

논문 2005-42SD-5-6

# 다중 시스템 클럭과 이종 코어를 가진 시스템 온 칩을 위한 연결선 지연 고장 테스트 제어기

(At-speed Interconnect Test Controller for SoC  
with Multiple System Clocks and Heterogeneous Cores)

장 연 실\*, 이 현 빈\*, 신 현 철\*\*, 박 성 주\*\*

(Yeonsil Jang, Hyunbin Lee, Hyunchul Shin, and Sungju Park)

## 요 약

본 논문은 SoC 상에서 정적인 고장 뿐 아니라 동적인 고장도 점검하고 진단할 수 있는 새로운 At-speed Interconnect Test Controller (ASITC)를 소개한다. SoC는 IEEE 1149.1과 P1500 래퍼의 코어들로 구성되고 다중 시스템 클럭에 의해 동작될 수 있으며, 이러한 복잡한 SoC를 테스트하기 위해 P1500 래퍼의 코어를 위한 인터페이스 모듈과 update부터 capture까지 1 시스템 클럭으로 연결선의 지연 고장을 점검할 수 있는 ASITC를 설계하였다. 제안한 ASITC는 FPGA로 구현하여 기능검증을 하였으며 기존의 방식에 비해 테스트 방법이 쉽고, 면적의 오버헤드가 적다는 장점이 있다.

## Abstract

This paper introduces a new At-speed Interconnect Test Controller (ASITC) that can detect and diagnose dynamic as well as static defects in an SoC. SoC is comprised of IEEE 1149.1 and P1500 wrapped cores which can be operated by multiple system clocks. In order to test such a complicated SoC, we designed an interface module for P1500 wrapped cores and the ASITC that makes it possible to detect interconnect delay faults during 1 system clock from launching to capturing the transition signal. The ASITC proposed requires less area overhead than other approaches and the operation was verified through the FPGA implementation.

**Keywords :** ASITC, at-speed, EXTEST, IEEE 1149.1, P1500

## I. 서 론

최근 SoC 설계 기술이 나노미터 수준으로 발전하면서 SoC의 크기는 점점 소형화되는 반면 복잡도는 크게 늘어났다. SoC를 동작시키는 시스템 속도 또한 점점 빨라지고, 하나의 SoC 안에 여러 시스템 클럭을 사용하게 되었다. 이와 같이, 크게 늘어난 SoC의 복잡도와 속도로

인해 테스트가 보다 어려워지고 있다.

SoC 내에는 다양하고 수많은 코어들이 내장되기 때문에 코어간의 연결선 수 또한 증가하게 된다. 따라서, 정적고장(고착고장 (stuck-at-fault), 개방고장 (open-net-fault), 단락고장 (shorted-net-fault) 등) 테스트뿐 아니라, 고속으로 동작하는 SoC에서는, 연결선 상의 신호 지연으로 인한 시스템 오작동이 큰 문제이기 때문에, 동적고장을 검출해 낼 수 있는 at-speed 테스트가 반드시 필요하다. 그러한 at-speed 테스트는 다중의 시스템 클럭을 사용하는 SoC에서는 연결선에 사용되는 각각의 클럭에 맞춰 at-speed 테스트를 수행할 수 있어야 한다.

칩이나 코어의 고장을 점검하는 방법 중 IEEE 1149.1 (경계스캔)과 P1500이 있다. 경계스캔은 보드나 시스템에서 칩 테스트를 위한 칩 수준의 래퍼이고<sup>[1]</sup>

\* 정희원, 한양대학교 컴퓨터공학과  
(Department of Computer Science & Engineering,  
Hanyang Univ.)

\*\* 정희원, 한양대학교 전자컴퓨터공학과  
(Department of Electrical Engineering Computer  
Science, Hanyang Univ.)

※ 본 논문은 한국과학재단 특정 기초과제(R01-2003-000-101-50-0)로부터 지원 받아 진행하였습니다.  
접수일자: 2005년3월16일, 수정완료일: 2005년4월20일

P1500은 코어 테스트를 위한 코어 수준의 래퍼이다<sup>12, 31</sup>. SoC는 하나의 칩이므로 경계스캔 회로로 구성되어 있고, 내부의 비메모리 코어는 경계스캔 혹은 P1500 래퍼를 포함하고 있어서 SoC의 내부 코어를 테스트하려면 이들 코어와 SoC 경계스캔 회로와의 인터페이스가 필요하다. 그러나 인터페이스를 개발하여 SoC 내부 코어를 제어하더라도, 경계스캔의 근본적인 문제로 인해, 이를 이용하여 at-speed 테스트를 수행하는 것은 매우 어려운 일이다.

본 논문은 SoC 내 P1500 래퍼의 코어를 칩 수준에서 테스트할 수 있도록 P1500 래퍼 제어 신호를 내보내주는 인터페이스 모듈인 Wrapper Interface Port Controller (WIP Controller) 제안하고, 경계스캔의 Test Access Port (TAP) 제어를 이용하여 정적고장 뿐만 아니라 다중 시스템 클럭을 사용한 SoC에서 연결선의 지연고장까지 점검하고 진단할 수 있는 At-speed Interconnect Test Controller (ASITC)를 제안한다.

본 논문의 II장에서는 경계스캔과 P1500에 대해 좀더 상세히 설명하고, III장에서는 경계스캔에서의 at-speed 테스트에 대한 문제점과 기존 연구의 문제점에 대해서 이야기한다. IV장에서는 본 논문에서 제안하는 WIP Controller와 ASITC를 소개하며, V장에서는 검증 결과 및 기존 연구와의 성능을 비교하고, 마지막으로 VI장에서는 결론을 기술한다.

## II. IEEE 1149.1과 P1500

IEEE 1149.1은 칩의 입출력 판에 전용 경계스캔 레지스터를 사용하여 칩 테스트 패턴을 직접 주입 및 관측할 수 있게 하는 보드수준에서의 테스트 설계 기술이다. 90년대에 경계스캔이 업계의 표준으로 선택된 후부터 비메모리 칩에 널리 내장되고 있다.

경계스캔의 구조는 그림 1과 같이 Test Access Port(TAP), TAP 제어기, 명령어 및 각종 테스트 데이터 레지스터로 구성되어 있다.

TAP은 경계스캔에서 필요로 하는 칩의 핀으로 Test Data Input (TDI), Test Data Output (TDO), Test Clock input (TCK), Test Mode Select (TMS), Test Reset input\* (TRST\*)이고, TAP 제어기는 각종 레지스터의 동작을 제어하기 위한 신호를 생성한다. 칩의 각 입력 핀은 연계된 경계스캔 셀을 통하여 칩 내부로 신호를 주입하고, 회로 내부에서 생성된 신호는 경계스캔 셀을 통하여 칩 외부로 출력하게 되어 있다.

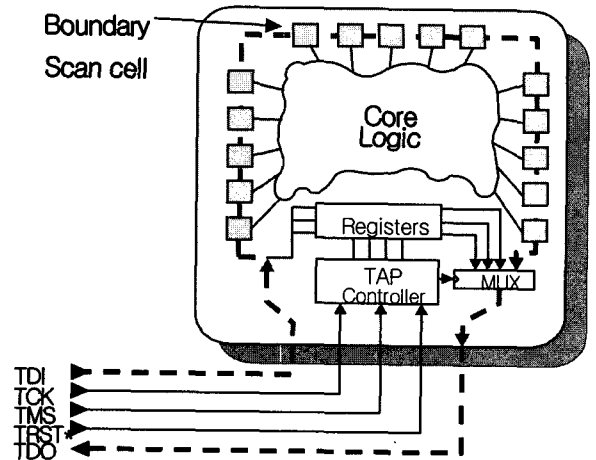


그림 1. IEEE 1149.1의 구조

Fig. 1. Architecture of IEEE 1149.1.

테스트 데이터 레지스터로는 경계스캔 레지스터 이외에도 bypass 레지스터 및 idcode 레지스터 등이 있으며 특수한 용도로 사용하고자 할 경우 사용자의 추가가 가능하다. 명령어 레지스터는 입력되는 명령어에 따라 TDI에서 TDO 경로 사이에 테스트 데이터 레지스터중 해당되는 레지스터를 선택하여 배치해 준다.

경계스캔에서 사용되는 명령어로는 필수 명령어인 BYPASS, EXTEST, SAMPLE/PRELOAD와 선택적인 명령어인 CLAMP, INTEST, RUNBIST 등이 있다. 필수 명령어중 BYPASS는 TDI로 들어오는 데이터를 경계스캔 셀을 통하지 않고 한 클럭만에 바로 TDO로 내보내도록 하는 명령어으로써, 테스트가 필요 없는 칩을 통과하여 테스트 시간을 단축시킬 수 있게 한다. EXTEST는 칩과 칩 사이의 연결선 점검을 위해 사용되는 명령어이고, SAMPLE/PRELOAD는 정상동작 하는 칩의 값을 스캔셀을 통해 쉬프트 시켜 관찰함으로써 칩의 정상동작여부를 살펴 보거나, 정상동작이 수행되는 동안에 미리 경계스캔 셀에 테스트 패턴을 적재할 수 있게 하는 명령어 들이다. 경계스캔과 비슷한 구조를 가지고 비슷한 기능을 하지만 다양한 코어들간의 테스트를 지원하여 경계스캔방식보다 코어 테스트에 더 적합한 테스트 방법으로 P1500이 제안되었다.

P1500에 의해 정의되는 코어 테스트 wrapper는 다음과 같은 특징을 갖는다.

- \* 경계스캔에서 제공되는 모드의 부분집합으로서 코어 내부 테스트, 연결선 테스트, 그리고 바이패스 모드를 지원한다.

- \* 코어 테스트 래퍼와 코어 내부 스캔 체인을 연결

하여 SoC 내부 테스트 용도로 사용한다 (다양한 Test Access Mechanism (TAM) 지원).

\* 경계스캔의 TAP 제어기같은 제어신호를 주는 모듈이 없다.

그림 3은 P1500 Wrapper 레지스터, TAM 연결부, 명령어 레지스터 및 외부에서 제공받아야 하는 Wrapper Interface Port (WIP)를 보여주고 있다.

WIP는 그림 3에서와 같이 6가지 신호들로 구성되어 있다. WRCK는 WIR (Wrapper Instruction Register), WBY (Wrapper Bypass Register), WBR (Wrapper Boundary Register)을 위한 테스트 클럭이고 WRSTN 은 Wrapper Reset이다. SelectWIR은 WSI (Wrapper Serial Input)와 WSO (Wrapper Serial Output) 상에, 명령어 레지스터 또는 테스트 데이터 레지스터를 선택 하는 신호로 1일 때 WIR이 선택된다. CaptureWR, ShiftWR, UpdateWR은 선택된 레지스터에서 WRCK에

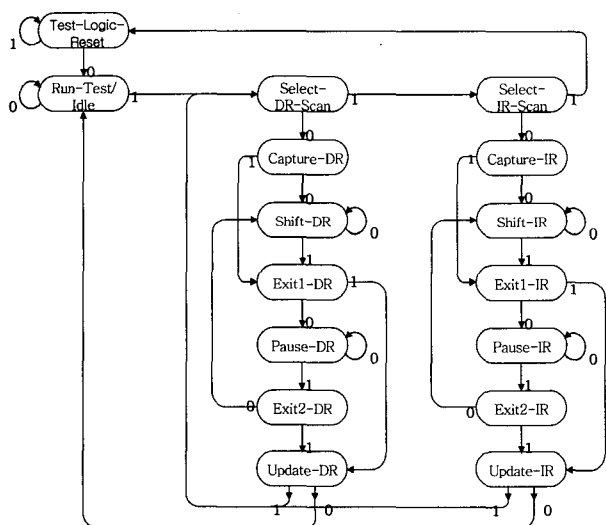


그림 2. TAP 제어기의 상태천이도  
Fig. 2. State Transition Diagram of TAP Controller.

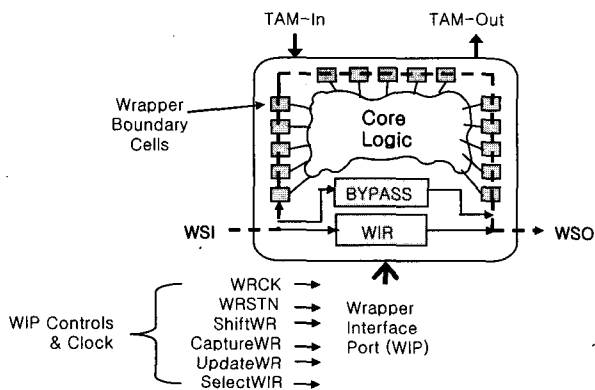


그림 3. P1500의 구조  
Fig. 3. Architecture of P1500.

동기화되어, 각각 capture, shift, update 동작을 수행하도록 한다.

IP(Intellectual Property) 코어를 제공하는 제공자의 선택에 따라 SoC 내에는 경계스캔 래퍼의 코어와 P1500 래퍼의 코어가 동시에 존재 할 수 있다.

### III. IEEE 1149.1에서의 at-speed 테스트에 대한 문제점과 기존의 기술

경계스캔을 사용해 EXTEST를 수행한다면 다음과 같은 순서로 테스트해야 한다.

\* 테스트 패턴 주입 → 연결선의 시작부분 코어의 Update → 연결선 끝부분 코어에서 전달된 테스트 데이터 Capture → 직렬로 쉬프트 하여 결과 관측

이 때, update 동작은 TAP 제어기의 UpdateDR 신호에 의해 일어나고 capture 동작은 ClockDR의 첫 번째 상승 에지 시에 일어난다. UpdateDR 신호는 UpdateDR 상태에서 TCK의 하강에지에 0에서 1로 천이되며, ClockDR 신호는 Capture-DR 상태와 Shift-DR 상태에 TCK를 그대로 받아 각각의 상태로 들어간 후 다음 상태로 천이 할 때 상승에지를 가지게 된다. 따라서, 그림 4와 같이, EXTEST 시에, Update 부터 Capture 까지 2.5 TCK가 소요된다. 정적인 고장을 테스트하기 위해서는 몇 TCK가 소요되어도 문제가 되지 않지만, at-speed 테스트를 위해서는 정상동작시의 시스템 클럭에 따른 지연고장을 점검해야 한다.

이러한 문제를 해결하기 위한 기존 연구 중, Early Capture<sup>[4]</sup>는 원래 석위쳐 있는 코어나 칩의 바운더리 스캔 셀을 변경하여야 하며, 정상 동작 시에도 칩의 출력 신호가 래치를 통과하여 처리속도가 저하된다. 그리고 보드 수준에서의 칩 테스트를 위한 것이기 때문에 SoC에 적용 시 단순히 경계스캔 래퍼의 코어만 있을

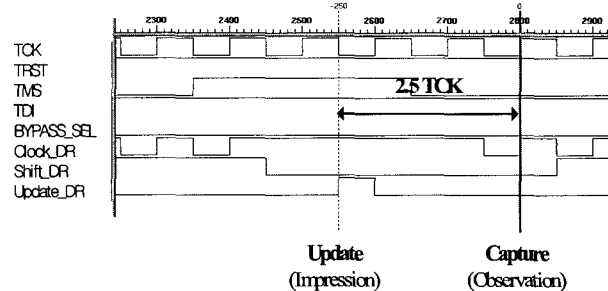


그림 4. IEEE 1149.1 EXTEST 시뮬레이션 파형  
Fig. 4. Waveforms of TAP Controller for EXTEST.

때에 사용이 가능하다. Late Update<sup>[5]</sup> 역시 경계스캔 래퍼의 코어에서만 사용 가능하고 at-speed 테스트를 수행하기 위해 새로운 TMS 테스트 패턴을 알아야 하는 문제점과, 경계스캔과 완전히 호환되지 못하고 있는 문제점을 지니고 있다. At-speed Boundary-scan Interconnect Testing<sup>[6]</sup>은 다중 시스템 클럭을 지원하지 않지만 이 방식은 모든 입력 셀을 변경하고 ECCR을 추가하여야 하며, at-speed 테스트를 위해 ECCR을 셋업한 후에나 테스트를 수행할 수 있으므로 테스트 시 번거로운 점이 있다. 그리고 역시 경계스캔 래퍼의 칩이나 코어에만 적용가능하다.

#### IV. At-speed Interconnect Test Controller

##### 1. At-speed Interconnect Test Controller(ASITC)

SoC에 내장된 코어간 연결선의 at-speed 테스트를 위해서는 update된 신호를 1 시스템 클럭 (SCK) 이내에 capture 해야 한다.

제안한 ASITC는 at-speed 테스트 시에는 기존의 TAP 제어기에서 나오는 UpdateDR 신호와 ClockDR 신호 대신 그림 5의 late\_update\_DR 신호와 sck\_DR 신호를 각각 ASITC의 UpdateDR, ClockDR 신호로 출력하여 update된 신호를 1 SCK 이내에 capture 할 수 있도록 해야한다.

그림 6이, 제안한 ASITC의 전체 블록 다이어그램이다. SCK\_DR\_GEN은 at-speed 테스트를 위한 sck\_DR 신호를 생성하고 LUP\_DR\_GEN은 at-speed 테스트를 위한 late\_update\_DR 신호를 생성한다.

ASITC\_EN\_GEN은 ASITC의 동작을 제어하는 신호들을 생성한다. ASITC\_EN\_GEN에서 생성되는 신호는 cap\_EN과 updr\_EN 신호로, 두 가지 신호에 대해 자세

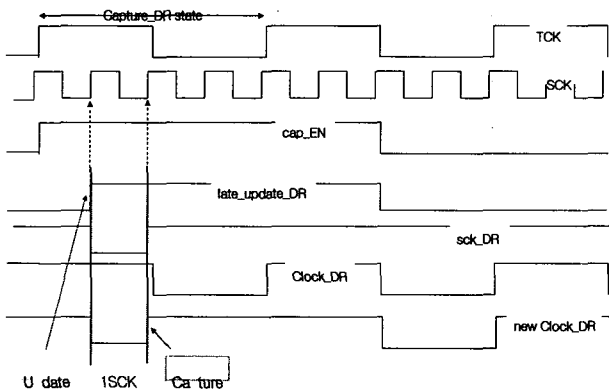


그림 5. ASITC의 목표 파형  
Fig. 5. Waveforms of ASITC.

히 말하면 다음과 같다.

\* cap\_EN : 기본 0으로 출력되다 at-speed 테스트 시에 Capture\_DR 상태에서부터 바로 그 다음 상태의 하강에지까지 1이 출력된다.

· 역할 :

- ① SCK\_DR\_GEN 블록과 LUP\_DR\_GEN 블록 제어
- ② TAP 제어기의 Clock\_DR 대신 SCK\_DR\_GEN의 sck\_DR을 ASITC의 Clock\_DR로 선택

이 때, Update-DR 상태를 거친 후의 Capture-DR 상태만 enable되도록 하여 테스트 패턴 입력 시의 Capture-DR 상태에서는 불필요한 동작이 일어나지 않도록 하였다.

\* updr\_EN : 기본 0값을 가지다가 at-speed 테스트 시에 테스트패턴이 입력된 후에 Update\_DR 상태부터 Capture\_DR 상태까지만 1값을 가진다.

· 역할 :

- ① TAP 제어기의 Update\_DR 대신 LUP\_DR\_GEN의 late\_update\_dr을 Update\_DR로 선택

그림 7을 통해 updr\_EN이 Update-DR 상태에 들어

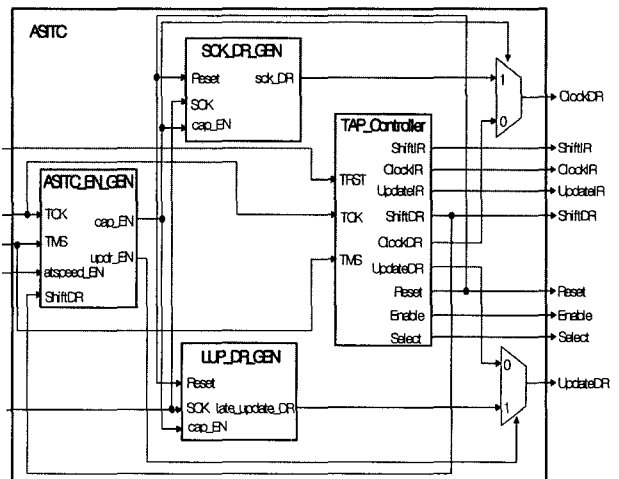


그림 6. ASITC의 전체 블록 다이어그램  
Fig. 6. Block Diagram of ASITC.

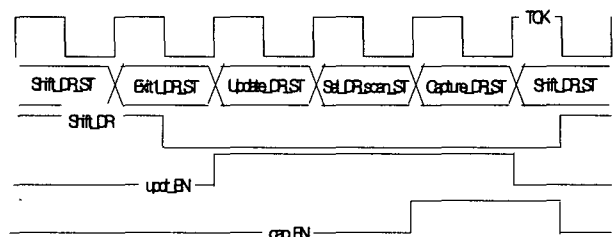


그림 7. updr\_EN과 cap\_EN의 출력 파형도  
Fig. 7. Waveforms of signals updr\_EN and cap\_EN.

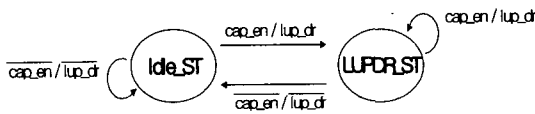


그림 8. LUP\_DR\_GEN의 상태 천이도  
Fig. 8. State Transition Diagram of LUP\_DR\_GEN.

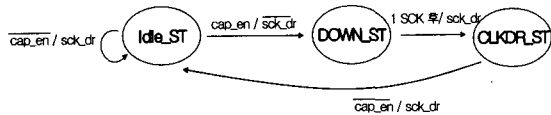


그림 9. SCK\_DR\_GEN의 상태 천이도  
Fig. 9. State Transition Diagram of SCK\_DR\_GEN.

가자마자 1로 됨으로써, 그 이후에 나오는 TAP 제어기의 Update-DR은 무시되고 at-speed 테스트를 위한 late-update-DR이 선택된다. SCK\_DR\_GEN과 LUP\_DR\_GEN 모두 각각 SCK의 상승에지에 상태천이를 하는 유한상태기로 cap\_EN에 의해 초기상태에서 다음 상태로 천이 될 수 있다.

그림 8이 LUP\_DR\_GEN의 상태천이도이고, 그림 9가 SCK\_DR\_GEN의 상태천이도이다.

초기상태에서 계속 cap\_EN의 값을 체크하고 있다가 1이 되면 LUP\_DR\_GEN에서는 그 값을 인식한 첫 SCK의 상승에지에 late\_update\_DR을 1로 내보내고 cap\_EN이 0이 되면 다시 초기상태로 돌아가면서 late\_update\_DR을 0으로 만들어준다. SCK\_DR\_GEN에서는 cap\_EN이 1이 되면 sck\_DR을 0으로 다운시킨 후 1 SCK 후에 1로 다시 출력시켜준다. 이렇게 생성된 신호에 의해 ASITC는 update 동작부터 capture 동작까지 1 SCK 안에 at-speed 테스트를 가능하게 한다.

## 2. Wrapper Interface Port Controller (WIP Controller)

SoC 내의 이중의 코어를 지원하기 위해서는 P1500 래퍼의 코어를 제어하는 신호인 WIP를 칩의 제어부에서 주는 신호를 이용하여 변환해주는 인터페이스 모듈이 필요하다. WIP Controller는 TAP 제어기 뿐만 아니라 ASITC를 사용하여서도 P1500 래퍼를 제어할 수 있도록 9가지 신호를 WIP의 6가지 신호로 변경하여 출력해준다.

그림 10은 WIP Controller의 입출력 신호이다. WIP Controller의 ShiftWR은 ShiftIR와 ShiftDR의 논리합(OR), UpdateWR도 마찬가지로 UpdateIR과 UpdateDR의 논리합(OR)으로 생성하였다. WRSTN은 Reset 신호를 그대로, SelectWIR 신호도 Select 신호를 그대로

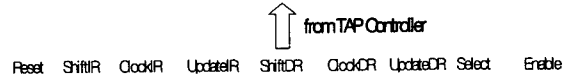
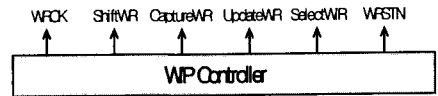


그림 10. WIP Controller의 입출력 신호  
Fig. 10. WIP Controller.

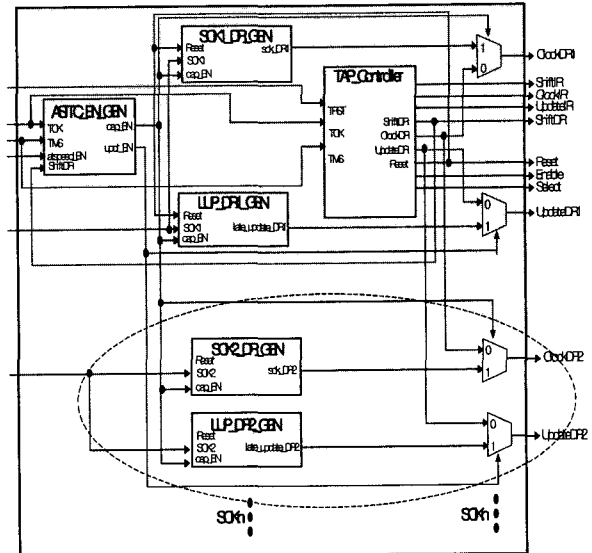


그림 11. 다중 시스템 클럭을 지원하는 ASITC 블록도  
Fig. 11. ASITC Configuration for supporting multiple system clocks.

내보낸다. 그러나 CaptureWR은 ClockIR, ClockDR 신호와는 다르게 클럭 신호가 아니라 capture 상태라는 것을 알려주어야 하는 신호이다. 그래서 CaptureWR은 ClockIR과 ShiftIR, ClockDR과 ShiftDR 신호를 조합하여 기본 0값을 유지하다 Capture-IR 상태와 Capture-DR 상태에만 1이 되도록 하였다. WRCK는 at-speed 테스트 시에도 적용할 수 있도록 Capture-DR 시에는 TCK와 ClockDR 신호를 멀티플렉싱하고 다른 상태에서는 TCK를 그대로 전달되도록 하였다. P1500 래퍼는 CaptureWR로 capture 래치가 동작되지 않고 CaptureWR이 1로 되어 있는 동안 WRCK의 상승에지에 의해 동작되기 때문이다.

## 3. 다중 시스템 클럭을 위한 ASITC와 WIP Controller

그림 11은 ASITC를 다중 시스템 클럭일 경우 확장한 전체 블록 다이어그램이다.

SCK의 수만큼 SCK\_DR\_GEN, LUP\_DR\_GEN과 멀티플렉서 두 개만 추가해주면 된다. 회로 추가에 따라 각각의 SCK에 해당되는 ClockDR과 UpdateDR이 생성되고 테스트를 할 때에는 그 신호들을 연결선의 SCK에

맞춰 보내주면 된다. 즉, 타원형안에 있는 회로만 SCK 수에 맞춰 추가해주면 된다. WIP Controller 또한, 추가된 SCK만큼 CaptureWR과 WRCK를 만드는 회로부분만 추가해주면 된다.

### V. 검증 결과 및 기존연구와의 비교

#### 1. ASITC와 WIP Controller의 검증

ASITC와 WIP Controller의 검증을 위해 경계스캔 래퍼의 코어와 P1500 래퍼의 코어가 함께 있는 SoC를 설계하였다. 설계한 SoC에서 경계스캔 래퍼의 코어는 코어 내부의 TAP 제어기를 ASITC로 변경해 제어신호를 줄 수 있도록 하고 P1500 래퍼의 코어에는 칩 ASITC의 신호를 WIP Controller를 통해 전달해 줄 수 있도록 구성하였다.

그림 12의 SoC를 FPGA를 통해 합성하였고 그 시뮬레이션 결과는 그림 13에서 보여준다. 그림 13의 시뮬레이션 결과를 통해 코어2와 코어3의 연결선에 실어준 값이(test pattern) 1 SCK 만에 제대로 전달된 것을 TDO로 출력된 테스트 결과를 보고 알 수 있다. 즉 update부터 capture까지 1 SCK 소요되는 것을 확인할 수 있다. 또한 CaptureWR 신호가 1로 되어 있는 동안 WRCK가 ClockDR과 멀티플렉싱 되어 update 동작 후 1 SCK 만에 제대로 capture 동작을 하는 것을 이 시뮬레이션 파형을 통해 확인할 수 있다. 다중 시스템 클럭을 지원하는 ASITC를 검증하기 위하여 그림 12의 SoC를 코어2와 코어1 사이의 연결선은 SCK1로 동작되고 코어3과 코어1 사이의 연결선은 SCK2로 동작되도록 바꾸어 보았다.

그림 14가 다중 시스템 클럭을 지닌 SoC에서 ASITC와 WIP Controller를 적용하여 실험한 시뮬레이션 결과이다. 시뮬레이션 결과를 통해 각각의 SCK에 따라서 1 SCK만에 update부터 capture 동작이 일어나는 것을 확인할 수 있다.

이상의 검증을 통하여 제안한 ASITC와 WIP Controller가 다양한 SoC 환경 하에서 at-speed 테스트를 수행하는 것을 확인할 수 있었다.

#### 2 기존 연구와의 비교 분석

기존 연구와 본 논문에서 제안하는 ASITC의 주요 차이점이 표 1에 요약되어 있다.

기존에 연구되었던 방식들에 비해 ASITC는 경계스캔과 완벽하게 호환되고 경계스캔과 P1500 모두를 지

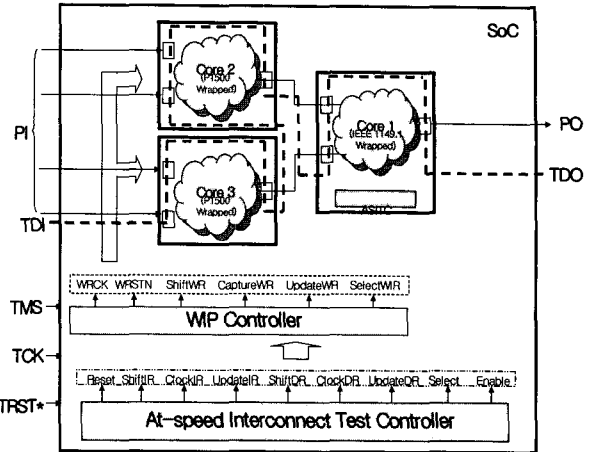


그림 12. 이종 코어가 내장된 SoC에서 ASITC와 WIP Controller

Fig. 12. ASITC and WIP Controller with Heterogeneous Cores.

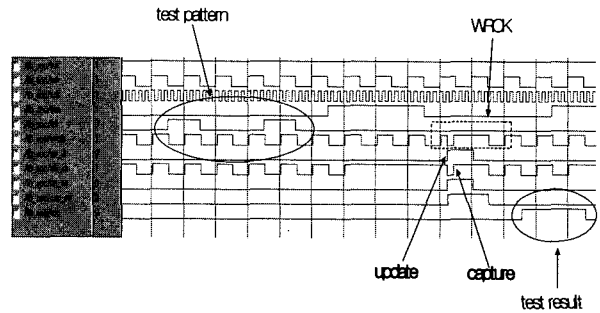


그림 13. ASITC의 시뮬레이션 파형도

Fig. 13. Simulation Result of ASITC.

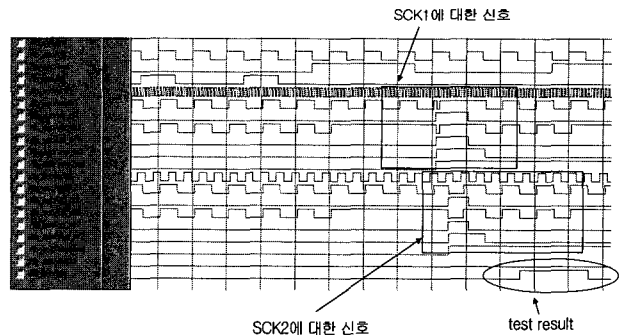


그림 14. 다중 시스템 클럭을 지원하는 ASITC의 시뮬레이션 파형도

Fig. 14. Simulation result of ASITC with Multiple System Clock.

원하여 이종코어가 있는 SoC에서 테스트를 수행할 수 있다. 코어의 정상 동작 시에도 방식 1과 3은 주입력이 셀에 추가된 래치를 계속해서 통과해야 하므로 칩의 처리속도를 저하시키지만 ASITC는 정상 동작 시에는 전혀 영향을 끼치지 않는다. 또한 방식 1과 3 모두 at-speed 테스트를 위해 기존의 셀을 변경하여야 하지

표 1. 기존 연구와 ASITC의 비교  
Table 1. Comparison 1.

	IEEE 1149.1 Compliance	P1500 support	No Effect to Normal mode	multiple SoC support	Key Changes
Method 1[4]	x	x	x	x	BS cells
Method 2[5]	x	x	o	x	Only TAP Controller
Method 3[6]	x	x	o	o	BS cells & ECCR
ASITC	o	o	o	o	Only TAP Controller

표 2. ASITC와 기존 방식과의 비교  
Table 2. Comparison 2.

	Xeon Processor (INTEL)	TMS320DM642 (TI)	TMS320C6713 (TI)	Pentium III 0672 (INTEL)
셀 수	304개	363개	363개	457개
기존 TAP + 추가 gate 수 (총 gate size)	377+1824 (2201)	377+2178 (2555)	377+2178 (2555)	377+2742 (3119)
ASITC	608 (72% area recuction)	608 (76% area recuction)	608 (76% area recuction)	608 (80% area recuction)

만 ASITC는 SoC에 있던 기존 TAP 제어기에 ASITC의 다른 블록들을 붙여 ASITC로 만들어주지만 하면 된다. 상용되는 SoC는 많은 수의 래퍼 셀을 가지므로 래퍼 셀 자체를 변경하는 방식은 면적 면에서 큰 오버헤드를 가지게 된다.

표 2가, 상용되고 있는 INTEL과 TI의 SoC에서 셀 변경 방식과 ASITC 방식을 적용하였을 경우, 추가되는 면적을 보여주고 있다.

기존 TAP 제어기는 377 gate size를 가지며 ASITC는 총 608 gate size를 가진다. 만약 셀을 변경하여야 한다면 SoC의 모든 셀을 변경하여야 할 것이고 그에 따른 추가 면적은 셀 수가 많아질수록 급격하게 증가한다. 표에 나타난 추가 gate 수는 셀에 멀티플렉서가 추가될 경우의 gate size이며 추가된 회로가 래치나 플립플롭이면 배 이상의 gate size가 추가된다. 표에 나타난 결과를 통해 ASITC가 셀 변경 방식에 비해 훨씬 적은 면적의 오버헤드를 가졌음을 알 수 있다.

만약 P1500 래퍼의 코어를 위해 WIP Controller를 추가한다 하더라도 단지 46 gate size를 가지기 때문에 이 때에도 ASITC 방식이 셀 변경 방식에 비해 SoC의 전체 면적을 현격히 감소시킬 수 있다.

## VI. 결 론

본 논문에서 제안한 ASITC는 SoC에 직접 적용 시

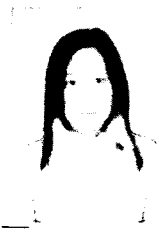
다른 부분은 변경할 필요가 없이 최소한의 회로로 구성할 수 있다. 또한 SoC의 TAP 제어기를 그대로 이용하여 ASITC를 그리고 경계스캔과 호환성을 유지하여 쉽게 테스트를 수행할 수 있고 WIP controller를 함께 제안하여 이중코어가 존재하는 SoC에서도 테스트를 수행할 수 있도록 하였다. 나아가 다중 시스템 클럭을 가진 SoC에서도 ASITC를 쉽게 확장하여 사용할 수 있게 하여, 다중 시스템 클럭을 사용한 SoC에서도 at-speed 테스트를 수행할 수 있도록 하였다.

코어의 재사용, 정적인 고장 테스트 및 코아간 연결선의 at-speed 테스트까지 할 수 있으므로 테스트 비용 면에서도 효율적인 ASITC는 이중 코아, 다중 시스템 클럭을 가진 SoC 즉 다양하고 복잡한 SoC를 테스트하는데 있어 많은 실질적인 도움이 될 것이라 확신한다.

## 참 고 문 헌

- [1] IEEE Standard 1149.1-2001, "IEEE Standard Test Access Port and Boundary-Scan Architecture," IEEE, June 2001.
- [2] IEEE P1500 SECT Wep site. <http://grouper.ieee.org/groups/1500/>.
- [3] ERIK JAN MARINISSEN et al., "On IEEE P1500's Standard for Embedded Core Test," JOURNAL OF ELECTRONIC TESTING, pp. 365-383, 2002.
- [4] K. Lofstrom, "EARLY CAPTURE FOR BOUNDAR SCAN TIMING MEASUREMENTS," Proceedings of IEEE International Test Conference, pp.417-422, 1996.
- [5] S Park and T Kim, "A New IEEE 1149.1 BOUNDARY SCAN DESIGN FOR THE DETECTION OF DELAY DEFECTS," Design Automation and Test in Europe Conference, pp.458-462, 2000.
- [6] Jongchul Shin, Hyunjin Kim and Sungho Kang, "At-speed Boundary-Scan Interconnect Testing in a Board with Multiple System Clocks," Design Automation and Test in Europe Conference, 1999.

저 자 소 개



장 연 실(정회원)  
 2003년 한양대학교 전자컴퓨터  
 공학 학사.  
 2005년 한양대학교 컴퓨터공학과  
 석사.  
 2005년~현재 삼성전자 System  
 LSI 사업부

<주관심분야 : SoC 테스트, ASIC 설계.>



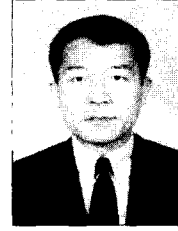
이 현 빈(정회원)  
 2001년 한양대학교 전자컴퓨터  
 공학 학사.  
 2003년 한양대학교 컴퓨터공학과  
 석사.  
 2003년~현재 한양대학교 컴퓨터  
 공학과 박사 과정.

<주관심분야 : SoC 테스트, ASIC 설계, 네트워크 시스템 설계.>



신 현 철(정회원)  
 1978년 서울대학교  
 전자공학과 학사.  
 1980년 한국과학기술원 전기 및  
 전자공학 석사.  
 1983년~1987년 U.C. Berkeley  
 Ph.D

1987년~1989년 MTS, AT&T Bell Lab's,  
 Murray Hill N.J., USA,  
 1989년~현재 한양대학교 전자컴퓨터공학부 정교수  
 1997년~현재 IDEC 한양대학교 지역센터 센터장.  
 <주관심분야 : CAD&VLSI, 통신용 반도체 설계,  
 저전력설계.>



박 성 주/교신저자(정회원)  
 1983년 한양대학교  
 전자공학과 학사  
 1983년~1986년 금성사  
 소프트웨어 개발.  
 1992년 Univ. of Massachusetts  
 전기 및 컴퓨터공학과  
 박사졸업.

1992년~1994년 IBM Microelectronics 연구스텝,  
 1992년~1994년 한양대학교 전자컴퓨터공학부  
 정교수.

<주관심분야 : 테스트 합성, Built-In Self Test, Scan Design, ATPG, ASIC설계, 고속 신호처리 시스템 설계, 그래프이론 등.>