

Network-on-Chip 시스템을 위한 새로운 내장 자체 테스트 (Built-In Self-Test) 구조

이건호, 김인수, 민형복
성균관대학교

The Novel Built-In Self-Test Architecture for Network-on-Chip Systems

Keon-Ho Lee, Insoo Kim, Hyoung Bok Min
Sungkyunkwan University

Abstract - NoC 기반 시스템이 적용되는 설계는 시스템 크기가 커짐에 따라 칩 테스트 문제도 동시에 제기 되고 있다. 이에 따라 NoC 기반의 시스템의 테스트 시간을 줄일 수 있는 internal test 방식의 새로운 BIST(Built-in Self-Test) 구조에 관한 연구를 하였다. 기존의 NoC 기반 시스템의 BIST 테스트 구조는 각각의 router와 core에 BIST logic과 random pattern generator로 LFSR(Linear Feedback Shift Register)를 사용하여 연결하는 individual 방식과 하나의 BIST logic과 LFSR를 사용하여 각각의 router와 core에 병렬로 연결하는 distributed 방식을 사용한다. 이때, LFSR에서 생성된 테스트 벡터가 router에 사용되는 FIFO 메모리를 통과하면서 생기는 테스트 타임 증가를 줄이기 위하여 shift register 형태의 FIFO 메모리를 변경하였다 제안된 방법에서 테스트 커버리지 98%이상을 달성하였고, area overhead면에서 효과를 볼 수 있다.

1. 서 론

최근 SoC 설계는 GIGA Hertz범위 안의 동작 주파수에서 1억 개의 트랜지스터가 집적되어 있는 것을 볼 수 있다. 또한 10년 안에 약 10억 개의 트랜지스터가 집적된 설계가 이루어 질 것이며, 이와 같은 설계는 한 개의 칩 안에 CPU(Central Processing Unit), DSP(Digital Signal Processor), memory(RAM and ROM), mixed-signal장치, buses/interconnect, glue logic, ASIC(Application-Specific Integrated Circuit), IP들로 구성되어 있다.

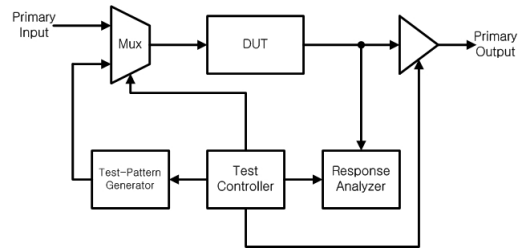
이와 같은 칩의 설계 기술 및 공정 기술이 최근 급격히 발전함에 따라서 완성된 시스템의 정상 동작 여부를 판단하는 테스트의 중요성도 급속히 증가 하고 있으며, 그에 따른 시간(time)과 비용(cost) 등도 동시에 증가 하고 있다. 기존에 존재하던 테스트 기술인 DFT(Design for Testability), BIST(Built-in Self-Test), Boundary Scan등이 설계 기법과 함께 발전하고 있지만 칩이 SoC화 되면서 완벽한 테스트를 기대할 수 없게 되었다. 이에 따라 SoC 테스트를 위한 새로운 표준 방식인 IEEE 1500(IEEE Standards 1500)이 소개 되었고 현재 대부분 SoC 칩은 테스트를 위해 IEEE 1500에 맞게 설계되고 있다.[1]

NoC 기반의 시스템 또한 복잡한 설계에 따른 새로운 테스트 방식이 요구되고 있지만, 기존의 SoC 테스트 방법을 NoC를 구성하고 있는 라우터, 네트워크 인터페이스(라우터와 임베디드 코어의 연결 부분), 연결선(interconnect) 등의 테스트에 바로 적용할 수 없다. 이에 따라 NoC 기반의 시스템의 테스트 방식은 아직까지 활발히 연구 중이고 표준화의 필요성이 대두 되고 있다.

2. 본 론

2.1 BIST 구조

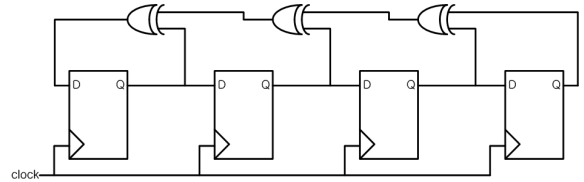
그림 1은 일반적인 BIST(Built-In Self-Test) 구조를 나타내고 있다. normal mode 일 때는 DUT(Design under Test)는 primary input에 인가된 신호에 의하여 동작된다. test mode 일 때는 내장된 test-pattern generator에 의해 test pattern이 DUT 내에 인가된다. 인가된 input pattern에 의해 기대되는 pattern을 response analyzer에 의해 판단하여 test-fault 유무를 판단한다.[2]



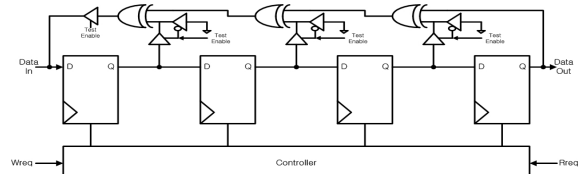
<그림 1> BIST Architecture

2.2 Proposed Linear Feedback Shift Register

BIST 구조에서 사용되는 테스트 패턴 생성기로는 그림 2와 같은 LFSR(Linear Feedback Shift Register)를 사용한다. NoC에서 사용하는 라우터의 shift register 형태의 FIFO memory를 BIST 수행시에 LFSR로 동작 할 수 있는 회로를 제안하려고 한다. 제안된 회로는 그림 3과 같다.



<그림 2> LFSR(Linear Feedback Shift Register)

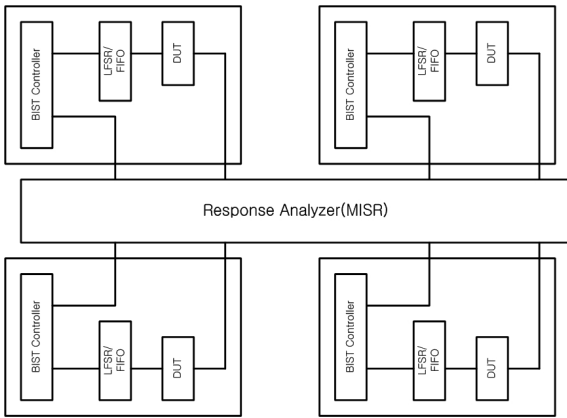


<그림 3> 제안된 LFSR의 구조

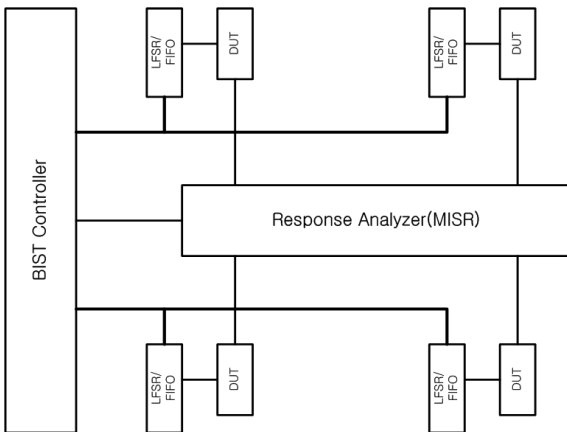
2.3 Network-on-Chip Systems를 위한 BIST 구조

2.2에서 제안된 LFSR를 이용하여 NoC 시스템에 적용한 BIST 구조를 그림 4와 그림 5 두 가지 형태로 나타낼 수 있다. 그림 4는 각각의 라우터와 코어에 BIST 회로를 구성하여 테스트를 수행하는 방식으로 테스트를 수행하기 원하는 코어를 선택하여 테스트 수행을 할 수 있다.

그림 5는 한 개의 BIST controller로 라우터와 코어로 연결된 NoC 시스템 전체를 동시에 테스트를 수행을 할 수 있다.



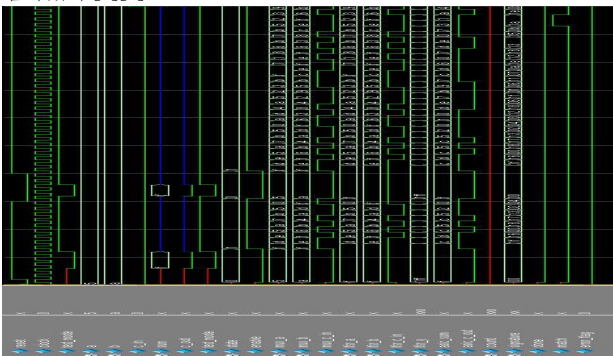
<그림 4> Individual 방식의 NoC BIST 구조



<그림 5> Distribution 방식의 NoC BIST 구조

3. 결 론

제안된 LFSR를 사용한 NoC BIST 구조의 검증을 위하여 ITC99 벤치마크 회로와 opencores에서 제공되는 OpenRISC 1200을 사용하였다.[3][4] 논문에서 다루어지는 모든 회로는 Mentor Graphics의 ModelSim Simulator를 사용하여 동작 검증을 하였고, synthesis 및 scan chain insertion은 Synopsys사의 Design Compiler를 사용하였다. 또한, 테스트 time 시간 및 테스트 커버리지(coverage) 측정은 Synopsys사의 ATPG tool인 TetraMax를 사용하였다. Synthesis를 위한 standard cell은 TSMC 90nm 공정을 사용하였다. 모든 실험은 Intel Core Duo 2.0GHz CPU와 4Gbyte memory가 장착된 Linux 시스템에서 실행하였다.[5][6]



<그림 6> BIST Controller의 시뮬레이션 파형

테스트 회로로 사용한 ITC99 벤치마크회로는 많은 scan chain 구성을 위하여 Flip-Flop 개수가 많은 b18과 b19 회로를 사용하였다. 또한, OpenRISC 1200을 사용하여 32bit RISC Core에 적용하여 실험을 하였다.

그림 7은 32bit LFSR 사용하기 위하여 테스트 회로에 design compiler를 사용하여 synthesis한 후 scan chain을 insertion한 결과이다.

Scan_path	Len	ScanDataIn	ScanDataOut	ScanEnable	MasterClock
I 1	7	test_s1	test_s0	test_se	clock
I 2	7	test_s13	test_s02	test_se	clock
I 3	7	test_s16	test_s03	test_se	clock
I 4	7	test_s14	test_s04	test_se	clock
I 5	7	test_s15	addr[1]	test_se	clock
I 6	7	test_s16	addr[8]	test_se	clock
I 7	7	test_s17	addr[15]	test_se	clock
I 8	7	test_s18	data0[0]	test_se	clock
I 9	7	test_s19	data0[7]	test_se	clock
I 10	7	test_s110	data0[14]	test_se	clock
I 11	7	test_s111	data0[21]	test_se	clock
I 12	7	test_s112	data0[28]	test_se	clock
I 13	7	test_s113	rd	test_se	clock
I 14	7	test_s114	test_s014	test_se	clock
I 15	7	test_s115	test_s015	test_se	clock
I 16	7	test_s116	test_s016	test_se	clock
I 17	7	test_s117	test_s017	test_se	clock
I 18	7	test_s118	test_s018	test_se	clock
I 19	7	test_s119	test_s019	test_se	clock
I 20	7	test_s120	test_s020	test_se	clock
I 21	7	test_s121	test_s021	test_se	clock
I 22	7	test_s122	test_s022	test_se	clock
I 23	7	test_s123	test_s023	test_se	clock
I 24	6	test_s124	test_s024	test_se	clock
I 25	6	test_s125	test_s025	test_se	clock
I 26	6	test_s126	test_s026	test_se	clock
I 27	6	test_s127	test_s027	test_se	clock
I 28	6	test_s128	test_s028	test_se	clock
I 29	6	test_s129	test_s029	test_se	clock
I 30	6	test_s130	test_s030	test_se	clock
I 31	6	test_s131	test_s031	test_se	clock
I 32	6	test_s132	wr	test_se	clock

<그림 7> ITC99 b18 회로의 Scan Chain

검증된 BIST controller와 scan chain insertion을 한 테스트 회로를 사용하여 그림 4와 그림 5와 같이 두 가지 형태의 BIST 구조로 설계하였으며 Random Pattern Generator로 32bit LFSR를 사용하였다. 설계된 BIST회로의 test time과 test coverage를 측정하기 위하여 TetraMax를 사용하여 측정하였고 표 1과 표 2는 결과를 보여주고 있다.

Test Circuit	Gates	Primary Input	Primary Output	Flip-Flops	Test Vector	Test Fault	Test Time(sec)	Test Coverage (%)
ITC99 b18	114,621	36	23	3,320	7,760	234,904	5.04	99.82
ITC99 b19	231,320	21	30	6,642	7,460	236,648	5.36	99.96
OpenRISC 1200	474,501	167	207	67,827	97,056	6,588,672	883.48	99.99

<표 1> Individual 방식의 NoC BIST 구조 실험 결과

Test Circuit	Gates	Primary Input	Primary Output	Flip-Flops	Test Vector	Test Fault	Test Time(sec)	Test Coverage (%)
ITC99 b18	114,621	36	23	3,320	7,760	234,904	2.44	99.82
ITC99 b19	231,320	21	30	6,642	7,460	236,648	2.34	99.96
OpenRISC 1200	474,501	167	207	67,827	97,056	6,588,672	458.44	99.99

<표 2> Distribution 방식의 NoC BIST 구조 실험 결과

표 1과 2의 실험 결과로 제안된 BIST 구조의 test coverage는 99%이상 나타나는 것을 알 수 있고, 전체 칩 test time에서는 distribution 방식이 individual 방식에 비하여 test time 단축효과를 나타내는데 알 수 있다.

3. 결 론

본 논문에서는 NoC 시스템에서 사용하는 라우터의 FIFO 메모리를 LFSR로 이용한 BIST 구조를 제안하였다. FIFO 메모리를 LFSR로 사용할 수 있게 하여 각 코어의 테스트 수행을 추가적인 별도의 LFSR없이 수행 할 수 있고, BIST시 문제가 되는 area overhead 문제에도 도움이 될 수 있다. 또한, 각각의 코어에 개별적인 LFSR를 사용하므로 인해 individual 방식에서는 별도의 테스트를 수행하기 원하는 코어의 테스트 수행을 가능하게 할 수 있다. Distribution 방식은 일괄적인 전체 칩 테스트 모드에서 individual 방식에 비해 test time을 단축시킬 수 있고, 코어 전체가 동일한 multi-processor 형태가 아닌 다양한 기능의 코어를 가질 시에는 두 방식을 동시에 사용하는 mixed 방식을 사용할 수 있다.

NoC 시스템 형태에 따라 individual 방식과 distribution 방식 중 적합한 형태의 BIST를 구조를 선택하므로 인해 효율적인 테스트를 수행할 수 있을 것으로 판단된다.

[참고 문헌]

- [1] Laung-Terng Wang "System-on-Chip Test Architectures", Morgan Kaufmann Publishers, 2007
- [2] Miron Abramovici "Digital Systems Testing and Testable Design", IEEE Press, 1990
- [3] <http://www.opencores.org/>

- [4] <http://www.cad.polito.it/tools/itc99.html>
- [5] <http://www.synopsys.com/>
- [6] <http://www.mentor.com/>