Interconnect Delay Fault Test on Boards and SoCs with Multiple Clock Domains

Hyunbean Yi, Jaehoon Song, and Sungju Park

This paper introduces an interconnect delay fault test (IDFT) controller on boards and system-on-chips (SoCs) with IEEE 1149.1 and IEEE 1500 wrappers. By capturing the transition signals launched during one system clock, interconnect delay faults operated by different system clocks can be simultaneously tested with our technique. The proposed IDFT technique does not require any modification on boundary scan cells. Instead, a small number of logic gates needs to be plugged around the test access port controller. The IDFT controller is compatible with the IEEE 1149.1 and IEEE 1500 standards. The superiority of our approach is verified by implementation of the controller with benchmark SoCs with IEEE 1500 wrapped cores.

Keywords: Design for testability, system-on-chip, IEEE 1149.1, IEEE 1500, multiple clock domains.

I. Introduction

As board and system-on-chip (SoC) integration density and the speed of system clocks increase, it is crucial to test delay as well as conventional static faults on interconnect wires. Boundary scan design is a design-for-testability technique to simplify the application of test patterns for the detection and diagnosis of different faults at levels of packages, such as chips, modules, boards, and backplanes. Both the IEEE 1149.1 standard for board testing and IEEE 1500 standard for embedded core testing have become increasingly prevalent as industrial standards [1], [2].

The IEEE boundary scan architecture consists of test access ports (TAPs), a TAP controller, and instruction and data registers. Test data input (TDI), test data output (TDO), test clock (TCK), test mode selector (TMS), and test reset (TRST) constitute the TAP, and TRST is optional. All of the input and output pins of a chip are connected to input and output boundary scan cells, respectively. The IEEE boundary scan instructions can be classified into compulsory instructions, such as BYPASS, external test (EXTEST), and SAMPLE /PRELOAD, and optional instructions, such as CLAMP, HIGHZ, and RUNBIST [1].

As shown in Fig. 1, the TAP controller is a finite state machine with 16 states. It mainly enables the application of patterns to data and instruction registers as well as the observation of test responses. The interconnect faults on a board can be classified into stuck-at, stuck open, shorted net, and delay. Interconnect delay defects prevent rising or falling values from reaching the receiver within a system clock interval.

This paper introduces a new technique which makes it possible to test delay defects in addition to the static interconnect faults with the EXTEST instruction. The method

Manuscript received Sept. 20, 2007; revised Dec. 2, 2007.

This work was supported by the "System IC2010" project of Korea Ministry of Commerce, Industry and Energy.

Hyunbean Yi (phone: + 82 31 408 4723, email: bean@mslab.hanyang.ac.kr), Jaehoon Song (email: song@mslab.hanyang.ac.kr), and Sungju Park (email: parksj@mslab.hanyang.ac.kr) are with the Department of Computer Science and Engineering, Hanyang University, Seoul, Rep. of Korea.

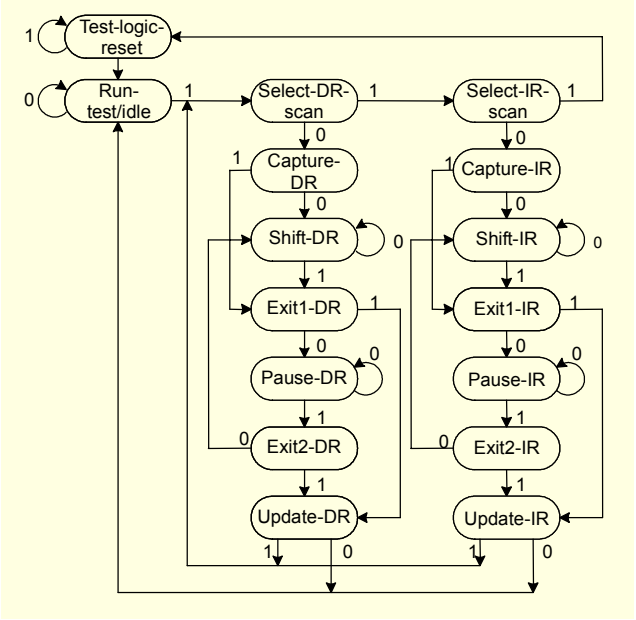

Fig. 1. 16 states of IEEE 1149.1 TAP controller.

to apply and observe interconnect test patterns and state transitions of the TAP controller upon the TMS values can be summarized as follows.

Step 1. EXTEST instruction is read and decoded. The state transition is the following: RESET → IDLE → Select-DR → Select-IR → Capture-IR → Shift-IR → ⋯ → EXIT1-IR → Update-IR →.

Step 2. Interconnect test patterns are serially applied through the boundary scan register. The corresponding state transitions are the following: Select-DR → Capture-DR → Shift-DR → ⋯ → EXIT1-DR →.

Step 3. Test patterns read are applied to the Update latch and the signals are propagated to input boundary scan cells (BSCs) in parallel. The corresponding state transitions are the following: Update-DR → Select-DR → Capture-DR →.

Step 4. Test responses captured are shifted out through BSCs to TDO. The corresponding state transitions are the following: Capture-DR → Shift-DR → ⋯ → EXIT1-DR →.

Update-DR and Update-IR states are active on the falling edge of the TCK while all the others are active on the rising edge. The timing diagram of step 3 is shown in Fig. 2. Update-DR (UpdateDR) and Capture-DR (ClockDR) are active on the Update and Capture dotted lines, respectively; thus, 2.5 TCKs are required from Update-DR to Capture-DR. That is, it takes 2.5 TCKs from the application of the interconnect test patterns through the output BSCs to the observation of test responses to the input BSCs. Although the number of TCK cycles has no relevance to the detection of static interconnect faults, the test

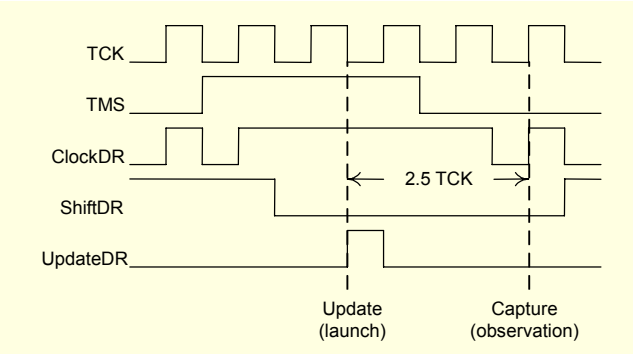

Fig. 2. Launch and observation timing interval of 2.5 TCKs by IEEE 1149.1 TAP controller.

responses must be observed in a system clock cycle to detect delay defects. Since the delay defects cannot be tested with the IEEE boundary scan design, either the state diagram of the TAP controller or boundary scan cells must be changed to capture the responses within associated system clock intervals, especially if multiple system clocks are involved. Even if a test control technique allows the launched signal to be captured within an associated system clock interval, the skews on the TCK or system clocks may invalidate the test results.

This paper presents interconnect delay fault test (IDFT) solutions on boards and SoCs operating with multiple system clocks. Related works are introduced in section II, and section III describes IDFT circuits and the test procedure with IEEE 1149.1. In section IV, the IDFT for an SoC with IEEE 1500 wrappers is presented. Comparative analysis from design experiments shows the superiority of our approach in section V, and the conclusion follows in section VI.

II. Related Work

The IEEE 1149.1 standard has an inherent deficiency in testing delay defects on interconnect wires due to the required 2.5 test clock interval from launching to capturing the test patterns. Boundary scanning has been used to test delay defects on I/O pads at wafer or package levels [3]; however, at-speed testing by multiple system clocks has not been properly addressed.

An early capture latch was added to each input BSC to sample the test responses [4], and an early capture control register (ECCR) composed of flip-flops and inverters was additionally required to test interconnects operated by different system clocks [5]. However, extra hardware for each boundary scan cell is required to adopt this method.

Programmable delay logic was introduced to capture the responses early or to update the stimuli later [6], but in practice it is very hard to analyze and implement the delay proposed.

An additional test pin is used to synchronize the test clock

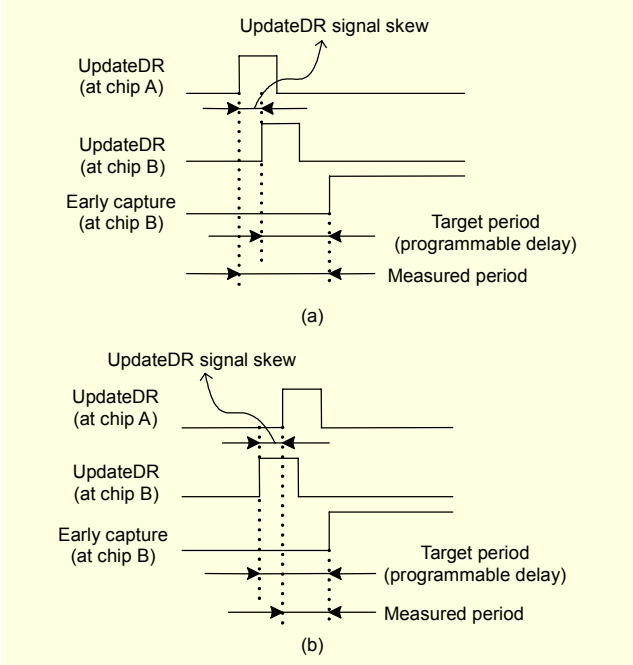

Fig. 3. Clock skew problem on the interconnect delay test.

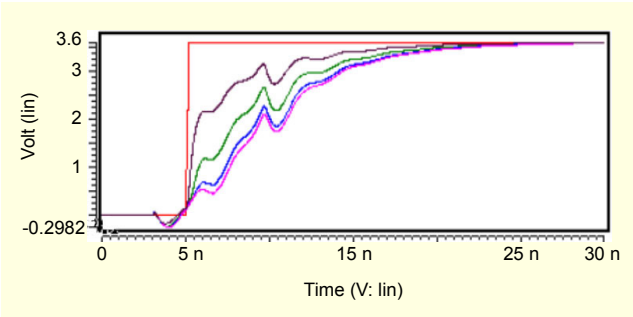

Fig. 4. Skews of UpdateDR control signals.

and system clocks to precisely measure the launching value within a system clock interval, but the incompatibility with the standard may prevent scalability in system boards and SoCs [7].

Without modifying the BSC, a late update and system clock multiplexing technique was proposed, but multiple system clocks and IEEE 1500 wrappers were not addressed [8]. Multi-frequency core wrappers were presented for internal at-speed testing without handling interconnects among cores [9].

Suppose an interconnect delay fault from chip A to chip B is to be tested by implementing the programmable delay logic proposed in [6]. The effects of skews on test control signals are analyzed in Fig. 3. Figure 3(a) shows a case in which the measured duration is longer than the target programmable delay; that is, any interconnect delay may go undetected due to the early update at chip A. A different case can also occur as shown in Fig. 3(b) where the early launching signal from chip A results in a shorter target clock period. Hence, although there is no interconnect delay, testing results may indicate that such a delay does exist. The converse situation also arises. Recently a static timing analysis (STA) tool has been extensively used to synchronize the test signals on an SoC.

Figure 4 shows an example design in which the skews of the IEEE 1149.1 TAP UpdateDR signals on different modules are precisely shown. By adding buffers, the skews can be drastically minimized. Likewise, a similar STA tool can be extensively adopted to reduce clock skews on boards to compensate the optimistic and pessimistic results of [6]. Instead of adding different programmable delay logics to the TAP controller [6], which is in practice hard to attain, this paper introduces a simple IDFT controller whose details will be described in the following section.

III. IDFT on Boards with IEEE 1149.1 Chips

The key feature of our IDFT is to design an edge generator corresponding to each system clock. One board with two chips (chips 1 and 2) operating with the same system clock is introduced in Fig. 5. The IDFT controller transforms the signals from a standard TAP controller to late update UpDR and early capture CapDR with the Delay_EXTEST instruction. Note that the UpDR and CapDR control signals driven by the IDFT controller shown in Fig. 5 are different from the UpdateDR and ClockDR signals feeding each BSC. The output BSCs launch test stimuli at the rising edge of UpDR and the input BSCs capture the test responses exactly one system clock later by the CapDR signal. Most boards use several different system clocks to optimize the overall system performance [5]. To test delay defects on interconnects operated by different system clocks, the associated test data must be updated and captured only once according to the corresponding system clocks. We adopt the stretched Update-DR state [4] to capture the responses early, but instead of changing each BSC, we attach a simple controller to the TAP controller for each system clock to provide modified update and capture control signals to the BSCs.

An IDFT controller described as a finite state machine in Fig. 6 has to perform update and capture operations during the Update-DR state and it must perform a shift operation during the Shift-DR state. Figure 7 shows the detailed architecture of the IDFT controller. In normal mode, M1 and M2 choose UpdateDR and ClockDR by the IDFT_Mode value of 0. In the Delay_EXTEST mode, the stretched IDFT_UpDR and IDFT_CapDR generated by the system clock, IDFT_Mode, and UpdateDR signals are chosen by the multiplexers. The key

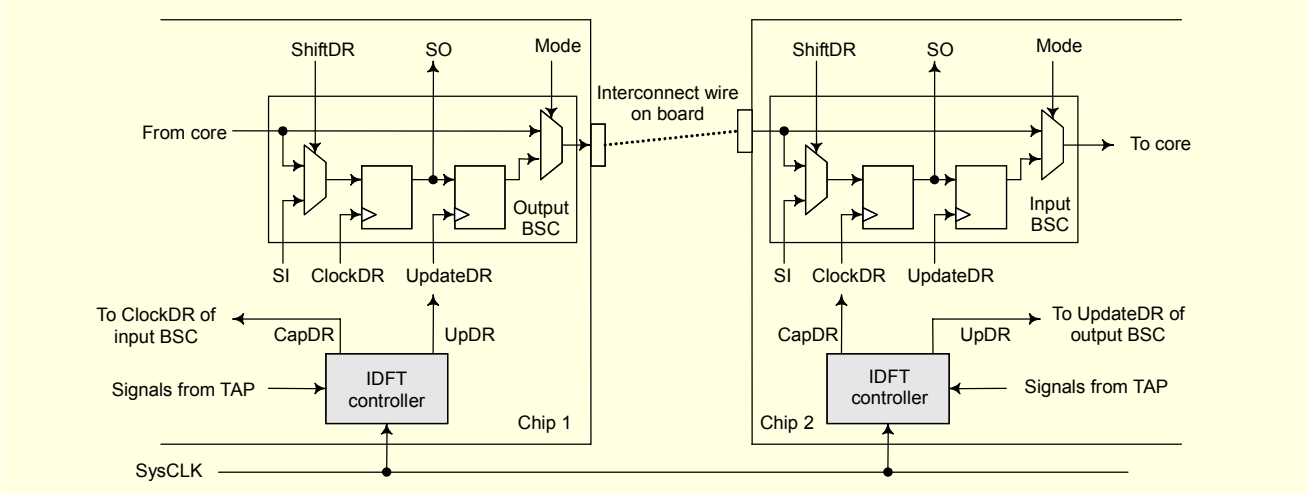

Fig. 5. Interconnect delay fault test using IDFT controller.

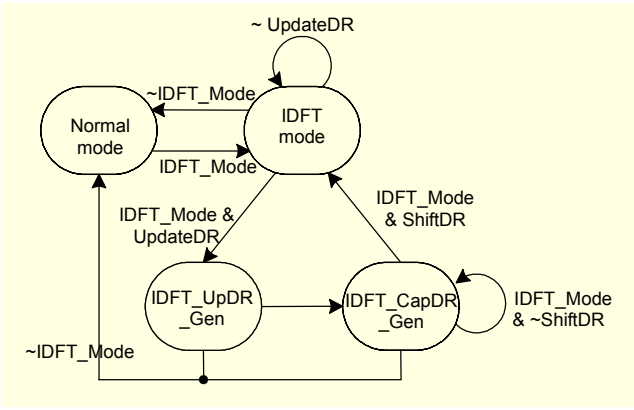

Fig. 6. IDFT controller state diagram

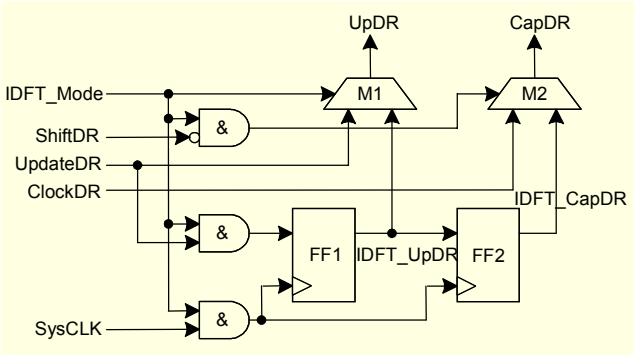

Fig. 7. IDFT controller.

flip-flops are made inactive during normal mode by degating the SysCLK. The operation of our IDFT controller can be described in the following steps.

Step 1. Shift in Delay_EXTEST instruction and IDFT_Mode control signal is set to 1 after decoding.

Step 2. Shift in test stimuli.

Step 3. Stretch Update-DR state to generate IDFT_UpDR and IDFT_CapDR from the UpdateDR, IDFT_Mode, and SysCLK.

Step 4. Shift out test responses.

By applying the above procedures, the timing diagram of Fig. 8 is obtained. The TCK, TMS, UpdateDR, ClockDR and ShiftDR signals are the standard signals from the IEEE 1149.1 TAP controller. During the Update-DR state prolonged by TCK and TMS, UpDR and CapDR are transited from 0 to 1 only once in a SysCLK interval. When the Update-DR state is over, UpDR and CapDR are transited from 1 to 0 at the rising edge of the system clock, and CapDR resumes the test signal of ClockDR during the Shift-DR state.

To test delay defects on interconnects operated by multiple system clocks, only this simple IDFT controller is needed for each system clock. The UpDR and CapDR signals from each IDFT controller are connected to the set of output and input BSCs operated by the corresponding system clock. Figure 9 shows a timing diagram example for IDFT with multiple system clocks. Signals UpDR1 and CapDR1 are the outputs of one IDFT controller operated by SysCLK1, and the signals UpDR2and CapDR2 are the outputs of the other IDFT controller operated by SysCLK2.

During the stretched Update-DR, UpDR1 and CapDR1 are transited from 0 to 1 only once in a SysCLK1 interval, and UpDR2 and CapDR2 are also transited from 0 to 1 once in a SysCLK2 interval. When the Update-DR state is over, signals UpDR1 and UpDR2 are transited from 1 to 0 at the rising edge of each system clock, and CapDR1 and CapDR2 resume the test signal of ClockDR during the Shift-DR state.

The IDFT controller dedicated to each system clock is

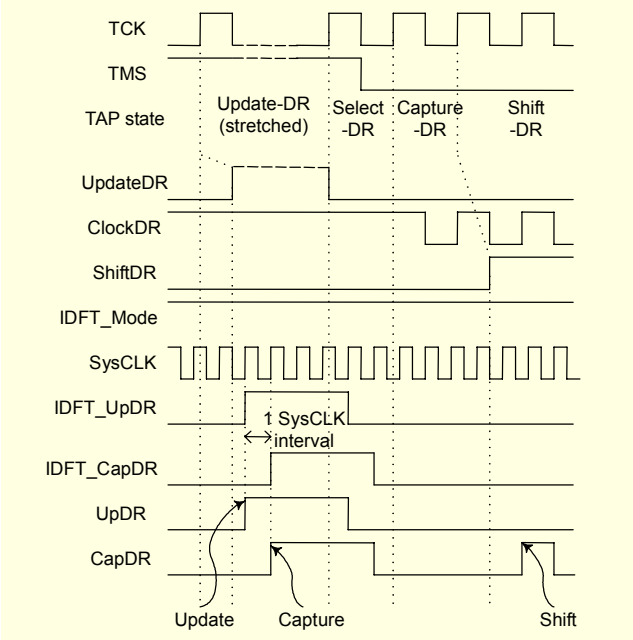

Fig. 8. Timing diagram of IDFT controller.

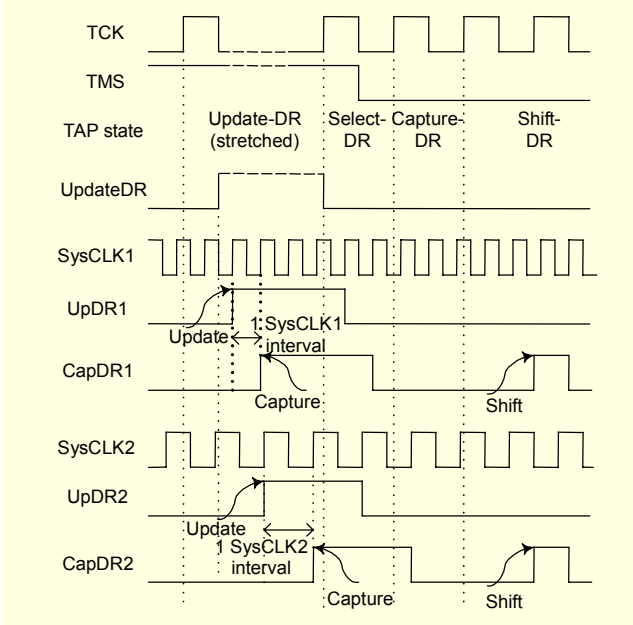

Fig. 9. IDFT timing diagram with multiple system clocks.

designed not to produce any glitch by activating the FF1 and FF2 only upon the IDFT_mode and UpdateDR signals. Even if the test and system clocks are not synchronized, since those clocks are not multiplexed in our scheme, it is fairly unlikely that the clock would include any glitch. Thus, we can have a high level of confidence that any delay defect on interconnect nets operated with different system clocks can be properly tested by tolerating any skew among test and system clocks through our IDFT controller.

IV. IDFT in SoCs with IEEE 1500

1. Interface of IEEE 1149.1 to IEEE 1500

The IEEE 1500 standard is used for testing embedded cores while preserving scalability for hierarchical test access. The IEEE 1500 standard provides a flexible hardware interface between an embedded core and its environment so that the predefined test patterns can be efficiently delivered to and from the embedded core. The core test wrapper standardized by IEEE 1500 has the following features [2].

- It provides the core test, interconnect test, and bypass mode which are subsets of IEEE 1149.1 modes.
- It may connect the core boundary chain (wrapper) to any internal scan chain to perform internal testing of the cores.
- The various modes of the core test wrapper are operated by several control signals in general generated through an SoC TAP controller.

The IEEE 1500 architecture consists of the IEEE 1500 wrapper register, TAM connection, instruction register, and control signals provided externally. The wrapper boundary cell (WBC), wrapper serial input (WSI), wrapper serial output (WSO) and wrapper serial control (WSC) are shown in Fig. 10. In general, control signals to access the WSP are provided by an existing IEEE 1149.1 TAP controller with an interface glue logic as shown in Fig. 11, where some mechanisms are required to select the cores or the chip and to set up, release, and change serial scan paths [2], [10]-[13]. The WRCK is directly connected to the TCK and SelectWIR, whose value of 1 indicates the selection of WIR and 0 the selection of WBR, is connected to the Select signal of the TAP controller. From the WSC signals, the control signals for each WBC are generated by the logic of Fig. 12. When WBCs are selected (SelectWIR= 0),

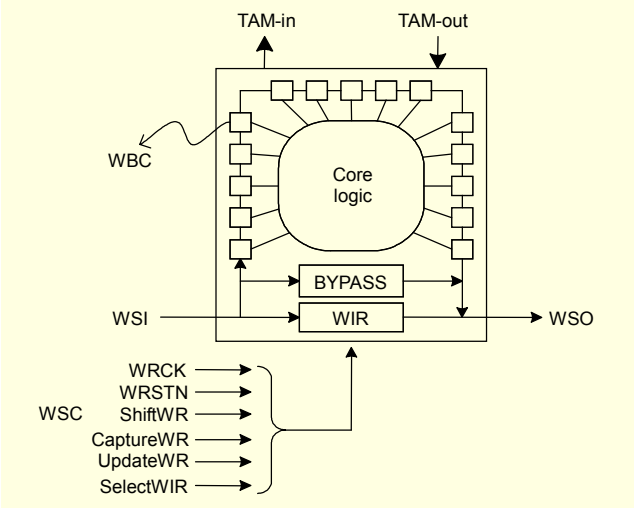

Fig. 10. IEEE 1500 wrapper and WSP.

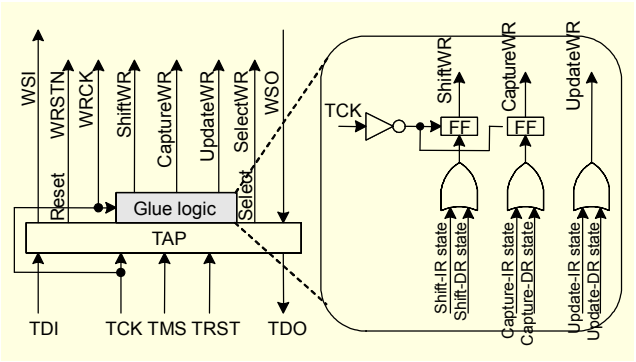

Fig. 11. TAP to WSP glue logic.

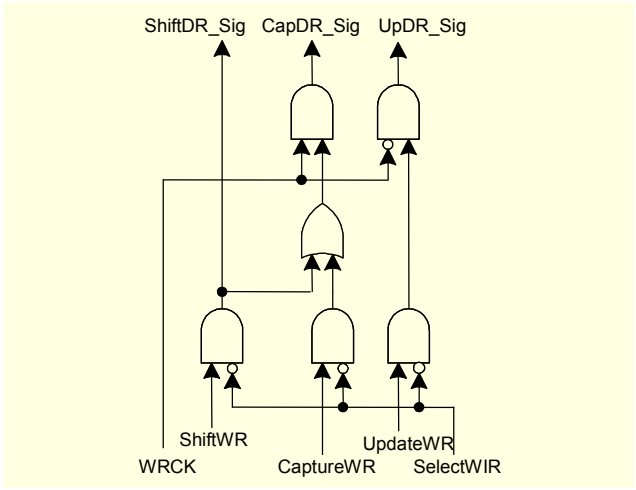

Fig. 12. WSC to WBC logic.

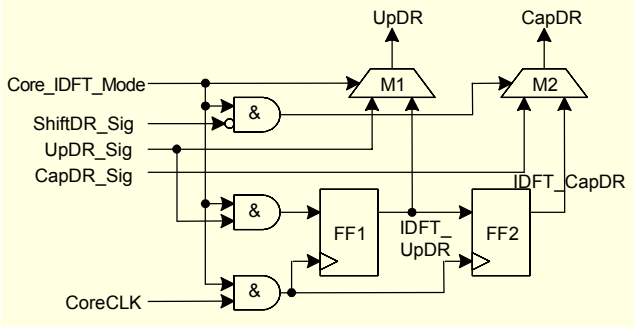

Fig. 13. IDFT controller for IEEE 1500.

this logic enables each WBC to perform the update, shift, and capture functions by passing ShiftWR to ShiftDR_Sig and WRCK to CapDR_Sig and UpDR_Sig, respectively [2], [14].

2. IDFT among IEEE 1500 Wrapped Cores

The IEEE 1500 standard defines various types of WBCs such as single flip-flop types without update storage, two flip-flops types with update storage, and multiple flip-flops types

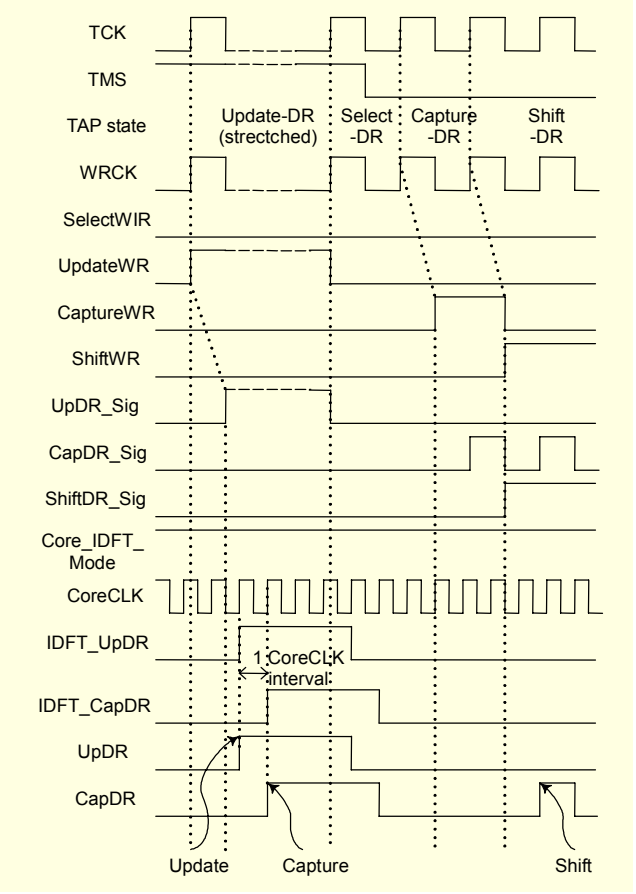

Fig. 14. Timing diagram of IDFT controller.

with a number of shift-path storages [2]. For WBCs without update storage, more complex clock gating and control logics have to be added to support the shift, update, and capture events with a single flip-flop. Therefore, in this paper, we only consider the WBCs with more than one flip-flop, such as the BSC shown in Fig. 5. When an IEEE 1500 wrapped core includes WBCs with update storage, the delay defects on interconnects among IEEE 1500 wrapped cores can be easily tested by implementing our IDFT controller and applying the test procedures described in section III. Figure 13 shows the detailed connections of the IDFT controller to an IEEE 1500 wrapped core. The timing diagram of TAP to WSP interface logic, WSC to WBC logic, and the IDFT controller is precisely described in Fig. 14. If multiple system clocks are involved, the IDFT controller for each system clock is needed.

In summary, interconnects on any SoC with IEEE 1149.1 and IEEE 1500 cores operated by multiple system clocks can be effectively tested with our technique.

V. Design Experiment and Comparative Analysis

An SoC with IEEE 1500 wrapped cores was designed to

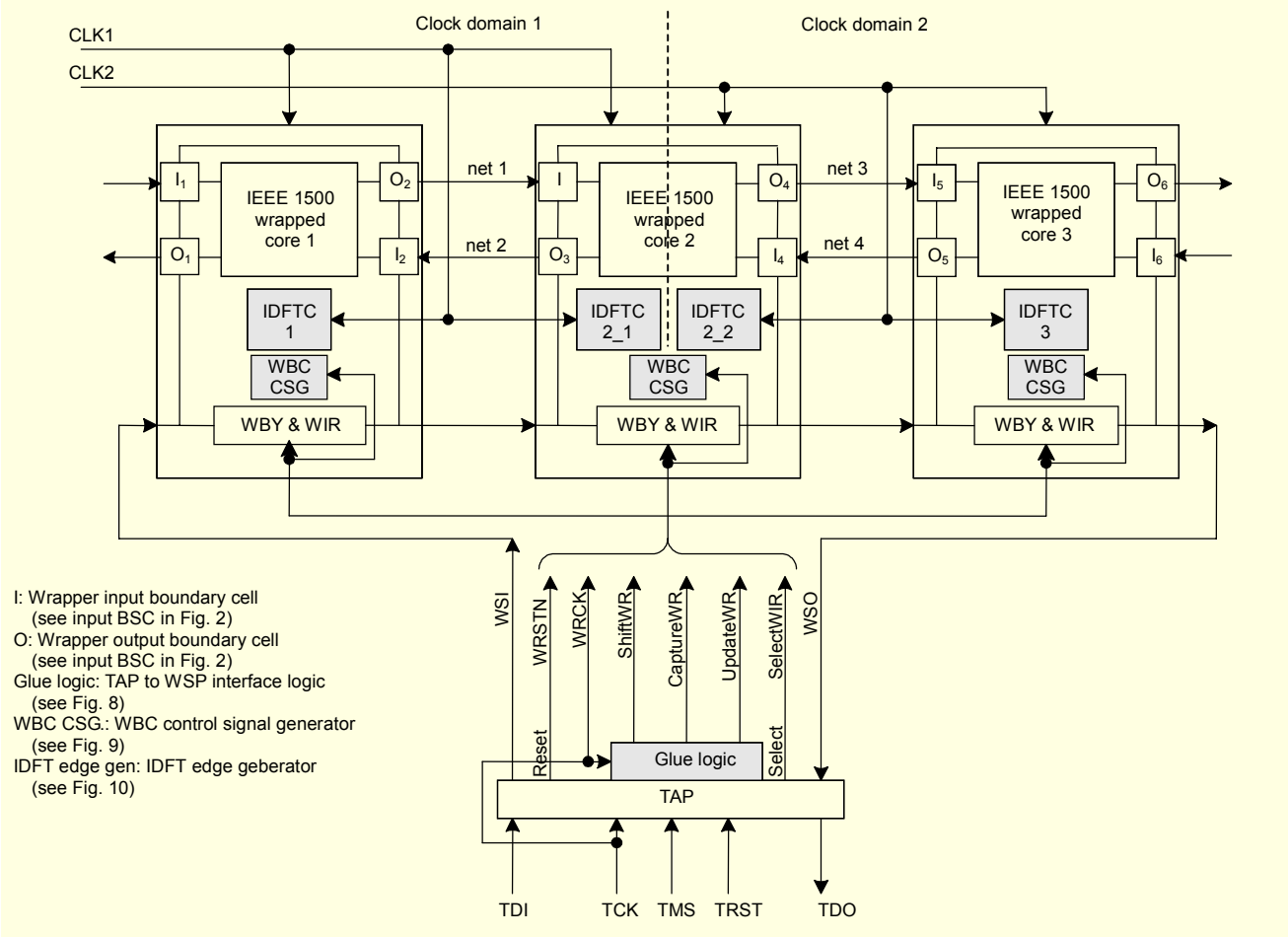

Fig. 15. Example of an SoC with multiple clock domains.

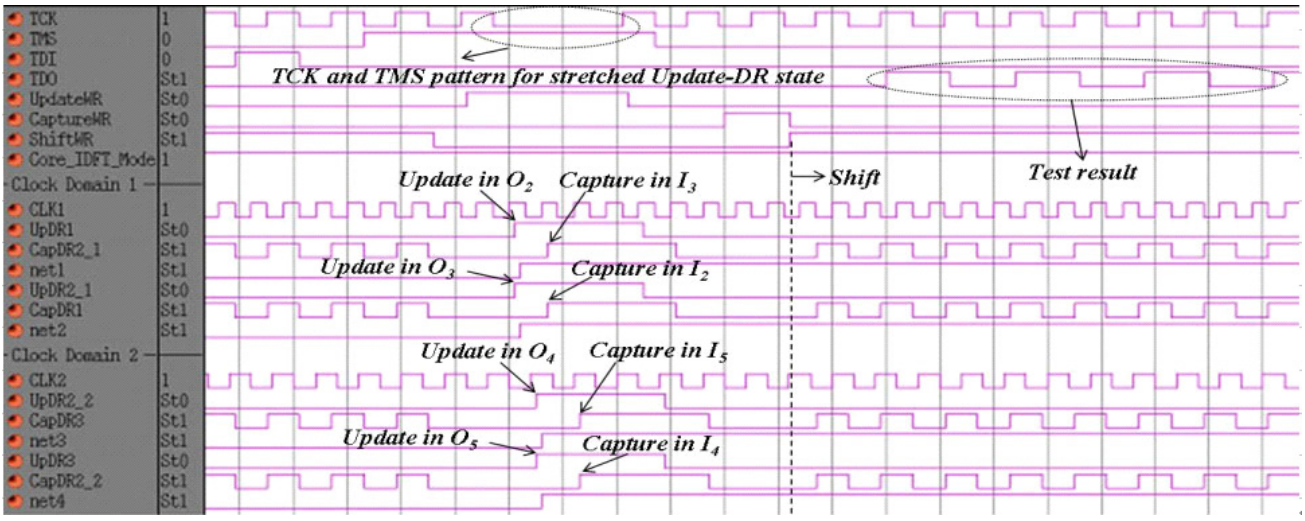

Fig. 16. IDFT simulation results between IEEE 1500 wrapped cores.

verify the operation of IDFT. The block diagram is shown in Fig. 15. Three IEEE 1500 wrapped cores are controlled by an SoC TAP, and two system clocks, CLK1 and CLK2, are adopted. CLK1 interfaces between core 1 and core 2, and CLK2 interfaces between core 3 and core 1. Both core 1 and core 3 have one IDFT controller each. Since core 2 includes

Table 1. Comparison with existing techniques.

	BSC modification	TAP modification	Multiple clock domains	Delay on normal operation
Early capture [4]	R	NR	NS	E
ECCR [5]	R	NR	S	E
Programmable delay [6]	NR	NR	S	NE
Late update [8]	NR	R	NS	NE
IDFTC	NR	NR	S	NE

R: Required NR: Not required S: Supported
E: Exist NE: Does not exist NS: Not supported

Table 2. Comparison of total cell area.

Method \ Chips	TI TMS320 C6713 # of IBSC = 126 # of clk dom. = 3		Intel Xeon processor # of IBSC = 165 # of clk dom. = 4		Intel Pentium III Xeon processor # of IBSC = 284 # of clk dom. = 4	
	TCA	RR	TCA	RR	TCA	RR
Early capture [4]	9429.84	9.25	12348.60	9.08	21255.56	15.64
Programmable delay [6]	1693.44	1.66	2257.92	1.66	2257.92	1.66
IDFTC	1019.52	1	1359.36	1	1359.36	1

two separate parts, one operating at CLK1 and another at CLK2, two IDFT controllers are needed. A benchmark SoC is described as Verilog-HDL and is synthesized through Synopsys with the Samsung 0.18 μm technology library. The clocks for TCK, CLK1, and CLK2 are adopted as 100 MHz, 200 MHz, and 125 MHz, respectively. In Fig. 16, the simulation results show that the patterns for net 1, net 2, net 3, and net 4 among different cores are successfully captured within one system clock cycle.

Compared with previous studies [4]-[6], [8], our IDFT controller fully complies with IEEE standards and supports an SoC with IEEE 1149.1 and IEEE 1500 wrapped cores. Normal operation is not affected by our technique, and no additional power is required by activating the IDFT controller only during the update stage of Delay_EXTEST.

Table 1, where IDFTC denotes our IDFT controller, precisely compares various aspects with other techniques. In Early capture [4], a latch must be added in each BSC adding extra delay during normal operation. ECCR [5] supports multiple clock domains with a BSC similar to early capture. Programmable delay [6] does not require BSC and TAP modification with supporting multiple clocks; however, precise analysis and implementation of the delay is in practice hard to attain. In late update [8], TAP needs to be modified to control the 1.5 TCK late UpdateDR generator and system clock multiplexers. Compared with other techniques, our IDFT has the advantages of not modifying BSC and TAP and supporting multiple clocks without affecting normal operation.

Table 2 shows additional area overhead incurred by using different DFT techniques for three commercial chips. Since the IDFT requires only the wrapper cell information among cores or chips, the results in this table are achieved only by considering the BSCs and TAP control logics. The numbers of input BSCs are 126, 165 and 284. The numbers of clocks are 3, 4, and 4, respectively for the three chips. The TCA indicates the total cell area obtained after synthesizing the chip design by Synopsys Design Analyzer. The TCA of early capture [4] depends only on the number of BSCs, while the TCAs of programmable delay [8] and the proposed IDFT controller depend only on the number of clock domains. For the chip TI TMS320C6713, the area overheads of early capture and programmable delay are 9.25 and 1.66 times higher than our IDFT controller, respectively.

VI. Conclusion

An efficient interconnect delay fault test (IDFT) controller was introduced for boards or SoCs operated by multiple system clocks. The IDFT controller enables the launched test stimuli to be captured by input cells within one system clock cycle and for multiple system clocks. Only the simple IDFT controller is needed for each system clock. Compared with other techniques, our IDFT controller fully complies with IEEE 1149.1 and IEEE 1500 standards, and the system speed and power are not affected.

References

[1] IEEE Computer Society, *IEEE Standard Test Access Port and Boundary Scan Architecture*, June 2001.

[2] IEEE Computer Society, *IEEE Standard Testability Method for Embedded Core-Based Integrated Circuits*, Aug. 2005.

[3] P. Gillis, F. Woytowich, K. McCauley, and U. Baur, "Delay Test of Chip I/Os using Lssd Boundary Scan," *Proc. IEEE Int'l Test Conf.*, 1998, pp. 83-90.

[4] K. Lofstrom, "Early Capture for Boundary Scan Timing Measurements," *Proceedings of IEEE Int'l Test Conf.*, 1996, pp. 417-422.

[5] J. Shin, H. Kim, and S. Kang, "At-Speed Boundary-Scan Interconnect Testing in a Board with Multiple System Clocks," *Proc. Design, Automation and Test in Europe Conf.*, 1999, pp. 473-477.

[6] Y. Wu and P. Soong, "Interconnect Delay Fault Testing with IEEE 1149.1," *Proc. IEEE Int'l Test Conf.*, 1999, pp. 449-457.
[7] B. Nadeau-Dostie, J.F. Cote, H. Hulvershorn, and S. Pateras, "An Embedded Technique For At-Speed Interconnect Testing," *Proc. IEEE Int'l Test Conf.*, 1999, pp. 431-438.
[8] S. Park and T. Kim, "A New IEEE 1149.1 Boundary Scan Design for The Detection of Delay Defects," *Proc. Design, Automation and Test in Europe Conf.*, 2000, pp. 458-462.
[9] Q. Xu, N. Nicolici, "Wrapper Design for Multifrequency IP Cores," *IEEE Trans. Very Large Scale Integration (VLSI) Systems*, vol. 13, June 2005, pp. 678-685.
[10] E.J. Marinissen, R. Kapur, and Y. Zorian, "On Using IEEE P1500 SECT for Test Plug-n-Play," *Proc. IEEE Int'l Test Conf.*, 2000, pp. 770-777.
[11] B.I. Dervisoglu, "A Unified DFT Architecture for Use with IEEE 1149.1 and VSIA/IEEE P1500 Compliant Test Access Controllers," *Proc. Design Automation Conf.*, 2001, pp. 53-58.
[12] L. Whetsel, "Inevitable Use of TAP Domains in SOCs," *Proc. IEEE Int'l Test Conf.*, 2002, pp. 1191.
[13] J. Song and S. Park, "A Simple Wrapped Core Linking Module for SoC Test Access," *Proc. the 11th Asian Test Symp.*, 2002, pp. 344-349.
[14] L. Whetsel, "An IEEE 1149.1 Based Test Access Architecture for ICs with Embedded Cores," *Proc. IEEE Int'l Test Conf.*, 1997, pp. 69-78.

Hyunbean Yi (M'05) received the BS, MS, and PhD degrees in computer science and engineering from Hanyang University, Korea, in 2001, 2003, and 2007, respectively. He worked at Korea Electronics Technology Institute (KETI) as an associate researcher from 2002 to 2007, designing high-speed communication chipsets. Currently, he is a postdoctoral fellow with the Department of Electrical and Computer Engineering at University of Massachusetts at Amherst. His research interests include high-speed communication system design, system-on-chip (SoC) and network-on-chip (NoC) testing, and design-for-testability (DfT). Mr. Yi received the Korea Test Conference 2005 best paper award from the Korea Test Association. He is a member of the Institute of Electronics Engineers of Korea (IEEK); the Korea Information Science Society (KISS); the Institute of Electronics, Information and Communication Engineers (IEICE); and the Institute of Electrical and Electronics Engineers (IEEE).

Jaehoon Song (M'05) received the BS and MS degrees in computer science and engineering from Hanyang University, Korea, in 2000 and 2002, respectively. He worked for the System-on-a-Chip (SoC) Design Center of Seoul National University as a development staff member in charge of platform-based design (2003). He is currently working toward the PhD degree in computer science and engineering from Hanyang University. His main research interests are VLSI testing, signal integrity, and low-power design. Mr. Song is a member of the Institute of Electronics Engineers of Korea (IEEK), the Korea Information Science Society (KISS), and the Institute of Electrical and Electronics Engineers (IEEE).

Sungju Park (M'92) received the BS degree in electronics from Hanyang University, Korea, in 1983, and the MS and PhD degrees in electrical and computer engineering from the University of Massachusetts at Amherst in 1988 and 1992, respectively. He was with the Gold Star Company from 1983 to 1986. He worked for IBM Microelectronics, Endicott, NY, as a development staff member in charge of boundary scan and LSSD scan design from 1992 to 1995. Since then, he has been a professor with the Department of Computer Science and Engineering of Hanyang University. His research interests lie in the area of VLSI testing, including scan design, built-in self test, test pattern generation, fault simulation, and synthesis of test. Additional interests include graph theory and design verification.