consumption increases because of these intermediate patterns.

Minimizing the transition count in scan chains by efficient thermal-aware don't-care filling is a feasible solution to reduce peak temperature, as well as circuit power.

Thermal-aware don't-care filling technique presented in [3] produces reasonably low temperature test pattern set. In [3], the entire circuit has been divided into several blocks consisting of several gates and flip-flops. A power estimation tool [3] identifies the flip-flops with high transitions in each block. These flip-flops are called critical flip-flops as they play a leading role in high power consumption of the block. Each block has been given a weight as per the power consumption of the block. However, as the power of neighboring blocks may have a large impact on the temperature of a particular block, floorplan information of the circuit has to be incorporated with the power estimation tool to estimate the temperature of the blocks. The temperature of the hottest block with highest estimated temperature has been tried to minimize with an aim to minimize the peak temperature of the circuit.

In this section, we present a *Thermal-Aware Clique Partitioning Heuristic* which can take care of both compression as well as the temperature of the chip.We have used the test pattern set generated using the method described in [3] as the base case of thermal-aware solution. The don't-care filling technique mentioned in [3] can only handle the circuits with single scan chain set-up. Our modified thermal-aware don't-care bit filling technique can fill up don't-care bits taking care of the multiple scan chains. We start with two test

Algorithm 1: ***Thermal-Aware Clique Partition Heuristic***

input : Original test patterns TP_O with don't-care bits
output: Final test patterns TP_{FINAL} with a trade-off between temperature and compression ratio

begin
- Fill up don't-care bits presented in the test patterns using thermal-aware don't-care bit filling technique [3] to get thermal-aware test patterns TP_{TH};
- Store the thermal-aware test slices obtained from TP_{TH} in thermal-aware $Slice\_Info$ (SI_{TH});
- Apply clique partitioning heuristic [1] on TP_O; Get corresponding compression-aware $Clique\_List$ (CL_{CM}) and $Slice\_Info$ (SI_{CM}); Compute compression-aware test pattern TP_{CM} from SI_{CM};
- Sort the compression-aware $Clique\_List$ (CL_{CM}) in descending order on the basis of the number of elements present in a clique;
- Select a weightage factor Wt within a range of 0 to 1. (0 $\rightarrow$ thermal-aware don't-care filling; 1 $\rightarrow$ compression-aware don't-care filling);
- Calculate total slice conversion $TSC = \lfloor Wt \times n \times l \rfloor$;
- The slices of SI_{TH} corresponding to top most TSC clique members of sorted $Clique\_List$ (CL_{CM}) are replaced with the slices from SI_{CM}. Modified slice info is SI_{NEW};
- Compute new test patterns TP_{NEW} from SI_{NEW};
- Apply clique partitioning heuristic [1] on TP_{NEW}; Get corresponding $Clique\_List$ (CL_{FINAL}) and $Slice\_Info$ (SI_{FINAL}) ;Compute final test patterns TP_{FINAL} from SI_{FINAL}; Calculate compression ratio of TP_{FINAL}; Apply TP_{FINAL} to the circuit; calculate peak temperature using temperature calculation method mentioned in Section IV;

pattern sets, one with all the don't-care bits filled up using thermal-aware don't-care bit filling technique and the other one is the original test pattern set TP_O with all the don't-care bits retained. The test pattern set with filled don't-cares can be named as *thermal-aware test pattern set* (TP_{TH}). Our objective is to find the final set of test patterns TP_{FINAL} with a trade-off between temperature and compression ratio. As no don't-care bit is present in TP_{TH}, it cannot use the flexibility of having don't-care bits, which can be filled up suitably to increase the number of compatible slices. Hence, it produces a poor compression ratio. On the other hand, TP_O gets the full flexibility to fill up the don't-care bits to achieve a high compression ratio. Clique partitioning heuristic [1] is applied on TP_O to generate a compression-aware $Clique\_List$ (CL_{CM}) and corresponding $Slice\_Info$ (SI_{CM}). The slices belonging to SI_{CM} are called *compression-aware slices* as the test pattern set TP_{CM} formed from SI_{CM} produces a good compression ratio. The slices obtained from TP_{TH} are called *thermal-aware slices* due to their low-temperature effect. However, these *compression-aware slices* and *thermal-aware slices* differ in the process of their don't-care bit filling. A new $Slice\_Info$ (SI_{NEW}) can be formed taking some of the *thermal-aware slices* from SI_{TH} and some of the *compression-aware slices* from SI_{CM} to have the advantage of both good compression as well as low temperature. A new set of test patterns is generated using the slice information of SI_{NEW}. Clique partitioning heuristic [1] is applied on TP_{NEW} to get a final clique list CL_{FINAL} and corresponding $Slice\_Info$ (SI_{FINAL}). Final test pattern set TP_{FINAL}, computed from SI_{FINAL}, produces a balance between compression ratio and temperature. However, the percentage of *compression-aware slices* that will be there in SI_{NEW} (with the rest of the *thermal-aware slices*), can be chosen flexibly using a weightage factor Wt with a value in the range of '0' to '1'. A '0' value of Wt chooses all the *thermal-aware slices*, while a '1' value of it refers to all *compression-aware slices* being chosen. Any value between '0' and '1' shows a trade-off between temperature and compression ratio.

VI. Experimental Results and Discussions

In this section, we present the results of our thermal-aware test data compression technique on several ISCAS'89 and ITC'99 benchmarks. Table III shows the trade-off between temperature and compression ratio for different weightage factor Wt. Temperature calculated using the method mentioned in Section IV. Compression ratio (CR) for the dictionary size D = 128 for different N_{SC} values are shown in the table. Wt is varied from '0' to '1' and for each value of Wt corresponding temperature in Kelvin (T(K)) and CR (in %) are noted for different N_{SC}. It may be noted that, for the value of Wt = 0, all don't-cares are filled thermally, hence, the test pattern set produces minimum temperature, but the corresponding compression ratio is very poor, while for Wt = 1, although the compression ratio is impressive, it produces a high peak temperature, which can be a serious threat to the thermal safety of the chip. Other values of Wt show a balance between compression ratio and temperature. For Wt = 0, all the slices being *thermal-aware slices* are expected to produce minimum temperature. With the gradual increase of Wt from '0' to '1', more numbers of *compression-aware slices* replace *thermal-aware slices*, which increase the compression ratio, but fail to reduce the temperature of the chip. It may be noted that in some cases, the trade-off may not be available. In Table III, one such situation could be observed for circuit $b14$ with Wt larger than 0.5. This we believe has happened due to the heuristic nature of the clique partitioning process.

Figure 2 shows a graphical representation of CR and temperature for different values of Wt for different scan chain set-ups of benchmark $s38584$. It may be noted from the figure that temperature increases to a large extent with the increase of the value of N_{SC}. This is due to transfer of more number of test data at a time, which increase transition counts in the scan chains and hence more power consumption. Increasing the number of scan chains may transfer test data faster reducing test time of the chip, but at the same time increase in the temperature may cause permanent damage of the chip. However, the variation of CR with N_{SC} varies from circuit to circuit. Although few benchmarks like $s13207$ and $s15850$ show better compression for larger N_{SC}, some other benchmark like $s38417$ shows opposite effect, while for other benchmarks CR do not vary much with the increase of N_{SC}.

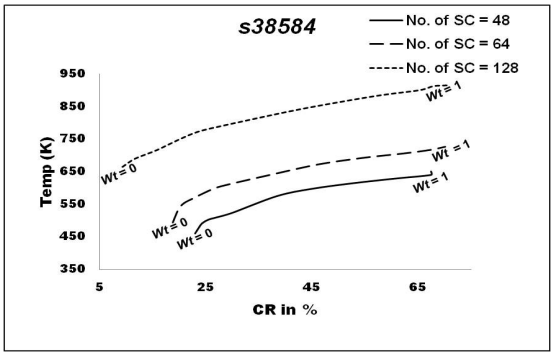

Figure 2. Variation of temperature and CR with Wt for different scan chain set-up for benchmark $s38584$.

As none of the previous compression works reported in the literature have addressed the thermal issue of compression, we could not make a direct comparison of our work with those. However, our best case CR is comparable with the results presented in [1]. Although some other works [13], [14] achieve a better compression ratio with more computational complexity and extra hardware overhead, all are based on the clique partitioning. Method like dividing the test slices further into smaller sub-slices [13] can also be incorporated with our method to improve compression ratio without hampering the balance between CR and temperature of the chip.

Table III
VARIATION OF TEMPERATURE AND COMPRESSION RATIO FOR DIFFERENT VALUES OF Wt FOR DIFFERENT SCAN CHAINS (FOR D = 128)

Circuit name	N_{SC}		Weightage factor Wt												CR [1]
			0	**0.05**	**0.1**	**0.15**	**0.2**	**0.25**	**0.4**	**0.5**	**0.6**	**0.7**	**0.8**	**1.0**	
$s13207$	48	T(K)	372.56	383.31	393.01	402.8	412.1	418.8	436.2	437.03	451.92	473.48	484.77	497.5	81.1
		CR(%)	63.81	63.81	63.86	63.86	64.05	64.05	65.57	65.79	67.55	69.63	71.92	81.01	
	64	T(K)	398.3	420.96	447.65	468.91	488.31	506.85	556.13	558.25	583.72	592.8	602.35	616.12	85.4
		CR(%)	63.28	63.4	63.4	63.58	63.83	64.34	65.61	66.95	69.29	71.68	75.8	85.01	
	128	T(K)	448.32	492.8	520.89	553.25	576.23	594.66	652.87	698.52	726.36	745.78	758.96	772.24	91.5
		CR(%)	63.50	63.64	63.90	64.77	66.38	66.77	69.05	72.05	74.66	78.19	83.40	91.3	
$s15850$	48	T(K)	392.41	409.88	426.95	443.07	448.5	458.47	486.25	495.74	505.53	514.8	521.28	524.43	74.4
		CR(%)	42.18	42.24	42.34	42.76	43.23	43.96	47.3	50.74	56.47	63.1	69.25	74.2	
	64	T(K)	414.29	436.94	455.75	469.9	482.47	491.27	520.27	531.11	540.86	550.7	557.61	557.75	75.8
		CR(%)	38.79	38.84	39.58	41.55	41.76	43.46	48.76	50.95	54.20	67.62	73.43	75.84	
	128	T(K)	508.67	530.66	546.64	558.57	567.14	569.07	590.95	601.54	611.76	618.98	622.86	623.14	81.9
		CR(%)	35.25	35.8	38.38	41.53	43.94	45.88	57.13	64.64	72.44	78.59	80.84	81.59	
$s38417$	48	T(K)	431.21	447.9	458.14	469.68	477.87	482.53	492.92	497.1	501.78	505.63	508.75	511.22	61.8
		CR(%)	11.12	13.08	16.17	19.3	22.42	25.43	35.16	42.38	49.47	52.43	55.43	57.92	
	64	T(K)	461.66	491.1	508.3	522.71	531.08	538.65	551.69	556.23	561.19	567.31	569.5	573.72	42.7
		CR(%)	10.89	12.91	14.98	17.68	20.79	23.63	35.21	37.19	37.93	38.49	41.73	43.30	
	128	T(K)	635.94	659.79	672.36	685.7	701.0	709.56	739.3	751.57	753.0	753.99	757.98	759.18	36.7
		CR(%)	9.28	13.39	17.28	20.96	24.33	27.49	32.85	33.58	34.95	36.57	36.57	36.57	
$s38584$	48	T(K)	460.76	477.85	492.33	503.1	510.02	524.4	575.41	597.81	615.11	629.32	640.58	651.07	67.9
		CR(%)	23.03	23.69	24.37	25.51	27.04	30.06	38.43	45.01	52.81	60.74	67.47	67.47	
	64	T(K)	493.85	532.39	552.93	569.26	587.73	605.15	643.52	673.24	692.82	706.65	717.69	727.08	70.8
		CR(%)	18.82	19.82	20.87	22.78	25.06	27.82	37.95	46.35	54.53	62.35	67.37	70.13	
	128	T(K)	664.72	689.91	716.01	745.47	772.28	790.48	834.82	862.02	884.83	900.48	912.64	915.01	70.7
		CR(%)	9.39	11.76	15.83	19.6	23.55	28.11	40.75	49.68	58.07	65.43	67.97	70.87	
$b14$	48	T(K)	388.26	394.64	400.77	408.38	416.31	422.81	438.17	446.21	453.59	459.93	465.10	469.58	-
		CR(%)	22.74	23.58	24.57	25.6	27.56	29.95	37.15	42.96	46.47	46.47	46.47	45.61	
	64	T(K)	416.33	416.56	416.48	424.68	435.75	443.4	471.1	487.35	501.31	512.58	521.11	525.33	-
		CR(%)	26.8	26.91	26.97	27.58	29.01	30.6	37.94	44.32	46.31	45.96	44.74	43.78	
	128	T(K)	499.8	500.52	500.84	501.16	501.67	501.98	530.25	562.32	584.31	599.38	606.01	607.05	-
		CR(%)	38.1	38.1	38.1	38.1	39.4	39.4	42.4	45.57	51.69	53.79	55.76	61.72	
$b15$	48	T(K)	348.41	350.5	356.51	377.8	393.45	409.38	419.85	427.5	435.95	445.5	451.44	459.48	-
		CR(%)	45.44	45.51	45.68	47.0	48.27	50.74	53.78	56.92	60.03	65.01	69.82	78.95	
	64	T(K)	351.17	365.75	377.13	390.82	399.76	407.72	429.05	443.04	449.39	458.77	466.63	473.88	-
		CR(%)	33.98	35.25	36.93	39.73	41.6	43.62	49.57	54.58	59.75	65.67	72.12	80.65	
	128	T(K)	398.35	417.13	436.27	452.36	464.68	477.05	491.94	506.31	516.1	523.75	529.44	532.13	-
		CR(%)	18.71	21.4	24.55	28.23	31.59	34.95	44.82	52.63	60.08	67.63	73.53	73.53	

VII. CONCLUSIONS

In this paper we have presented a thermal-aware test data compression technique, which does a trade-off between compression ratio and temperature of the chip by efficiently filling the don't-care bits present in the test patterns. Clique partitioning guided dictionary based coding technique has been adopted for test data compression. The flexibility of our technique helps to choose a suitable balance between thermal safety and test cost to store huge amount of test data.

REFERENCES

[1] L. Li and K. Chakrabarty, "Test data compression using dictionaries with fixed-length indices [soc testing]," in *VLSI Test Symposium, 2003. Proceedings. 21st*, April 2003, pp. 219–224.

[2] K. Butler, J. Saxena, A. Jain, T. Fryars, J. Lewis, and G. Hetherington, "Minimizing power consumption in scan testing: pattern generation and dft techniques," in *Test Conference, 2004. Proceedings. ITC 2004. International*, Oct 2004, pp. 355–364.

[3] A. Dutta, S. Kundu, and S. Chattopadhyay, "Thermal aware don't care filling to reduce peak temperature and thermal variance during testing," in *Test Symposium (ATS), 2013 22nd Asian*, Nov 2013, pp. 25–30.

[4] T. Yoneda, M. Nakao, I. Inoue, Y. Sato, and H. Fujiwara, "Temperature-variation-aware test pattern optimization," in *European Test Symposium (ETS), 2011 16th IEEE*, May 2011, pp. 214–214.

[5] M. Chen and A. Orailoglu, "Scan power reduction for linear test compression schemes through seed selection," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 20, no. 12, pp. 2170–2183, Dec 2012.

[6] X. Liu and Q. Xu, "On x-variable filling and flipping for capture-power reduction in linear decompressor-based test compression environment," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 31, no. 11, pp. 1743–1753, Nov 2012.

[7] I. Hamzaoglu and J. Patel, "Test set compaction algorithms for combinational circuits," in *Computer-Aided Design, 1998. ICCAD 98. Digest of Technical Papers. 1998 IEEE/ACM International Conference on*, Nov 1998, pp. 283–289.

[8] *TetraMax ATPG Guide*, Synopsys Inc., 2006.

[9] "Faraday," http://www.faraday.com.tw/AIP/ips/90library.html, 2011.

[10] *Design Vision "User Guide", Version 2002.05*, Synopsys Inc., 2002.

[11] *Encounter "User Guide", Product Version 7.1.2*, Cadance Inc., 2008.

[12] M. R. Stan, K. Skadron, M. Barcella, W. Huang, K. Sankaranarayanan, and S. Velusamy, "Hotspot: a dynamic compact thermal model at the processor-architecture level." *Microelectronics Journal*, vol. 34, no. 12, pp. 1153–1165, 2003.

[13] P. Sismanoglou and D. Nikolos, "Input test data compression based on the reuse of parts of dictionary entries: Static and dynamic approaches," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 32, no. 11, pp. 1762–1775, Nov 2013.

[14] K. Basu and P. Mishra, "Test data compression using efficient bitmask and dictionary selection methods," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 18, no. 9, pp. 1277–1286, Sept 2010.