A Scalable and Parallel Test Access Strategy for NoC-based Multicore System

Taewoo Han, Inhyuk Choi, Hyunggoy Oh, Sungho Kang
Department of Electrical and Electronic Engineering
Computer systems & reliable SoC Lab., Yonsei University
Seoul, Korea
{twhan, ihchoi, kyob508}@soc.yonsei.ac.kr, shkang@yonsei.ac.kr

Abstract—This paper proposes a new parallel test access strategy for multiple identical cores in a network-on-chip (NoC). The proposed test strategy takes advantage of the regular design of NoC to reduce both test area overhead and test time. The proposed NoC reused test access mechanism (TAM) adopted a pipelining structure and a deterministic test data routing algorithm in order to reuse the full bandwidth of links in the NoC. Also, the architecture has complete scalability according to the number of cores and applications for 3D environment are also represented. Experimental results show that the proposed TAM can test multiple cores with the same test time as a single core and negligible hardware overhead.

Keywords—parallel test; multiple identical cores; NoC; TAM

I. INTRODUCTION

The number and complexity of cores in a system-on-chip (SoC) are predicted to increase continuously and rapidly in the next few years. It makes a major challenge to design and test reliable systems and thus efficient design-for-test (DFT) technologies are required [1]. Many difficulties in testing arise from the use of deeply embedded cores. A test access mechanism (TAM) is needed that allows the core to be efficiently accessed and tested [2]. IEEE std. 1500 provides an efficient solution for testing cores by wrapping cores to prevent interaction from outside data sources and allowing tests for a single core to be generated and applied to each of the instances of the core [3]. With technology scaling, the communication architecture cause severe on-chip synchronization errors, unpredictable delays, and high power consumption. One emerging solution for such communication bottleneck is network-on-chip (NoC) as a promising alternative to bus-based and point-to-point communication architectures [4]. Furthermore, the reuse of the NoC as TAM was presented as a cost-effective strategy for the testing of embedded cores, with reduced area, pin count, and test time costs [5]. Infrastructures of the NoC must be tested before reusing the NoC as TAM [6]. Routers in a NoC are homogeneous structure and concurrent test strategy for them was studied [7]. Previous researches about test strategy for NoC-reused TAM were targeting heterogeneous cores and the optimization of test pin-count, test scheduling and test access was represented [8, 9].

Recently, multicore and manycore designs have evolved to include multiple identical cores. This trend has been increasingly observed for CPU cores in modern microprocessor designs [10]. The use of multiple identical cores achieves several goals: in addition to the benefits of multiprocesing, some cores can be used as redundant cores to guarantee a highly reliable system. Especially, NoC helps implementing reconfigurable system and a topology reconfiguration for defect-tolerant NoC-based homogeneous manycore systems is researched [11]. However, it did not focus about the effective testing of homogeneous cores. Some parallel TAMs for concurrent testing of multiple identical cores in not NoC but non-NoC based SoC have been described previously. The AZSCAN architecture tests multiple identical cores in parallel, and the responses are compared with the expected data in the chip by using on-chip comparators [12]. A pipeline-based TAM allows a great deal of flexibility in test applications, and the pipelining helps to improve test times and to reduce the capture power requirements [13]. Also, a majority-based TAM uses the majority value of test response data and all cores can be tested using the same test pins and the same test time as required for testing a single core [14]. These parallel TAMs are based on bus-based architecture and they have not supported NoC-reuse TAM.

In this paper, a parallel TAM which specialized for multiple identical cores in NoC-based system is proposed. It aims to test multiple cores in parallel for minimizing test time and test pins. In addition, it targets most common NoC architecture [15] and has flexibility in design, configuration, and application. By reusing the infrastructures of NoC, the hardware overhead is minimized and abundant interconnections help to reduce the test time. In order to use the full bandwidth of NoC interconnection as the width of TAM, a new deterministic routing algorithm to transfer test data is designed. The proposed TAM can be used to perform a complete core-level diagnosis in the case of faults by multiple cores.

This paper is organized as follows: Section 2 reviews more detailed relating works. Section 3 presents an overview of NoC design and a description of the NoC-reused TAM. It includes a new architecture for 3D environments. Section 4 discusses experimental results, while Section 5 concludes the paper.
II. RELATED WORKS

On-chip access to wrapped cores embedded in a SoC is provided by a TAM. The TAM is used to transport the test stimuli data from a test pattern source to the core under test and to transport the test response data from the core under test to a test pattern sink. In multicore or manycore systems, TAM can be implemented in various ways, which has test pins and test time as the main factors. Especially, multiple identical cores are expected to respond identically to given test patterns. This allows TAM to compare the test response data from cores either to the expected response or to the other cores’ response using on-chip comparators [16].

When the test response data of cores are compared with the expected response at on-chip, additional test pins are required for the expected data input [12]. To reduce the number of test pins to that of a single core, a pipelined TAM has been proposed [13]. In this design, the test response data of multiple identical cores are compared on chip with the test response data from a primary core. The test response data of the primary core are compared with the expected data in the automated test equipment (ATE); if they agree, the primary core is considered non-faulty, and any other cores whose test response data differ from that of the primary core are thus considered faulty. On the contrary, if the primary core has a fault, the other core is selected as a primary core and the test is restarted with the new primary core.

The proposed NoC-reused TAM adopted the pipelined structure and implemented the pipelining registers by reusing input buffers of the routers in NoC. Therefore, with minimized hardware overhead, the proposed TAM can test multiple identical cores with the same number of test pins and test time as that of a single core. However, if the pipeline-based TAM is applied to NoC-reused TAM as it is, the test stimuli data and the response data of primary core are transferred in one direction. It will reuse only half of the bidirectional links in NoC and it lead to loss of test time. In order to reuse the full bandwidth of the links and reduce the test time, a novel routing algorithm of test data (test stimuli and response of primary core) is proposed in addition to the pipelined NoC-reused TAM architecture.

III. PROPOSED NOC-REUSED PARALLEL TAM

NoCs typically use the packet-passing communication model, where the IP cores attached to the network communicate by sending and receiving request and response messages. It can be described by the approaches used to implement the mechanisms of flow control, routing, arbitration, switching, and buffering [17]. Figure 1 presents a typical structure of NoC-based system with the proposed NoC-reused parallel TAM (which is denoted as NRP-TAM in this paper). More detailed descriptions about NRP-TAM are presented in the following chapters and the experimental results show that it has negligible hardware overhead.

A. Architecture

NRP-TAM reuses the buffers in inputs of the routers and some multiplexers and comparators (XOR gates) are added, but the comparators can be shared with the testing of routers in [7].

Fig. 1. Architecture of the proposed TAM in NoC

Figure 1 shows a partial hardware architecture of the proposed TAM in NoC. It is assumed that the Router0 is a source & sink node and Core0 is a primary core. External ATE sends the test stimuli data to Router0 and receives the test response data from Router0 and confirms whether the response data is identical with expected data. Router0 transfers the test stimuli data from ATE to Core0 and at the same time, it is transferred to Router1. One input buffer in Router 1 is reused as the pipelining register for test stimuli data. The red lines indicate the transmission of test stimuli data which transfers from Router0 to Router1, Router5 and Router4 according to the proposed routing algorithm. Core0 is selected as the primary core and the test response data of Core0 is transferred to both Core4 and ATE. The blue lines indicate the transmission of Core0’s test response data which transfers from Router0 to Router4, router5 and Router1.

All routers and cores can send and receive the test stimuli data and test response data of primary core in the same way. And then, each router compares the test response data of its core and the test response data of primary core. In some cases, additional buffering is necessary for matching the timing of test response data of each core and primary core.

Router1 receives the test stimuli data prior to the test response data of primary core, so the test response data of Core1 must be buffered until receiving the primary core’s test response data. This buffering reuses the input buffer as the pipelining register and the routing algorithm will show that the buffering depths are always two clocks. As shown in Figure1, one of the input buffers which connected to Core1 in Router1 is reused as a pipelining register and two more buffers are reused as buffering register (the buffering in this case is called as core buffering in this paper). Therefore, Core1 can be tested by comparing the buffered test response data and the received test response data of primary core through north input path.
Router4 receives the test response data of primary core prior to the test stimuli data, therefore the primary core’s test response data must be buffered until receiving the test response data of Core4 (the buffering in this case is called as primary_buffering in this paper). Core4 can be tested by comparing the test response data of Core4 and the received and buffered test response data of primary core through south input path.

On the other hand, Router5 does not need any buffering because it receives the test response data of Core5 (stimulated by the test stimuli data through south path) and the test response data of primary core at the same time. Core5 can be tested by comparing the test response data of Core5 and the received test response data of primary core through east path. According to the proposed test data routing algorithm in the next chapter, most routers do not need buffering like as Router5.

B. Routing

![Routing algorithm of the proposed TAM in NoC](image)

Fig. 2. Routing algorithm of the proposed TAM in NoC

In order to reuse the bidirectional links in NoC for an abundant bandwidth of TAM, NRP-TAM uses a novel deterministic routing algorithm for test data. Figure 2(a) and Figure 2(b) represent 4*3 NoC systems and routings according to the proposed algorithm. Cores connected to the routers are omitted in these figures.

The main rule of the proposed routing algorithm for test data is to route with the shortest path, while test stimuli data having priority in a row routing and test response data of primary core having priority in a column routing.

In Figure 2(a), Router0 is a source & sink node and Core0 is selected as a primary core. Router0 transfers the test stimuli data through a row line and Router1 receives this data. To transfer the test response data of primary core to Router1, it bypasses Router4 and Router5 since the east link of Router0 is preoccupied by the test stimuli data. As a result, Router1 requires the core_buffering and the depth of this buffering is two clocks for bypassing two Routers. On the other hand, Router0 transfers the test response data of primary core through a column line and Router4 receives this data. To transfer the test stimuli data to Router4, it bypasses Router1 and Router5 because the north link of Router0 is preoccupied by the test response data. Therefore, Router4 has the primary_buffering and the depth of this buffering is also two clocks for bypassing two Routers. The test stimuli data and test response data of the primary core for Router5 are routed from Router1 and Router4 respectively according to the rule of shortest path.

After routing all the routers in a NoC by the proposed test data routing algorithm, only routers in the same row (Router1, Router2 and Router3) and column (Router4 and Router8) with the primary core have buffering and regardless of the number of cores, the depth of buffering are always two clocks. The remaining routers do not need the buffering and all cores in the NoC can be tested concurrently by the pipelined test data.

If the primary core is faulty in the pipeline based TAM, that core is derated and an arbitrary core is decided as a new primary core. Figure 2(b) shows the state of Core0 which was a primary core has a fault and Core1 is selected as a new primary core. NRP-TAM selects the primary core in a numerical order (Core0, Core1, Core2 …). The source & sink node is not changed, but the routings are mirrored on the column of Router1 (connected to the new primary core) as axis. The proposed routing algorithm offers the same routing mentioned above to the routers on the axis and the right side. On the contrary, cores on the left side are routed as mirrored.

In Figure 2(b), Router0, Router4 and Router8 are on the left side of the primary core. Router1 receives the test stimuli data from ATE through Router0. Then, Router1 sends the test stimuli data to Router2 and at the same time, Router1 sends it to Router4 through Router0. In other words, Router0 multicasts by connecting east input link to west output link and west input link to north output link. Router4 receives the test response data of the primary core from Router1 through Router5. The timing of test response data of Core4 and the primary core at Router4 are matched and no buffering is required. Router8 can be routed in the same manner. Only routers in the same row (Router2 and Router3) and column (Router5 and Router9) with the primary core have buffering as before.

The proposed deterministic test data routing algorithm is arranged as pseudo-code in Figure 3. It is designed clearly and simply according to the position of routers. The routing algorithm of NRP_TAM consists of Primary_routing (routing of the primary core) and General_routing (routing of the non-primary cores).
transfers the test stimuli data in the mirrored direction. If the router is on the same column, and the down side of the primary core, it can be considered that the core connected to this router was a primary core and a fault was detected in this core. In this case, this router is inactive.

Last is the case when the remaining routers are on neither same row nor column of the primary core. If the router is on the right and upper side of the primary core, it receives the test stimuli data from south link and the test response data of the primary core from west link. On the other hand, if the router is on the left and upper side of the primary core, it is routed in the mirrored direction (receives the test stimuli data from south and the test response data from east). The cores connected to the other routers are faulty and they are inactive.

The proposed deterministic test data routing algorithm can be extended to cases of alternative primary core or the number of cores which is changed. Likewise general NoC, the proposed routing algorithm in NoC-reused TAM design gives importance to the complete scalability. The characteristic of routing algorithm always requires two additional buffers for some routers in order to use the entire bidirectional links. These buffers can be implemented by reusing the input buffers in general routers at NoC. Even though a particular NoC has less than three input buffers in router (one for pipelining and two for additional buffering), this overhead would be negligible.

**C. Timing diagram**

The core buffering plays an important role to the complete scalability. The characteristic of routing algorithm always requires two additional buffers for some routers in order to use the entire bidirectional links. These buffers can be implemented by reusing the input buffers in general routers at NoC. Even though a particular NoC has less than three input buffers in router (one for pipelining and two for additional buffering), this overhead would be negligible.

**C. Timing diagram**

To represent pipelined test data and concurrent test process of the NRP-TAM, Figure 4 shows the timing diagram of test data in the proposed TAM. The test stimuli data is transferred from Router0 to Router1, Router5 and Router4. Therefore, they receive the test response data of their core in the same order as above. The test response data of the primary core is transferred from Router0 to Router4, Router5 and Router1. Router1 needs core_buffering for comparing the first test response of its core (TR1) and the first test response of the primary core (PR1).
Router4 needs primary_buffering and the depths of both buffering are two clocks as described earlier.

For example, when the clock time is clk4, Router0 receives the fourth test response data of Core0 (TR4) and transfers it as the fourth test data of the primary core (PR4). Router1 receives the third test response data of Core1 (TR3) and the first test response data of primary core (PR1). TR3 is buffered and PR1 is compared with the core_buffered TR1 for testing Core1. Router5 receives the second test response data of Core5 (TR2) and the primary core (PR2). They are compared for testing Core5. Router4 receives the first test response data of Core4 (TR1) and the third test response data of primary core (PR3). PR3 is buffered and TR1 is compared with the primary_buffered PR1 for testing Core4.

D. 3D Extention

![Fig. 5. Routing results with the proposed TAM 3D NoC](Image)

Advances in chip design and test technology have led to 3D stacked chips and there are many recent studies about 3D NoC topology and 3D TAM technologies for reliable and high yield 3D ICs [18, 19, 20, 21]. NRP-TAM can extend to such 3D environments and Figure 5 shows the proposed TAM in a 4*3*2 NoC system.

Each layer can be tested by the proposed test data routing algorithm in NRP-TAM as mentioned earlier. This is a pre-bond testing process in the 3D stacked ICs, and the proposed TAM is extended for the post-bond testing. In Figure 5, the routing result from the bottom die is the same as the routing result of the 2D NoC in Figure 2(a). The upper die receives the test stimuli data (TS) and the test response data of the primary core (TR) from the bottom die. In this case, only two vertical links are reused for transmission of the test data and Router12 operates as a semi-primary router in the upper layer.

Router12 receives the test response of the primary core from Router0 (primary router) and the test stimuli data from ATE through Router0, Router1 and Router13. In the upper layer, Router12 sends the test stimuli data and the test response data of primary core to the other routers and the cores in this layer can be tested concurrently. The routers on the same row or column with the semi-primary router require two depths of buffering as the bottom layer, but the semi-primary router also needs the buffering.

IV. EXPERIMENTAL RESULTS

Several experiments are performed to verify the effectiveness of the proposed NoC-reused parallel TAM. The general NoC [22] is synthesized by Synopsys 90nm generic library [23] for analyzing and implementing the proposed TAM in real NoC systems.

A. Test time

<table>
<thead>
<tr>
<th>TAM</th>
<th>Test time (cycles)</th>
</tr>
</thead>
<tbody>
<tr>
<td>width</td>
<td>M5 in d695</td>
</tr>
<tr>
<td>8</td>
<td>24,197</td>
</tr>
<tr>
<td>16</td>
<td>12,209</td>
</tr>
<tr>
<td>32</td>
<td>6,215</td>
</tr>
</tbody>
</table>

In the NoC with the bidirectional link, NRP-TAM reuses one directional link for the test stimuli data and another directional link for the test response data of primary core. Therefore, multiple cores can be tested in the same amount of test time when testing a single core by a typical TAM which has the unidirectional link. The bandwidth of the proposed TAM is twice as large as the NoC-reused TAM applied by pipeline-based TAM because it reuses the one directional link for both the test stimuli data and the test response data of primary core.

In order to show the relation between test time and width of the NoC-reused TAM, some modules in ITC’02 benchmarks [24] are tested according to wrapper & TAM optimization method [25]. Some hypothetical homogeneous systems are implemented by duplicating a specific module in the benchmarks. Therefore, the test time of a specific core in the benchmarks are used as the total test time of multiple identical cores with NRP-TAM. As the width increases as twice as large, the test times tends to decrease nearly as half. For example, the result value 48.81% in Table 1 indicates the average reduction rate of each test time as the width increases from 8 to 16. This experiment shows that the proposed TAM helps to reduce the test time in nearly half due to the bidirectional links.

B. Hardware overhead

<table>
<thead>
<tr>
<th>NoC</th>
<th>Hardware area</th>
<th>Proposed TAM</th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>buffer</td>
<td>flit</td>
<td>NoC</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>16</td>
<td>90,426</td>
<td>1,744</td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>136,680</td>
<td>3,439</td>
</tr>
<tr>
<td>64</td>
<td></td>
<td>231,623</td>
<td>6,866</td>
</tr>
<tr>
<td>12</td>
<td>16</td>
<td>141,164</td>
<td>1,743</td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>224,772</td>
<td>3,451</td>
</tr>
<tr>
<td>64</td>
<td></td>
<td>394,518</td>
<td>6,990</td>
</tr>
<tr>
<td>18</td>
<td>16</td>
<td>193,380</td>
<td>1,765</td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>314,118</td>
<td>4,487</td>
</tr>
<tr>
<td>64</td>
<td></td>
<td>565,113</td>
<td>14,127</td>
</tr>
</tbody>
</table>
### Table III. Hardware Overhead of the Proposed TAM in 4*3*2 (3D) NoC

<table>
<thead>
<tr>
<th>NoC buffer</th>
<th>Hardware area</th>
<th>Flit</th>
<th>NoC</th>
<th>Proposed TAM</th>
<th>Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td></td>
<td>16</td>
<td>180,853</td>
<td>3,626</td>
<td>1.97%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>32</td>
<td>273,361</td>
<td>7,142</td>
<td>2.55%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>64</td>
<td>463,246</td>
<td>14,269</td>
<td>2.99%</td>
</tr>
<tr>
<td>12</td>
<td></td>
<td>16</td>
<td>282,329</td>
<td>3,623</td>
<td>1.27%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>32</td>
<td>449,544</td>
<td>7,183</td>
<td>1.57%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>64</td>
<td>789,048</td>
<td>14,512</td>
<td>1.81%</td>
</tr>
<tr>
<td>18</td>
<td></td>
<td>16</td>
<td>386,760</td>
<td>3,671</td>
<td>0.94%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>32</td>
<td>628,224</td>
<td>9,261</td>
<td>1.45%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>64</td>
<td>1,130,232</td>
<td>28,402</td>
<td>2.45%</td>
</tr>
</tbody>
</table>

Table II and Table III represent the hardware size of the proposed TAM with NoC in the number of NAND gates. The feature of NRP-TAM is reusing the input buffers for pipelining registers and additional buffering. Additional multiplexers and comparators relating to the flit size of NoC are needed. Therefore, the hardware experimental results indicate that the hardware overhead of the proposed TAM is decreased when the number of buffers in the original NoC are increased. Also, the hardware overhead is increased in the larger size of flit at NoC, which is the width of the TAM.

The hardware overhead of NRP-TAM is less than 3% in the worst case of the experiments, but it does not include the hardware size of cores. Considering the fact that the number of gates of a modern multicore processor system is much more than million gates, the hardware overhead of the proposed TAM in a chip is negligible. In conclusion, the proposed hardware architecture needs a few more multiplexers (other architectures can reuse the infrastructures of NoC) than the aforementioned unidirectional NoC-reused TAM applied by pipeline-based TAM, but the experimental results show that this hardware overhead is imperceptible.

### V. CONCLUSION

This paper describes a novel NoC-reused parallel TAM for concurrent testing of multicore or manycore system. All the cores can be tested simultaneously by reusing infrastructures of the NoC and test time as required for a single core. A new deterministic test data routing algorithm is designed for reusing the full bandwidth of bidirectional NoC links as the TAM. Experimental results show that the proposed TAM has a minimized test time with abundant TAM width and negligible hardware overhead. The architecture has complete scalability and it can be extended to 3D environments for 3D NoC topology and 3D stacked ICs. The proposed NRP-TAM is only related to the delivery of test data and it can be compatible and improved with the existing DFT technologies.

### ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 2012R1A2A1A03006255).

### REFERENCES