Code Multiplexed VLSI Test Architecture for SOC Testing

Mubarak Ali Meerasha[1*], S.Radhakrishnan[2], T.Nirmalraj[2], S.Saravanan[2]
[1]Engineering and Research Services Division, HCLTech, Chennai, Tamil Nadu, India
[2]Dept. of ECE, SRC Campus, SASTRA University, Tamil Nadu, India

Abstract—This work presents a code multiplexed test architecture for system-on-a-chip (SOC) testing utilizing simultaneous test data from test generators (TGs) transaction on common bus to the embedded core in the SOC. To improve the SOC testing performance without increasing the testing channel resources and complexity, this work presents an efficient test architecture that exploits parallelism in core-level testing, resulting in shorter testing time and higher concurrency on a shared test bus. The proposed code division multiple access (CDMA) enables multiple concurrent transactions on a shared bus. The CDMA utilizes n-bit orthogonal code for n-embedded cores, which exploits parallel testing with reduced number of test buses and complexity. The multiple access mechanism of the CDMA improves real-time communication between multiple embedded cores or semiconductor intellectual property (SIP) blocks on a shared bus. This technique is experimentally verified with Xilinx's Virtex-5 XC5VLX50FF676 and Xilinx ISE 12.1 Software environment.

Index Terms—SOC testing, CDMA, Concurrent, orthogonality, direct sequence spread spectrum (DSSS)

I. Introduction

The manufacturing defects in very large scale integrated (VLSI) circuits lead to improper circuit operation, and which must be screened by test patterns. Due to the ever growing complexity of SOC, the testing of soc becomes tedious. To support different embedded cores in SOC, capability of different test pattern generator will be incorporated within the testing mechanism to test the different embedded core. Different test pattern generators are configured during run time using dynamic reconfiguration concept. Due to the presence of different embedded cores, the testing platform needs the different test generator. In order to test the different embedded cores simultaneously the system requires the dedicated test mechanism or channel. This leads to more communication resources. To reduce the communication resources and achieve the concurrent test data transactions to the embedded core the proposed system adopts the CDMA concept. The objective of our work is to propose a new method for SOC testing with less communication resources and dynamic reconfiguration of test generators. To achieve higher computational performance at the core level, instead of just increasing processing clock speed, parallelism in computation is widely used for chip. These systems achieve computational concurrencies by using scheduling techniques such as multitasking or multithreading [13]. However, the on- and off-chip communication architectures are still mostly based on conventional time-division multiplexing (TDM)-based communication protocols [typically known as time-division multiple access (TDMA) or time-interleaving] that do not allow real communication parallelism. So, the key motivation of this paper is to explore how to exploit parallelism and concurrency in communication without increasing resources and complexity. In order to achieve this goal this paper presents a new shared bus architecture called CDMA technique. This paper is organized as follows. Section II presents the overview of different communication bus architectures. Section III discusses the proposed communication bus architecture and discusses the related issues. Section IV describes the experimental results in terms of some selected criteria. Section V discusses the conclusions.

II. Overview of different communication bus architectures

Fig. 1, shows the principle of testing multiple cores in an SOC. in Fig. 1(a) all the cores are given independent tester channels, while in Fig. 1(b) the number of tester channels required is much fewer as one set of channels is allocated and used by the cores in sequence, i.e. The cores are tested in series, whereas in Fig. 1(c) a small number of tester channels are allocated and the cores are tested simultaneously. This is possible if a single core needs the channels at a time. However this will require some control information to be passed from the test generator to the embedded cores that indicates when each core will "sample" the seed placed on the tester channels. From the Fig. 1 it is clear that there will be trade-off between tester channels and test time for each embedded core. In contrast with Fig.1 (a) and Fig.1 (b), the Fig. 1(c) shows the some significant advantages over the previous two architectures, that is short test time and few tester channels, this is applicable when testing a single core at a time. Therefore for testing the multiple cores in the SOC at the same time the Fig. 1(c) architecture is to be modified as Fig. 1(d). Here in this architecture all the test data are applied concurrently to each core, by the way all the test responses analyzed simultaneously using the main controller. The detailed description of modified bus architecture is explained in the next section.

[*]Corresponding author: mubarakali.meerasha@hcl.com

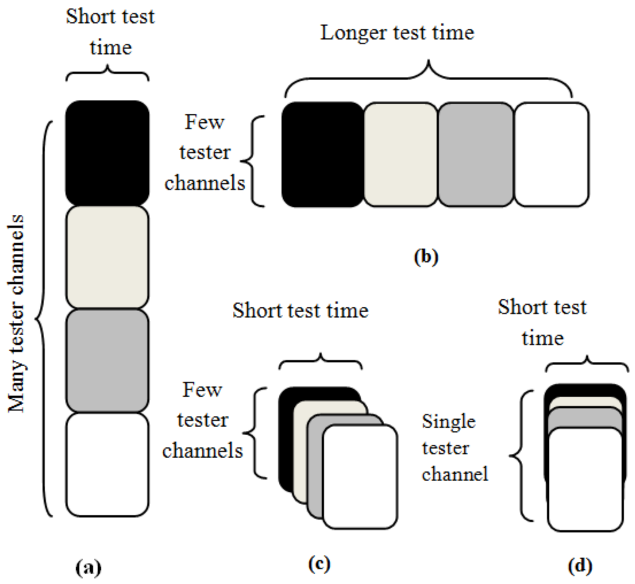
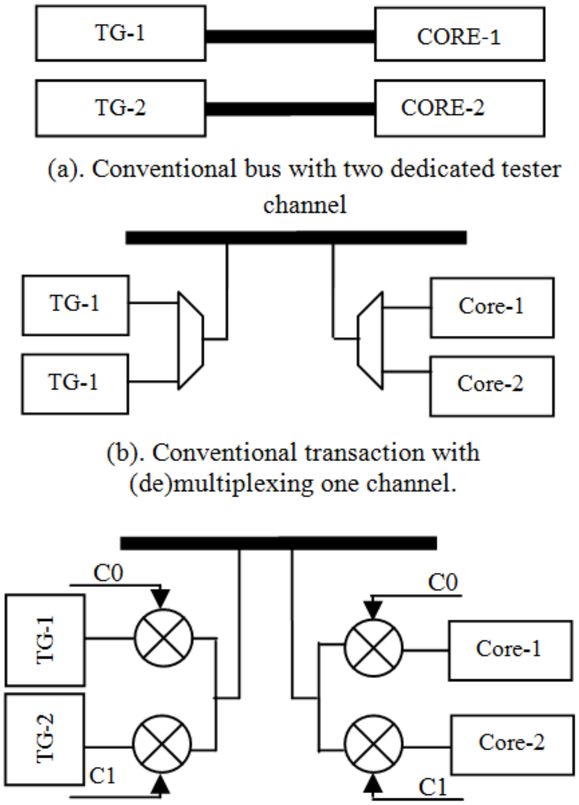


Fig. 1. The proposed system: Flight section.



(a). Conventional bus with two dedicated tester channel

(b). Conventional transaction with (de)multiplexing one channel.

(c). Proposed CDMA bus with one tester channel.

Fig. 2. Bus architectures for simultaneous test data transferring.

III. Proposed CDMA based modified bus architecture

Fig.2 (a) shows the two tester channels are required for two cores (core 1 and 2) to transmit two separate test data simultaneously in a conventional TDM architecture. It is also possible to transfer those two test data through a single channel by using split transaction or time multiplexing as shown in Fig.2 (b). However, this split transaction bus technique introduces bus contention latency and, therefore, increases the tester channel request latency. Fig.2 (c) shows the proposed CDMA bus allows the two cores to access the shared bus simultaneously: two test data are transmitted on a single bus trace concurrently. This means that a single CDMA bus consists of dual virtual TDM channels capable of operating simultaneously but isolated by each other. The principle of the CDMA technique is illustrated in Fig. 3. At the TG end, the data from different TGs are encoded using a set of orthogonal spreading codes. The encoded data from different TGs are added together for transmission without interfering with each other because of the orthogonal property of spreading codes. The orthogonal property means that the normalized autocorrelation value and the cross-correlation value of spreading codes are 1 and 0, respectively.

Autocorrelation of spreading codes refers to the sum of the products of a spreading code with itself, while cross-correlation refers to the sum of the products of two different spreading codes. Because of the orthogonal property, at the receiving end, the data can be decoded from the received sum signals by multiplying the received signals with the spreading code used or encoding.

A. Digital encoding and decoding scheme

Several on-chip bus schemes that apply the CDMA technique have been presented in [4]–[7]. Those schemes are implemented by analog circuits, namely, the encoded data are represented by the continuous voltage or capacitance value of the circuits. Therefore, the data transfers in the analog bus are challenged by the coupling noise, clock skew, and the variations of capacitance and resistance caused by circuit implementation [8]. In order to avoid the challenges faced by the analog circuit implementation, digital encoding and decoding schemes developed for the SOC are illustrated in Figs. 4 and 5, respectively. In the encoding scheme illustrated in Fig. 4, test data from different TGs fed into the encoder bit by bit. Each data bit will be spread into S bits by XOR logic operations with a unique S-bit spreading code as illustrated in Fig. 4. Each bit of the S-bit encoded data generated by XOR operations is called a data chip. Then, the data chips which come from different TGs are added together arithmetically according to their bit positions in the S-bit sequences. Namely, all the first data chips from different TGs are added together and all the second data chips from different TGs are added together, and so on. Therefore, after the add operations, we will get S sum values of S-bit encoded data. Finally, as proposed in [9], binary equivalents of the S sum values are transferred to the receiving end. An example of encoding two data bits from two TGs is illustrated in Fig. 5 in order to illustrate the proposed encoding scheme in more detail. Fig. 5(a) and Fig. 5(b) illustrate

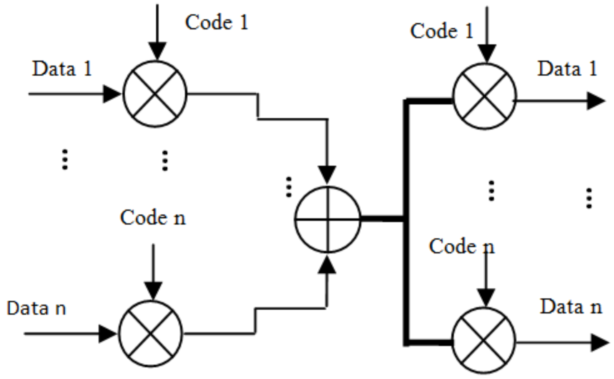


Fig. 3. CDMA technique.

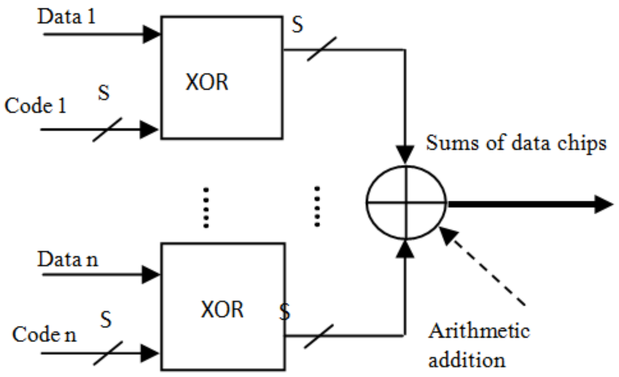


Fig. 4. Digital CDMA encoding scheme

two original test data bits from different TGs and two 8-bit spreading codes, results after data encoding (XOR operations) for the original data bits. The bottom figure presents the eight sum values after addition operations. Then the binary equivalents of each sum value will be transferred to the receiving end. In this case, two binary bits are enough to represent the three possible different decimal sum values, “0,” “1,” and “2.” For example, if a decimal sum value “2” needs to be transferred, we need to transfer two binary digits “10.” The digital decoding scheme applied in the CDMA is depicted in Fig. 6. The decoding scheme accumulates the received sum values into two separate parts, a positive part and a negative part, according to the bit value of the spreading code used for decoding. For instance, as illustrated in Fig. 6, the received first sum value will be put into the positive accumulator if the first bit of the spreading code for decoding is “0,” otherwise, it will be put into the negative accumulator. The same selection and accumulation operations are also performed on the other received sum values. The principle of this decoding scheme can be explained as follows. If the original data bit to be transferred is “1,” after the XOR operations in the encoding scheme illustrated in Fig. 4, it can only contribute nonzero value to the sums of data chips when a bit of spreading code is “0.” Similarly, the 0-value original data bit can only contribute nonzero value to the sums of data chips when a bit of spreading code is “1.” Therefore, after accumulating the sum values according to the bit values of the spreading code, either the positive part or negative part is larger than the other if the spreading codes are orthogonal and balance. Hence, the original data bit can be decoded by comparing the values between the two accumulators. Namely, if the value of the positive accumulator is larger than the value in the negative accumulator, the original data bit is “1”; otherwise, the original data bit is “0.”

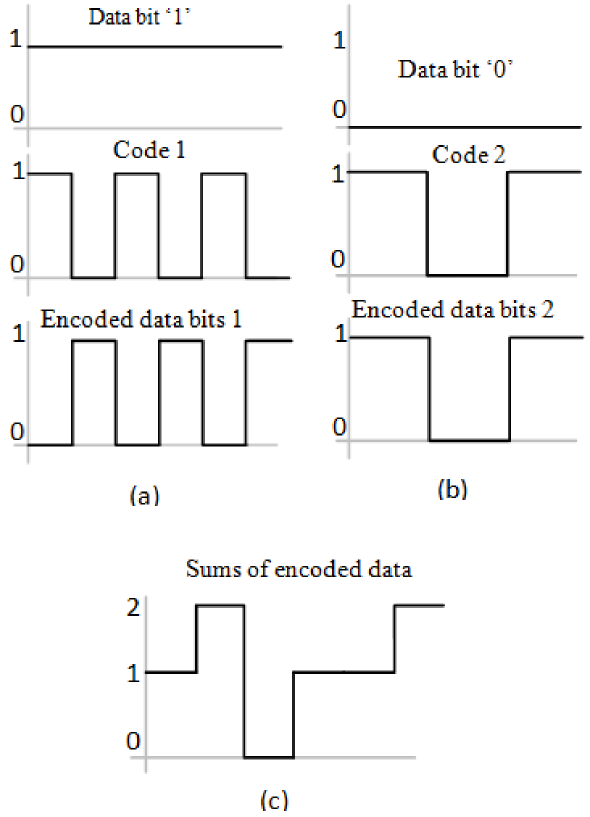


Fig. 5. Data encoding example

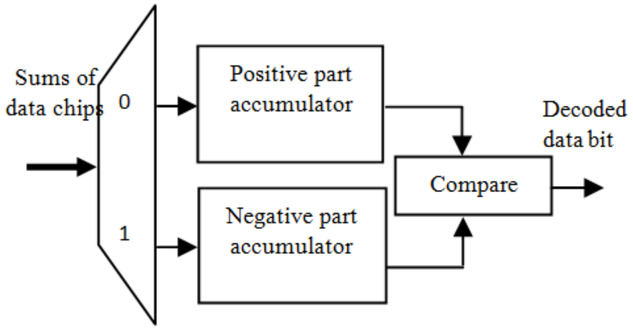


Fig. 6. Digital CDMA decoding scheme

B. Spreading code selection

The proposed decoding scheme requires the spreading codes used in the CDMA to have both the orthogonal and balance properties. The orthogonal property has been explained in the previous paragraphs. The balance property means that the number of bit “1” and bit “0” in a spreading code should be equal. Several types of spreading codes have been proposed for CDMA communication, such as Walsh code, M-sequence, Gold sequence, and Kasami sequence, etc. [8]. However, only Walsh code [8] has the required orthogonal and balance properties. Therefore, Walsh code family is chosen as the spreading code library for the CDMA. In an S-bit (S=2N, integer N>1) length Walsh code set, there are S-1sequences that have both the orthogonal and balance properties. Hence, the proposed CDMA based bus can have at most S-1 core connection. The length of applied Walsh code set should be kept as small as possible according to the number network nodes. The purpose is to reduce the number of data chips generated during data encoding operations as illustrated in Fig. 4.

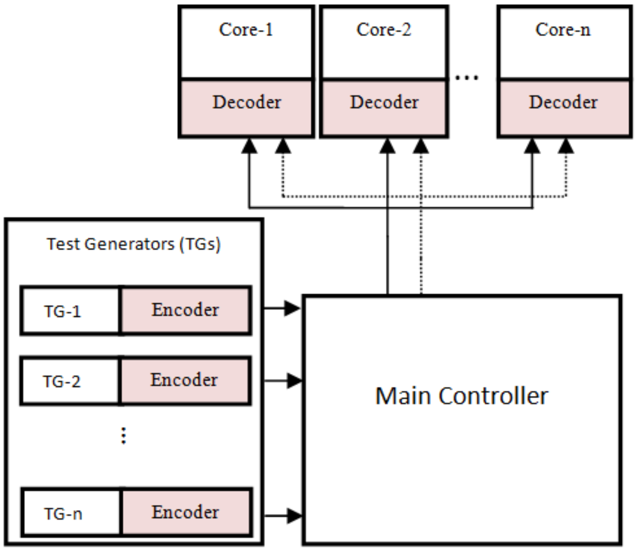


Fig. 7. Proposed code multiplexed SOC testing.

Fig. 7 shows the overview of the code multiplexed SOC testing. This comprises the following main parts. It consists of test generators, encoders, main controller, decoders and embedded cores. Each test data is encoded using CDMA principle; here each test data is mixed up with dedicated orthogonal codes. The main controller is responsible for many purposes. In encoder side the main controller adds all the encoded test data to a single test data. Here all the test data are added together without mixing up each other, this is possible due to the orthogonal property of Walsh code. The main controller sends the bulk test data to all the embedded cores. Each embedded core is incorporated with specific Walsh code sequence. Based on the specific sequence the embedded core receives the corresponding test data by decoding process. The main controller also performs the comparison test and analyzes the test results.

IV. Experimental results

For experimental set up hypothetical SOC (comprised of the 5 ISCAS’89 benchmark circuits [14] is considered. Here the following benchmark circuit s344, s349, s820, s832 and s1494 are considered as embedded core. For the comparison with the existing methods following criteria are considered as shown in Table I. Due to our proposed technique the testing platform can achieve significant performance improvement in terms of latency. The latency calculation in our proposed system is based on each embedded core, but calculated simultaneously.

Table 1: Criteria satisfied for compared approaches

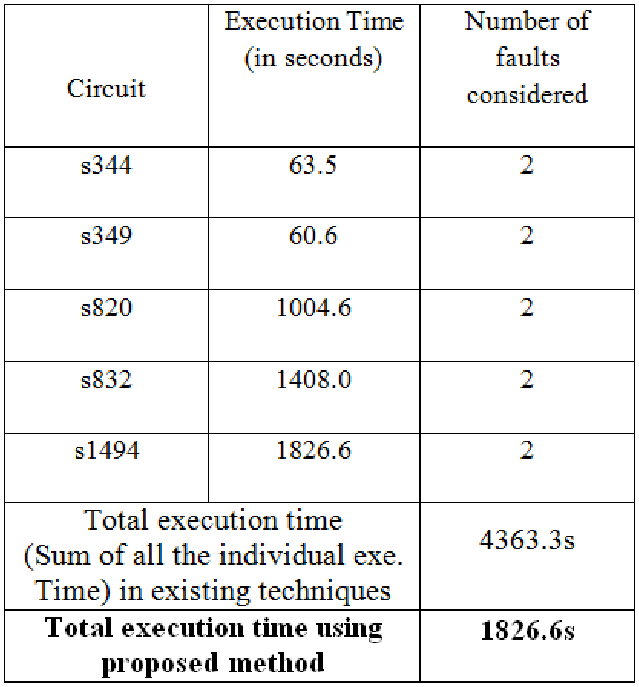
	[9]	[10]	[11]	[12]	[1]	Proposed
IP-Consistent	√	X	√	X	√	√
Modular	X	X	X	X	√	√
Scalable	X	X	√	X	√	√
Programmable	√	√	√	X	√	√
Low cost ATE	√	√	√	√	√	√
Simultaneous Multiple core testing	X	X	X	X	X	√

Table II shows the execution time of ISCAS’89 circuits for 2 faults. Total execution time for all the circuits to execute 2 faults is sum of all the individual execution times. i.e, 4363.3s But the proposed technique takes only maximum execution time i.e. 1826.6s, within this maximum execution time all the circuits to be executed their faults. This can be achieved due to the parallel execution of all the circuits using common tester channel. By this way the proposed technique can significantly reduces the execution time of each core. The total execution time of the particular SOC depends on the maximum execution time of particular core. In contrast to existing methods it accumulates the execution time.

V. Conclusion

To alleviate heavy traffic load in traditional TDM-based bus architectures, this paper presented latency-aware bus

Table 2: Execution time of ISCAS'89 circuits



Circuit	Execution Time (in seconds)	Number of faults considered
s344	63.5	2
s349	60.6	2
s820	1004.6	2
s832	1408.0	2
s1494	1826.6	2
Total execution time (Sum of all the individual exe. Time) in existing techniques		4363.3s
Total execution time using proposed method		**1826.6s**

architecture to exploit parallelism in communication. By utilizing n-bit orthogonal coding techniques, the proposed CDMA bus enables n-concurrent transactions on a shared bus without bus contention. The proposed bus architecture is compared with the conventional existing bus architectures.

References

[1] Adam B. Kinsman, and Nicola Nicolici, "Time-Multiplexed Compressed Test of SOC Designs" in IEEE Trans.on VLSI Systems, VOL. 18, NO. 8, AUGUST 2010

[2] Jongsun Kim et al,"Design of an Interconnect Architecture and Signaling Technology for Parallelism in Communication" in IEEE Trans.on VLSI Systems, VOL. 15, NO. 8, AUGUST 2007

[3] Xin Wang, Tapani Ahonen, and Jari Nurmi, "Applying CDMA Technique to Network-on-Chip",in IEEE Trans.on VLSI Systems, VOL. 15, NO. 10, OCTOBER 2007

[4] R. Yoshimura, T. B. Keat, T. Ogawa, S. Hatanaka, T. Matsuoka, and K. Taniguchi, "DS-CDMA wired bus with simple interconnection topology for parallel processing system LSIs," in Dig. Tech. apers IEEE Int. Solid-State Circuits Conf., 2000, pp. 370–371.

[5] T. B. Keat, R. Yoshimura, T. Matsuoka, and K. Taniguchi, "A novel dynamically programmable arithmetic array using code division multiple access bus," in Proc. 8th IEEE Int. Conf. Electron., Circuits Syst., 2001, pp. 913–916.

[6] S. Shimizu, T. Matsuoka, and K. Taniguchi, "Parallel bus systems using code-division multiple access technique," in Proc. Int. Symp. Circuits Syst., 2003, pp. 240–243.

[7] M. Takahashi, T. B. Keat, H. Iwamura, T. Matsuoka, and K. Taniguchi, "A study of robustness and coupling-noise immunity on simultaneous data transfer CDMA bus interface," in Proc. IEEE Int. Symp. Circuits Syst., 2002

[8] E. H. Dinan and B. Jabbari, "Spreading codes for direct sequence CDMA and wideband CDMA cellular networks," IEEE Commun. Mag., vol. 36, no. 9, pp. 48–54, Sep. 1998.

[9] S. Reda and A. Orailoglu, "Reducing test application time through test data mutation encoding," Proc. IEEE/ACM DATE, 2002.

[10] K. Miyase, S. Kajihara, and S. M. Reddy, "Multiple scan tree design with test vector modification," in Proc. IEEE ATS, 2004.

[11] L. Li and K. Chakrabarty, "Test data compression using dictionaries with fixed-length indices," in Proc. IEEE VLSI Test Symp. (VTS), 2003, pp. 219–224.

[12] W. Rao, I. Bayraktaroglu, and A. Orailoglu, "Test application time and volume compression through seed overlapping," in Proc. IEEE/ACM DAC, 2003, pp. 732–737.

[13] W. Stallings, Computer Organization and Architecture; Designing for Performance. Englewood Cliffs, NJ: Prentice-Hall, 2003.

[14] F. Brglez, D. Bryan, and K. Kozminski, "Combinational profiles of sequential benchmark circuits," in Proc. IEEE ISCAS, 1989, pp. 1929–1934.