Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 8 (11): 2083-2090, 2008
ISSN 1812-5654
© 2008 Asian Network for Scientific Information

Parallel Circular-Scan Architecture

Mehdi Azimipour and Mohammad Eshghi
Department of Electrical and Computer Engineering, Shahid Beheshti University, Tehran, Iran

Abstract: One approach to deal with today's complicated SOCs test procedure is using scan-based designs. This serial access mechanism increases the test application time and cost. The circular-scan architecture reduces test data volume, test application time and the number of scan input counts. The basic idea of circular-scan architecture is using the captured response of the previously applied test pattern as a template for the next pattern. In this architecture, only the conflicting bits of previously captured response are updated through a data input pin. This research presents a new circular-scan architecture that makes it possible to select several scan chains in parallel. In the proposed architecture, multiple conflict bits are selected and updated simultaneously In comparison with original architecture which it is possible to select only one scan chain at each time using a regular decoder, the parallel updating of conflict bits results in more reduction in test data volume and test application time. Scan chains are selected in parallel, using a multiple-hot decoder. Experimental results show an average improvement of 26% in test data volume and test application time, in the 5 largest ISCAS'89 benchmark circuits.

Key words: System-on-chip (SOC), design-for-test (DFT), circular-scan architecture, multiple-hot decoder

INTRODUCTION

One of the major challenges in testing SOC's is dealing with the large size of test data that is stored in the tester and transferred between the tester and chip (Zorian *et al.*, 1998). As complexity increases, the volume of test data also increases. The data volume in 2014 is expected to be as much as 150 times the data volume in 1999 (Khoche and Rivoir, 2002).

One approach to deal with today's complicated SOCs test procedure is using scan-based designs. Scan-based testing is a very widely used Design-For-Testability (DFT) technique that aims at enhancing controllability and observability of internal nodes in digital integrated circuits (McCluskey, 1986). This serial access mechanism increases test application time and cost. The problem of reducing test cost for core-based SOCs has been considered in many recent literatures (Touba, 2006; Al-Yamani *et al.*, 2005; Arslan and Orailuglu, 2004). An attractive approach for reducing test data volume for SOCs is based on use of data compression techniques. In this approach, the test set T_D is compressed (encoded) to a smaller test set T_E and stored in the ATE memory. An on-chip decoder is used for pattern decompression to generate T_D from T_E during test procedure. Several techniques are proposed to achieve test vector compression. The vector compression schemes (Touba, 2006) fall broadly into three classes:

- Code-based schemes use data compression codes to encode test cubes
- Linear-decompression-based schemes decompress the data using only linear operations (that is LFSRs and XOR networks)
- Broadcast-scan-based schemes rely on broadcasting the same values to multiple scan chains

From first class, Run-length coding, Huffman coding and Lampel-Ziv are some of the compression techniques (Touba, 2006). The second class of test vector compression schemes involves linear expansion which uses only linear operations to decompress the test vectors. Linear expansion circuit exploits the unspecified bit positions in test cubes to achieve a large amount of compression. One to five percent of test data is typically specified and the rest are considered as don't care (Hiraide *et al.*, 2003). This method is based on Linear Feedback Shift Register (LFSR) reseeding and combinational linear expansion circuits consisting of XOR gates (Balakrishnan and Touba, 2006). In traditional serial scan-based architecture, increasing the length of scan chain causes long and costly test process. One method to deal with these problems is scan-chain partitioning. Illinois scan architecture (ILS) (Hamzaoglu and Patel, 1999) is used to overcome parallel scan chains broadcasting architecture, using two modes of operation. The first mode is called the broadcast scan mode in which

Corresponding Author: Mehdi Azimipour, Department of Electrical and Computer Engineering, Shahid Beheshti University, P.O. Box 14455-463, Tehran, Iran Tel: +989112742543 Fax: +982122417940

same vector is shifted into multiple scan chains through one scan-in pin. The second mode is called the serial scan mode, in which all the scan chains are connected in series. Segmented addressable scan architecture proposed in Al-Yamani *et al.* (2005), which uses some of the basic concepts from Illinois scan architecture (Hamzaoglu and Patel, 1999). The authors in Arslan and Orailuglu (2004) propose a circular-scan chain architecture to reduce test time and cost. In this architecture, captured response of previously applied pattern is used as a template for the next pattern by updating only conflicting bits (Arslan and Orailuglu, 2004). In another study (Azimipour *et al.*, 2007), we proposed a new scan input selection unit which updates conflict bits internally and eliminates loading conflict bits from ATE, results in reduction in test data volume.

The proposed architecture in this paper uses the basic concepts of Circular-Scan Architecture (CSA) (Arslan and Orailuglu, 2004), but it has the following distinguishing factors:

- In proposed architecture, the modified scan selection unit (Azimipour *et al.*, 2007) is used which allows internal updating of conflict bits, which in original architecture in done using a data input pin.
- Instead of a regular decoder, Multiple-Hot Decoder (MHD) is used, which allows simultaneous updating of conflict bits, results in more reduction in test data volume and test application time.

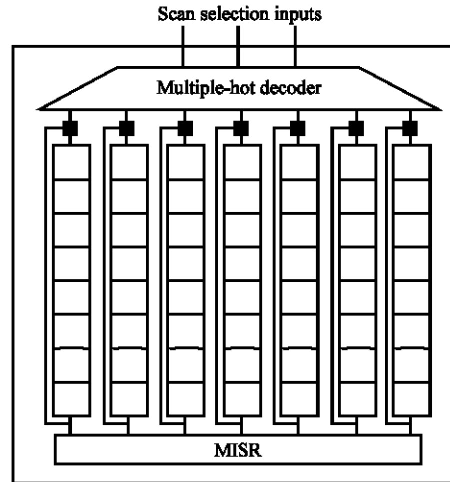

Fig. 1: Circular-scan Architecture (Arslan and Orailuglu, 2004) with MHD

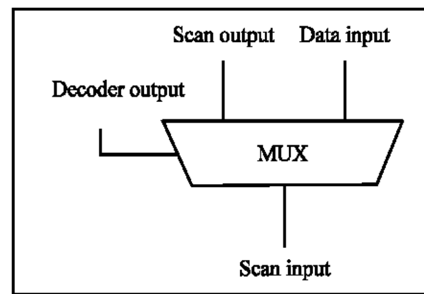

Fig. 2: Scan input selection unit (Arslan and Orailuglu, 2004)

MATERIALS AND METHODS

Circular-scan architecture: Circular-scan architecture makes it possible to use the captured response of the previously applied test pattern as a template to generate the next pattern (Arslan and Orailuglu, 2004). In this architecture, it is needed to update necessary conflicting bits between captured response and current test vector. This architecture uses this fact that in today's test vectors, the ratio of specified bits is much smaller than unspecified bits (Hiraide *et al.*, 2003). The architecture proposed in this study is similar to the one proposed in (Arslan and Orailuglu, 2004) and showed in Fig. 1, with this difference that regular decoder in original architecture is replaced by a multiple-hot decoder. As shown in Fig. 1, the output of each scan chain is connected to the input of the same scan chain. In original-circular scan architecture, each scan could be selected by regular decoder that is connected to ATE pins. By using one decoder with N inputs, 2^N scan chains are selected. This architecture reduces the required ATE pins from 2^N to N pins. As a result, the cost of the test is reduced considerably.

In this method, first test pattern is loaded to scan chains and its response is captured. If in the current test vector a set of bits differ from its corresponding bits in scan cells, then one of them is updated through the data input (Fig. 2). Since only one of the conflicting bits of the current test slice can be changed at each cycle, multiple full rotation of the scan chain content may be required. In the traditional scan architecture, if perfectly balanced scan chains and a scan chain per input pin is assumed (Arslan and Orailuglu, 2004), the number of required test data volume, v and test application time, t, can be calculated by Eq. 1 and 2:

$$v = T\times[N/I]\times I \quad (1)$$

$$t = T\times([N/I]+1)\times F^{-1} \quad (2)$$

where, T is number of test vectors, N number of scan cells and I is the number of input pins connected to ATE. If the specified bits density of the test set is p, the probability that single bit of a test vector have conflicting value to the captured response bit would be p/2. Let k_j is

the number of bits that conflicts with the captured test response in the jth test slice and K_{max} is the maximum of k_j's. The probability of having i conflicting bits in the jth test slice ($p(k_j) = i$) is given by the Eq. 3 (Arslan and Orailuglu, 2004):

$$p(k_j = i) = \binom{S}{i}\left(\frac{p}{2}\right)^i\left(1-\frac{p}{2}\right)^{s-i} \quad (3)$$

For S scan chains, the number of test slices in each test vector is N/S. So, the probability to have maximum i conflicting bits (i.e., $k_{max} = i$) is (Arslan and Orailuglu, 2004):

$$p(k_j \le m) = \left(\sum_{k=0}^{m}\binom{S}{K}\left(\frac{p}{2}\right)^k\left(1-\frac{p}{2}\right)^{S-k}\right) \quad (4)$$

In Arslan and Orailuglu (2004), it is supposed that specified bits are uniformly distributed. So, the average number of rotations is equal to the expected value of the k_{max} ($E(k_{max})$), as shown in Eq. 5 (Arslan and Orailuglu, 2004):

$$E(K_{max}) = \sum_{i=1}^{S} i \times p(K_{max} = i) = S - \sum_{i=0}^{S}\left(\left(\sum_{k=0}^{i}\binom{S}{k}\left(\frac{p}{2}\right)^k\left(1-\frac{p}{2}\right)^{S-k}\right)^{\left\lceil\frac{N}{S}\right\rceil}\right) \quad (5)$$

And the test data volume, v_s, is calculated by Eq. 6:

$$v_s = N + (T-1) \times E(K_{max}) \times \lceil N/S \rceil \times I \quad (6)$$

So, the reduction in test data volume and test application time is calculated by using Eq. 7 and 8, respectively.

$$t_r = 1 - \left(\frac{t_s}{t}\right) \quad (7)$$

$$v_r = 1 - \left(\frac{v_s}{v}\right) \quad (8)$$

Modified circular-scan architecture: A new circular-scan architecture that eliminates the loading data from ATE for conflicting bits is proposed in Azimipour *et al.* (2007). This architecture saves the total of conflicting bits and reaches the more compression rate. In the scan input selection unit which is shown in Fig. 3, circular mode had two different paths, the buffer path and the inverter path. In each time, only one scan chain comes from the inverting path and the rest from the buffer paths. The inverter path is selected for scan chain that have conflicting bit. Using these two paths, the conflict bits are updated internally.

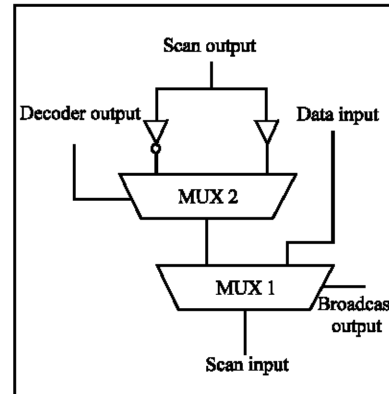

Fig. 3: Modified scan input selection unit (Azimipour *et al.*, 2007)

Equation 9(6) is comprised of three different terms: the First Term, [N], is the number of the first test vector bits, the second term, [(T-1)E(Kmax)×[N/S]×(I-1)], is the number of control bits and the third term, [(T-1)E(Kmax)×[N/S]], is the number of conflicting bits that are loaded from ATE in the original circular scan architecture. The approach taken in Azimipour *et al.* (2007) saves conflicting bits in the test vectors by internally updating of conflict bits.

$$\begin{aligned} v_s &= N + (T-1) \times E(K_{max}) \times \lceil N/S \rceil \times I \\ &= \underbrace{N}_{1} + \underbrace{(T-1)E(K_{max}) \times \lceil N/S \rceil \times (I-1)}_{2} \\ &\quad + \underbrace{(T-1) \times E(K_{max}) \times \lceil N/S \rceil}_{3} \end{aligned} \quad (9)$$

As a result, in the proposed architecture in Azimipour *et al.* (2007), the test application data volume, v_s, is reduced to Eq. 10:

$$v_s = \lceil N/S \rceil + (T-1) \times E(K_{max}) \times \lceil N/S \rceil \times (I-1) \quad (10)$$

In this architecture, the number of required bits for loading the first test vector is [N/S], because data input pin is connected to all scan chains and the behavior of data input pin in this mode is similar to the broadcasting architecture, in which one bit loaded to all scan chains.

Multiple-hot decoder: As mentioned in the previous section, the proposed architecture in this study relies on the circular-Scan architecture (Arslan and Orailuglu,

Table 1: Positional-cube notation (Al-Yamani *et al.*, 2005)

Symbol	Binary encoding
Ø	00
0	10
1	01
d	11

2004), the modified scan input selection unit (Azimipour *et al.*, 2007) and the multiple-hot decoders (MHD) (Al-Yamani *et al.*, 2005).

For the regular decoders, by each address in the decoder input, only one specified output could be selected. In the MHD, the scan chain address could be included don't care bits (d) that allows selecting multiple scan chains in parallel. The symbols used in the input part are {0, 1, d}. For example, suppose that there are 8 scan chains and scan chain #1 and #3 must be selected. By combining the addresses of the scan chain #1 {001} and scan chain #3{011}, the 0d1 address is obtained, which means by applying 01d in the MHD input, scan chain #1 and 3 could be selected in parallel.

In Al-Yamani *et al.* (2005), positional cube notation is used for encode 0s, 1s and don't care bits. The positional-cube notation encodes each symbol by 2-bit fields as shown in Table 1. By using the multiple-hot decoder, it is possible to have $3^{\lceil \log_2^S \rceil}$ different configurations in the decoder output, which S is the number of scan chains.

In this study, the multiple-hot decoder (MHD) similar to the one in Al-Yamani *et al.* (2005) is used to take the address of the scan chains and update the conflict bits in parallel.

PARALLEL CIRCULAR-SCAN ARCHITECTURE

In the previous proposed architectures (Arslan and Orailuglu, 2004; Azimipour *et al.*, 2007), for conflict bit updating, a regular decoder is used. In these architectures, at each time only one conflict bit is selected and updated. The basic idea of circular-Scan architecture is using the captured response of the previously applied test pattern as a template for the next pattern while allowing the full observation of the captured response (Arslan and Orailuglu, 2004). In this architecture, only the conflicting bits of previously captured response are updated through a data input pin shown in Fig. 2. In this study, parallel circular-scan architecture based on MHD is proposed. For the regular decoders, by each address in the decoder input, only one specified output could be selected. In the MHD, the scan chain address could be included don't care bits (d) that allows selecting multiple scan chains in parallel. In the proposed architecture, multiple conflict bits are selected and updated simultaneously. First we present the definitions.

Definition 1. Test matrix: Circular-scan architecture is viewed as a Matrix of Flip-Flops. In this architecture, a serial scan chain is divided into some multiple parallel chains. The test matrix is an (n×S) dimensional matrix, in which n is the number of test slices (rows) and S is the number of scan chains (columns). So, each test vector generated by Automatic Test Pattern Generator (ATPG) tools, is divided to the (n×S) dimensional test matrix (T[n×S]).

Definition 2. Captured-response matrix: Captured response matrix, R[i×j], is an (n×S) dimensional matrix which contains the response of the previously applied test matrix to the circuit.

Definition 3. Conflict-bit matrix: Conflict-bit matrix, C[n×S], is an (n×S) dimensional matrix which is defined on the R[n×S] and the T[n×S]. Conflict is defined on the difference between the T(i, j) and R(i, j), when the T(i, j) is not don't care. If there is conflict between the next test matrix and its corresponding captured response matrix, then the C(i, j) is 1; otherwise is 0. When the test matrix element is don't care, the C(i, j) is considered to be don't care (x). As a result, C(i, j) is defined as a Eq. 11.

$$C(i, j) = \begin{cases} T(i, j) \oplus R(i, j) & \text{if } T(i, j) \neq x \\ x & \text{if } T(i, j) = x \end{cases} \tag{11}$$

In proposed algorithm which is depicted in Fig. 4, we defined the [n×S] dimensional conflict-bit matrix. The aim of the proposed algorithm is to update all conflict bits. In the first step, construct the conflict-bit matrix using Eq. 11. Second, consider the ith slice of conflict-bit matrix. Then, all possible configurations of MHD ($3^{\lceil \log_2^S \rceil}$) is applied to this slice. Third, select the best configuration. The best configuration is the MHD output which updates the maximum number of conflict-bits. Then, apply the best configuration to the selected conflict-bit slice, in order to update the conflict-bits. This procedure continuous until all conflict bits in given slice are updated.

The above loop is repeated for all slices of conflict-bit matrix (for I = 1:n), until the conflict-bit matrix is completely updated. Figure 4 shows a pseudo code for the proposed algorithm. The aim of this algorithm is to find the minimum number of required configurations for updating each conflict slice. As a result of this parallel updating, the test data volume and the test application time decreases, significantly.

Example 1: Consider a captured response of previously applied test matrix as shown in Fig. 5a and the next test matrix as shown in Fig. 5b. As a result of these test matrix

```
1.        Construct the conflict-bit Matrix as defined in Eq. (11)
3.        for (i = 1:n)
4.             Select i^th conflict-bit slice
5.               While (there is any conflict bits in the given slice)
6.                    Apply all MHD configurations
7.                    Select the best configuration, based on highest updating of
                      conflict bits
8.                    Update given conflict-bit slice according to selected
                      configuration
9.               End
10.       End
```

Fig. 4: The proposed Heuristic algorithm for parallel updating of conflict bits

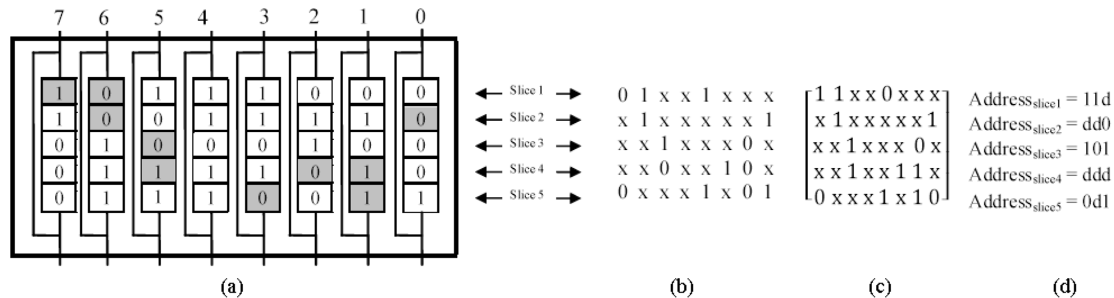

Fig. 5: (a) Captured test response, (b) Next pattern, (c) Conflict-bit Matrix and (d) MHD configurations

and response matrix, the conflict-bit matrix is calculated using Eq. 11 and is shown in Fig. 5c. In this example, the test matrix consists of 8 scan chains (columns) and 5 test slices (rows).

Since we have 8 scan chains, there are $3^{\lceil \log_2^8 \rceil}$ configurations in the MHD output.

As shown in Fig. 5c, in slice #1 of the conflict-bit matrix, bits #6 and #7 have conflicting values (1). By combining the address of scan chain #6 (110) and #7 (111), the 11d address is obtained. The output of MHD for this address is 11000000. Therefore, by using 11d as a MHD input, the scan chain #6 and 7 is selected in parallel. By using the scan selection unit architecture which we proposed in Azimipour *et al.* (2007), the scan chain #6 and #7 are inverted. In slice #2, the bits #0 and 6 have conflicting values. By using dd0 as a MHD address, the output of MHD is 01010101. By using dd0 address in the MHD input, not only the scan chain #0 and 6 with conflict-bits are selected, but also, the scan chain #4 and 2 without conflict bits are inverted. Since bits #2 and 4 in the slice #2 are don't care (x), inverting of these bits is also don't care. Notice that different MHD configurations could be used for updating of the each slice. For slice #2, instead of the dd0 address, the ddd address could be used (because none-conflict bits are 'x').

In slice #3, the bit #5 has conflicting value. Like the regular decoder, scan chain #5 is selected by 101 address. In the slice #4, the bits #1, 2 and 5 have conflicting values. For this slice, ddd address is selected. And for slice #5, the bits #1 and 3 have conflicting values. By using 0d1 as a MHD address, the output of MHD is 00001010 and all conflict bit positions are selected and updated.

The MHD configurations for updating conflict-bit matrix are shown in Fig. 5d. In this example, each slice of C[n×S] is updated with only one configuration. The number of required full rotations in the original circular-scan architecture (Arslan and Orailugu, 2004) is equal to the maximum number of conflict bits in slices. The maximum number of conflict-bits in the slices in this example is 3. So, in comparison with original circular-scan architecture which 3 full rotation is needed for updating all conflict bits, in the proposed architecture based on MHD, only one rotation is needed, because all conflict bits in each slice is updated by only one MHD configuration.

Example 2: In test slices with large number of conflict bits, in order to update conflict bits, several MHD configurations and consequently several full scan chain rotations may be required. In this situation, the aim is to find minimum number of required configurations in order to reduce the required control bits for MHD.

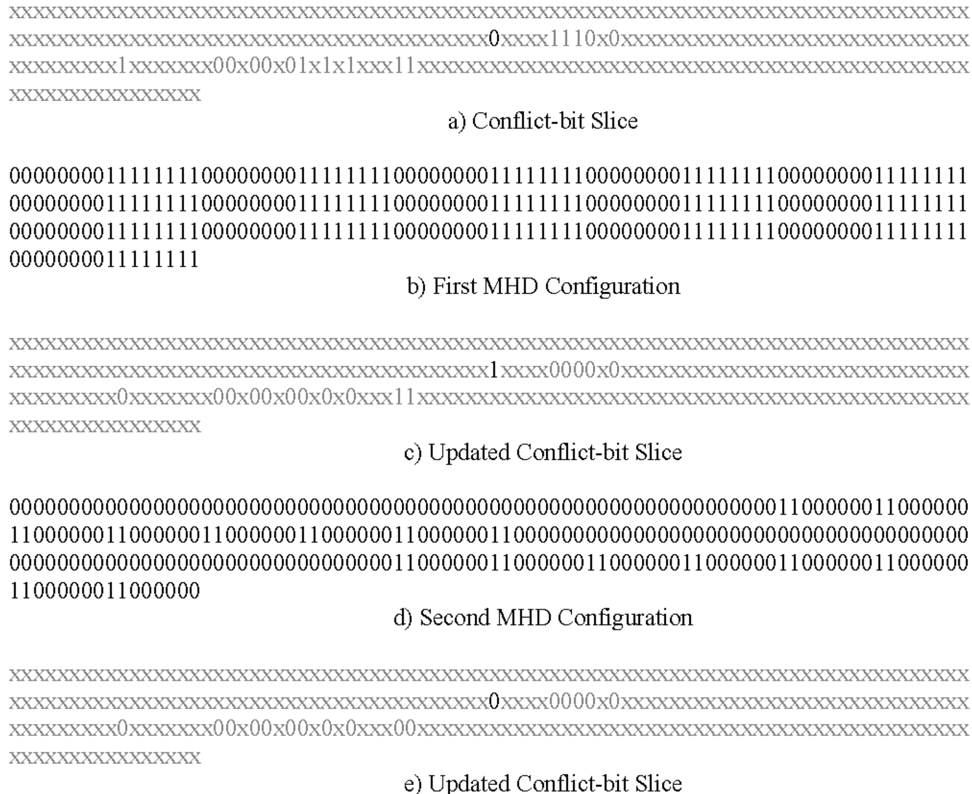

Fig. 6: Updating the conflict-bit slice

Consider the example shown in Fig. 6. For constructing the conflict-bit slice which depicted in Fig. 6a, the test matrix (T[7×256]) and its corresponding captured response matrix (R[7×256]) is used. The test matrix and its corresponding captured response matrix are generated using ATALANTA test pattern generator (Lee and Ha, 1993) and HOPE fault simulation (Lee and Ha, 1992) tools, for s38417 circuit, one of the ISCAS'89 benchmarks. In Fig. 6a, due to the space limitations, only one slice is shown.

In the conflict-bit slice, there are 17 specified bits; 9 conflict bits which are shown with 1; 8 bits are shown with 0 and the other bits remain don't care (x). The aim is to find minimum number of configurations to update conflict-bits in the given slice (changing all 1 bits to 0). In this example, there are 256 scan chains; as a result, there are $3^{\lceil \log_2^S \rceil}$ = 6561 different configurations in the MHD output. The first selected configuration for updating conflict-bit slice is shown in Fig. 6b. This configuration is chosen among 6561 different MHD configurations and 7 conflict-bits are correctly updated.

As shown in Fig. 6c, beside the conflict-bits, one of the none-conflicts bits (0) (which is shown in bold) is selected and inverted. This change has to be compensated in the next scan chain full rotations, using another configuration. It is clear that any change in x bits is acceptable. As shown in Fig. 6d and e, after applying the second configuration, all conflict-bits are changed to the 0.

In this example, all conflict bits are updated after applying 2 configurations, where in circular-scan architecture (Arslan and Orailuglu, 2004), 9 configurations are needed, because by using a regular decoder, it is possible to update only one bit in each rotation. Since the length of test slice is 256, by using an 8 to 256 decoder, the control bits volume, v, in original circular-scan architecture (Arslan and Orailuglu, 2004) is 72, Eq. 12.

$$v = I \times NC = 8 \times 9 = 72 \qquad (12)$$

In this equation, I is the number of regular decoder or MHD inputs and NC is the number of required configurations for updating conflict bits. In the parallel circular-scan architecture proposed in this study, the March 11, 2008 number of control bits reduced to 32, Eq. 13:

$$v = I \times NC = 16 \times 2 = 32 \quad (13)$$

Beside the reduction of the required control bits volume which results in more test data compression rate, the proposed parallel circular-scan architecture reduces the test application time, significantly. In this example, the number of scan chains full rotation is reduced to 2, where in the original circular-scan architecture is 9.

It is worth to mention that the MHD could be used in the original circular-scan architecture (Arslan and Orailuglu, 2004). However, in the proposed architecture, we used the modified scan selection unit (Azimipour *et al.*, 2007) for two reasons. First, it reaches more compression in test data volume by updating conflict bits internally. Second, it allows simultaneous updating of conflict bits with opposite logics. In the original circular-scan architecture, conflict bits are updated by using the data input pin; so, in each time it could update the group of the conflict bits with the same logic.

Consider the slice #4 in the example 1 and suppose that the MHD is used in the original circular-scan architecture with scan selection unit depicted in Fig. 2. In this slice, bits #1 and 5 are '1' and have to be inverted to '0'. These bit positions could be in the same group and selected and updated using the MHD with d01 address. The updating of these conflict-bits is done by loading 0 through the data input pin shown in Fig. 2. The bit #2 contains 0 and has to be updated to 1; therefore, this bit position could be selected and updated by 010 address in another rotation of scan chains. As a result, the minimum number of configurations for updating MHD is reduced to 2, which in the original circular-scan architecture are 3. By using the modified scan input selection unit (Azimipour *et al.*, 2007) which is depicted in Fig. 3, it is possible to select all 3 conflict bits using one configuration and consequently update all conflict-bits in only one scan chains full rotation. This provides more reduction in the test data volume and test application time for updating conflict bits.

RESULTS AND DISCUSSION

The basic idea of circular-Scan (Arslan and Orailuglu, 2004) is to use the captured response of the previously applied test pattern as a template for the next pattern. In this architecture, conflicting bits are updated using data input pin. In this paper a new circular-Scan architecture that makes it possible to select several scan chains in parallel is presented. The proposed architecture is relied on the circular-scan architecture proposed in Arslan and Orailuglu (2004), the modified scan input selection unit proposed in Azimipour *et al.* (2007) and multiple-hot decoders proposed in Al-Yamani *et al.* (2005). In the presented algorithm, the aim was to update all conflict bits in conflict-bit matrix according to MHD different possible configurations. Updating conflict bits has done by applying all possible configurations, selecting the best configuration and updating conflict-matrix according to selected configuration. These procedures continued until updating all conflict bits.

Experiments are performed on the 5 largest ISCAS'89 circuits. In these experiments, the ATALANTA test generation tool and the HOPE fault simulation tool (Lee and Ha, 1993) are used. The general information and density of specified bits (p) in test vectors used in these experiments is shown in Table 2. Results for test data volume reduction for scan chains counts (N_{sc}) of 64, 128 and 256 are shown in Table 3.

Table 3 shows that by using the proposed algorithm in this study, the test data volume in the s13207, s15850, s35932, s38417 and s38584 circuits, for 256 scan chains, are reduced by a factor of 17.8, 18.9, 53.6, 17.5 and 22.9%, respectively. This result is obtained in comparison with the original-circular scan architecture (Arslan and Orailuglu, 2004).

In Table 4, reductions in test application time is shown. In the proposed architecture for 256 scan chains, the improvements in test application time for s13207, s15850, s35932, s38417 and s38584 circuits are 14.1, 23.3, 54.7, 19.8 and 22%, respectively. This result also is obtained in comparison with the original-circular scan architecture (Arslan and Orailuglu, 2004).

Table 2: Iscas'89 benchmark specifications

Circuit	Flop#	T	f	P (%)
s13207	700	291	9085	5.1
s15850	611	190	10852	10.8
s35932	1763	39	33247	10.4
s38417	1664	294	30251	13.7
s38584	1464	267	31569	8.3

Table 3: Test data volume reduction in 5 largest iscas'89 circuits

	N_{sc} = 64			N_{sc} = 128			N_{sc} = 256		
Circuit	(Arslan and Orailuglu, 2004)	(Azimipour *et al.*, 2007)	Proposed	(Arslan and Orailuglu, 2004)	(Azimipour *et al.*, 2007)	Proposed	(Arslan and Orailuglu, 2004)	(Azimipour *et al.*, 2007)	Proposed
S13207	53.6	58.3	68.4	65.1	67.8	77.3	61.6	69.3	79.4
S15850	28.9	33.7	45.5	39.5	44.5	62.1	44.2	51.6	63.1
S35932	5.1	7.2	33.3	16.2	20.7	66.4	21.7	26.0	75.3
S38417	28.3	32.6	48.1	36.8	42.9	59.2	42.2	48.3	59.7
S38584	30.9	35.4	51.5	40.7	48.1	60.3	48.9	55.6	71.8

Table 4: Test application time reduction in 5 largest iscas'89 circuits

	Nsc = 64		Nsc = 128		Nsc = 256	
Circuit	(Arslan and Orailuglu, 2004)	Proposed	(Arslan and Orailuglu, 2004)	Proposed	(Arslan and Orailuglu, 2004)	Proposed
S13207	51.2	70.3	64.1	73.7	66.2	80.3
S15850	31.5	50.6	38.0	65.4	43.8	67.1
S35932	9.6	44.7	15.1	68.9	17.8	72.5
S38417	30.3	50.6	35.8	56.4	43.4	63.2
S38584	31.7	59.4	43.3	65.1	51.6	73.6

It must be mentioned that proposed architecture provides the considerable reduction in test data volume and the test application time with minimal hardware overhead for implementation of MHD and the few extra number of scan input counts. In the original circular-scan architecture (Arslan and Orailuglu, 2004), the number of required scan input pins for implementation of a regular decoder is $\lceil \log_2^S \rceil$, where S is the number of scan chains. In the parallel circular scan chain, proposed in this research, the number of scan input pins for implementation of MHD is increased to $2 \times \log_2^S$.

CONCLUSION

One of the major challenges in testing SOC is dealing with the large size of test data that are stored in the tester and transferred between the tester and chip (Zorian *et al*., 1998). Circular-Scan architecture addresses some of the problems in SOCs testing. These problems are test data volume and test application time. In the circular-scan architecture, conflicting bits are updated using data input pin. In the architecture proposed in this study, the multiple-hot decoder (MHD) is used instead, of the regular decoder and allows simultaneous updating of conflict bits. In the presented algorithm, the aim was to update all conflict bits in conflict-bit matrix, according to MHD different possible configurations. Experimental results showed an average improvement of 26% in test data volume and test application time in 5 largest ISCAS'89 benchmark circuits, in comparison with original circular-scan architecture (Arslan and Orailuglu, 2004).

ACKNOWLEDGMENT

This study was supported by the Iran Telecommunication Research Center (ITRC) under grant No. T500/1917.

REFERENCES

Al-Yamani, A., E. Chmelar and M. Grinchuk, 2005. Segmented addressable scan architecture. In: Proceedings of VLSI Test Symposium, pp: 405-411.

Arslan, B. and A. Orailuglu, 2004. Circular scan: A scan architecture for test cost reduction. In: Proceedings of DATE Conference, pp: 1290-1295.

Azimipour, M., M. Eshghi and A. Khademzadeh, 2007. A modification to circular-scan architecture to improve test data compression. IEEE CS Proceedings of 15th International Conference Advance Computing and Communication, pp: 27-33.

Balakrishnan, K.J. and N.A. Touba, 2006. Improving linear test data compression. IEEE Trans. VLSI Syst., 14 (11): 1227-1237.

Hamzaoglu, I. and J.H. Patel, 1999. Reducing test application time for full scan embedded cores. In: Proceedings of 29th International Symposium on Fault Tolerant Comp., pp: 260-267.

Hiraide, T., K.O. Boateng, H. Konishi, K. Itaya, M. Emori and H. Yamanaka, 2003. BIST-aided scan test-a new method for test cost reduction. In: Proceedings of VLSI Test Symposium, pp: 359-364.

Khoche, A. and J. Rivoir, 2002. I/O bandwidth bottleneck for test: Is it real. In: Proceedings of Test Resource Partitioning Workshop.

Lee, H.K. and D.S. Ha, 1992. HOPE: An efficient parallel fault simulator. In: Proceedings of Design and Automation Conference, pp: 336-340.

Lee, H.K. and D.S. Ha, 1993. On the generation of test patterns for combinational circuits. Technical report, department of Electrical Engineering, Virginia Polytechnic Institute and State University, pp: 12-93.

McCluskey, E.J., 1986. Logic Design Principles with Emphasis on Testable Semicustom Circuits. Prentice-Hall, Englewood Cliffs, NJ, USA.

Touba, N.A., 2006. Survey of test vector compression techniques. IEEE Design and Test of Computers, pp: 294-303.

Zorian, Y., E.J. Marinissen and S. Dey, 1998. Testing embedded-core based system Chips. In: Proceedings of International Test Conference, pp: 130-143.