Accepted for publication in IEEE Transactions on CAD/ICAS

Low-Power Scan Testing and Test Data Compression for System-on-a-Chip[1]

Anshuman Chandra and Krishnendu Chakrabarty

Department of Electrical and Computer Engineering
Duke University
130 Hudson Hall, Box 90291
Durham, NC 27708, USA

Contact author: Anshuman Chandra
Phone: (919) 660-5230, Fax: (919) 660-5293
E-mail: achandra@ee.duke.edu

ABSTRACT

Test data volume and power consumption for scan vectors are two major problems in system-on-a-chip testing. Since static compaction of scan vectors invariably leads to higher power for scan testing, the conflicting goals of low-power scan testing and reduced test data volume appear to be irreconcilable. We tackle this problem by using test data compression to reduce both test data volume and scan power. In particular, we show that Golomb coding of precomputed test sets leads to significant savings in peak and average power, without requiring either a slower scan clock or blocking logic in the scan cells. We also improve upon prior work on Golomb coding by showing that a separate cyclical scan register is not necessary for pattern decompression. Experimental results for the larger ISCAS 89 benchmarks show that reduced test data volume and low power scan testing can indeed be achieved in all cases.

Keywords: Embedded core testing, Golomb codes, precomputed test sets, scan testing, switching activity, test set encoding, power reduction.

[1]This research was supported in part by the National Science Foundation under grant number CCR-9875324, and in part by an equipment grant from Intel Corporation. An abridged version of this paper was published in *Proc. Design Automation Conference*, pp. 166-169, Las Vegas, June 2001.

1 Introduction

Pre-designed intellectual property (IP) cores are now commonly used in large system-on-a-chip (SOC) designs [1]. An SOC design integrates multiple cores (e.g., microprocessor, memory, DSPs, and I/O controllers) on a single piece of silicon. Despite these benefits, IP cores pose several difficult test challenges. Two problems that are becoming increasingly important are power consumption during manufacturing test and test data volume. The precomputed test patterns provided by the core vendor must be applied to each core within the power constraints of the SOC. In addition, test data compression is necessary to overcome the limitations of the automatic test equipment (ATE), e.g. tester data memory and I/O channel capacity.

Power consumption during testing is important since excessive heat dissipation can damage the circuit under test. Since power consumption in test mode is higher than during normal operation, special care must be taken to ensure that the power rating of the SOC is not exceeded during test application [2]. A number of techniques to control power consumption in test mode have been presented in the literature. These include test scheduling algorithms under power constraints [3], low-power built-in self-test (BIST) [4, 5, 6, 7], and techniques for minimizing power during scan testing [8, 9, 10]. Power consumption and the resulting heat dissipation are especially important for SOCs since test scheduling techniques and test access architectures for system integration attempt to reduce testing time by applying scan/BIST vectors to several cores simultaneously [11–15]. Therefore, it is extremely important to decrease power consumption while testing the IP cores in an SOC.

Test data volume is another problem faced in SOC test integration. One way to alleviate this problem is to use BIST. However, BIST can only be applied to SOCs if the IP cores in them are BIST-ready. Since most currently-available IP cores are not BIST-ready, the incorporation of BIST in them requires considerable redesign. Hence test data compression techniques that facilitate low-power scan testing are desirable for SOC testing.

The conflicting goals of low-power scan testing and reduced test data volume appear to be irreconcilable. Test generation for low-power scan testing usually leads to an increase in the number of test vectors [8]. On the other hand, static compaction of scan vectors causes significant increase in power consumption during testing [10]. The compacted vectors are rendered useless if they exceed power constraints. Clearly, uncompacted vectors cannot be used since they require excessive tester memory. This problem is addressed in a recent paper on power-constrained static compaction of scan vectors [10]. However, while [10] provides

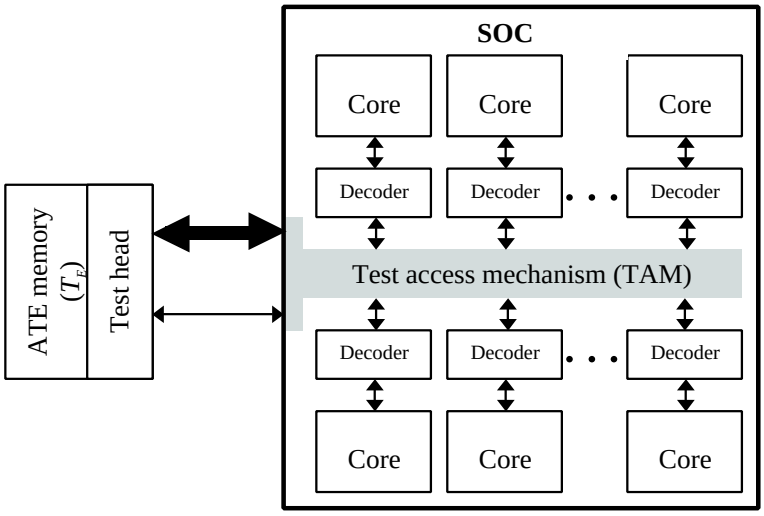


Figure 1: A conceptual architecture for testing a system-on-chip by storing the encoded test data T_E in ATE memory and decoding it using on-chip decoders.

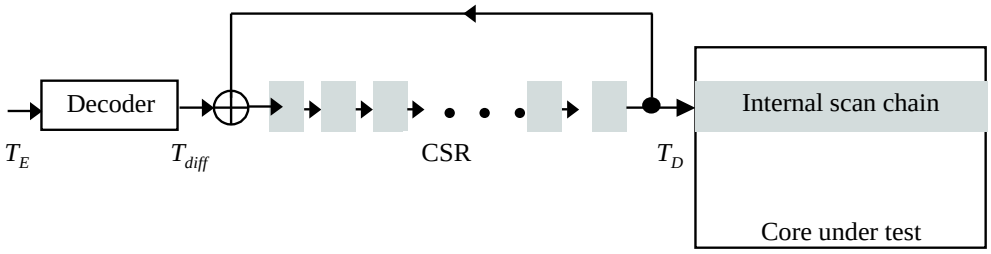


Figure 2: Decompression architecture based on a cyclical scan register (CSR).

2-3 times reduction in power consumption for several ISCAS benchmark circuits, it does not lead to any appreciable reduction in test data volume—in fact, it does not provide any improvement over standard static vector compaction techniques. Furthermore, the scheme presented in [10] only addresses scan-in power and it does not consider power dissipation during the scan-out operation.

Recently, a number of data compression techniques have been proposed for reducing SOC test data volume [17–24]. In this approach, the precomputed test set T_D provided by the core vendor is compressed (encoded) to a much smaller test set T_E and stored in ATE memory; see Figure 1. An on-chip decoder is used for pattern decompression to generate T_D from T_E during pattern application. It was shown in [19, 20, 24] that compressing a "difference vector" sequence T_{diff} determined from T_D results in smaller test sets and reduces testing time. Figure 2 shows the test architecture based on T_{diff} and a cyclical scan registers (CSR). An obvious drawback of this approach is that it requires a separate CSR.

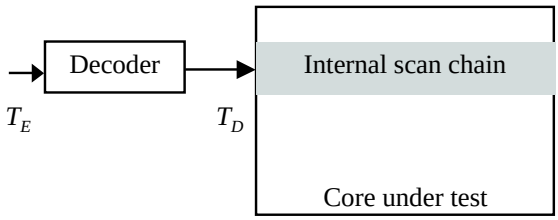


Figure 3: Test architecture based on Golomb coding of the precomputed test set T_D.

In this paper, we dispel the notion that scan vector compaction always leads to higher power consumption. Since static compaction invariably leads to higher power, we explore test data compression for overcoming this problem. We show that test data compression leads to significant reduction in power consumption during scan testing. In particular, we show that we can decrease both peak and average power by using Golomb codes for compressing the scan vectors of IP cores. In this way, there is no need to either reduce the scan clock rate for low power or add blocking logic to the scan cells [5]. The use of a low-cost on-chip decoder allows us to achieve significant test data compression, and the decompressed scan vectors cause very little switching activity in the scan chains during test application. While we only target scan-in power in our compression scheme, we show experimentally that significant savings are also obtained in scan-out power. In addition, we show that it is not necessary to use a separate CSR; we can directly encode T_D and thereby obviate the need for the CSR (Figure 3).

The organization of the paper is as follows. In Section 2, we first review Golomb coding. We then describe the data compression procedure and the decompression architecture, and highlight the key differences from [20]. Section 3 shows how we can combine low-power scan testing with test data compression. Finally, in Section 4 we present experimental results for the large ISCAS 89 benchmark circuits as well as for a real-life microprocessor circuit from IBM.

2 Compression method and test architecture

We first review Golomb coding and its application to test data compression in [20]. The major advantages of Golomb coding of test data include very high compression, analytically-predictable compression results, and a low-cost and scalable on-chip decoder. In addition, the novel interleaving decompression architecture allows multiple cores in an SOC to be tested concurrently using a single ATE I/O channel.

If the difference vector T_{diff} is used for compression, the first step is to derive it from T_D, where T_D = $\{t_1, t_2, t_3, ..., t_n\}$, is the (ordered) precomputed test set. The ordering is determined using a heuristic procedure described later. T_{diff} is defined as follows:

$$T_{diff} = \{d_1, d_2, \ldots, d_n\} = \{t_1, t_1 \oplus t_2, t_2 \oplus t_3, ..., t_{n-1} \oplus t_n\}.$$

where a bit-wise exclusive-or operation is carried out between patterns t_i and t_j. This assumes that the CSR starts in the all-0 state. (Other starting states can be considered similarly.)

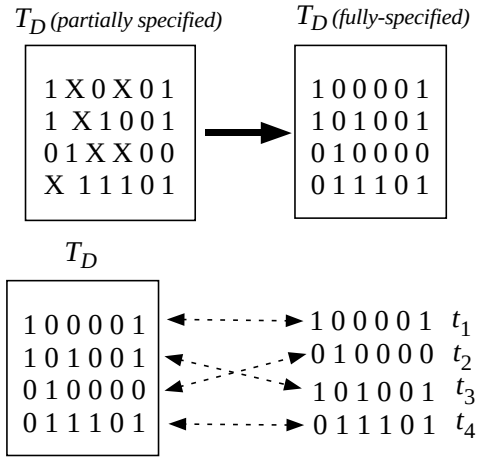
In this work however, we encode T_D directly, hence there is no need to generate T_{diff}. All the don't-care bits in T_D are mapped to 0s to obtain a fully-specified test sequence. We show in Section 4 that better compression is obtained using T_D instead of T_{diff} in almost all cases.

The next step in the encoding procedure is to select the Golomb code parameter m, referred to as the group size [21]. Once m is determined, e.g. using the methods described in [21], the runs of 0s in the test data stream are mapped to groups of size m (each group corresponding to a run length). The number of such groups is determined by the length of the longest run of 0s in T_D. The set of run-lengths $\{0, 1, 2, \ldots, m-1\}$ forms group A_1; the set $\{m, m+1, m+2, \ldots, 2m-1\}$, group A_2; etc. In general, the set of run-lengths $\{(k-1)m, (k-1)m+1, (k-1)m+2, \ldots, km-1\}$ comprises group A_k. To each group A_k, we assign a group prefix of $(k-1)$ 1s followed by a 0. We denote this by $1^{(k-1)}0$. If m is chosen to be a power of 2 i.e., $m = 2^N$, each group contains 2^N members and a $\log_2 m$-bit sequence (tail) uniquely identifies each member within the group. Thus, the final code word for a run-length L that belongs to group A_k is composed of two parts—a group prefix and tail. The prefix is $1^{(k-1)}0$ and the tail is a sequence of $\log_2 m$ bits. The encoding process is illustrated in Table 1 for $m = 4$.

Scan vectors can be reordered to decrease test data volume. The problem of determining the best ordering of test vectors is equivalent to the NP-Complete Traveling Salesman problem. Therefore, a greedy algorithm is used to generate an ordering and the corresponding T_D. Suppose a partial ordering $t_1 t_2 \ldots t_i$ has already been determined for the patterns in T_D. To determine t_{i+1}, we calculate the Hamming distance $HD(t_i, t_j)$ between t_i and all patterns t_j that have not been placed in the ordered list. We define $HD(t_i, t_j)$ as the number of 0s in the pattern t_j. This metric is chosen since it tends to produce longer runs of 0s. We select the pattern t_j for which $HD(t_i, t_j)$ is maximum and add it to the ordered list, denoting it by t_{i+1}. In this way, a fully-specified test pattern is obtained and the smallest number of 1s is added to the ordered vector sequence. We continue this process until all test patterns in T_D are placed in the ordered list. Figure 4 illustrates the procedure for obtaining fully specified ordered T_D.

Group	Run-length	Group prefix	Tail	Codeword
A_1	0	0	00	000
	1		01	001
	2		10	010
	3		11	011
A_2	4	10	00	1000
	5		01	1001
	6		10	1010
	7		11	1011
A_3	8	110	00	11000
	9		01	11001
	10		10	11010
	11		11	11011
...	...	...	...	...

Table 1: An example of Golomb coding for $m = 4$.



$T_D = \{t_1, t_2, t_3, t_4\} = \{100001, 010000, 101001, 011101\}$

Figure 4: An example to illustrate the procedure of deriving fully specified ordered T_D.

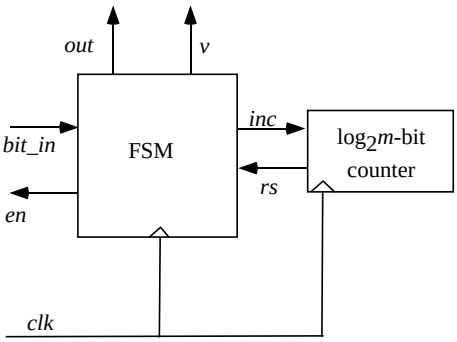


Figure 5: The decoder block diagram for Golomb code parameter $m = 4$ [20].

An on-chip decoder decompresses the encoded test set T_E and produces T_D. Even though T_D contains more patterns than test sets obtained after static compaction of ATPG vectors, the testing time is reduced since pattern decompression can be carried out on-chip at higher clock frequencies. As discussed in [20], the decoder can be efficiently implemented by a $\log_2 m$-bit counter and a finite-state machine (FSM) and is independent of the precomputed test set and the circuit under test. The block diagram of the decoder for $m = 4$ is shown in Figure 5. The synthesized decode FSM circuit contains only 4 flip-flops and 34 combinational gates. For any circuit whose test set is compressed using $m = 4$, the given logic is the only additional hardware required other than the 2-bit counter. This is especially the case if, unlike in [20], T_D is directly used for encoding and a CSR is not required for decompression.

Since the decoder for Golomb coding needs to communicate with the tester, and both the codewords and the decompressed data can be of variable length, proper synchronization must be ensured through careful design. In particular, the decoder must communicate with the tester to signal the end of a block of variable-length decompressed data. These and other related decompression issues are discussed in detail in [20].

3 Power estimation for scan vectors

In this section, we examine the impact of test set encoding on power consumption during scan testing. We then show how power consumption can be minimized by appropriately assigning binary values to the don't-care bits in T_D and then applying Golomb coding for test data compression.

For a CMOS circuit, power consumption can be classified as either static or dynamic. Static power consumption, which is caused by leakage current, is usually negligible and therefore ignored. Dynamic power is consumed when the the outputs of circuit elements from high-to-low and low-to-high transitions.

This constitutes the predominant fraction of CMOS power consumption.

For scan vectors, the dynamic power consumption during testing depends on the number of transitions that occur in the scan chain as well as on the number of circuit elements that switch during the scan in and scan out operations. Power estimation models based on the switching activity of circuits have been presented in the literature [6, 10]. We use the weighted transitions metric (WTM) introduced in [10] to estimate the power consumption due to scan vectors. This model was validated in [10], hence we do not report on its accuracy in this paper. The WTM metric models the fact that the scan in power for a given vector depends not only on the number of transitions in it but also on their relative positions. For example, consider a scan vector $v_1v_2v_3v_4v_5 = 01000$, where v_1 is first loaded into the scan chain. The 0-to-1 transition between v_1 and v_2 causes more switching activity in the scan chain than the 1-to-0 transition between v_2 and v_3. We use the same model to estimate the power consumption during scan out operation.

The weighted transitions count metric is also strongly correlated to the switching activity in the internal nodes of the core under test during the scan in operation. It was shown experimentally in [10] that scan vectors have have higher weighted transition metric dissipate more power in the core under test.

Consider a scan chain of length l and a scan vector $t_j = t^\star_{j,1}t^\star_{j,2}\ldots t^\star_{j,l}$, with $t^\star_{j,1}$ scanned in before $t^\star_{j,2}$, and so on. The *weighted transitions metric* for t_j, denoted WTM_j, is given by $WTM_j = \sum_{i=1}^{l-1}(l-i)\cdot(t^\star_{j,i} \oplus t^\star_{j,i+1})$. If the test set T_D contains n vectors $t_1, t_2, \ldots, t_n$ then the average scan in power P_{avg} and peak scan in power P_{peak} are estimated as follows:

$$P_{avg} = \frac{\sum_{j=1}^{n}\sum_{i=1}^{l-1}(l-i)\cdot(t^\star_{j,i} \oplus t^\star_{j,i+1})}{n}$$

$$P_{peak} = \max_{j\in\{1,2,\ldots,n\}}\{\sum_{i=1}^{l-1}(l-i)\cdot(t^\star_{j,i} \oplus t^\star_{j,i+1})\}.$$

If the peak power exceeds a threshold value, it can cause structural damage to the silicon or to the package. Likewise, elevated average power can also cause structural damage to the silicon, bonding wires or the package. It also adds to the thermal load that must be transported away from the device under test.

We next show how Golomb codes can be used to minimize the volume of test data and at the same time, minimize P_{avg} and P_{peak}. Scan-in power is influenced by the manner in which the don't-cares in T_D are mapped to binary values. While P_{avg} and P_{peak} can be minimized by choosing an appropriate mapping, such a mapping is not guaranteed to provide high test data compression. In fact, our experiments show that the encoded test sets in such cases are often larger than the uncompacted test sets. Instead, it is far more

Partially-specified scan vector	Fully-specified vector (Minimum WTM)	Fully-specified vector (Don't-cares mapped to 0s)
$t_i = 01XXX10XXX01$	011111000001 Golomb code length: 19 bits ($m = 4$) $WTM_i = 18$	010001000001 Golomb code length: 10 bits ($m = 4$) $WTM_i = 37$
$t_j = 0XX1010XXXX1$	000101000001 Golomb code length: 10 bits ($m = 4$) $WTM_j = 31$	010101000001 Golomb code length: 13 bits ($m = 4$) $WTM_j = 52$

Table 2: Mapping of don't-cares in T_D to binary values.

efficient to simply map all the don't-cares in T_D to 0s as shown in Figure 4. While this approach does not minimize P_{avg} and P_{peak}, it provides significant reductions in power consumption, and at the same time, decreases the test data volume considerably. The fully-specified test set thus obtained is then compressed using Golomb codes. Since uncompacted test cubes contain a large number of don't-cares, mapping these don't-cares to 0s results in long runs of 0s. These long runs of 0s provide very high test data compression, as well as reduced transitions during scan in. Even though we do not directly address scan-out power, our experiments with benchmark circuits show that this approach reduces the number of transitions and the resulting WTM during scan out. This is an added advantage of using encoded test sets for scan testing.

For example, Table 2 shows two partially-specified scan vectors $t_i = 01 - - - 1 - - - -01$ and $t_j = 01 - 1010 - - - -1$ with scan chain length $l = 12$, where $-$ denotes a don't-care bit. If the don't-cares are mapped to binary values to minimize the weighted transition metric, then $d - - - - d'$, $d \in \{0, 1\}$, must be mapped to $dddddd'$. Similarly, $d - - - -$ must be mapped to $ddddd$. This ensures that the few unavoidable transitions occur "late" during scan in. Table 2 shows the values of WTM_i and WTM_j and the Golomb codes for the corresponding fully-specified vectors ($m = 4$). The weighted transitions metric is clearly higher if the don't-cares are always mapped to 0. However, Golomb coding is much more effective in reducing test data volume if this strategy is used. We show in the next section that mapping don't-cares to 0s reduces test data volume considerably without any significant decrease in power savings.

First we derive the following theorem which characterizes the maximum WTM for a given test length n, scan chain length l, and the number of 1s r in the test set. This yields the maximum value for the average power P_{avg} and it can be used to predict average power by using limited information about T_D. We use this theorem in Section 4 to derive the upper bound on P_{avg} for ISCAS 89 benchmark circuits. The

maximum value for the peak power is obtained for a test pattern that has alternating 1s and 0s and thus has the maximum switching activity. Hence peak power P_{peak} is given by

$$\begin{aligned} P_{peak} &= (l-1) + (l-2) + (l-3) + \ldots + 1 \\ &= \frac{l(l-1)}{2}, \end{aligned}$$

as long as $r \geq l/2$.

Theorem 1 *For a given test length n, scan chain length l, and the number of 1s r in the test set, the average power is given by*

$$P_{avg} \leq \frac{lr}{n} - \frac{r^2}{n^2} + \frac{r}{2n^3}(\frac{r}{n} + 1).$$

Proof: Let $r = Qn + R$ such that $Q = \lfloor r/n \rfloor$. The WTM for the entire test set is maximum when the 1s are distributed in alternating fashion as discussed above. Hence, the maximum WTM is given by

$$\begin{aligned} maxWTM &= ln + (l-1)n + \cdots + (l-Q+1)n + (l-Q)R \\ &= (Ql - \frac{Q(Q-1)}{2})n + (l-Q)R \\ &= lr - \lfloor \frac{r^2}{n} \rfloor + \frac{1}{2n} \lfloor \frac{r}{2n} \rfloor (\lfloor \frac{r}{n} \rfloor + 1) \\ &\leq lr - \frac{r^2}{n} + \frac{r}{2n^2}(\frac{r}{n} + 1). \end{aligned}$$

The maximum average power is now obtained by dividing $maxWTM$ by n. □

4 Experimental results

In this section, we evaluate the effect of Golomb coding of T_D on test data volume and power consumption during scan testing for the ISCAS 89 benchmark circuits. The experiments were conducted on a Sun Ultra 10 workstation with a 333 MHz processor and 256 MB of memory. We only considered the large full-scan circuits with a single scan chain each. The test vectors for these circuits were reordered to increase compression. The amount of compression obtained was computed as follows:

$$Compression\ (percent) = \frac{(Total\ \ no.\ \ bits\ \ in\ \ T_D - Total\ \ no.\ \ bits\ \ in\ \ T_E)}{(Total\ \ no.\ \ bits\ \ in\ \ T_D)} \times 100$$

Table 3 presents the experimental results for test cubes T_D obtained from the Mintest ATPG program with dynamic compaction [25]. In order to compare with [20], we also present compressed results obtained

Circuit	No. of bits in T_D	Size of Mintest test set (bits)	Golomb coding using T_{diff}		Golomb coding using T_D		
			Comp-pression (percent)	No. of bits in T_E	Comp-pression (percent)	No. of bits in T_E	Improvement over Mintest (percent)
s5378	23754	20758	40.70 ($m = 4$)	14085	37.11 ($m = 4$)	14937	28.04
s9234	39273	25935	43.34 ($m = 4$)	22250	45.25 ($m = 4$)	21499	17.10
s13207	165200	163100	74.78 ($m = 16$)	41658	79.74 ($m = 16$)	33467	79.48
s15850	76986	57434	47.11 ($m = 4$)	40717	62.82 ($m = 8$)	28618	50.17
s38417	164736	113152	44.12 ($m = 4$)	92054	28.37 ($m = 4$)	117987	-4.27
s38584	199104	161040	47.71 ($m = 4$)	104111	57.17 ($m = 8$)	85275	47.05
Average	—	—	49.63	—	51.74	—	36.26

Table 3: Experimental results on test data compression using Golomb codes.

using the difference vector sequences T_{diff} for the same test sets. The table lists the sizes of the precomputed (uncompacted) test sets, the amount of compression achieved for the best value of the Golomb code m, and the size of the smallest encoded test set obtained after static compaction using Mintest.

As is evident from Table 3, T_D yields better compression than T_{diff} in four out of the six cases. For these circuits, we achieve better compression without requiring a separate CSR. (The best value of the code parameter m is shown in parenthesis.) Therefore, there is a significant reduction in hardware overhead as compared to the compression scheme presented in [19, 20]. The results also show that ATPG compaction may not always be necessary for saving memory and reducing testing time. In five out of the six cases, the size of the encoded test set is less than the smallest ATPG-compacted test sets known for these circuits. This comparison is essential in order to show that storing T_E in ATE memory is more efficient than simply applying static compaction to test cubes and storing the resulting compact test sets. On average, the size of T_E is 36.26% less than that of the compacted test sets obtained using Mintest.

We next present results on the peak and average power consumption during the scan-in operation. These results show that test data compression can also lead to significant savings in power consumption. As described in Section 3, we estimate power using the weighted transitions metric. Let P^C_{peak} (P^C_{avg}) be the peak (average) power with compacted test sets obtained using Mintest. Similarly, let P^G_{peak} (P^G_{avg}) be the peak (average) power when Golomb coding is used by mapping the don't-cares in T_D to 0s. Table 4 compares the average and peak power consumption for Mintest test sets with T_D when Golomb coding is used. The percentage reduction in power was computed as follows:

$$Reduction\ in\ peak\ power\ consumption = \frac{P^C_{peak} - P^G_{peak}}{P^C_{peak}} \times 100$$

Circuit	Mintest test sets after static compaction		Uncompacted test sets			
	Peak **power** P^C_{peak}	Average **power** P^C_{avg}	Peak **power** P^G_{peak}	**Peak power reduction (percent)**	Average **power** P^G_{avg}	**Average power reduction (percent)**
s5378	13423	11081	10127	24.55	3336	69.89
s9234	17494	14630	12994	25.72	5692	61.09
s13207	135607	122031	101127	25.42	12416	89.82
s15850	100228	90899	81832	18.35	20742	77.18
s35932	707280	583639	172834	75.56	73080	87.47
s38417	683765	601840	505295	26.10	172665	71.31
s38584	572618	535875	531321	7.21	136634	74.50
Average	—	—	—	28.98	—	75.89

Table 4: Experimental results on peak and average scan-in power consumption.

$$Reduction\ in\ average\ power\ consumption = \frac{P^C_{avg} - P^G_{avg}}{P^C_{avg}} \times 100.$$

Table 4 shows that the peak power and average power are significantly less if Golomb coding is used for test data compression and the decompressed patterns are applied during testing. On average, the peak (average) power is 28.98% (75.89%) less in this case than for the Mintest test sets.

We next present results on the peak and average power consumption during the scan-out operation. Table 5 shows that the peak power and average power are significantly less if Golomb coding is used for test data compression. On average, the peak (average) power is 23.54% (57.31%) less than for the Mintest test sets. Thus our results demonstrate that the substantial reduction in test data volume is also accompanied by significant reduction in power consumption during scan testing. The reduction in scan-out power is an important added advantage since we do not directly target scan-out power in our compression scheme.

Next, we justify the the strategy of mapping all don't-cares in T_D to 0s before Golomb coding. As discussed in Section 3, the power consumption can be minimized if the don't-cares are assigned to binary values to minimize the weighted transitions metric. Unfortunately, this strategy does not lead to any significant decrease in the test data volume—in fact, we found that in many cases, the encoded test set was larger than the original test set. We therefore carried out a set of experiments to demonstrate that if all don't-cares are mapped to 0s, the test data volume decreases substantially (Table 3) and at the same time, power savings are significant.

Our experimental results for the larger ISCAS 89 circuits are shown in Table 6. We note that while

Circuit	Mintest test sets after static compaction		Uncompacted test sets			
	Peak **power** P^C_{peak}	Average **power** P^C_{avg}	Peak **power** P^G_{peak}	**Peak power reduction (percent)**	Average **power** P^G_{avg}	**Average power reduction (percent)**
s9234	16777	14555	13855	17.41	9218	36.66
s13207	132129	116695	116644	11.72	39816	65.88
s15850	99647	88385	83914	15.78	32972	62.69
s35932	745282	390622	145163	80.52	53520	86.29
s38417	619929	553491	572593	7.63	294266	46.83
s38584	577365	521882	530042	8.19	284135	45.55
Average	—	—	—	23.54	—	57.31

Table 5: Experimental results on peak and average scan-out power consumption.

Circuit	Uncompacted test sets with don't-cares mapped to 0s				Uncompacted test sets with don't-cares mapped to minimize WTM			
	Peak power P^G_{peak}	**Peak power reduction (percent)**	**Average power** P^G_{avg}	**Average power reduction (percent)**	**Peak power** P^G_{peak}	**Peak power reduction (percent)**	**Average power** P^G_{avg}	**Average power reduction (percent)**
s5378	10127	24.55	3336	69.89	9531	28.99	2435	78.02
s9234	12994	25.72	5692	61.09	12060	31.06	3466	76.30
s13207	101127	25.42	12416	89.82	97606	28.02	7703	93.68
s15850	81832	18.35	20742	77.18	63478	36.66	13381	85.27
s35932	172834	75.56	73080	87.47	125490	82.25	46032	92.11
s38417	505295	26.10	172665	71.31	404617	40.82	112198	81.35
s38584	531321	7.21	136634	74.50	479530	16.25	88298	83.52
Average	—	28.98	—	75.89	—	37.72	—	84.32

Table 6: Impact of the mapping of don't-cares to binary values on power consumption.

the average power consumption is greater compared to the "optimal" mapping of don't-cares, it is still significantly less than the power for ATPG-compacted test sets. In some cases, the difference is as low as 4%, while on average, the average power consumption increases by only 8%. Likewise, the difference in peak power consumption is only 9% on average. Nevertheless, compared to Mintest, we achieve 51% test data compression on average with 76% reduction in average power consumption for scan testing. This provides a strong justification for the proposed test data compression approach.

We next present experimental results for a real test set from industry. We obtained a set of 32 scan vectors from IBM (a total of 362922 bits of test data per vector) for a design with 3.6 million gates and 726000 latches. The compression results for the 32 scan vectors is shown in Table 7. These vectors are statically-compacted tests and we mapped the remaining don't-cares to 0s to reduce scan power and increase the amount of compression. Note that we obtain a staggering 97.10% compression on average. We do not have any direct means to compare the WTM measure here with a base case—nevertheless we expect significant power savings due to the presence of long runs of 0s in the patterns fed to the scan chain after on-chip decompression.

Finally, we compare the upper bound on P_{avg} provided by Theorem 1 provided with the actual value of P_{avg} for the ISCAS 89 circuits. Table 8 shows that in most cases, Theorem 1 can be used a reasonable predictor of average power consumption for a precomputed test set.

5 Conclusions

We have addressed the problems of test data volume and power consumption for scan vectors for system-on-a-chip testing. Since static compaction of scan vectors invariably leads to higher power for scan testing, the conflicting goals of low-power scan testing and reduced test data volume appear to be irreconcilable. In this paper, we have employed test data compression to tackle these problem. This approach allows us to reduce test data volume and scan power simultaneously. In particular, we have shown that Golomb coding of precomputed test sets leads to significant savings in peak and average power, without requiring either a slower scan clock or blocking logic in the scan cells. We have also improved upon prior work on Golomb coding by showing that a separate cyclical scan register is not necessary for pattern decompression. Experimental results for the larger ISCAS 89 benchmarks and for an IBM production circuit show that reduced test data volume and low-power scan testing can indeed be achieved in all cases.

Scan Vector	Size of T_D (bits)	Golomb codes	
		Size of T_E ($m = 128$)	Percentage compression
1	362922	27708	92.36
2	362922	20406	94.37
3	362922	20406	94.37
4	362922	20244	94.42
5	362922	18267	94.96
6	362922	14081	96.12
7	362922	14418	96.02
8	362922	14370	96.04
9	362922	12875	96.45
10	362922	10414	97.13
11	362922	9729	97.31
12	362922	11199	96.91
13	362922	8618	97.62
14	362922	8675	97.60
15	362922	8123	97.76
16	362922	6508	98.20
17	362922	7629	97.89
18	362922	7974	97.80
19	362922	7668	97.88
20	362922	6791	98.12
21	362922	8149	97.75
22	362922	7120	98.03
23	362922	6959	98.08
24	362922	5856	98.38
25	362922	5911	98.37
26	362922	6254	98.27
27	362922	6903	98.09
28	362922	9212	97.46
29	362922	6460	98.22
30	362922	5580	98.46
31	362922	5994	98.34
32	362922	6118	98.31

Table 7: Compression obtained for test data from industry.

Circuit	l	n	r	P_{avg} (computed)	Upper bound on P_{avg}
s5378	214	111	3538	3336	5795
s9234	247	159	4817	5692	6564
s13207	700	236	5021	12416	14436
s15850	611	126	5336	20742	24056
s38417	1664	99	29473	172665	406749
s38584	1464	136	16814	136634	165710

Table 8: Comparison of upper bound on P_{avg} (predicted by Theorem 1) with the actual value of P_{avg}.

Acknowledgments

We thank Brion Keller and Carl Barnhart of IBM Corporation for providing scan vectors for a production circuit. We also thank Rafael Medina for helping us in our experiments with the IBM test data.

References

[1] Y. Zorian, E. J. Marinissen and S. Dey, "Testing embedded-core based system chips", *Proc. International Test Conference*, pp. 130–143, 1998.

[2] Y. Zorian, "A distributed BIST control scheme for complex VLSI devices", *Proc. IEEE VLSI Test Symposium*, pp. 4-9, 1993.

[3] R. M. Chou, K. K. Saluja and V. D. Agarwal, "Scheduling tests for VLSI systems under power constraints", *IEEE Transactions on VLSI Systems*, vol. 5, pp. 175–185, June 1997.

[4] S. Wang and S. K. Gupta, "LT-RTPG: A new test-per-scan BIST TPG for low heat dissipation", *Proc. International Test Conference*, pp. 85-94, 1999.

[5] S. Gerstendörfer and H.-J. Wunderlich, "Minimized power consumption for scan-based BIST", *Proc. International Test Conference*, pp.77-84, 1999.

[6] P. Girard, L. Guiller, C. Landrault and S. Pravossoudovitch, "A test vector inhibiting technique for low energy BIST design", *Proc. IEEE VLSI Test Symposium*, pp. 407-412, 1999.

[7] F. Corno, M. Rebaudengo and M. S. Reorda, "Low power BIST via non-linear hybrid cellular automata", *Proc. IEEE VLSI Test Symposium*, pp. 29-34, 2000.

[8] S. Wang and S. K. Gupta, "ATPG for heat dissipation minimization during scan testing", *Proc. Design Automation Conference*, pp. 614-619, 1997.

[9] V. Dabholkar, S. Chakravarty, I. Pomeranz and S. M. Reddy, "Techniques for minimizing power dissipation in scan and combinational circuits during test application", *IEEE Transactions on Computer-Aided Design*, vol. 17, No. 12, pp. 1325-1333, Dec. 1998.

[10] R. Sankaralingam, R. R. Oruganti and N. A. Touba, "Static compaction techniques to control scan vector power dissipation", *Proc. IEEE VLSI Test Symposium*, pp. 35-40, 2000.

[11] K. Chakrabarty, "Test scheduling for core-based systems using mixed-integer linear programming", *IEEE Transactions on Computer-Aided Design*, vol. 19, pp. 1163–1174, October 2000.

[12] M. Sugihara, H. Date and H. Yasuura. A novel test methodology for core-based system LSIs and a testing time minimization problem. *Proc. International Test Conference*, pp. 465–472, 1998.

[13] K. Chakrabarty, "Design of system-on-a-chip test access architectures using integer linear programming", *Proc. IEEE VLSI Test Symposium*, pp. 127-134, 2000.

[14] K. Chakrabarty, "Optimal test access architectures for system-on-a-chip", *ACM Transactions on Design Automation of Electronic Systems*, vol. 6, pp. 26-49, January 2001.

[15] K. Chakrabarty, "Design of system-on-a-chip test access architectures under place-and-route and power constraints", *Proc. IEEE/ACM Design Automation Conference*, pp. 432-437, 2000.

[16] S. Wang and S. K. Gupta, "ATPG for heat dissipation minimization during scan testing", *Proc. Design Automation Conference*, pp. 614-619, 1997.

[17] V. Iyengar, K. Chakrabarty and B. T. Murray, "Deterministic built-in pattern generation for sequential circuits", *Journal of Electronic Testing: Theory and Applications (JETTA)*, vol. 15, pp. 97–115, August/October, 1999.

[18] A. Jas, J. Ghosh-Dastidar and N. A. Touba, "Scan vector compression/decompression using statistical coding", *Proc. IEEE VLSI Test Symposium*, pp. 114-120, 1999.

[19] A. Jas and N. A. Touba, "Test vector decompression via cyclical scan chains and its application to testing core-based design", *Proc. International Test Conference*, pp. 458-464, 1998.

[20] A. Chandra and K. Chakrabarty, "Test data compression for system-on-a-chip using Golomb codes", *Proc. IEEE VLSI Test Symposium*, pp. 113-120, 2000.

[21] A. Chandra and K. Chakrabarty, "System-on-a-chip test data compression and decompression architectures based on Golomb codes", *IEEE Transactions on Computer-Aided Design*, vol. 20, no. 3, pp. 355-368, March 2001.

[22] A. Chandra and K. Chakrabarty, "Test resource partitioning for SOCs", *IEEE Design & Test of Computers*, vol. 18, pp. 80-91, September-October 2001.

[23] A. Chandra and K. Chakrabarty, "Efficient test data compression and decompression for system-on-a-chip using internal scan chains and Golomb coding", *Proc. Design, Automation and Test in Europe Conference*, pp. 145-149, 2001.

[24] A. Chandra and K. Chakrabarty, "Frequency-directed run-length (FDR) codes with application to system-on-a-chip test data compression", *Proc. IEEE VLSI Test Symposium*, pp. 42–47, 2001.

[25] I. Hamzaoglu and J. H. Patel, "Test set compaction algorithms for combinational circuits", *Proc. International Test Conference on CAD*, pp. 283-289, 1998.