

# 다중 보드 접근을 위한 새로운 IEEE 표준 1149.1 Backplane 테스트 확장

임 용 태<sup>†</sup> · 김 현 진<sup>††</sup> · 강 성 호<sup>††</sup>

## 요 약

보드 내 소자들의 테스트를 지원하기 위해 제안된 IEEE 표준 1149.1은 보드 수준에서의 테스트를 완벽히 수행할 수 있게 해준다. 하지만 보드가 시스템에 장착되고 난 후의 시스템 수준에서의 테스트는 여전히 문제로 남아 있다. 이를 해결하기 위해 IEEE 표준 1149.1 시스템 backplane 테스트 확장이 제안되었다. 기존의 IEEE 표준 1149.1 시스템 backplane 테스트 확장은 테스트 데이터를 이동시키기 위해 한 번에 하나의 보드만을 제어하였다. 그 결과 테스트 데이터의 이동에 필요한 시간의 증가로 전체 테스트 시간이 길어졌다. 본 논문에서는 테스트에 소요되는 시간을 줄이기 위해 하나의 IEEE 표준 1149.1 테스트 버스로 두 개의 보드를 동시에 제어할 수 있는 새로운 IEEE 표준 backplane 테스트 확장 프로토콜을 제안한다. 새로운 프로토콜에서는 두 개의 보드를 동시에 접근하여 테스트 데이터를 이동시킬 수 있으므로 시스템 수준에서의 보드 테스트에 필요한 시간이 대폭 감소될 수 있다. 또한 새 알고리즘에 기초한 보드 연결부(Board Link Unit Port: BLU Port) 포트는 적은 트랜지스터의 사용으로 구현될 수 있으므로 적은 테스트 비용으로 효율적인 시스템 수준에서의 보드 테스트를 수행할 수 있다.

## A New IEEE Standard 1149.1 Backplane Test Extension for Multiple Board Accesses

Yong Tae Yim<sup>†</sup> · Hyun Jin Kim<sup>††</sup> · Sungho Kang<sup>††</sup>

## ABSTRACT

The IEEE standard 1149.1, which was proposed to support to the test of elements within the boards, makes it possible to perform the board level tests. But the problems of the system testing when the boards are equipped to the system, still remain. To solve these problems, IEEE standard 1149.1 system backplane test extension was proposed. The traditional IEEE standard 1149.1 backplane test extension controls only one board at a time to shift the test data to the output cell. Therefore, the total test time increases because of the increase of the time required to shift the data. In this paper, we propose new IEEE standard 1149.1 backplane test protocol, in which two boards can be controlled by using a IEEE standard 1149.1 test bus at a time. By introducing the new protocol, the time required for the board tests can be greatly reduced since it is possible to shift the test data by accessing two boards at a time. And the Board Link Unit(BLU) port based on the new protocol can be implemented with the small number of transistors, so it is possible to perform the efficient board tests in system level with the small test costs.

※ 이 연구는 96년도 한국과학재단 연구비 지원에 의한 결과임  
(과제 번호:96-0102-16-01-3).

† 준 회원: LG반도체 System Device 연구소

†† 정 회원: 연세대학교 전기공학과

논문접수: 1997년 8월 27일, 심사완료: 1997년 12월 15일

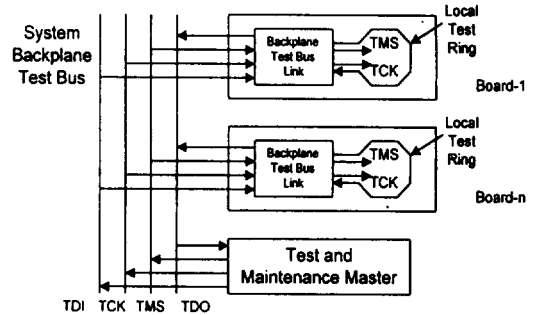
### 1. 기존의 시스템 수준의 보드 테스트 방법

전통적으로 인쇄 회로 기판(printed circuit board: PCB)의 테스트에는 회로 기판 내 소자의 입력 핀에 탐침(probe)을 이용하여 입력 신호를 인가하고 출력 핀에서 응답 신호를 역시 탐침을 사용하여 테스트하는 인서킷(in-circuit) 테스트 방법이 사용되어 왔다[1]. 그러나 최근 VLSI 설계 기술의 발달로 인한 칩의 소형화와 하나의 칩에 보다 많은 기능을 포함함에 따라 칩의 입출력 핀이 증가하게 되었다. 이에 따라 전통적인 인서킷 방법에 의한 테스트는 사실상 불가능해졌다. 또 각각의 특성에 맞는 테스트 입력을 사용해야만 하는 ASIC(Application Specific Integrated Circuit)을 이용한 설계가 증가함에 따라 인서킷 방법에 사용되던 표준 테스트 입력의 합성에 의한 테스트 방식은 더 이상 사용될 수 없게 되었다. 이러한 문제를 극복하기 위해 보드 내의 각 소자의 입출력 핀에 주사 이동 레지스터(scan shift register)를 첨가하고 이들을 하나의 주사 체인(scan chain)으로 연결하여 각 소자들의 제어 가능성(controllability)과 관측 가능성(observability)을 증가시킨 경계 주사 구조(boundary-scan architecture)가 제안되었다[2]. IEEE 표준 1149.1의 등장은 보드 내의 IC 소자들의 테스트에 뛰어난 해결책을 제시하였지만 이러한 보드들이 시스템에 장착된 경우 시스템 수준에서의 통일된 테스트는 여전히 해결해야 할 문제로 남아 있다.

시스템 수준에서의 보드 테스트 문제를 해결하기 위해 제안된 표준안인 IEEE 표준 1149.5[3]는 복잡한 시스템 테스트를 위해 IEEE 표준 1149.1과는 다른 프로토콜을 사용한다. 따라서 IEEE 표준 1149.1을 지원하는 보드가 시스템에 장착된 경우, 이를 테스트하기 위해서는 IEEE 표준 1149.5와 IEEE 표준 1149.1 사이의 프로토콜 변환이 필요하다[3, 4]. 하지만 이러한 연구들은 모두 IEEE 표준 1149.5 테스트 버스를 기준으로 하기 때문에 단순한 IEEE 표준 1149.1 보드의 테스트만을 수행할 경우 테스트 비용의 증가가 너무 크다.

IEEE 표준 1149.1의 프로토콜만을 사용하여 적은 비용으로 시스템에 장착된 보드를 테스트하는 방법들이 제안되었는데 이를 IEEE 표준 1149.1 backplane 테스트 확장이라고 한다[5-7]. 기본적인 backplane 테

스트 확장 구조를 (그림 1)에 나타내었다. 테스트 유지 제어기(Test and Maintenance Master)는 BT-Link 부(Backplane Test Bus Link Unit)를 이용하여 각 보드를 시스템 테스트 버스에 연결하여 필요한 테스트 작업을 수행한다. Backplane 테스트는 TDI 버스를 통해 연결 요청 신호(link request signal)와 주소(address)를 전파하고 해당 주소를 가진 보드로부터 연결 응답 신호(link acknowledge signal)를 받아서 시스템에 장착된 보드를 테스트 버스에 연결하여 일련의 테스트 작업을 수행하는 것이다.



(그림 1) IEEE 표준 1149.1 backplane 테스트 확장 (Fig. 1) IEEE standard 1149.1 backplane test extension

[5]에서는 보드가 Test-Logic-Reset 상태에 있을 때 TDI 버스를 통해 연결 요청 신호를 주소와 함께 전파하여 보드의 테스트를 수행하였다. 하지만 이 경우에는 테스트를 위한 보드를 하나 밖에 선택할 수 없게 되어 보드들 사이의 연결 테스트를 수행할 수 없다는 단점이 있다. 다시 말하면, 테스트 유지 제어기는 연결하고자 하는 보드가 Test-Logic-Reset 상태에 있을 때만 연결 요청 신호를 TDI 버스를 통해 전파할 수 있다. 하지만 보드들 사이의 연결을 테스트하기 위해서는 첫 번째 보드를 연결한 상태에서 다음 보드를 연결하기 위해서는 시스템에 장착된 보드들을 Test-Logic-Reset 상태로 만들어야 하기 때문에 이전에 연결된 보드 역시 Test-Logic-Reset 상태로 천이해야 한다. 따라서 두 개 이상의 보드들 사이의 연결을 테스트할 수 있는 방법이 없다.

[6]에서는 [5]의 이와 같은 문제를 해결하기 위해 shadow protocol[6, 7]이라는 새로운 연결 프로토콜을

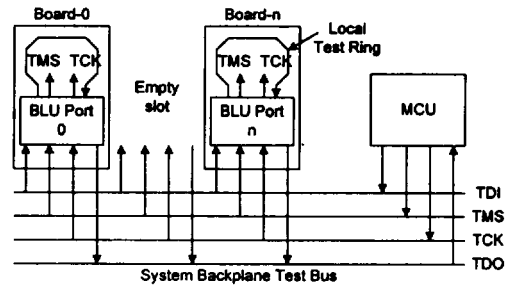
정의하여 사용하였다. Shadow protocol의 기본적인 개념은 보드 상의 IC들이 Test-Logic-Reset 상태뿐만 아니라 Run-Test/Idle, Pause-IR, Pause-DR의 4 가지 무위 상태(idle state)들 중의 한 상태에 있다면, 어떠한 신호가 TDI 버스를 통해 전파되더라도 보드 상의 IC들은 이를 무시할 수 있다는 점을 이용한 것이다. 따라서 [6]에서는 두 개 이상의 보드들 사이의 연결을 테스트하기 위해 첫 번째 보드를 테스트 버스에 연결하여 보드를 Pause-DR 상태로 천이시킨 후, 다음 보드를 연결하여 같은 방식으로 Pause-DR 상태로 천이시킨다. 테스트를 원하는 모든 보드들이 Pause-DR 상태에 도달하고 난 후, 모든 보드들을 일제히 Run-Test/Idle 상태로 천이시켜 연결 테스트를 수행하는 방식을 취하고 있다. 이와 같이 shadow protocol은 보드 사이의 연결 테스트를 가능하게는 하였지만, 한 보드에 대한 테스트 입력과 테스트 결과의 이동이 끝나기 전에는 다른 회로 기관을 시스템 테스트 버스에 연결할 수 없다. 따라서 시스템에 장착된 보드들을 위한 테스트 데이터의 이동에 필요한 시간이 길어져 전체 테스트 시간이 길다는 단점이 있다.

본 논문에서는 이전의 테스트 방법이 지원하는 사항을 모두 충족시키면서 전체 시스템의 테스트에 필요한 시간을 줄일 수 있는 새로운 backplane 테스트 알고리즘을 제안한다. 새 알고리즘에서는 하나의 테스트 버스로 두 개의 보드를 위한 테스트 데이터를 동시에 이동시킬 수 있다. 따라서 전체 시스템 테스트에 필요한 시간을 크게 줄일 수 있다. 새 알고리즘은 IEEE 표준 1149.1을 지원하므로 시스템 수준의 테스트를 위해 프로토콜 변환이 필요 없으며, 보드 상에 장착된 IC들이 IEEE 표준 1149.1을 지원한다면 아무런 수정 없이 이들을 테스트할 수 있다. 따라서 전체 테스트 비용의 상승을 최소화하면서 빠른 시간 내에 시스템에 장착된 보드들을 테스트할 수 있다.

**2. 새로운 시스템 수준의 보드 테스트**

새 알고리즘은 기본적으로 shadow protocol과 유사한 개념에서 접근한다. 하지만 그 내용은 shadow protocol과는 다른 형태를 띠고 있다. 새 알고리즘은 시스템에 장착된 보드들을 접근하기 위해 TAP 제어기 상태도[2]의 6개 안정된 상태인 Test-Logic-Reset, Run-

Test/Idle, Shift-DR, Shift-IR, Pause-DR, Pause-IR 상태를 이용한다. 이는 보드 상의 IC들이 이 6개의 상태들 중 한 상태에 있을 때는 해당 상태를 유지할 수 있도록 TMS 신호를 입력시키고(Test-Logic-Reset 상태의 경우 TMS=1, 그 외의 경우는 TMS=0) 보드들을 테스트 버스에서 연결을 해제하면, TMS 버스에 어떠한 값을 전파시키더라도 보드 상의 IC들의 상태 천이에는 영향을 미치지 않는다는 사실을 이용한다. (그림 2)는 본 논문에서 사용되는 시스템 수준의 테스트를 위한 구조도를 나타낸다. 기본적으로 (그림 1)과 유사한 구조이다. 보드 연결부(board link unit:BLU) 포트는 주 제어부(master control unit:MCU)의 연결 요청 신호에 따라 해당 보드를 테스트 버스에 연결하는 역할을 수행한다. 주 제어부는 테스트 버스에 연결된 보드를 테스트하기 위해 필요한 일련의 작업을 수행한다.



(그림 2) 시스템 backplane 테스트를 위한 구조 (Fig. 2) Architecture for system backplane test extension

**2.1 연결/인터럽트 신호의 구성**

주 제어부는 보드를 backplane 테스트 버스에 연결하기 위해 TMS 버스를 사용한다. 보드를 테스트 버스에 연결하기 위해 사용되는 연결/인터럽트 신호의 구성을 (그림 3)에 나타내었다.

2-bit	1-bit	1-bit	Variable	2-bit
Start	Link/Intr.	Unlink	Address	End

(그림 3) 연결/인터럽트 신호의 구성 (Fig. 3) Configuration of link/interrupt signal

연결을 위한 신호와 인터럽트 신호는 시작(Start)과 끝(End) 사이에 연결/인터럽트 구분 필드(Link/Interrupt), 연결 해제 요청 필드(Unlink), 주소 필드(Address)를 삽입한 프레임 형태로 구성된다. 시작과 끝은 11로 인코딩 된다. 주소 필드의 데이터 인코딩 방법은 <표 1>에 나타낸 것과 같다.

<표 1> 데이터 인코딩  
<Table 1> Data encoding

데이터	MSB	LSB
0	1	0
1	0	1

연결/인터럽트 필드는 0일 경우 연결 요청 신호나 연결 확인 신호임을 나타내고 1일 경우 인터럽트 신호임을 나타낸다. 연결 해제 필드는 0일 경우 BLU 포트는 테스트 버스에 연결된 상태를 유지하고, 1일 경우 현재 TAP 제어기의 상태에 따라 Test-Logic-Reset 상태일 때는 지역 링의 TMS 신호를 1로 유지하고, Run-Test/Idle, Shift-DR, Shift-IR, Pause-DR, Pause-IR 상태일 때는 0으로 유지한 채 테스트 버스로부터의 연결을 해제한다. 주소 필드는 시스템에 장착된 보드들의 주소를 나타낸다. 주소 필드가 모두 0인 경우와 1인 경우는 특수한 경우로서, 주소 필드가 모두 0인 신호가 TMS 버스를 통해 전파될 경우, 시스템에 장착된 보드들은 모두 Test-Logic-Reset 상태로 천이한다. 주소 필드가 모두 1인 신호가 전파될 경우, 시스템에 장착된 보드들은 모두 일제히 테스트 버스에 연결되어 주 제어기의 제어에 따라 동작한다.

2.2 BLU 포트의 연결과 명령어의 이동

기본적인 BLU 포트의 연결은 TMS 버스를 통한 연결 요청 신호의 전파와 연결 확인 신호의 수신으로 이루어진다. 시스템이 처음 부팅 하였을 때 모든 BLU 포트들은 테스트 버스로부터 분리된 상태에 있으며 MCU는 주소 필드를 0으로 채운 연결 신호를 TMS 버스를 통해 전파하여 모든 BLU 포트들이 자신의 TMS 신호를 1로 유지하게 하여 Test-Logic-Reset 상태로 천이시킨다. 각 BLU 포트는 자신의 고유한 주소를 가지고 있다. MCU는 테스트하고자 하는 보드

를 테스트 버스에 연결하기 위해 연결/인터럽트 신호의 주소 필드를 해당 보드의 주소로 설정하고 연결/인터럽트 필드를 0으로 설정한 연결 요청 신호를 TMS 버스를 통해 전파하고 곧바로 해당 보드의 TAP 제어기 상태로 천이한다. 각 BLU 포트들은 연결이 해제된 상태에 있으므로 TMS 버스를 통해 어떠한 신호가 전파되어 오더라도 보드 내의 IC들의 동작에는 영향을 미치지 않는다. TMS 버스로부터 전파된 연결 요청 신호를 수신한 각 BLU 포트들은 수신된 주소를 자신에게 설정된 주소와 일치하는지 비교한다. 만약 자신의 주소와 일치할 경우 해당 BLU 포트는 (그림 3)과 같이 구성된 연결 확인 신호를 TDO 버스를 통해 MCU로 보내고 자신의 주사 경로를 테스트 버스에 연결한다. MCU는 수신된 연결 확인 신호를 통해 연결의 성공 여부를 확인한다. 정확한 보드가 연결됐을 경우 MCU는 명령어의 이동을 위해 연결된 BLU 포트를 Shift-IR 상태로 천이시키고 명령어를 이동시킨다. 명령어의 이동이 끝나고 나면 다음 보드를 위한 명령어의 이동을 위해 현재 연결된 보드의 TAP 제어기 상태를 Pause-IR 상태로 천이시키고 그 상태를 유지하도록 지역 주사 경로의 TMS 입력을 0으로 설정하고 backplane 테스트로부터 연결을 해제한다. 이전과 같은 방식으로 새로운 보드를 연결하여 필요한 명령어를 이동시킨다. 테스트를 원하는 모든 보드들의 명령어를 이동시키고 난 후에는 주소 필드를 모두 1로 설정한 연결 요청 신호를 TMS 버스를 통해 전파하고 MCU는 Pause-IR 상태로 천이한다. 주소 필드가 모두 1인 연결 요청 신호(CAB)를 받은 보드들은 현재 자신의 상태가 Pause-IR 상태일 경우 일제히 테스트 버스에 연결한다. 이 때는 버스 충돌을 방지하기 위해 연결 확인 신호를 보내지 않는다. MCU는 다시 Update-IR 상태로 천이하여 연결된 보드들이 실제로 이동된 명령어를 적재하게 한다. MCU는 다시 Run-Test/Idle 상태로 천이하여 테스트 데이터의 이동을 준비한다.

2.3 테스트 데이터의 이동

하나의 보드만을 테스트할 경우 MCU는 연결 해제 요청 필드를 0으로 설정한 연결 요청 신호를 TMS 버스를 통해 전파하고 연결하고자 하는 BLU의 TAP 제어기 상태로 천이한다. 연결 요청 신호의 주소 필드

와 같은 주소를 가진 BLU 포트는 주소 필드를 자신의 주소로 설정한 연결 확인 신호를 TDO 버스를 통해 MCU로 보내고 자신의 지역 주사 경로를 테스트 버스에 연결한다. 연결 해제 요청 필드가 0으로 설정되어 있으므로 해당 BLU 포트는 테스트 버스에 연결된 상태를 유지한다. MCU는 연결 확인 신호를 받은 후 실제 테스트를 수행하기 위해 천이하여 연결된 보드의 TAP 제어기 상태가 Shift-DR 상태가 되게 하여 테스트에 필요한 데이터를 이동시킨다. 데이터의 이동이 끝난 후 MCU는 다시 천이하여 연결된 보드를 Run-Test/Idle 상태로 천이시키고 이 상태에서 실제 테스트를 수행한다. 첫 테스트가 끝난 후에는 앞의 과정을 반복하여 다음 테스트 입력과 테스트 결과를 이동시켜 테스트를 계속한다.

본 논문에서는 테스트에 소비되는 시간을 줄이기 위해 하나의 테스트 버스로 두 개의 보드를 동시에 연결하여 테스트 데이터를 이동한다. 따라서 하나의 보드만을 테스트할 때보다는 다소 복잡한 과정이 필요하다.

IEEE 표준 1149.1은 IC의 TDI에서의 데이터의 이동은 TCK의 상승 천이에 이루어질 것을, TDO에서의 데이터의 이동은 TCK의 하강 천이에 이루어질 것을 의무화하고 있다. 두 보드의 테스트 데이터가 모두 같은 클럭에서 이동되어야 하므로 기존의 방식만으로는 테스트 버스에서 발생하는 데이터의 충돌을 피할 수 없다. 이를 해결하기 위해 본 논문에서는 두 데이터를 하나의 테스트 버스로 동시에 전송할 수 있는 <표 2>와 같은 데이터 전송 모드를 정의한다.

<표 2> 테스트 데이터의 전송 모드  
<Table 2> Test data transmission mode

전송 모드	수행 작업	클럭 사용
A	테스트 입력의 수신	TCK의 ↑
	테스트 결과의 전송	TCK의 ↓
B	테스트 입력의 수신	TCK의 ↓
	테스트 결과의 전송	TCK의 ↑

두 보드를 위한 데이터를 하나의 테스트 버스를 통해 전송해야 하므로 TCK의 상승 천이와 하강 천이를 모두 사용해야 한다. 모드 A로 동작하는 보드는 자신

의 주사 경로를 테스트 버스에 직접 연결하여 테스트 버스를 통해 MCU로부터 전송되어온 테스트 입력을 곧바로 주사 경로 상의 첫 번째 IC로 전달한다. TDI를 통해 테스트 입력을 전달받은 IC는 TCK의 상승 천이에서 데이터를 이동시킨다. TCK의 하강 천이에서 IC의 TDO로부터 이동되어온 테스트 결과는 곧바로 테스트 버스를 통해 MCU로 전송한다. 따라서 모드 A의 동작은 IEEE 표준 1149.1에서 요구하는 방식대로 동작할 수 있다. 하지만 모드 B로 동작하는 보드는 모드 A로 동작하는 보드와 테스트 버스를 공유해야 하므로 모드 A와는 다른 방식으로 동작해야 한다. 모드 B로 동작하는 보드는 모드 A로 동작하는 보드와는 달리 자신의 주사 경로를 테스트 버스에 직접 연결하지 않는다. 모드 B로 동작하는 보드의 BLU는 내부의 수신 레지스터를 경유하여 주사 경로를 테스트 버스에 연결한다. 내부 레지스터는 1비트 풀림풀림으로 구성되며 TCK의 하강 천이에서 테스트 버스를 통해 전송되어온 테스트 입력을 수신한다. 수신된 테스트 입력은 보드 내의 주사 경로로 전달된다. TCK의 하강 천이에서 IC의 TDI로 전달된 테스트 입력은 뒤따르는 TCK의 상승 천이에서 이동된다. 따라서 B모드로 동작하는 보드 내의 IC들도 IEEE 표준 1149.1의 요구대로 동작할 수 있다. 또한 TCK의 하강 천이에서 이동되어온 테스트 결과를 곧바로 테스트 버스로 전송할 경우 모드 A로 동작하는 보드와 데이터 충돌을 일으키므로 시간적인 차이를 두고 데이터를 전송해야 한다. 따라서 모드 B로 동작하는 보드에서 이동되어온 테스트 결과는 TCK의 상승 천이에서 테스트 버스를 통해 MCU로 전송한다.

MCU는 테스트를 원하는 첫 번째 보드들 테스트 버스에 연결하기 위해 연결 해제 요청 필드를 1로 설정한 연결 요청 신호를 TMS 버스를 통해 전파하고 해당 보드의 TAP 제어기 상태로 천이한다. 연결 요청 신호의 주소 필드와 일치하는 BLU 포트는 MCU의 요청에 따라 연결 확인 신호를 TDO 버스를 통해 전파하고 자신의 주사 경로를 테스트 버스에 연결한다. 이 때 연결된 보드는 모드 A로 동작하므로 연결 확인 신호를 TCK의 하강 천이에 TDO 버스를 통해 MCU로 전송한다. 연결 확인 신호를 받은 MCU는 테스트 데이터를 이동시키기 위해 TAP 제어기의 상태를 Shift-DR 상태로 천이한다. 연결 요청 신호의 연

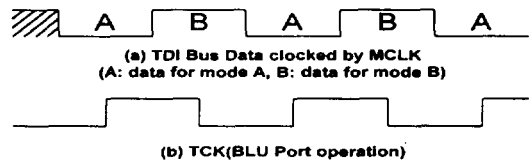
결 해제 요청 필드가 1로 설정되어 있으므로 MCU에 의해 Shift-DR 상태로 천이한 BLU 포트는 현재의 상태를 유지하도록 TMS 신호를 0으로 설정하고 TMS 버스로부터 연결을 해제한다. Shift-DR 상태에서 TDI와 TDO 버스를 통해 보드의 테스트 데이터가 설정된 동작 모드에 따라 전송된다.

첫 번째 보드의 BLU 포트가 Shift-DR 상태에서 TMS 버스로부터 연결을 해제한 후, MCU는 두 번째 보드를 연결하기 위해 주소 필드를 해당 주소로 설정하고 연결 해제 요청 필드를 역시 1로 설정한 연결 요청 신호를 TMS 버스를 통해 전파하고 해당 보드의 TAP 제어기 상태로 천이한다. 전파된 연결 요청 신호의 주소 필드와 같은 주소로 설정된 BLU 포트는 연결 확인 신호를 TDO 버스를 통해 MCU로 보내고 자신의 주사 경로를 테스트 버스에 연결한다. 이 때 현재의 BLU 포트는 B 모드로 설정되어 있으므로 A 모드로 동작하고 있는 보드와의 데이터 충돌을 피하기 위해 TCK의 상승 천이에서 연결 확인 신호를 보낸다. BLU 포트로부터 연결 확인 신호를 받은 MCU는 테스트 데이터의 이동을 위해 TAP 제어기의 상태를 Shift-DR 상태로 천이하고 TCK의 하강 천이에서 BLU 포트가 테스트 입력을 수신할 수 있도록 테스트 입력을 TDI 버스를 통해 전송한다. BLU 포트는 연결 요청 신호의 연결 해제 요청 필드가 1로 설정되어 있으므로 현재의 상태를 유지할 수 있도록 TMS 신호를 0으로 설정하고 TMS 버스로부터의 연결을 해제한다. 테스트 데이터는 자신의 전송 모드에 따라서 테스트 버스를 통해 전송된다.

IEEE 표준 1149.1에 따르는 테스트 버스 제어기들은 TCK의 하강 천이에서 테스트 입력을 TDI 버스를 통해 전송하여 보드 내의 IC들이 TCK의 상승 천이에서 테스트 입력을 이동시킬 수 있도록 한다. 하지만 본 논문에서는 TCK의 하강 천이와 상승 천이를 모두 사용해야 하므로 이러한 방식을 따를 수 없다. MCU는 테스트 입력을 전송하기 위해 TCK의 새로운 클럭을 사용해야 하며 이는 TCK로부터 쉽게 얻을 수 있다. MCU의 데이터 전송 클럭을 (그림 4)에 나타내었다. (그림 4)의 (a)는 MCU의 데이터 전송 클럭인 MCLK에 의해 테스트 입력이 TDI 버스를 통해 전송되고 있는 상황을 나타내고 있다. A와 B는 각각 A 모드로 동작하는 보드와 B 모드로 동작하는 보드

를 위한 테스트 입력을 나타낸다. MCU는 MCLK의 하강 천이에서 A 모드를 위한 테스트 입력을 전송하고 상승 천이에서 B 모드를 위한 테스트 입력을 전송한다. (그림 4)의 (b)는 TCK를 나타낸다. A 모드로 동작하는 보드의 경우에는 TDI 버스를 통해 전송되어 온 테스트 입력이 TCK의 상승 천이에서 IC의 TDI를 통해 이동된다. B 모드로 동작하는 보드의 경우에는 TDI 버스를 통해 전송되어 온 테스트 입력이 BLU 포트의 수신 레지스터를 통해 TCK의 하강 천이에서 수신되고 뒤따르는 상승 천이에서 IC의 TDI를 통해 이동된다.

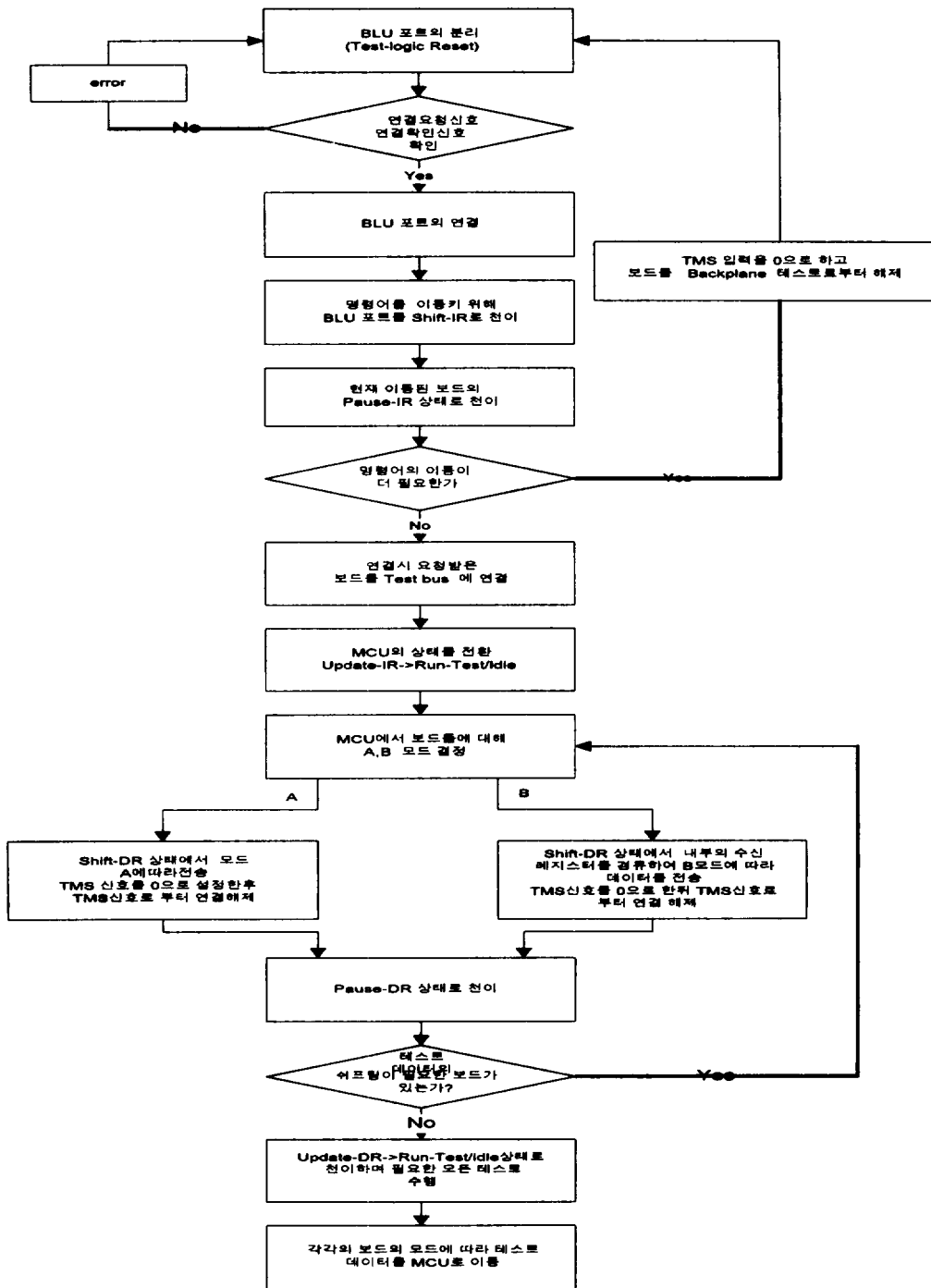
MCU에서 TDO 버스를 통해 테스트 결과를 수신하는 과정도 유사한 방식으로 할 수 있다.



(그림 4) MCU 데이터 전송 클럭  
(Fig. 4) MCU data transmission clock

### 2.4 데이터의 이동이 끝난 BLU 포트의 상태 천이

테스트 데이터의 이동이 끝난 보드는 테스트 수행을 위해 TAP 제어기의 상태 천이를 필요로 한다. 이를 위해서는 MCU가 해당 보드의 데이터 이동이 끝나기 전에 이를 다시 테스트 버스에 연결하는 과정을 필요로 한다. 테스트 버스에 연결할 보드는 연결에 앞서 TDO 버스를 통해 MCU로 연결 확인 신호를 보내야 한다. 하지만 현재 TDO 버스는 두 보드의 테스트 결과를 이동시키기 위해 사용되고 있으므로 연결 확인 신호를 보낼 수 있는 방법이 없다. 이는 보드의 자동 상태 천이를 이용하면 해결될 수 있다. 테스트 데이터의 이동이 끝난 보드의 BLU 포트는 MCU의 연결 요청 신호를 기다리지 않고 자체적으로 TMS 신호를 생성하여 TAP 제어기의 상태를 천이시킨다. 상태 천이에 필요한 TMS 신호는 TAP 제어기의 상태도 [2]를 참조하면 01임을 알 수 있다. 따라서 BLU 포트 내에 필요한 TMS 신호 생성기의 구조는 매우 간단하



(그림 5) BLU 포트의 동작에 대한 플로우 차트  
(Fig. 5) Flow chart about BLU port behaviors

게 구현될 수 있다. TAP 제어기의 상태 천이는 TCK의 상승 천이에서 이루어지므로 상태 천이에 필요한 TMS 신호는 TCK의 하강 천이에서 생성되어야 한다.

테스트 데이터의 이동이 끝난 보드는 데이터를 이동 중인 보드들과 동기를 맞추기 위해 Pause-DR 상태로 천이한다. 이 상태에서 BLU 포트는 자신이 Pause-DR 상태로 천이하였음을 MCU에게 알리기 위해 연결 확인 신호의 연결/인터럽트 필드를 1로 설정한 인터럽트 신호를 TDO 버스를 통해 전송한다. 인터럽트 신호의 의미는 현재 하나의 보드만이 테스트 버스를 사용중임을 알리는 신호로 해석할 수 있다. 인터럽트 신호를 받은 MCU는 다시 새로운 보드를 테스트 버스에 연결하고 필요한 테스트 데이터를 이동시킨다. 이와 같은 방식으로 MCU는 하나의 테스트 버스를 통해 두 개의 보드를 위한 테스트 데이터를 이동시킬 수 있다.

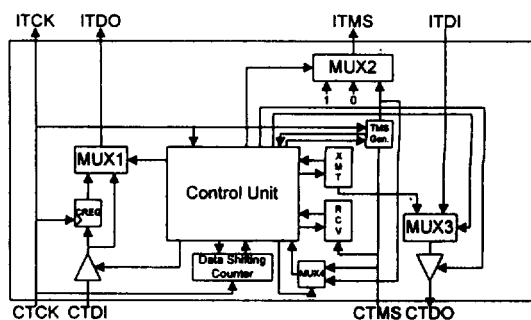
테스트 중인 모든 보드의 데이터 이동이 끝나고 Pause-DR 상태에 도달하면 실제 테스트를 수행해야 한다. MCU는 테스트 수행을 위해 CAB 연결 요청 신호(주소 필드를 모두 1로 설정)를 TMS 버스를 통해 전송하고 Pause-DR 상태로 천이한다. CAB 연결 요청 신호를 받은 보드들은 현재 자신의 상태가 Pause-DR 상태일 경우 테스트 버스에 연결한다. 이 과정에서는 여러 보드들 사이의 버스 충돌을 피하기 위해 연결 확인 신호를 보내지 않는다. MCU는 다시 TAP 제어기를 Update-DR 상태를 거쳐 Run-Test/Idle 상태로 천이시켜 이동된 테스트 입력으로 실제 테스트를 수행한다. 이 과정에서 보드 내의 연결 테스트, 보드 내 소자들의 BIST, 보드들 사이의 연결 테스트 등 필요한 모든 테스트를 수행한다.

(그림 5)는 보드들을 테스트하는 경우의 보드와 MCU의 상태의 천이에 따른 BLU 포트의 동작에 대해 나타낸 플로우 차트이다.

### 3. BLU 포트의 구조

본 논문의 알고리즘에 따라 구현된 BLU 포트의 구조를 (그림 6)에 나타내었다.

BLU 포트는 CTCK, CTDI, CTMS, ITDI로 구성된 네 개의 입력과 ITCK, ITDO, ITMS, CTDO로 구성된 네 개의 출력을 가진다. CTCK, CTDI, CTMS,



(그림 6) BLU 포트의 구조  
(Fig. 6) Architecture of BLU port

CTDO는 각각 TCK 버스, TDI 버스, TMS 버스, TDO 버스에 연결되는 신호들이고, ITCK, ITDI, ITDO, ITMS는 각각 IC의 TCK, TDO, TDI, TMS 입력에 연결되는 신호들이다.

제어부(Control Unit)는 멀티플렉스, 버퍼, 데이터 이동 카운터, 연결 요청 신호 수신부(RCV), 연결 확인 신호 전송부(XMT), TMS 신호 생성기를 위한 제어 신호를 발생시키는 역할을 수행한다. 제어부는 연결 요청 신호 수신부의 데이터 입력과 인터럽트 신호 입력, 연결 확인 신호 전송부, TMS 생성기와 데이터 이동 카운터로부터의 상태 또는 인터럽트 신호 입력, MUX4로부터의 TMS 입력을 가지며, 연결 확인 신호 전송부에 대한 데이터 출력과 제어 출력, 연결 요청 신호 수신부, TMS 생성기, 데이터 이동 카운터, MUX1, MUX2, MUX3, MUX4, CTDI 버퍼, CTDO 버퍼를 위한 제어 신호 출력을 갖는다. 제어부는 현재 연결된 보드의 TAP 제어기 상태를 알기 위해 상태 머신을 가지고 있다. 제어부는 상태 머신이 6개의 안정된 상태(Test-Logic-Reset, Run-Test/Idle, Shift-DR, Shift-IR, Pause-DR, Pause-IR) 중의 하나에 있을 때, 연결 요청 신호 수신부를 활성화시켜 연결 요청 신호의 수신을 가능하게 한다. 이 때 제어부는 상태 머신을 비활성 상태로 만들어 TMS 버스를 통해 전송되어 오는 연결 요청 신호에 의해 임의로 천이하는 것을 방지한다. 상태 머신은 비활성 상태에서는 마지막 상태를 유지한다. 연결 요청 신호의 수신에 종결되고 나면 상태 머신은 다시 활성화 되어 정상적으로 동작한다. 제어부는 연결 요청 신호의 주



소 필드가 설정된 주소와 일치할 경우나 인터럽트 신호를 전송할 경우 연결 확인 신호 전송부를 활성화시킨다. 제어부는 연결 확인 신호 전송이 끝나면 연결 확인 신호 전송부를 다시 비활성 상태로 만들고 테스트를 위해 보드와 테스트 버스의 연결을 위해 MUX와 버퍼에 제어 신호를 보낸다. MUX1은 A 모드 동작일 경우에는 TDI 버스를 직접 IC의 TAP에 연결하고, B 모드 동작일 때는 CREG의 출력을 IC의 TAP에 연결한다. CREG는 1-비트 수신 레지스터로 TCK의 하강 천이에서 동작하며 CTDI 입력과 MUX1으로의 출력을 갖는다. CREG는 BLU 포트가 B 모드로 동작할 경우 테스트 입력을 TDI 버스로부터 TCK의 하강 천이에서 수신하는 역할을 수행한다. MUX2는 테스트 버스에 연결된 상태에서는 TMS 버스 신호를 IC의 TAP에 전달하고 연결이 해제된 상태에서는 Test-Logic-Reset 상태일 때는 1을, 그 외의 상태일 때는 0을 유지한다. MUX3는 연결 확인 신호나 인터럽트 신호를 보낼 때는 연결 확인 신호 전송부의 출력을 TDO 버스로 전달하고 테스트 결과를 보낼 때는 IC의 TDO로부터 이동되어온 데이터를 TDO 버스로 전달한다. MUX4는 Shift-DR 상태에서 Pause-DR 상태로 천이하기 위해 TMS 신호 생성기가 활성화되었을 때는 TMS 신호 생성기의 출력을 상태 머신에 전달하여 제어부의 상태 머신과 보드의 IC들의 상태를 일치시킨다. 그 외의 경우에는 CTMS 입력을 상태 머신으로 전달하여 정상적으로 동작한다. CTDI 버퍼는 Shift-DR이나 Shift-IR 상태일 때 활성화되어 테스트 입력을 보드 상의 IC에 전달할 수 있게 한다. CTDO 버퍼는 연결 확인 신호나 인터럽트 신호를 보낼 때, 보드 상의 IC로부터 테스트 결과를 테스트 버스에 전송할 때 활성화 된다. 이 때의 동작은 설정된 모드(A 모드 또는 B 모드)에 따라 동작한다.

데이터 이동 카운터(Data Shifting Counter)는 16-비트 카운터로 구성되며 TCK 입력, 활성화 신호 입력, 데이터 이동 상태 신호 출력을 가진다. 보드 상의 주사 셀(scan cell)의 총수로 초기화되고 TAP 제어기가 Shift-DR 상태가 되면 제어부는 활성화 신호를 보내어 데이터 이동 카운터를 동작시킨다. 활성화된 데이터 이동 카운터는 주사 경로 상의 데이터가 이동될 때마다 하나씩 감소하게 된다. 주사 경로 상의 모든

데이터의 이동이 끝나면 데이터 이동 상태 신호는 1이 되어 데이터 이동의 종결을 제어부에 알린다. 데이터 이동 상태 신호의 발생에 따라 제어부는 TMS 신호 생성기(TMS Gen.)를 활성화시킨다. TMS 신호 생성기는 TCK 입력, 제어부로부터의 활성화 신호 입력, TMS 신호 생성 상태 출력 신호를 가진다. Pause-DR 상태로 천이하는데 필요한 TMS 신호가 생성된다. TMS 신호는 TCK의 하강 천이에서 생성되며 필요한 신호의 값은 01이다. TMS 신호 생성기는 비활성 상태일 때는 TMS 버스의 신호를 그대로 MUX2에 전달한다.

연결 요청 신호 수신부는 직렬 입력·병렬 출력의 장치이다. 활성화된 연결 요청 수신부는 모드에 관계없이 TMS 버스로부터 전송되는 신호를 TCK의 상승 천이에서 수신한다. 데이터의 수신은 두 비트 단위로 이루어지며 첫 두 비트가 11이면 연결 요청 신호가 시작됨을 나타내고 다음 11이 수신될 때까지 데이터의 수신이 계속된다. 연결/인터럽트 필드와 연결 해제 요청 필드는 동시에 11이 될 수 없고 주소 필드의 데이터는 10 또는 01이므로 데이터의 수신은 정상적으로 이루어 질 수 있다. 만약 주소 필드가 11로 시작되면 CAB 연결 요청 신호로 해석된다. 연결 요청 수신부는 프레임의 끝(11)이 수신되면 제어부에 인터럽트 신호를 보내 연결 요청 신호의 수신이 종결되었음을 알린다. 연결 요청 신호 수신부로부터 인터럽트 신호를 받은 제어부는 데이터 버스를 통해 연결 요청 신호 수신부로부터 연결/인터럽트 필드와 주소 필드를 수신하여 설정된 보드의 주소와 비교한다. 제어부는 수신된 주소와 설정된 주소가 일치할 경우 정상적인 동작을 위해 연결 요청 신호 수신부를 비활성 상태로 만들고 연결 확인 신호를 전송하기 위해 연결 확인 신호 전송부(XMT)를 활성화시킨다.

연결 확인 신호 전송부는 병렬 입력·직렬 출력 장치이다. 제어부는 설정된 보드의 주소가 연결 요청 신호의 주소와 같을 경우나 Shift-DR 상태에서 Pause-DR 상태로 천이한 후 인터럽트 신호를 보낼 때 연결 확인 신호 전송부를 활성화시킨다. 활성화된 연결 확인 신호 전송부는 데이터 버스를 통해 제어부로부터 연결 확인 신호나 인터럽트 신호를 수신하여 TDO 버스로 전송한다. 연결 확인 신호 전송부는 연결 확인 신호 수신부와는 달리 TDO 버스를 다른 BLU 포

트와 공유해야 하므로 설정된 동작 모드에 따라서 동작해야 한다. 따라서 A 모드로 동작할 경우 TCK의 하강 천이에서 신호를 전송하고 B 모드로 동작할 경우에는 TCK의 상승 천이에서 신호를 전송한다. 신호의 끝(11)을 전송하고 나면 연결 확인 신호 전송부는 신호 전송의 종결을 알리는 인터럽트 신호를 제어부로 보낸다. 연결 확인 신호 전송부로부터 인터럽트 신호를 받은 제어부는 실제 테스트 동작에 들어가기 위해 연결 확인 신호 전송부를 비활성 상태로 만든다.

## 4. 결과 및 고찰

### 4.1 BLU 포트 회로의 합성 결과

회로의 합성에는 역시 Synopsys 툴이 사용되었다. 이에 따른 합성된 결과를 Synopsys 툴을 이용해 면적을 분석한 결과 859개의 단위 소자가 사용되었다. 또한 임계 경로(critical path)에 대한 시간 분석의 결과 데이터의 도달 시간이 10.49 ns로 나타났다. 따라서 BLU 포트는 적은 면적을 차지하므로 전체 보드 하드웨어 오버헤드에 큰 영향을 주지는 않을 것으로 예상된다. 실제로 TI의 ASP[6, 7]와 비교해 보면, BLU 포트는 TMS 신호 생성기와 16-비트 데이터 이동 카운터가 더 사용된 형태이고 이의 면적 증가는 그다지 크지 않을 것으로 기대된다. 따라서 ASP와 비슷하거나 약간 높은 하드웨어 오버헤드로 두 개의 보드를 동시에 연결하여 테스트 데이터를 이동할 수 있으므로 backplane을 통한 보드의 테스트에 소요되는 시간을 대폭 줄일 수 있다. 최근에 와서 고성능 칩이 장착된 보드들이 안정적으로 동작하는지를 검증하는 것이 매우 중요한 사항으로 인식되어 지고 있음에 따라 실제 보드의 제작 과정에서 보드 테스트가 차지하는 비중이 커지고 있다. 또한 이러한 보드들이 시스템에 장착된 후에도 일관된 방식으로 빠르게 테스트를 수행할 수 있다면 테스트 비용 상승을 억제할 수 있어 저비용으로 고성능의 보드를 생산할 수 있다. 이러한 점으로 미루어 볼 때 BLU 포트는 이를 실현하기에 가장 적합하다고 할 수 있다.

### 4.2 시뮬레이션 결과

본 논문에서 제시한 알고리즘에 따라 동작하는 BLU 포트는 VHDL 언어를 이용하여 구현되었다. 구

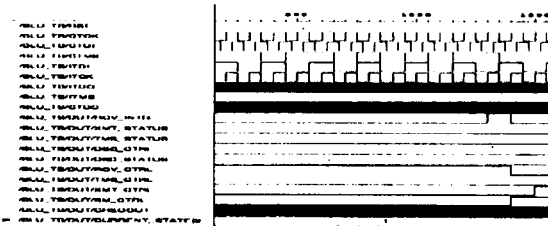
현된 BLU 포트는 Synopsys 툴을 사용하여 컴파일되었다. (그림 7)과 (그림 8)은 각각 BLU 포트의 A 모드 동작과 B 모드 동작을 역시 Synopsys 툴을 사용하여 검증한 결과를 나타내고 있다.

### A 모드 동작 시뮬레이션 검증

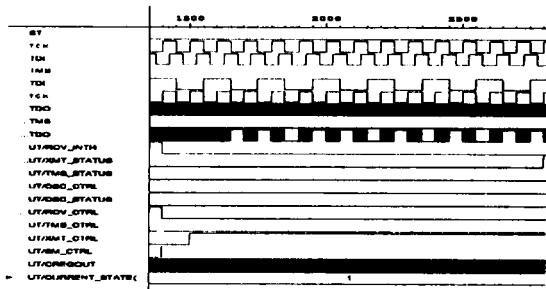
(그림 7)의 (a)는 주소 0으로 설정된 A 모드로 동작 중인 BLU 포트가 Run-Test/Idle 상태(CURRENT\_STATE=1)에서 연결 요청 신호를 수신하고 있는 과정이다. 빗금 부분은 Z(하이 임피던스)인 상태를 나타내고 시간의 단위는 ns이다. 현재 연결된 보드의 수가 10개라고 가정하였다. BLU 포트의 주소가 0으로 설정되어 있으므로 연결 요청 신호의 주소 필드가 101010일 경우 보드는 선택된다. TMS 버스(CTMS)를 통해 전송되어 온 연결 요청 신호가 110110101011이므로 연결 해제 요청 필드가 1로 설정되어 있으며 주소 필드가 101010이므로 현재의 BLU 포트가 선택되어야 한다. TMS 버스를 통해 전송되어 온 데이터에 의해 상태 머신이 불필요한 동작을 하지 못하도록 비활성 상태(SM\_CTRL=0)로 설정되어 있다. 1300 ns 후에 연결 요청 신호의 수신이 모두 종결되고 연결 요청 신호 수신부는 인터럽트 신호(RCV\_INTR=1)를 발생시켜 이를 제어부에 알리고 있다. 따라서 연결 요청 신호의 수신은 예상했던 대로 정확히 동작함을 확인할 수 있다.

(그림 7)의 (b)는 A 모드로 동작하는 BLU 포트가 연결 확인 신호를 보내고 있는 과정이다. 연결 요청 신호의 수신이 종결된 후 연결 요청 신호 수신부는 비활성 상태가 되고(RCV\_CTRL=0), 동시에 상태 머신은 활성 상태(SM\_CTRL=1)가 된다. (a)의 연결 요청 신호의 주소 필드가 현재 BLU 포트에 설정된 주소 필드와 일치하므로 제어부는 연결 확인 신호 전송부를 활성화 시켜(XMT\_CTRL=1), 110110101011로 설정된 연결 확인 신호를 TDO 버스(CTDO)를 통해 전송하고 있다. (c)는 연결 확인 신호의 전송이 종결된 후 테스트 버스에 연결되어 상태 이동을 하고 있는 과정이다. 연결 확인 신호 전송부는 연결 확인 신호의 전송이 끝났음을 알리는 상태 신호(XMT\_STATUS=1)를 발생시킨다. 연결 확인 신호 전송부의 상태 신호에 따라 제어부는 연결 확인 신호 전송부를 비활성 상태(XMT\_CTRL=0)로 만들고 테스트

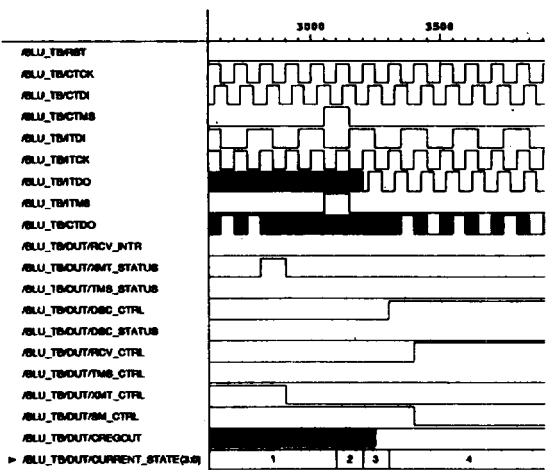
버스에 연결하여 MCU의 신호에 따라 TAP 제어기의 상태를 천이시킨다. BLU는 테스트 버스에 연결되어 있으므로 TMS 버스(CTMS)의 신호가 보드 상의 IC의 TMS 입력(ITMS)에 그대로 전달되고 있음을 확인할 수 있다. 전송되어 온 TMS 신호가 100..이므로 보드 상의 IC와 BLU 포트 내부의 상태 머신은 Select-DR-Scan → Capture-DR → Shift-DR의 순으로 이동된다. (CURRENT\_STATE = 2, 3, 4). 2는 TAP 제어기의 상태에서 중에서 Select-DR-Scan 상태를, 3은 Capture-DR 상태를, 4는 Shift-DR 상태를 나타낸다.



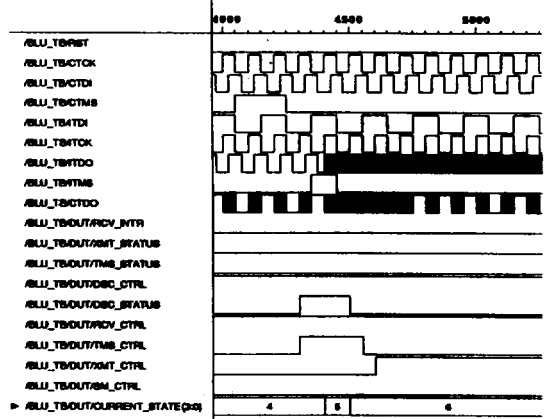
(a) 연결 요청 신호의 수신



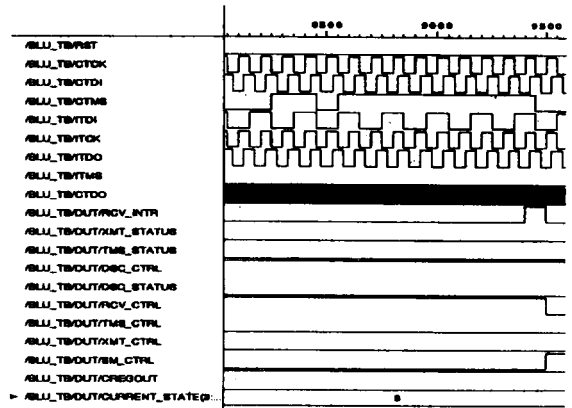
(b) 연결 확인 신호의 전송(A 모드)



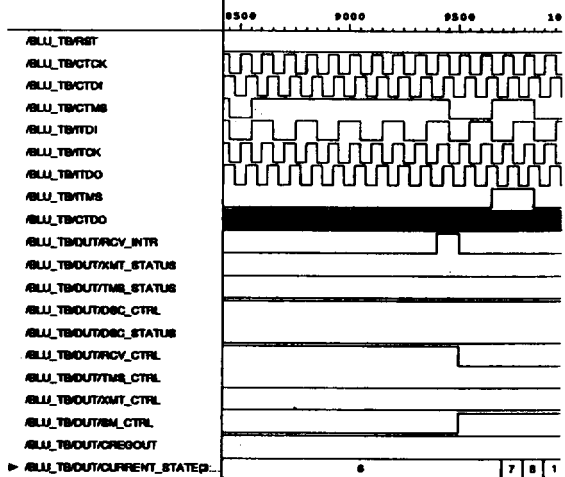
(c) 보드 연결과 상태 천이



(d) TMS 신호 생성과 인터럽트 신호 발생



(e) CAB 신호의 수신



(f) CAB 연결 요청 신호에 의한 테스트 수행과 상태 천이

(그림 7) A 모드 BLU 포트의 시뮬레이션 결과  
(Fig. 7) Simulation result of A mode BLU port



(RCV\_CTRL=1), 상태 머신은 비활성 상태(SM\_CTRL=0)로 있으므로 보드 상의 IC에는 영향을 미치지 않으면서 CAB 연결 요청 신호를 수신하고 있다. CAB 신호의 수신은 종결되고 난 후(9400 ns), 연결 요청 신호 수신부는 신호의 수신을 알리는 인터럽트 신호(RCV\_INTR=1)를 발생시키고 있다. (f)는 CAB 신호에 따라 테스트 버스에 연결되어 테스트 수행을 위한 상태 천이 과정을 나타내고 있다. CAB 신호가 수신된 후 연결 요청 신호 수신부는 비활성 상태(RCV\_CTRL=0)로 바뀌고 상태 머신은 활성 상태(SM\_CTRL=1)가 되어, TMS 버스를 통해 전송되어 온 TMS 신호(CTMS=110)에 따라 상태 머신과 보드 상의 IC(ITMS)들은 Exit1-DR → Update-DR → Run-Test/Idle로 이동된다(CURRENT\_STATE=7, 8, 1). 7은 TAP 제어기 상태도의 Exit1-DR 상태를, 8은 Update-DR 상태를, 1은 Run-Test/Idle 상태를 나타낸다. CTMS와 ITMS가 같은 신호값을 가지고 있으므로 TMS 버스를 통해 전송된 TMS 신호가 보드 상의 IC로 직접 전달됨을 알 수 있다. 이 과정에서 이동된 테스트 입력에 의해 실제 테스트가 수행된다. 이상의 시뮬레이션 결과를 통해 BLU 포트의 A 모드 동작은 예상했던 바와 정확히 일치하여 동작함을 확인할 수 있다.

### B 모드 동작 시뮬레이션 검증

B 모드의 동작 검증도 A 모드와 마찬가지로 각각의 중요한 과정들을 중심으로 검증하였다. 역시 연결된 보드의 수는 10개로 가정하였고 설정된 BLU 포트의 주소는 0(101010)로 가정하였다. 연결 요청 신호의 수신 과정은 A 모드와 같으므로 생략하였다. (그림 8)의 (a)는 연결 요청 신호에 따른 연결 확인 신호의 전송 과정을 나타낸다. B 모드의 연결 확인 신호의 전송도 TDO 버스를 통해 데이터를 전송하는 방법을 제외하고는 A 모드와 같음을 알 수 있다. B 모드에서는 A 모드로 동작 중인 BLU 포트와의 데이터 충돌을 방지하기 위해 TCK가 1인 동안에 TDO 버스로 데이터를 전송한다. 따라서 1101101011로 구성된 연결 확인 신호가 TCK가 1인 동안에 TDO 버스로 전송되고 있음을 확인할 수 있다. (b)는 연결 확인 신호의 전송이 끝나고 BLU 포트가 보드를 테스트 버스에 연결하여 MCU에 따라 상태를 이동하는 과정이다. 역시 Shift-

DR 상태에서의 데이터 이동 과정을 제외하면 A 모드와 똑같이 동작함을 알 수 있다. Shift-DR 상태에서는 A 모드로 동작 중인 BLU 포트와의 데이터 충돌을 방지하기 위해 TCK가 1인 동안에 TDO 버스로 테스트 결과를 전송하고 있다. 또한 테스트 입력은 CREG를 통해 TCK의 하강 천이에서 수신하여 이를 IC의 TAP(ITDO)으로 전달하여 뒤 따르는 TCK의 상승 천이에서 보드 상의 IC가 데이터를 이동시킬 수 있도록 한다. CTDI 버퍼는 BLU 포트가 Shift-DR 상태에 들어가기 전에 활성 상태가 되어 CREG가 첫 번째 테스트 입력을 정확하게 수신할 수 있도록 한다. 수신된 테스트 입력은 BLU 포트가 Shift-DR 상태로 이동함과 동시에 주사 경로 상의 첫 번째 IC에 의해 이동된다. 따라서 MCU의 설계에서도 이러한 클럭 타이밍을 고려해야 한다. 시뮬레이션의 검증을 쉽게 하기 위해 A 모드 테스트 입력은 모두 0이 되게, B 모드 테스트 입력은 모두 1이 되게 하였다. 따라서 ITDO는 Shift-DR 상태에서 1의 값을 유지하고 있다. (c)는 Shift-DR 상태에서 데이터의 이동이 종결된 후 TMS 신호 생성기에 의해 Pause-DR 상태로 이동하는 과정을 나타낸다. 역시 TDO 버스를 통한 데이터의 이동 상황을 제외하고는 A 모드와 똑같이 동작함을 확인할 수 있다. 주목할 사항은 Shift-DR 상태에서 IC의 TDO로부터 이동되어 온 마지막 테스트 결과의 처리에 있다. 실험 결과를 살펴보면 4350 ns에서 TMS 신호 생성기에 의해 TMS 신호가 생성되며 이 때 마지막 테스트 결과가 IC의 TDO를 통해 이동되어 온다. 따라서 이 데이터는 뒤 따르는 TCK의 상승 천이(4400 ns)에서 TDO 버스로 이동되어야 한다. 실험 결과는 이를 정확히 수행하고 있음을 보여주고 있다. MCU의 설계에서도 앞서와 마찬가지로 이러한 클럭 타이밍 역시 고려해야 한다. 이후의 과정인 CAB 신호의 수신과 이에 따른 MCU에 의한 상태 천이 과정은 역시 A 모드와 같으므로 생략하였다.

## 5. 결 론

보드 내에 장착된 소자들의 테스트를 지원하기 위해 제안된 IEEE 표준 1149.1은 보드 수준에서의 완벽한 테스트를 제공해 준다. 하지만 보드가 시스템에 장착되고 난 후의 테스트는 여전히 문제로 남아 있다.

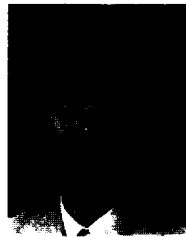
이를 해결하기 위해 제안된 방법이 IEEE 표준 1149.1 backplane 테스트 확장이다. 기존의 IEEE 표준 1149.1 backplane 테스트 확장 방법은 테스트 데이터를 이동시키기 위해 한 번에 하나의 보드만을 제어하는 방식을 취하였다. 그 결과 테스트 데이터의 이동에 필요한 시간의 증가로 전체 테스트 시간이 길어졌다. 본 논문에서는 테스트에 소요되는 시간을 줄이기 위해 하나의 IEEE 표준 1149.1 테스트 버스로 두 개의 보드를 동시에 제어할 수 있는 새로운 알고리즘을 개발하였다. 개발된 알고리즘은 두 개의 보드를 동시에 접근하여 테스트 데이터를 이동시킬 수 있게 하여 테스트에 소요되는 시간을 대폭 감소시킨다. 새 알고리즘에 기초하여 제작된 보드 연결부 포트(Board Link Unit Port: BLU Port)는 적은 트랜지스터의 사용으로 구현되어 하드웨어 오버헤드의 증가가 전체 보드의 면적 증가에 큰 영향을 미치지 않는다. 또한 보드 연결부 포트를 사용할 경우 테스트 버스 제어기는 두 개의 보드를 동시에 테스트 버스에 연결하여 테스트 데이터를 이동시킬 수 있으므로 테스트 시간을 절약하여 전체 테스트 비용의 상승을 억제하면서 효율적으로 테스트를 수행할 수 있다. 따라서 보드 연결부 포트는 현재 중요성을 더해 가는 저비용의 보드 테스트 문제 해결에 가장 적합하다고 할 수 있다.

**참 고 문 헌**

[1] K. P. Parker, "The Boundary-Scan Handbook", Kluwer Academic Publisher, USA, 1992.  
 [2] IEEE Computer Society, "IEEE Standard 1149.1 Test Access Port and Boundary-Scan Architecture", IEEE std. 1149.1-1990, IEEE Computer Society, New York, NY, May 21, 1990.  
 [3] D. Landis, C. Hudson, and P. McHugh, "Applications of The IEEE P1149.5 Module Test and Maintenance Bus," Proc. of International Test Conf., pp. 984-992, 1992.  
 [4] C. Su, S. J. Jou, and Y. T. Ting, "Decentralized BIST for 1149.1 and 1149.5 Based Interconnects," Proc. of Asian Test Symp., pp. 120-125, 1996.  
 [5] D. Bhavsar, "An Architecture for Extending the IEEE Standard 1149.1 Test Access Port to System

Backplanes," Proc. of International Test Conf., pp. 768-776, 1991.

[6] L. Whetsel, "A Proposed Method of Accessing 1149.1 in a Backplane Environment," Proc. of International Test Conf., pp. 206-216, 1992.  
 [7] L. Whetsel, "Hierarchically Accessing 1149.1 Applications," Proc. of International Test Conf., pp. 517-526, 1993.



**강 성 호**

1986년 2월 서울대학교 제어계측공학과 졸업(학사)  
 1988년 5월 The University of Texas Austin 전기 및 컴퓨터공학과(공학석사)

1992년 5월 The University of Texas Austin 전기 및 컴퓨터공학과(공학박사)  
 1989년 11월~1992년 8월 Schlumberger Inc. Research Scientist  
 1992년 9월~1992년 10월 The Univ. of Texas at Austin Post Doctoral Fellow  
 1992년 8월~1994년 6월 Motorola Inc. Senior Staff Engineer  
 1994년 9월~현재 연세대학교 전기공학과 조교수  
 관심 분야: CAD 및 VLSI, 테스트, ASIC 설계



**임 용 태**

1995년 2월 연세대학교 전기공학과 졸업(학사)  
 1995년 9월~1997년 8월 연세대학교 전기공학과 대학원 졸업(공학석사)  
 1997년 8월~현재 LG 반도체 연구원 근무중

관심분야: CAD 및 VLSI, 테스트



**김 현 진**

1997년 2월 연세대학교 전기공학과 졸업(학사)  
 1997년 3월~현재 연세대학교 전기공학과 대학원 석사과정(석사2학기)

관심분야: CAD 및 VLSI, 테스트