

IEEE 1149.1을 이용한 확장된 스캔 경로 구조

손 우 정[†] · 윤 태 진^{††} · 안 광 선^{†††}

요 약

본 논문에서는 다중 보드를 시험하기 위한 새로운 구조인 확장된 스캔 경로(ESP: Extended Scan Path)와 절차를 제안한다. 보드를 시험하기 위한 기존의 구조로는 단일 스캔 경로와 다중 스캔 경로가 있다. 단일 스캔 경로 구조는 시험 데이터의 전송 경로인 스캔 경로가 하나로 연결되므로 스캔 경로가 단락이나 개방으로 결합이 생기면 나머지 스캔 경로에 올바른 시험 데이터를 입력할 수 없다. 다중 스캔 경로 구조는 다중 보드 시험 시 보드마다 별도의 신호선이 추가된다. 그러므로 기존의 두 구조는 다중 보드 시험에는 부적절하다. 제안된 ESP 구조를 단일 스캔 경로 구조와 비교하면, 스캔 경로 상에 결합이 발생하더라도 그 결합은 하나의 스캔 경로에만 한정되어 다른 스캔 경로의 시험 데이터에는 영향을 주지 않는다. 뿐만 아니라, 비스트 (BIST: Built In Self Test)와 IEEE 1149.1 경계면 스캔 시험을 병렬로 수행함으로써 시험에 소요되는 시간을 단축한다. 또한 ESP 구조를 다중 스캔 경로 구조와 비교하면, 스캔 경로마다 신호선을 공통으로 사용함으로써 다중 보드 시험 시 추가되는 신호선이 없다. 본 논문에서는 제안한 ESP 구조와 기존 시험 구조의 성능을 비교하기 위해서, ISCAS '85 벤치마크 회로를 대상으로 각 구조의 시험 수행 시간을 비교하여 우수함을 보였다.

An Extended Scan Path Architecture Based on IEEE 1149.1

Woo Jung Son[†] · Tae Jin Yun^{††} · Gwang Seon Ahn^{†††}

ABSTRACT

In this paper, we propose a ESP(Extended Scan Path) architecture for multi-board testing. The conventional architectures for board testing are single scan path and multi-scan path. In the single scan path architecture, the scan path for test data is just one chain. If the scan path is faulty due to short or open, the test data is not valid. In the multi-scan path architecture, there are additional signals in multi-board testing. So conventional architectures are not adopted to multi-board testing. In the case of the ESP architecture, even though scan path is either short or open, it doesn't affect remaining other scan paths. As a result of executing parallel BIST and IEEE 1149.1 boundary scan test by using the proposed ESP architecture, we observed that the test time is short compared with the single scan path architecture. Because the ESP architecture uses the common bus, there are not additional signals in multi-board testing. By comparing the ESP architecture with conventional one using ISCAS '85 benchmark circuit, we showed that the architecture has improved results.

1. 서 론

최근에 와서 회로는 VLSI 설계 기술의 발전과 새로운 생산 공정 기술로 인해 집적도가 증가하고 있다. 이러한 집적도의 증가는 회로의 부피를 더욱 축소시키므로 기존의 인서킷(in-circuit) 방법으로는 시험이 어렵게 되었다. 그러므로 이러한 문제를 해결하기 위하여 하드웨어 설계 초기에 시험을 고려한 설계

† 중신회원: 대구전문대학 전자계산과 조교수
†† 준 회 원: 경북대학교 대학원 컴퓨터공학과
††† 중신회원: 경북대학교 컴퓨터공학과 정교수
논문접수: 1996년 10월 24일, 심사완료: 1996년 11월 22일

기법인 DFT(Design For Testability)의 연구가 활발하게 진행되고 있다[1-3]. 최근에 연구되는 대표적인 DFT인 IEEE 1149.1 경계면 스캔 구조[4, 5]는 1985년 네덜란드 필립사의 Frans Beenker에 의해 제안되었으며, 그 후 1989년 미국의 TI, MOTOROLA 등의 참여에 따라 세계적인 연구활동 후, 1990년에 IEEE에 의해 시험 구조의 표준으로 결정되었다. IEEE 1149.1 경계면 스캔 구조는 대상 회로의 모든 입력 및 출력단에 경계면 스캔 레지스터(BSR: Boundary Scan Register)를 두어 이들을 스캔 경로로 연결시킴으로써, 외부에서부터 관측 및 제어를 한다.

IEEE에서는 보드를 시험하는 구조로서 단일 스캔 경로[2, 4, 5]와 다중 스캔 경로[6, 7]를 제시하였다. 단일 스캔 경로 구조와 다중 스캔 경로 구조는 시험의 여러 분야에 응용되고 있다[8, 9, 10]. 그러나 단일 스캔 경로 구조에서는 시험 데이터의 전송 경로인 스캔 경로가 하나로 연결되므로 시험 데이터 경로가 단락이나 개방에 의해서 결함이 생기면, 나머지 스캔 경로의 시험 데이터는 올바른 값을 갖지 못하는 문제가 발생한다. 다중 스캔 경로 구조에서는 각 모듈이 서로 독립된 스캔 경로를 가지므로 모든 스캔 경로마다 신호선이 추가되는 문제가 발생한다. 그러므로 단일 스캔 경로 구조와 다중 스캔 경로 구조는 이러한 문제점 때문에 다중 보드 시험에는 적합하지 않다.

다중 보드 시험을 위하여 경계면 스캔 경로 구조를 이용한 방법이 시도되었다[11]. 그러나 이 방법은 계층적인 시험 방식이지만, 단일 스캔 경로 구조만을 지원하고 다중 스캔 경로 구조는 지원하지 않았다.

본 논문은 단일 스캔 경로 구조와 다중 스캔 경로 구조를 혼합하여 다중 보드 시험에 적합한 ESP 제어를 설계하고 이를 위한 시험 절차를 제안한다. ESP 구조는 단일 스캔 경로 구조와는 달리 스캔 경로 상에 결함이 발생하더라도 그 결함은 하나의 스캔 경로에만 한정되어 다른 스캔 경로의 시험 데이터에는 영향을 주지 않는다. 또한 다중 스캔 경로 구조와는 달리, 각 스캔 경로마다 신호선을 공통으로 사용함으로써 다중 보드 시험 시 신호선 추가는 없다.

본 논문의 구성은 다음과 같다. 2장에서는 IEEE 1149.1 경계면 스캔 구조와 단일 및 다중 스캔 경로 구조에 대해 설명하고, 3장에서는 ESP 구조를 설명한다. 또한 이 구조에서 사용하는 ESP 제어기를 설계

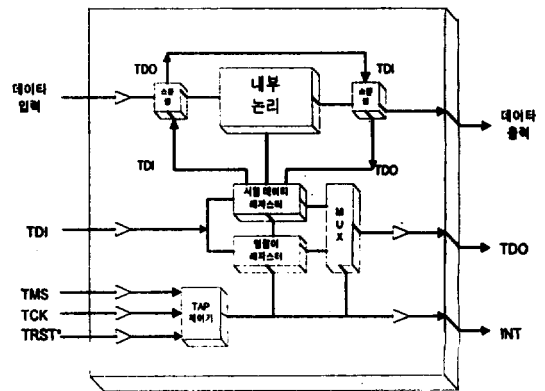
하고 이 제어기를 위한 시험 절차를 제안한다. 4장에서는 ESP 구조와 기존의 시험구조를 ISCAS '85 벤치마크 회로를 이용하여 시험 수행 시간을 실험한다. 5장에서는 ESP 구조와 기존의 시험구조의 성능을 비교 분석하고, 마지막으로 6장에서 결론을 맺는다.

2. 기존의 보드 시험 구조

2.1 IEEE 1149.1 경계면 스캔 구조

IEEE 1149.1 경계면 스캔 구조는 기존의 탐침판 (bed of nails)을 직접 연결하여 사용하던 인서킷 (in-circuit) 시험의 문제점을 해결하기 위해서 탐침판 없이, 내부에 시험 메카니즘을 구성하고 외부선만을 이용하여 시험한다. IEEE 1149.1 경계면 스캔 구조를 가지는 집적회로는 시스템 내부 논리와 시험 논리의 두 부분으로 나뉜다. 시스템 내부 논리는 칩 설계의 원래 목적대로 정상 동작을 수행하는 회로다. 시험 논리는 시스템 논리를 제외한 부분으로서, 시험을 위해 포함시킨 회로다.

IEEE 1149.1 경계면 스캔 구조[4]는 그림 1에서 보는 바와 같이 TAP 제어기 (TAPC: Test Access Port Controller), 명령어 레지스터 (IR: Instruction Register), 시험 데이터 레지스터 (TDR: Test Data Register)의 3가지 핵심요소로 구성된다. TAP 제어기는 TAP의 입력 신호들인 TMS(Test Mode Select), TCK(Test Clock), 그리고 TRST*(Test Reset)를 받아 명령어 레지스터



(그림 1) IEEE 1149.1 경계면 스캔 구조
(Fig. 1) IEEE 1149.1 boundary scan architecture

와 시험 데이터 레지스터에 필요한 클럭과 제어 신호를 발생한다. 명령어 레지스터에는 수행될 명령어가 적재된다. 시험 데이터 레지스터에는 시험에 필요한 데이터나 조건값이 적재된다.

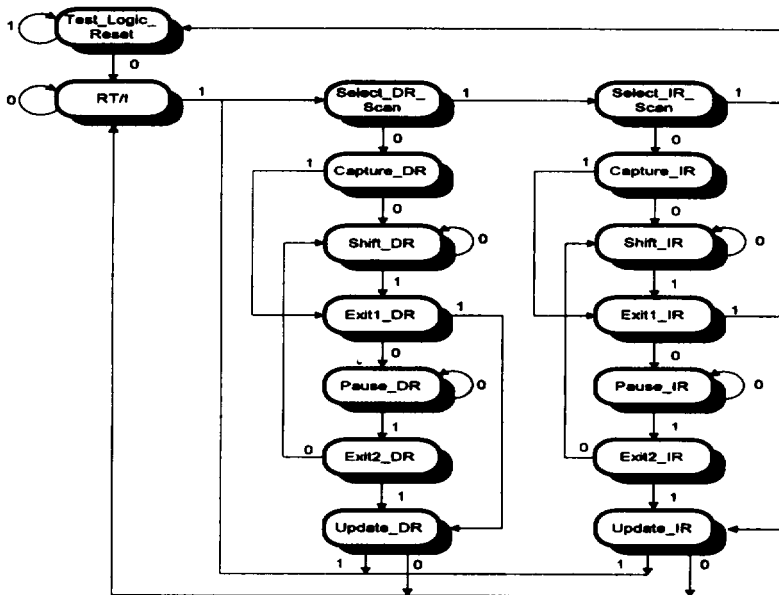
IEEE 1149.1 경계면 스캔 구조에서 사용되는 신호 [4]들은 TCK (Test Clock), TMS (Test Mode Select), TDI (Test Data Input), TDO (Test Data Output), 그리고 TRST*(Test Reset)이다.

TCK 신호는 여러 칩들 사이에 시험을 동기화하고 TAP 제어기를 구동시킨다. TMS 신호는 TAP 제어기가 명령어 레지스터와 시험 데이터 레지스터에 필요한 클럭과 제어신호를 발생하게 하고, TDI 신호는 TMS 신호에 따라 명령어 레지스터나 시험 데이터 레지스터에 데이터를 입력한다. TDO 신호는 TMS 신호에 따라 시험 데이터의 내용을 출력하고, TRST* 신호는 TAP 제어기를 시험의 시작 상태로 초기화한다. TRST*의 '*'은 로우 액티브(low active)를 의미한다. 즉, 초기화를 위한 입력 신호는 로직 로우(logic low)에서 활성화된다. TAP 제어기는 그림 2에서 보는 바와 같이 상태천이도에 따라 동작하며 16개의 상태를 가진다[4].

IEEE 1149.1 경계면 스캔 구조에서 지원하는 기본적인 명령어는 샘플(Sample)/사전 적재(Preload), 익스테스트(Extest), 인테스트(Intest), 그리고 바이패스(Bypass)이다. 샘플/사전 적재 명령어는 두 가지 기능을 수행한다. 샘플 메카니즘은 보드의 정상동작에는 영향을 미치지 않으면서 시스템 편으로부터 시험 데이터를 샘플한다. 사전 적재 메카니즘은 다른 시험 동작이 선택되기 전에 초기 시험 데이터를 경계면 스캔 레지스터 셀의 병렬 출력에 입력한다. 익스테스트 명령어는 대상 회로의 상호연결(interconnection) 시험에 사용되고 인테스트 명령어는 칩 내의 대상회로 시험에 사용된다. 바이패스 명령어는 소자가 시험을 필요로 하지 않을 때 외부로부터 입력된 TDI 신호를 전달하는데 있어 최단경로를 제공한다.

2.2 단일 및 다중 스캔 경로 구조

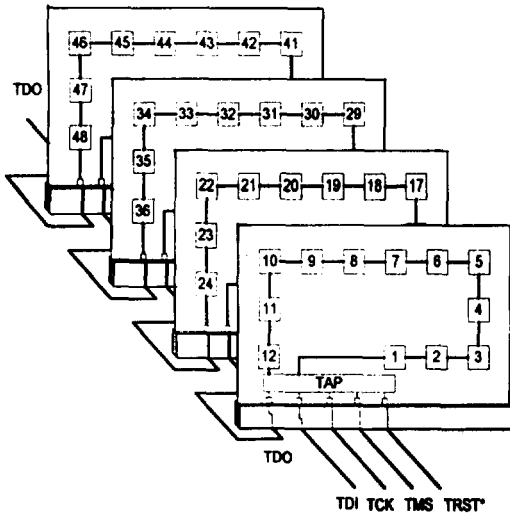
최근에 보드 시험[12, 13]과 다중 보드 시험[14-17]에 대한 연구가 활발히 진행되고 있다. IEEE 1149.1 경계면 스캔 구조를 가지는 칩 및 보드를 시험하는 구조에는 단일 스캔 경로와 다중 스캔 경로가 있다.



(그림 2) TAP 제어기의 상태 천이도
(Fig. 2) State transition diagram of TAP controller

2.2.1 다중 보드 시험을 위한 단일 스캔 경로 구조

단일 스캔 경로 구조에서는 TCK와 TMS 신호는 공유되고 칩들의 스캔 경로는 하나로 연결된다. 다중 보드 시험을 위한 단일 스캔 경로 구조는 그림 3과 같다. 그림 3에서, 모든 칩을 전부 시험할 경우, 첫 번째 보드의 TDI로부터 시험 데이터가 각 칩의 경계면 스캔 레지스터에 인가되고, 결과는 마지막 보드의 TDO로 출력된다. 첫 번째 보드와 두 번째 보드 사이의 경로, 두 번째 보드와 세 번째 보드 사이의 경로를 시험할 경우, 첫 번째 보드의 TDI로 인가된 시험 데이터가 첫 번째 보드와 두 번째 보드의 각각 12와 24 경계면 스캔 레지스터에 인가된다. 응답값은 두 번째 보드와 세 번째 보드의 각각 13과 25 경계면 스캔 레지스터에 입력되고, 마지막 보드의 TDO로 출력된다.

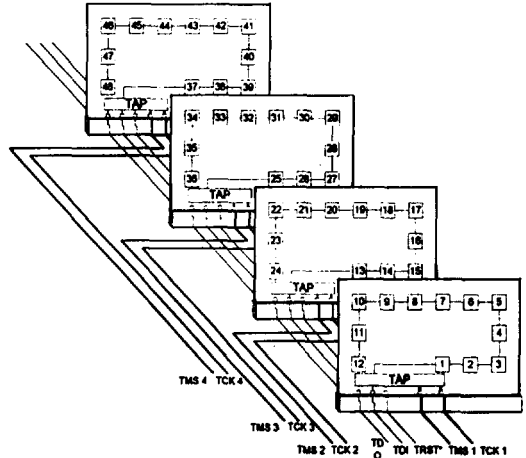


(그림 3) 단일 스캔 경로 구조
(Fig. 3) Single scan path architecture

2.2.2 다중 보드 시험을 위한 다중 스캔 경로 구조

다중 스캔 경로 구조에서는 TDI와 TDO 신호는 공유되고 TMS와 TCK 신호는 각 보드마다 추가된다. 그리고 각 보드들은 서로 독립된 스캔 경로를 가진다. 다중 보드 시험을 위한 다중 스캔 경로 구조는 그림 4와 같다. 그림 4에서 첫 번째 보드와 두 번째 보드 사이의 연결을 시험할 경우, 첫 번째 보드의 출력단 경계면 스캔 레지스터에 시험 데이터가 입력되기 위

해 첫 번째 보드의 스캔 경로가 선택되고, 첫 번째 보드의 TMS와 TCK에 신호가 입력된다. 그 다음, 두 번째 보드의 스캔 경로가 선택되고, 두 번째 보드의 TMS와 TCK에 신호가 입력된 후, 응답값이 TDO로 출력된다.



(그림 4) 다중 스캔 경로 구조
(Fig. 4) Multi-scan path architecture

2.2.3 다중 보드 시험을 위한 단일 및 다중 스캔 경로 구조의 문제점

단일 스캔 경로 구조의 경우, 회로구현은 간단하다. 그러나 각 모듈들의 스캔 경로가 하나로 연결되므로 스캔 경로가 결합으로 인해 단락이나 개방이 생기면, 나머지 스캔 경로에 올바른 시험 데이터를 입력할 수 없다. 뿐만 아니라 다중 보드 시험에서 슬롯들 중 어느 하나가 비게 되면 스캔 경로의 전체 연결이 끊어져 올바른 시험 데이터를 갖지 못한다. 이 구조는 TMS 신호를 공통으로 사용하므로 TAP 제어기의 RT/I (Run-Test/Idle) 상태에서 비스트(BIST: Built In Self Test)[18, 19]와 IEEE 1149.1 경계면 스캔 시험의 동시 수행은 불가능하다. 다중 스캔 경로 구조의 경우, 각 칩이 서로 독립된 스캔 경로를 가지므로 한 스캔 경로의 결합으로 시험 데이터가 올바른 값을 갖지 못하더라도 결합은 그 스캔 경로에 한정되어 나머지 칩들의 시험 데이터에는 영향을 미치지 않는다. 그러나 모든 스캔 경로마다 각각의 TCK와 TMS 신호를 가

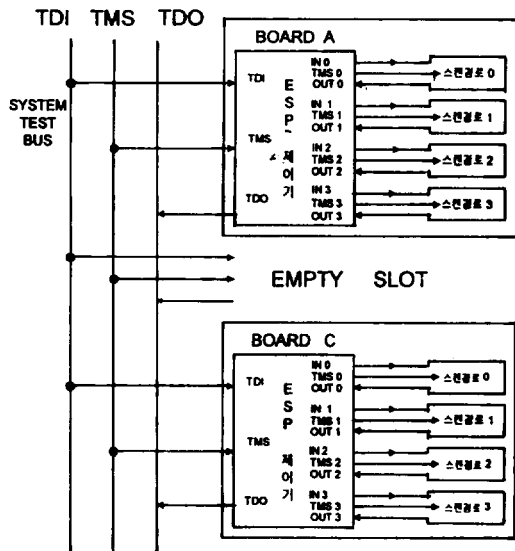
지기 때문에 단일 스캔 경로 구조에 비해 신호선 추가에 따른 하드웨어 부담이 커진다. 또한 TDI와 TDO 신호선을 공유하므로 스캔 경로사이에 동시에 데이터 신호를 보내게 되면 신호의 충돌을 가져오므로 이를 해결하기 위해 시험 절차가 복잡해지고, 시험을 제어하는 마스터와 칩들간의 동기를 맞추어야 한다.

3. 다중 보드 시험을 위한 ESP 구조

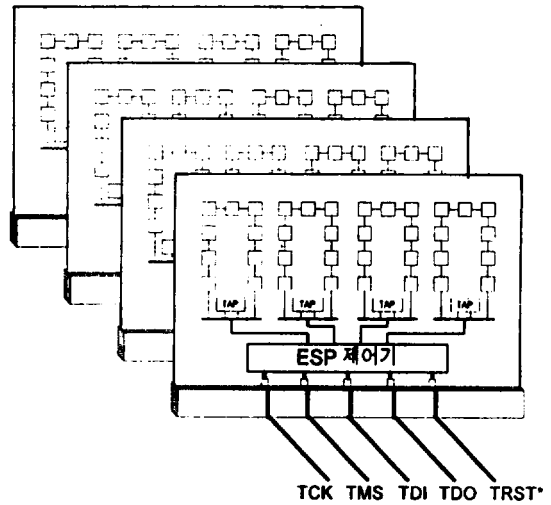
본 논문에서는 다중 보드를 시험하기 위한 ESP 구조를 제안한다. 또한 이 구조를 위한 ESP 제어기를 설계하고, ESP 제어기의 시험 절차를 제안한다.

3.1 ESP의 기본 구조

ESP 구조는 다중 보드 시험을 위해서 ESP 제어기와 시스템의 시험 버스를 연결한다. 다중 보드 시험을 위한 ESP 구조는 TDI, TCK, TMS, TRST* 그리고 TDO 신호선을 사용한다. 이 신호선은 모든 보드에서 공통으로 사용하는 버스 구조로서, 보드가 제거되거나 스캔 경로상의 결함이 발생하더라도 전체 보드에는 영향을 주지 않는다. 다중 보드 시험을 위한 ESP 제어기와 시스템 버스의 구성도는 그림 5와 같다.



(그림 5) ESP 제어기와 시스템 시험 버스의 구성도
(Fig. 5) Configuration of ESP controller and system test bus



(그림 6) ESP 구조
(Fig. 6) ESP architecture

ESP 구조는 다중 보드 시험 시에도 5개의 기본 신호선만을 사용한다. 뿐만 아니라 한 보드 당 4개의 독립된 스캔 경로를 가지며 각 스캔 경로는 TAP에 연결된다. 여기서 TAP은 5개의 기본 신호선을 가지는 포트로서 시험 로직인 TAP 제어기, 명령어 레지스터, 그리고 시험 데이터 레지스터를 포함한다. ESP 구조는 그림 6과 같다.

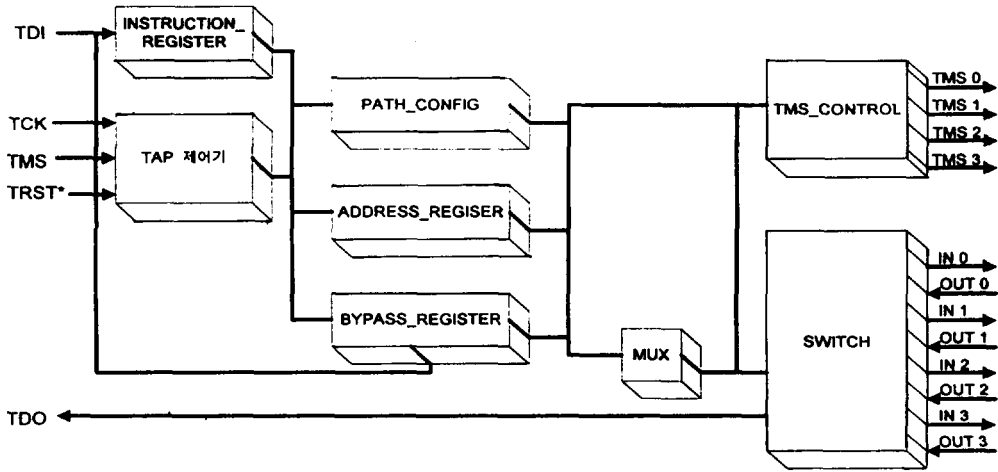
3.2 ESP 제어기의 설계

ESP 제어기의 블록도는 그림 7과 같다.

3.2.1 ESP 제어기의 구성요소

ESP 제어기는 INSTRUCTION_REGISTER, TAP 제어기, PATH_CONFIG, ADDRESS_REGISTER, BYPASS_REGISTER, MUX, TMS_CONTROL, SWITCH 블록으로 구성되며 각 블록은 다음과 같다.

INSTRUCTION_REGISTER 블록은 ESP 제어기에서 사용하는 5개의 명령어가 입력되는 명령어 레지스터이다. TAP 제어기 블록은 IEEE 1149.1 경계면 스캔 구조를 사용하는 보드에서 쉽게 이식할 수 있도록 하기 위해서 IEEE 1149.1 경계면 스캔 구조의 TAP 제어기를 그대로 사용한다. PATH_CONFIG 블록은 선택되어진 보드에 대해 스캔 경로를 구성한다. ADDRESS_REGISTER 블록은 시험할 보드의 주소



(그림 7) ESP 제어기 블록도
(Fig. 7) Block diagram of ESP controller

를 받아서 보드 선택 여부를 결정한다. BYPASS REGISTER 블록은 자신을 보드 시험을 위한 시험 데이터의 이동경로로서 이용한다. MUX 블록은, 경로의 모든 데이터가 하나의 직렬경로를 사용하므로 신호들을 하나씩 선택하여 출력한다. TMS_CONTROL 블록은 IEEE 1149.1 경계면 스캔 시험과 비스트를 동시에 수행하기 위해 4개 스캔 경로의 TMS 신호를 조절한다. SWITCH 블록은 스캔 경로 구성을 결정하는 PATH_CONFIG 블록의 출력을 받아 스캔 경로를 연결한다.

3.2.2 ESP 제어기의 명령어

ESP 제어기에서 사용하는 명령어는 Address_Select, Path_Config, TMS_Control, Bypass, 그리고 Group_Select이다. Address_Select 명령어는 시험하고자 하는 보드를 선택하고, Path_Config 명령어는 4개의 스캔 경로를 구성한다. TMS_Control 명령어는 각 스캔 경로의 TMS를 조절하고, Bypass 명령어는 TDI 신호가 다른 스캔 경로를 통과하지 않도록 함으로써 데이터의 전달 횟수를 감소시켜 시험 수행 시간을 줄인다. Group_Select 명령어는 시험하려는 그룹보드를 선택한다. Group_Select 명령어의 상위 2비트의 값이 '10'이면 그룹 선택을 나타내고, 하위 2비트의 값은 선택되어지는 4가지 그룹의 주소를 나타낸다. Group_Select 명령어는 표 1과 같다.

〈표 1〉 Group_Select 명령어
(Table 1) Group_Select instruction

명령어 비트				그룹 주소
3	2	1	0	
10		0	0	'00'
		0	1	'01'
		1	0	'10'
		1	1	'11'

3.3 ESP 제어기를 이용한 시험 절차

3.3.1 보드 선택

ESP 제어기는 명령어를 입력하기 위해서 Capture_IR 상태에서 Shift_IR 상태를 경유하여 Update_IR 상태로 천이한다. 이때 명령어 레지스터에 Address_Select 명령어가 입력되면 ADDRESS_REGISTER 블록으로 보드 주소를 입력받아서 Slot_ID와 동일인지 검사한다. 표 1과 같이 명령어 레지스터의 4비트 중에서 상위 2비트가 '10'이면 Group_Select 명령어이고, 명령어 레지스터의 하위 2비트의 값이 시험하려는 그룹 주소가 된다. 그러므로 명령어 레지스터에 Group_Select 명령어가 입력되면 명령어의 하위 2비트를 Group_ID와 비교한다. Address_Select 명령어의 경우에 ADDRESS_REGISTER 블록의 주소 값이 Slot_ID와 같거나 Group_Select 명령어의 경우에 Group_Select의 하위

```

/* 절차 1 : 보드 선택 절차 */
Procedure 보드_선택
{
TAP 제어기의 상태 = Test_Logic_Reset -> Run_Test/Idle -> Select_IR_Scan -> Capture_IR
if 명령어 레지스터 := Address_Select
then
for TAP 제어기의 상태 := Capture_DR to Update_DR do
ADDRESS_REGISTER := TDI
if ADDRESS_REGISTER = Slot_ID
then Select := 1
else
Select := 0
else if ( 명령어 레지스터[3.2] = Group_Select ) and ( 명령어 레지스터[1.0] = Group_ID )
then Select := 1
else
other operation /* ESP 제어기에서 제공하는 명령어 중 'Address_Select' 명령어를 제외한 명령어 실행 */
}

```

```

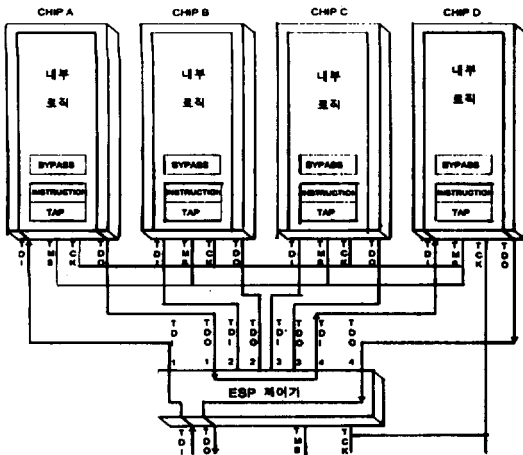
/* 절차 2 : 스캔 경로 구성 절차 */
Procedure 스캔경로_선택
{
TAP 제어기의 상태 = Test_Logic_Reset -> Run_Test/Idle -> Select_DR_Scan -> Capture_DR
if ( 명령어 레지스터 = Path_Config ) and ( Select = 1 )
then
for TAP 제어기의 상태 := Capture_DR to Update_DR do
PATH_CONFIG := TDI
case ( PATH_CONFIG의 데이터 ) of
'0000' : TDI->ESP 제어기->TDO;
'0001' : TDI->ESP 제어기->스캔경로1->ESP 제어기->TDO;
'0010' : TDI->ESP 제어기->스캔경로2->ESP 제어기->TDO;
'0011' : TDI->ESP 제어기->스캔경로1->ESP 제어기->스캔경로2->ESP 제어기->TDO;
'0100' : TDI->ESP 제어기->스캔경로3->ESP 제어기->TDO;
'0101' : TDI->ESP 제어기->스캔경로1->ESP 제어기->스캔경로3->ESP 제어기->TDO;
'0110' : TDI->ESP 제어기->스캔경로2->ESP 제어기->스캔경로3->ESP 제어기->TDO;
'0111' : TDI->ESP 제어기->스캔경로1->ESP 제어기->스캔경로2->ESP 제어기->스캔경로3->ESP 제어기->TDO;
'1000' : TDI->ESP 제어기->스캔경로4->ESP 제어기->TDO;
'1001' : TDI->ESP 제어기->스캔경로1->ESP 제어기->스캔경로4->ESP 제어기->TDO;
'1010' : TDI->ESP 제어기->스캔경로2->ESP 제어기->스캔경로4->ESP 제어기->TDO;
'1011' : TDI->ESP 제어기->스캔경로1->ESP 제어기->스캔경로2->ESP 제어기->스캔경로4->ESP 제어기->TDO;
'1100' : TDI->ESP 제어기->스캔경로3->ESP 제어기->스캔경로4->ESP 제어기->TDO;
'1101' : TDI->ESP 제어기->스캔경로1->ESP 제어기->스캔경로3->ESP 제어기->스캔경로4->ESP 제어기->TDO;
'1110' : TDI->ESP 제어기->스캔경로2->ESP 제어기->스캔경로3->ESP 제어기->스캔경로4->ESP 제어기->TDO;
'1111' : TDI->ESP 제어기->스캔경로1->ESP 제어기->스캔경로2->ESP 제어기->스캔경로3->ESP 제어기->
스캔경로4->ESP 제어기->TDO;
end.
else
other Operation: /* ESP 제어기에서 제공하는 명령어 중 'Path_Config' 명령어를 제외한 명령어 실행 */
}

```

2비트와 Group_ID와 같으면 보드를 선택하고 Select 신호를 활성화한다. 만일 명령어 레지스터에 Address_Select나 Group_Select가 아닌 명령어가 입력되면, ESP 제어기에서 수행하는 명령어 중에서 단일 및 그룹 주소를 선택하는 이 두 명령어를 제외한 다른 명령어를 수행한다. 단일 및 다중 보드의 선택은 절차 1과 같다.

3.3.2 스캔 경로 구성

시험을 위해 선택된 보드는 스캔 경로를 구성한다. ESP 제어기는 한 보드에 최대 4개의 스캔 경로를 구성 및 제어한다. ESP 제어기는 먼저 보드가 선택되어 Select 신호가 활성화되고, TAP 제어기의 상태가 Capture_DR 상태에서 Path_Config 명령어가 입력되면 PATH_CONFIG 블록에 입력된 데이터에 따라 경로를 구성한다. 만약 명령어 레지스터에 Path_Config가 아닌 명령어가 입력되면, ESP 제어기에서 수행하는 명령어 중에서 경로 선택 명령어를 제외한 다른 명령어를 수행한다. 보드 스캔 경로 구성은 절차 2와 같다. 예로서 데이터가 '1001'일 경우 스캔 경로 1과 4를 연결한 ESP 구조는 그림 8과 같다.



(그림 8) 스캔 경로 1과 4로 구성된 ESP 구조
(Fig. 8) ESP architecture configured scan path 1 and 4

3.3.3 IEEE 1149.1 경계면 스캔 시험과 비스트의 병렬 수행

ESP 제어기는 스캔 경로들에 대해서 비스트를 병

렬로 수행하기 위해 다음 두 단계로 동작한다. IEEE 1149.1 경계면 스캔 시험과 비스트의 병렬 수행을 위한 TMS 제어는 절차 3과 같다.

(1) 병렬 시험을 위한 상태 설정

ESP 제어기가 IEEE 1149.1 경계면 스캔 시험을 수행하면서 다른 스캔 경로에 대해 비스트를 수행하려면, 먼저 명령어 레지스터에 TMS_Control 명령어를 적재한다. 절차 2에서 보는 바와 같이 PATH_CONFIG 블록에는 '0000'부터 '1111'까지 4비트의 데이터가 입력되고 그 중 선택되었던 스캔 경로는 '1'로 설정된다. 여기서 '1'로 설정된 스캔 경로는 ESP 제어기의 연결에 의해 경로가 구성된다. 이때 PATH_CONFIG 블록에서 '1'로 설정된 스캔 경로들에 대해서는 IEEE 1149.1 경계면 스캔 구조의 TAP 제어기의 TMS 값에 따르고, '0'으로 설정된 스캔 경로에 대해서는 TMS 값을 '0'으로 고정시켜 비스트를 수행한다. IEEE 1149.1 경계면 스캔 구조에서는 비스트 수행이 TAP 제어기[6]의 RT/I 상태에서만 작동한다. TAP 제어기가 RT/I 상태를 유지하기 위해서는 TMS 값이 '0'이어야 한다. 그러므로 비스트 수행 중인 TMS 값을 '0'으로 고정시켜 ESP 제어기의 TMS 신호가 비스트 수행을 하는 스캔 경로에는 영향을 미치지 않게 한다. 만약 명령어 레지스터에 TMS_Control이 아닌 다른 명령어가 입력되면, ESP 제어기에서 수행하는 명령어 중에서 TMS 제어 명령어를 제외한 다른 명령어를 수행한다.

(2) 병렬 시험 해제

비스트 수행이 끝난 후 그 결과를 TDO로 출력하기 위해서는 고정된 스캔 경로의 TMS와 ESP 제어기의 TMS 사이의 동기를 맞춘다. ESP 제어기에 TMS_Control 명령어를 다시 인가하여 TMS 동기를 맞추고, 비스트 상태를 해제한다.

4. 실험 및 결과

본 연구에서 제안한 ESP 구조의 성능을 평가하기 위해, 먼저 ESP 제어기를 XILINX사의 XACT 툴의 DASH-LCA[20]를 이용하여 설계하였다. 또한 회로의 검증을 위해 SIMUCAD사의 SILOS 툴[21]을 이용하여 시뮬레이션하였다. 실험을 위해 ISCAS '85[22] 벤치마크 회로에 대해 ATALANTA(version 1.1)[23]의 ATPG(Automatic Test Pattern Generator)를 이용


```

/* 절차 3 : TMS 제어 절차 */
Procedure TMS 제어
(
  TAP 제어기의 상태 = Test_Logic_Reset -> Run_Test/Idle -> Select_IR_Scan -> Capture_IR
  for TAP 제어기의 상태 := Capture_IR to Update_IR do
    명령어 레지스터 := TDI
    if ( 명령어 레지스터 = TMS_Control ) and ( Select = 1 )
      then i := 0
         while i ≤ 3 do
           if PATH_CONFIG[i] = 1 /* 시험용 위해 PATH_CONFIG 괄목의 4비트 중에서 선택되어진 스캔경로 */
             then
               TMS[i] := IEEE 1149.1의 TMS /* 선택된 스캔경로는 IEEE 1149.1 TAP 제어기의 TMS 사용 */
             else
               TMS[i] := 0 /* 선택되지 않은 스캔경로는 TMS 값을 '0'으로 고정 */
               i = i + 1
           else
             other Operation: /* ESP 제어기에서 제공하는 명령어 중 'TMS_Control' 명령어를 제외한 명령어 실행 */
        )
    
```

하여 결정적 시험 데이터(deterministic test data)를 표 2와 같이 구하였다. 다음 절차 1, 2, 3을 위한 수행 순서 및 입력되어야 할 데이터를 출력하기 위한 프로그램을 C 언어로 작성하였다. 다음 시험 구조의 수행 시간을 측정하기 위해서 IBM PC에서 SILOS 툴에 프로그램의 출력값을 입력하여 수행시간의 결과치를 측정하였다.

다중 스캔 경로 구조는 그림 4에서 보는 바와 같이 보드 하나가 추가될 때마다 TCK, TMS 신호선이 추가된다. 수십 혹은 수백장으로 구성되는 다중 보드에서 이 신호선의 추가로 인하여 이 신호선을 위한 핀의 수가 계속 늘어나야 하므로 현실적으로 실현하기가 어렵다. 그러므로 제안한 ESP 구조는 기존의 시험 구조인 단일 스캔 경로 구조와 시험 수행 시간의 관점에서 비교하였다. 먼저 표 2의 ISCAS '85 회로를 조합하여 시험을 위한 대상 회로를 구성하였다. 시험 회로의 구성도를 표현하기 위해 $[X, Y]$, $[nX]$, 그리고 $[X]_n$ 을 다음과 같이 정의한다.

- $[X, Y] = \{ \text{ISCAS 회로 } X \text{와 } Y \text{가 연결된 회로} \}$
- $[nX] = \{ \text{ISCAS 회로 } X \text{가 } n \text{개 연결된 회로} \}$ 여기서 X 는 C_0 에서 C_7 까지 중 1개의 회로이다.
- $[X]_n = \{ \text{ISCAS 회로 } X \text{가 } n \text{개 연결된 회로} \}$ 여기서 X 는 C_0 에서 C_7 까지 중 2개 이상이 연결된 회로이다. 예로서 $[C_0, C_1, C_2, C_3]$ 은 ISCAS 회로 C_0, C_1, C_2, C_3 을 연결한 회로이고, $[2C_0]$ 은 ISCAS 회로 C_0 을 2개

연결한 회로이다. 그리고 $[C_0, C_1, C_2, C_3]_4$ 는 ISCAS 회로 C_0, C_1, C_2, C_3 으로 구성된 회로를 4개 연결한 회로이다.

〈표 2〉 ISCAS '85 벤치마크 회로
 〈Table 2〉 ISCAS '85 Benchmark circuits

회로번호	회로이름	게이트 수	입력신호 수	출력신호 수	시험데이터 수
C_0	C17	196	36	7	48
C_1	C432	243	41	32	54
C_2	C499	443	60	26	50
C_3	C1355	587	41	32	84
C_4	C1908	913	33	25	117
C_5	C2670	426	233	140	107
C_6	C3540	1719	50	22	152
C_7	C5315	2485	178	123	125

시험 회로의 구성도는 표 3과 같다.

표 3에서 구성한 시험 회로에 대해 표 4에서는 단일 스캔 경로와 ESP 구조에 대해 각각 5개의 시험을 수행하였다. 각 시험은 스캔 무결성, 익스테스트, 인테스트, 비스트, 그리고 비스트와 인테스트의 혼합 수행이다. 스캔 무결성 시험은 스캔 경로의 결함을 시험한다. 익스테스트는 대상 회로의 상호연결(interco-

〈표 3〉 시험 회로의 구성도
 〈Table 3〉 Configuration of test circuit

시험회로 이름	ISCAS '85 회로를 연결한 시험 회로 구성도	그룹 수	그룹 소자	전체 소자	전체 셀 수
TC ₀	[[4C ₀ , 4C ₁ , 4C ₂ , 4C ₃]]	4	4	16	1100
TC ₁	[[4C ₄ , 4C ₅ , 4C ₆ , 4C ₇]]	4	4	16	3216
TC ₂	[[2C ₀ , 2C ₁ , 2C ₂ , 2C ₃ , 2C ₄ , 2C ₅ , 2C ₆ , 2C ₇]]				
TC ₃	[[C ₀ , C ₁ , C ₂ , C ₃] ₄]	4	4	16	1100
TC ₄	[[C ₄ , C ₅ , C ₆ , C ₇] ₄]	4	4	16	3216
TC ₅	[[C ₀ , C ₁ , C ₂ , C ₃ , C ₄ , C ₅ , C ₆ , C ₇] ₂]	4	4	16	2158
TC ₆	[[4C ₀ , 4C ₁ , 4C ₂ , 4C ₃]]	2	8	16	1100
TC ₇	[[C ₀ , C ₁ , C ₂ , C ₃ , C ₄ , C ₅ , C ₆ , C ₇] ₂]	2	8	16	1100
TC ₈	[[4C ₄ , 4C ₅ , 4C ₆ , 4C ₇]]	2	8	16	3216
TC ₉	[[C ₀ , C ₁ , C ₂ , C ₃ , C ₄ , C ₅ , C ₆ , C ₇] ₂]	2	8	16	2158
TC ₁₀	[[C ₀ , C ₁ , C ₂ , C ₃ , C ₄ , C ₅ , C ₆ , C ₇] ₄]	4	8	32	2200
TC ₁₁	[[C ₄ , C ₅ , C ₆ , C ₇ , C ₄ , C ₅ , C ₆ , C ₇] ₄]	4	8	32	6432
TC ₁₂	[[4C ₀ , 4C ₁ , 4C ₂ , 4C ₃ , 4C ₄ , 4C ₅ , 4C ₆ , 4C ₇]]	4	8	32	4316
TC ₁₃	[[C ₀ , C ₀ , C ₁ , C ₁ , C ₂ , C ₂ , C ₃ , C ₃] ₂ [C ₄ , C ₄ , C ₅ , C ₅ , C ₆ , C ₆ , C ₇ , C ₇] ₂]	4	8	32	4316
TC ₁₄	[[C ₀ , C ₁ , C ₂ , C ₃] ₁₆]	16	4	64	4400
TC ₁₅	[[4C ₀ , 4C ₁ , 4C ₂ , 4C ₃ , 4C ₄ , 4C ₅ , 4C ₆ , 4C ₇] ₂]	16	4	64	8632

〈표 4〉 시험 수행 시간 비교
 〈Table 4〉 Comparison of test time

시험종류	스캔 무결성		전체 익스테스트		전체 인테스트		전체 비스트		비스트 및 인테스트	
	단일스캔 (TCK)	ESP (TCK)	단일스캔 (TCK)	ESP (TCK)	단일스캔 (TCK)	ESP (TCK)	단일스캔 (TCK)	ESP (TCK)	단일스캔 (TCK)	ESP (TCK)
TC ₀	2264	2513	3372	3470	61356	45192	5509	2670	39424	28827
TC ₁	6496	6745	9720	9818	275784	243068	10805	7106	147148	127067
TC ₂	4380	4629	6546	6644	168570	144130	8157	4990	140648	122725
TC ₃	2264	2557	3372	3470	61356	45192	5509	1780	33434	23787
TC ₄	6496	6745	9720	9818	275784	243068	10805	7106	143296	124841
TC ₅	4380	4629	6546	6644	168570	144130	8157	4990	88365	74314
TC ₆	2264	2429	3372	3438	61356	45616	5509	2614	39424	28732
TC ₇	2264	2429	3372	3438	61356	45616	5509	2035	33434	23692
TC ₈	6496	6661	9720	9786	275784	243492	10805	7050	147148	126972
TC ₉	4380	4545	6546	6612	168570	144554	8157	3622	88365	74219

TC ₁₀	4512	4813	6720	6786	121868	90512	10177	2071	66868	47375
TC ₁₁	12976	13277	19416	19482	550724	486264	20769	5449	286589	249483
TC ₁₂	8744	9045	13068	13134	386344	288388	15473	9322	281293	245218
TC ₁₃	8744	9045	13068	13134	386344	288388	15473	9322	281293	245218
TC ₁₅	9008	9929	13416	13658	306700	180768	23929	5296	133730	95025
TC ₁₅	17472	18393	26112	26354	769476	575632	34521	18400	353454	297373

〈표 5〉 시험 수행 시간 비교
(Table 5) Comparison of test time

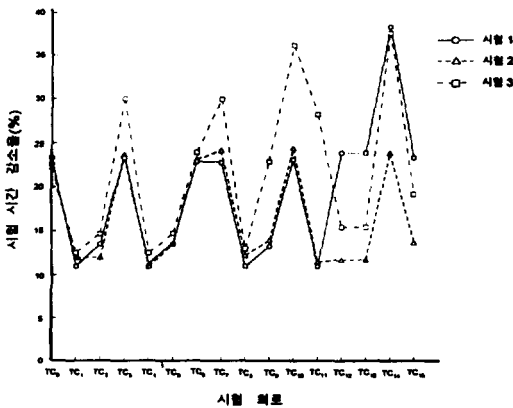
시험종류 시험회로	시험 1			시험 2			시험 3		
	기존시간 (TCK)	개선시간 (TCK)	감소율 (%)	기존시간 (TCK)	개선시간 (TCK)	감소율 (%)	기존시간 (TCK)	개선시간 (TCK)	감소율 (%)
TC ₀	66992	51175	23.61	45060	34810	22.74	11145	8653	22.35
TC ₁	292000	259631	11.08	163364	143630	12.07	27021	23669	12.40
TC ₂	179496	155403	13.42	151574	133998	11.59	19083	16263	14.77
TC ₃	66992	51219	23.54	39070	29814	23.69	11145	7807	29.95
TC ₄	292000	259631	11.08	159512	141404	11.35	27021	23669	12.40
TC ₅	179496	155403	13.42	99291	85587	13.80	19083	16263	14.77
TC ₆	66992	51483	23.15	45060	34599	23.21	11145	8481	23.90
TC ₇	66992	51483	23.15	39070	29559	24.34	11145	7902	29.09
TC ₈	TC8	292000	259939	10.97	163364	143419	12.20	27021	23404
TC ₉	179496	155711	13.25	99291	85376	14.01	19083	147791	22.55
TC ₁₀	133100	102111	23.28	78100	58974	24.48	21409	13670	36.14
TC ₁₁	583116	519023	10.99	318981	282242	11.51	53161	38208	28.12
TC ₁₂	408156	310567	23.90	303105	267397	11.78	37285	31501	15.51
TC ₁₃	408156	310567	23.90	303105	267397	11.78	27285	31501	15.51
TC ₁₄	329124	204355	37.90	156154	118612	24.04	46353	28883	37.68
TC ₁₅	813060	620379	23.40	397038	342120	13.83	78105	63147	19.15

nnection)을 시험한다. 인테스트 시험은 칩 내의 대상 회로의 결함을 시험한다. 비스트 방식은 결정적 시험 데이터(deterministic test data)를 룬에 저장하여 수행한다. 먼저 시작 주소를 초기값으로 경계면 스캔 셀에 입력한다. 그리고 시험 데이터를 읽고, 시험 회로에 인가하고 결과를 압축시키기 위해 TCK를 3회 인가한다. 시험 데이터 전체를 인가한 후 결과를 경계면 스캔 셀을 통해 출력한다. 여기서 TCK의 단위는

1μsec이다.

표 5에서는 두 구조에 대해 3가지 시험을 수행하였다. 시험 1에서는 스캔 무결성, 익스테스트, 그리고 인테스트 시험에 소요되는 전체 시간을 측정하였다. 시험 2에서는 스캔 무결성, 익스테스트, 그리고 인테스트와 50%의 비스트 시험을 혼합하여 수행하는데 소요되는 전체 시간을 측정하였다. 마지막 시험 3에서는 스캔 무결성, 익스테스트, 그리고 100%의 비스

트 시험에 소요되는 전체 시간을 측정하였다. 실험 결과인 표 5에서 3가지 시험 방식에 대해 전반적으로 좋은 감소율을 보이고 있다. 특히 각 모듈에 포함된 소자들의 구성이 동일한 구조에서 더 좋은 결과를 나타내고, 모든 구성 소자 수가 클수록 좋은 결과를 보였다. 단일 스캔 경로 구조와 ESP 구조에서 각 시험 1, 2, 3의 수행 시간 평균값은 그림 9와 같으며 각각 평균 19.25%, 19.84%, 14.14%의 수행 시간이 감소하였다.



(그림 9) 시험 회로에서 시험 시간 감소율
(Fig. 9) Decreasing ratio of test time in test circuits

5. 기존 시험 구조와의 비교 분석

다중 보드 시험을 위한 단일 스캔 경로 구조, 다중 스캔 경로 구조, 그리고 ESP 구조를 비교, 분석한다. N개의 스캔 경로로 구성된 보드가 M개인 다중 보드 시험에서 다중 스캔 경로 구조와 ESP 구조의 추가 부담은 표 6과 같다.

다중 보드 시험을 위한 다중 스캔 경로 구조는 버스 신호선 수는 $2(M \cdot N) + 3$ 개가 필요하고, ESP 구

조일 경우에 버스 신호선 수는 표준안에서 제시한 5개만이 필요하다. 다중 스캔 경로 제어 구조에서, 보드 제어를 위한 TMS 및 TCK 신호 제어기는 $(M \cdot N)$ 개의 TMS 및 TCK 병렬 제어를 위한 하드웨어와 소프트웨어의 부담이 증가하지만 ESP 구조는 TMS 신호를 직렬로 제어함으로써 하드웨어와 소프트웨어의 추가 부담이 감소한다. 다중 스캔 경로 구조인 경우 시험 보드 자체에 추가되는 하드웨어는 없으나 계층적인 시험을 지원하지 못한다. ESP 구조는 시험 보드 자체에 하드웨어가 추가된다. N개의 스캔 경로를 위한 ESP 제어기의 하드웨어 부담은 F/F인 경우에는 $3N + 2 \log N + 29$ 이고, 게이트인 경우는 $6N + 111$ 이다. 그러므로 $O(N)$ 의 하드웨어 부담이 발생하고 M개의 보드일 경우는 $O(MN)$ 이 된다. 그래서 설계한 ESP 제어기와 같이 4개의 스캔 경로를 제어하는 경우에는 45개의 D f/f와 135개의 게이트 하드웨어 부담이 생긴다. 회로도에 의해 계산된 게이트의 수를 비교하면 9.04%의 하드웨어의 부담이 증가한다. 그러나 대상 회로가 복잡해져 경계면 스캔 레지스터가 증가할 수록 하드웨어 부담이 줄어든다. 단일 스캔 경로 구조는 ESP 구조에 비해 여러 가지 시험 방법에서 시험 수행 시간이 많이 소요되고, 스캔 경로가 단락이나 개방으로 결함이 생기거나 빈 슬롯이 생기면 나머지 보드에 대한 올바른 시험 데이터를 입력할 수 없다. 또한 경계면 스캔 시험과 비스트의 병렬 수행이 불가능하다. 그러므로 이 구조는 다중 보드 시험에 적합하지 않다. ESP 구조는 다중 보드 시험 시 5개의 신호선만이 필요한 반면, 다중 스캔 경로 구조는 1개의 보드가 추가될 때마다 TMS 및 TCK의 신호선이 계속 추가된다. 다중 보드 시험에 적용할 경우에 TMS 및 TCK 신호를 계속해서 추가하면 시스템 버스의 신호선 수가 보드 수에 비례하여 증가한다. 그러므로 이러한 문제점 때문에 구현에 따른 어려움이 증가하므로 다중 보드 시험에는 적합하지 않다. 다중

<표 6> 다중 스캔 경로 구조와 ESP 구조의 추가 부담 비교
<Table 6> Comparison overhead of ESP and multi scan test architecture

다중 보드 시험을 위한 추가 부담	다중 스캔 경로 구조	ESP 구조
버스 신호선 수	$2(M \cdot N) + 3$	경계면 스캔 구조의 5개 신호
TMS 및 TCK 신호 제어기	$(M \cdot N)$ 개 TMS 및 TCK 신호의 병렬 제어를 위한 H/W 및 S/W 부담 증가	신호선 제어를 위한 H/W 및 S/W 추가 부담의 감소

보드 시험을 위한 ESP 구조는 스캔 경로상의 결합이 발생하거나, 보드가 제거되어도 전체 시스템에 영향을 주지 않는다. 그러므로 통합 시험이 가능하고, 256개의 보드까지 시험이 가능하다. 선택된 보드에 대해서는 4개의 그룹 분리와 전역 선택이 가능하고, 스캔 경로 4개당 1개의 제어기만이 필요하다.

6. 결 론

본 논문에서는 IEEE 1149.1 경계면 스캔 구조를 가지는 다중 보드 시험을 위한 구조와 절차를 제안하고, 이를 위한 ESP 제어기를 설계했다. 본 연구에서 제안한 ESP 구조의 성능을 평가하기 위해, 먼저 ESP 제어기를 XILINX사의 XACT 툴의 DASH-LCA를 이용하여 설계하였다. 또한 회로의 검증을 위해 SIMUCAD사의 SILOS 툴을 이용하여 시뮬레이션하였다. 제안한 ESP 구조는 기존의 시험 구조인 단일 스캔 경로 구조와 시험 수행 시간의 관점에서 비교하였다. 시험 수행 시간을 측정하기 위해 먼저 ISCAS '85 벤치마크 회로에 대해 ATALANTA 버전 1.1의 ATPG (Auto Test Pattern Generator)를 이용하여 결정적 시험 데이터를 구하였다. 그 다음 절차 1, 2, 3을 위한 수행 순서 및 입력되어야 할 데이터를 구하는 프로그램을 C 언어로 작성하였다. ISCAS '85 벤치마크 회로를 조합하여 시험 회로를 만든 후, IBM PC의 SILOS 툴에서 시험 수행 시간을 ESP 구조와 단일 스캔 경로 구조에 대해 측정하였다. 3가지 실험에서 모두 ESP 구조가 각각 평균 19.40%, 21.71%, 16.65%의 수행 시간이 감소하였다. ESP 구조는 다중 스캔 경로 구조의 스캔 경로마다 독립된 TMS와 TCK 신호선들이 각각 제어되는 것을 외부로는 TMS와 TDI 단일 신호선만으로 4개의 스캔 경로에 제공되는 신호선들의 역할을 수행하도록 설계함으로써 다중 스캔 경로 구조에 비해 신호선 추가가 없었다. 또한 기존의 IEEE 1149.1 경계면 스캔 구조를 가지는 보드에 용이하게 장착할 수 있어 이식성이 높았다. ESP 제어기는 최대 256개의 보드로 구성된 다중 보드 시스템에서의 단일 보드 선택, 그룹 보드 선택 그리고 전역 보드 선택을 지원하여 동일한 시험 동작을 동시에 실시할 수 있어 시험 시간을 단축시켰다. 현재 다중 보드로 구성되어진 통신 시스템의 보드와 시스템 테스트 기법에 대한

연구가 진행중이다. 향후과제로는 본 논문에서 설계, 구현한 ESP 제어기를 이 시스템에 장착하여 시험을 수행하고자 한다.

참 고 문 헌

- [1] H. Fujiwara, 'Logic Testing and Design for Testability', The MIT Press, 1985.
- [2] M. Abramovici, M. A. Breuer and A. D. Friedman, 'Digital Systems Testing and Testable Design', Computer Science Press, 1990.
- [3] P. S. Parikh and M. Abramovici, "On Combining Design for Testability Techniques," in Proc. Int'l Test Conf., pp. 423-429, 1995.
- [4] IEEE std. 1149. 1-1990, 'IEEE Standard Test Access Port and Boundary Scan Architecture', May 21, 1990.
- [5] R. G. Benneets and A. Osseran, "IEEE Standard 1149. 1-1990 on Boundary Scan: History, Literature Survey, and Current Status," JETTA, Vol. 2, No. 1, pp. 11-25, 1991.
- [6] D. Bhavsar, "An architecture for Extending the IEEE Standard 1149. 1 Test Access Port to System Backplanes," in Proc. Int'l Test Conf., pp. 768-776, 1991.
- [7] L. Whetsel, "A Proposed Method of Accessing 1149. 1 in a Backplane Environment," in Proc. Int'l Test Conf., pp. 206-216, 1992.
- [8] A. L. Crouch, M. Pressly and J. Circello, "Testability Features of the MC68060 Microprocessor," in Proc. Int'l Test Conf., pp. 60-69, 1994.
- [9] P. Varma, "On Path-Delay Testing in a Standard Scan Environment," in Proc. Int'l Test Conf., pp. 164-173, 1994.
- [10] S. Narayanan, R. Gupta and M. Breuer, "Configuring Multiple Scan Chains for Minimum Test Time," in Proc. IEEE International Conference on Computer Aided Design pp. 4-8, 1992.
- [11] L. Whetsel, "Hierarchically Accessing 1149. 1 Applications in a System Environment," in Proc.

Int'l Test Conf., pp. 17-21, 1993.

[12] P. Hansen, "The Impact of Boundary Scan on Board Test Strategies," in Proc. ATE and Instrumentation West, 1990.

[13] G. C. Gillette, "A Single Board Test System: Changing the Test Paradigm," in Proc. Int'l Test Conf., pp. 880-885, 1995.

[14] J. Maierhofer, "Hierarchical Self-Test Concept Based on the JTAG Standard," in Proc. Int'l Test Conf., pp. 127-134, 1990.

[15] J. Leenstra and L. Spaanenburg, "Hierarchical Test Assembly for Macro Based VLSI Design," in Proc. Int'l Test Conf., pp. 520-529, 1990.

[16] H. Harmanani, C. Papachristou, S. Chiu and M. Nourani, "An Environment for System-Level Design for Test," in Proc. European Conference on Design Automation, pp. 402-409, 1992.

[17] C. Maunder, "A Universal Framework for Managed Built-in Test," in Proc. Int'l Test Conf., pp. 21-29, 1993.

[18] G. L. Craig, C. R. Kime and K. K. Saluja, "Test Scheduling and Control for VLSI Built-In Self-Test," IEEE Trans. on Computers, Vol. 37, No. 9, pp. 1099-1109, 1988.

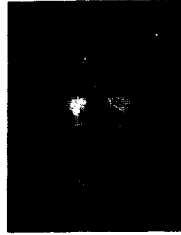
[19] V. D. Agrawal, "A Tutorial on Built-In Self-Test," IEEE Design & Test of Computers, pp. 69-77, 1993.

[20] FutureNet Co., 'Dash Schematic Designer User's Guide', 1987.

[21] Simucad Co., 'PC-SILOS User's manual', 1992.

[22] F. Brglez, D. Bryan, and K. Kozminski, "Combinational Profiles of Sequential Benchmark Circuits," in Proc. Int. Symp. on Circuits and Systems, pp. 1929-1934, June. 1989.

[23] H. K. Lee and D. S. Ha, "Atalanta: an Efficient ATPG for Combinational Circuits," Technical Report, 93-12, Electronic Testing Group, Dep't of Electrical Eng., Virginia Polytechnic Institute and State University, Blacksburg, Virginia.



손 우 정

1989년 경북대학교 공과대학 전자공학과 졸업(공학사)
 1993년 경북대학교 대학원 컴퓨터공학과 졸업(공학석사)
 1996년 경북대학교 대학원 컴퓨터공학과 박사과정 수료
 1994년~현재 대구전문대학 전자

계산과 조교수
 관심분야: VLSI 테스트, VLSI 설계, CAD 데이터베이스



윤 태 진

1994년 경북대학교 공과대학 컴퓨터공학과 졸업(공학사)
 1996년 경북대학교 대학원 컴퓨터공학과 졸업(공학석사)
 1996년~현재 경북대학교 대학원 컴퓨터공학과 박사과정
 관심분야: VLSI 테스트, VLSI 설

계, VHDL



안 광 선

1972년 연세대학교 전자공학과 졸업(공학사)
 1980년 연세대학교 대학원 전자공학과 졸업(공학박사)
 1975년~1976년 스페리유니버시티 근무
 1977년~현재 경북대학교 컴퓨터공학과 정교수

1981년~1982년 미국 아리조나시립대학 객원교수
 1990년~1991년 미국 텍사스주립대학 객원교수
 1995년~1996년 미국 버지니아대학 객원교수
 관심분야: VLSI 테스트, VLSI 설계, 컴퓨터구조