

# A Unified Approach for SOC Testing Using Test Data Compression and TAM Optimization<sup>1</sup>

Vikram Iyengar<sup>\*</sup>, Anshuman Chandra<sup>†</sup>, Sharon Schweizer<sup>†</sup>, and Krishnendu Chakrabarty<sup>†</sup>

<sup>\*</sup>IBM Microelectronics  
Essex Jct, VT 05452, USA  
vikrami@us.ibm.com

<sup>†</sup>Duke University, ECE Department  
Durham, NC 27708, USA  
{achandra,sms8,krish}@ee.duke.edu

## Abstract

We integrate for the first time test access mechanism (TAM) optimization and test data compression into a single test methodology. We show how an integrated test architecture based on TAMs and test data decoders can be designed. The proposed approach offers considerable savings in test data volume and testing time. Two case studies using the integrated test architecture are presented.

## 1 Introduction

Two critical challenges that test planning for system-on-chip (SOC) designs must address are (a) handling the increase in test suite sizes (“can we fit a new test suite on an existing ATE?”), and (b) transporting test data to cores embedded deep in the system (“can we get test data to where we want it on-chip, and can we do it on time?”). Both these challenges contribute to rising test cost. ATE cost is a significant part of the overall cost of manufacturing test. ATE cost is governed by speed and timing accuracy, performance, pin counts, memory depth and special functionality for mixed signal and RF cores.

TAM design and test data compression offer promising solutions to the problem of ballooning test data volume, rising ATE requirements and the challenge of transporting test data to cores [2, 4]. Here, we investigate the use of test data compression and TAM design (hitherto studied as independent problems) as an integrated approach to modular SOC test. We first use rectangle packing [3] to design the TAM architecture for the SOC. An appropriate total TAM width for the SOC is identified. This represents the “base case” test architecture with no test data compression. Next, two case studies are presented, in which test data compression using FDR codes [1] is combined with TAM design. In both case studies, the compressed test suite is stored off-chip on the ATE. However, while in the first case study only one on-chip decoder is implemented on each TAM wire, in the second case study there is one decoder per core. Testing time and test pin count results obtained for the two case studies demonstrate considerable savings in test resource requirements over the base case. Furthermore comparisons between the results of the two case studies reveal the advantages and trade-offs in testing time, test data volume, pin counts and decoder hardware overhead using the two different implementations of test data decompression.

We have put together an example SOC d2758 to validate our results. We use six different ISCAS-89 benchmark circuits to create the sixteen core instances of this hypothetical but non-trivial system. Core reuse, closely following the philosophy of design reuse in industrial SOC design, is therefore a fundamental part of our example SOC.

<sup>1</sup>This research was supported in part by the National Science Foundation under grants CCR-9875324 and CCR-0204077, and by an IBM Graduate Fellowship.

## 2 TAM design and test data compression

To design an effective SOC TAM architecture, we use rectangle packing [3] to determine the precise partition of the total TAM width among cores to minimize the testing time of the overall schedule. We applied the rectangle packing algorithm to SOC d2758 for values of  $W$  varying between 1 and 64 bits. In experiments, we observed that as  $W$  is increased from 1 bit, the testing time for d2758 decreases sharply until  $W$  equals 4 bits. The testing time levels off as  $W$  is increased beyond 4 bits. We therefore choose 4 bits as an effective TAM width for this benchmark. This represents the “base case” test architecture without test data compression.

We assume test reuse for different instances of the same core during system test suite creation. This means that only one copy of each unique core test set is stored on the ATE, to be applied to multiple instances of the same core. To facilitate test reuse, all instances of the same core must therefore be assigned to the same TAM wire(s) and tester channel(s).

A new class of compression codes called frequency-directed run-length (FDR) codes were crafted in [1] for test data compression. We use the FDR code to compress the test suite for d2758 in this work. The FDR code maps variable-length runs of 0s to variable-length codewords. An on-chip decoder decompresses the encoded test set  $T_E$  and produces the precomputed test set  $T_D$ . Even though  $T_D$  contains more patterns than the 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. The ATE and the scan chain operate at two different frequencies such that  $f_{ATE} = f_{scan}/\alpha$ ,  $\alpha \geq 1$ . The parameter  $\alpha$  should ideally be a power of 2 since it is easier to synchronize the ATE clock with the scan clock for such values of  $\alpha$ .

## 3 SOC case studies

We now introduce the two case studies based on SOC d2758 with on-chip FDR decoders for pattern application. The decoders can be inserted in d2758 in the following ways.

1. *Decoder per TAM*: In this scheme, one decoder is placed on each TAM line. We refer to this system as  $D_{TAM}$ ; see Figure 1.
2. *Decoder per core*: In this scheme, every core has a dedicated on-chip decoder; see Figure 2. We refer to this system as  $D_{core}$ .

To determine the test application time for the case studies, we developed an ATE testbench in VHDL to emulate ATE functionality. We also developed VHDL models for systems  $D_{TAM}$  and  $D_{core}$ . For all the experiments, we used a frequency ratio  $\alpha = 4$  between the scan frequency and the ATE frequency, i.e.,  $f_{scan} = 4f_{ATE}$ . The base case test application time for d2758 (i.e., without using FDR encoding of test sets) was found to be 383748 cycles.

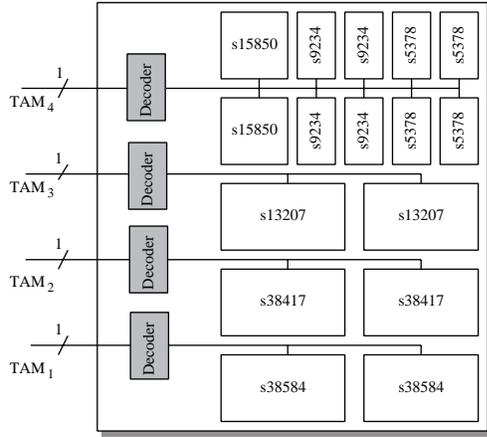


Figure 1. TAM architecture for SOC  $D_{TAM}$ .

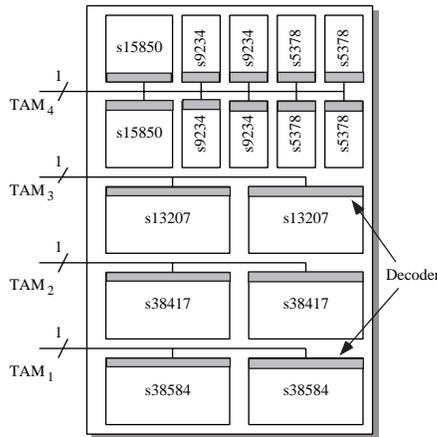


Figure 2. TAM architecture for SOC  $D_{core}$ .

Experimental results on test data volume compression for each core and for the complete SOC are presented in Table 1. The overall reduction in ATE memory requirement using FDR coding is 66.25%. This represents a significant saving in ATE memory.

Figure 3 shows the test schedule for the *Decoder per TAM* scheme. We note that the total test application time for system  $D_{TAM}$  is equal to 271,800 clock cycles, i.e., 111,948 clock cycles less than the base case test application time. Therefore, the total test application time is reduced by 29%. We also note that 271,800 clock cycles corresponds to the testing time of the cores assigned to TAM<sub>4</sub>. The test application time can be further reduced by careful reassignment of the cores to different TAM lines. For example, a further reduction of 19,360 clock cycles can be achieved by assigning all the C1 cores to TAM<sub>3</sub>. This

Core	Test set size			Percentage compression
	$T_D$ uncompactd	Padded with 0s	FDR encoded	
s5378	23,754	25,294	12,572	50.29
s9234	39,273	39,747	22,262	43.99
s13207	165,200	186,350	31,520	83.08
s15850	76,986	86,111	26,348	69.40
s38417	164,736	172,380	79,980	53.60
s38584	199,104	235,014	78,700	66.51
<b>SOC d2758</b>	669053	744896	251382	66.25

Table 1. Results on test data compression.

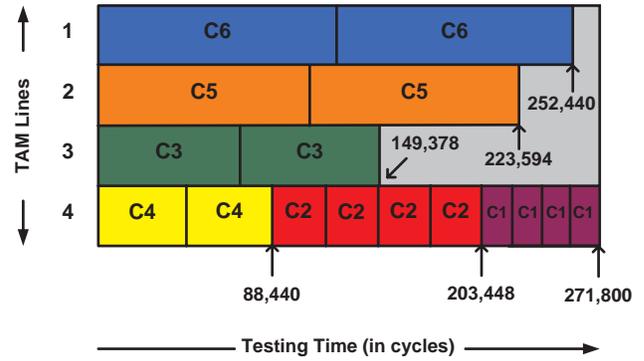


Figure 3. Test schedule for SOC  $D_{TAM}$ .

Case study	Test application time (cycles)	# of ATE channels	# of test control pins	Hardware overhead
Base case	383748	4	-	-
$D_{TAM}$	271800	4	8	12 F/F, 312 gates
$D_{core}$	192088	4	32	48 F/F, 1248 gates

Table 2. Comparison of  $D_{TAM}$  and  $D_{core}$ .

helps in reducing the idle time of TAM<sub>3</sub> and reduces the total testing time of  $D_{TAM}$  to 252,440 clock cycles.

The test application time for  $D_{core}$  is equal to 192,088 clock cycles. This scheme provides 38% reduction in test application time over the base case test application time.

Table 2 compares the testing time, number of pins and the hardware overhead for the case studies. To estimate the hardware overhead due to the on-chip decoding logic, we assume that the two FDR decoder counters [1] are configured using the existing counters in the core. While the testing time for  $D_{core}$  is the least, it requires a significant amount of decoding logic and a large number of pins to test the system. However, system  $D_{core}$  provides an effective means of testing multiple cores in parallel and therefore can be used for testing multiple sites simultaneously. Moreover, slow-speed serial channels can be used for control signals while high-speed interface pins can be used for data transfer. System  $D_{TAM}$  lies between  $D_{core}$  and the base case. As compared to  $D_{core}$ ,  $D_{TAM}$  provides a more practical solution for reducing test application time while keeping the pin count and hardware overhead low.

## 4 Conclusions

We have shown that combined test data compression and TAM design enhances test effectiveness and offers considerable savings in testing time and test cost. We have presented a trade-off between the testing time, number of test pins and the hardware overhead for two different decoder based designs. Furthermore, the method facilitates the use of less expensive ATEs.

## References

- [1] A. Chandra and K. Chakrabarty. Test resource partitioning for SOCs. *IEEE Design & Test of Computers*, vol. 18, pp. 80–91, Sept–Oct 2001.
- [2] Y. Huang et al. On concurrent test of core-based SOC design. *JETTA*, vol. 18, pp. 401–414, Aug–Oct 2002.
- [3] V. Iyengar, K. Chakrabarty, and E.J. Marinissen. On using rectangle packing for SOC wrapper/TAM co-optimization. *Proc. VLSI Test Symp.*, pp. 253–258, 2002.
- [4] A. Khoche, E. Volkerink, J. Rivoir and S. Mitra. Test vector compression using EDA-ATE synergies. *Proc. VLSI Test Symp.*, pp. 97–102, 2002.