

# RTL수준의 데이터패스 모듈을 위한 상위 수준 테스트 합성 기법

김성일<sup>U</sup>                      김석윤                      장훈  
승실대학교                  컴퓨터구조연구실  
kimsi@watt.soongsil.ac.kr, {ksy, hoon}@comp.ssu.ac.kr

## A Priority based Non-Scan DFT Method for Register-Transfer Level Dapapaths

Sung-il Kim<sup>U</sup>                  Seok-Yun Kim                  Hoon Chang  
Dept. of Computing, Soongsil University

### 요 약

본 논문에서는 RTL 회로의 데이터패스에 대한 테스트 용이도 분석방법과 테스트 용이화 설계방법을 제안한다. 데이터패스에 대한 테스트 용이도 분석은 컨트롤러에 대한 정보없이 RTL 회로의 데이터패스만으로 수행한다. 본 논문에서 제안한 테스트를 고려한 설계방법은 내장된 자체 테스트(BIST)나 주사(scan)방식이 아니며, 주사 방식을 적용했을 때에 비해 본 논문에서 제안한 테스트 용이화 설계 방법을 적용했을 때에 보다 적은 면적 증가율(area overhead)을 보인다는 것을 실험을 통해 확인하였다. 또한, 회로 합성 후 ATPG를 통해 적은 면적 증가만으로 높은 고장 검출율(fault coverage)을 얻을 수 있음을 보인다.

### 1 서론

칩의 집적도가 날로 커짐에 따라서 칩에 대한 높은 신뢰도 보장은 더욱 중요한 문제로 대두되고 있다. 이러한 칩의 신뢰도 보장을 위해서 테스트 점점 삽입(test point insertion)방식, 내장된 자체 테스트(BIST) 방식, 주사(scan) 방식과 같은 많은 테스트를 고려한 설계(DFT) 방식들이 제안되었다.

그러나 위에 열거한 방식들은 게이트 수준의 회로에 적용하기 위한 방식들이었기 때문에 회로에 집적되는 게이트 수가 커짐에 따라 위의 방식들을 적용하는데 걸리는 시간과 컴퓨터 자원이 방대해지게 된 문제점을 해결하기 위해서 최근에 게이트 수준이 아닌 RTL(Register Transfer Level) 회로에 대해서 문제에 접근하려는 많은 연구들이 진행되어 왔다. RTL 회로는 게이트 수준의 회로에 비하여 구성모듈들의 숫자가 훨씬 적기 때문에 적은 시간과 비용으로 테스트 용이도를 분석할 수 있다는 장점이 있다.

RTL 회로에 대한 테스트를 고려한 설계방법으로서 내장된 자체 테스트 방식[1], 주사방식[2]과 같은 방식들이 제안되었다. 주사 방식은 순차회로에 대한 ATPG에 비해서 테스트 패턴을 생성하는데 걸리는 시간이 적다는 장점이 있으나, 동작속도테스팅(at speed testing)이 불가능하다는 단점을 갖고 있다.

본 논문에서는 RTL 회로에 대해서 내장된 자체테스트 방식이나 주사 방식을 적용하지 않고 테스트 용이도를 향상시키는 방안을 제안한다. 제안하는 방식은 컨트롤러에 대한 정보 없이 데이터패스만을 고려하여 테스트 용이도를 분석한다. 이 때 컨트롤러로부터 데이터패스로 인가되는 제어신호들은

완벽하게 제어할 수 있다고 가정한다. 본 논문에서 제안된 방식을 적용할 때 필요한 입력은 RTL 데이터패스와 그 데이터패스에서 어떠한 모듈이 기능(functional)모듈인가에 대한 정보이다.

본 논문에서 제안한 방식은 결과로서 각각의 테스트용 멀티플렉서가 데이터패스에 삽입될 위치와 함께 삽입 우선순위(priority)를 제공하며, 이러한 결과 정보를 이용하여 우선순위가 높은 테스트용 멀티플렉서를 우선적으로 삽입함으로써 테스트 용이도가 높은 회로를 설계할 수 있다.

### 2 RTL 회로의 데이터패스

RTL 회로는 주입력, 주출력과 기능모듈(덧셈기, 뺄셈기, 곱셈기 등), 멀티플렉서, 그리고 메모리 모듈(latch, register)로 구성된다. 즉, RTL 회로는 주입력의 집합  $I$ , 주출력의 집합  $O$ , 기능모듈의 집합  $F$ , 메모리 모듈의 집합  $M$ , 그리고 멀티플렉서의 집합  $X$ 로 구성된다. RTL회로에서  $N$ 을 주입력의 수,  $M$ 을 주출력의 수,  $A$ 를 기능모듈의 수,  $B$ 를 메모리모듈의 수, 그리고  $C$ 를 멀티플렉서의 수라 할 때 각각의 집합  $I, O, F, M, X$ 는

$$I = \{ i_n \mid i_n \text{은 주입력}, 1 \leq n \leq N \},$$

$$O = \{ o_m \mid o_m \text{은 주출력}, 1 \leq m \leq M \},$$

$$F = \{ f_a \mid f_a \text{은 기능모듈}, 1 \leq a \leq A \},$$

$$M = \{ m_b \mid m_b \text{은 메모리모듈}, 1 \leq b \leq B \}$$

$$X = \{ x_c \mid x_c \text{은 멀티플렉서}, 1 \leq c \leq C \} \text{가 된다. 기능모듈 } f_a$$

의 입력의 수를  $K$ , 출력의 수를  $P$ 라고 할 때  $f_a$ 의 입력과 출력은 다음과 같은 두 집합으로 각각 나타낼 수 있다.

$$f_{a,i} = \{ i_{k, f_a} \mid i_{k, f_a} \text{는 } f_a \text{의 입력}, 1 \leq k \leq K \},$$

$$f_{a,o} = \{ o_{p, f_a} \mid o_{p, f_a} \text{는 } f_a \text{의 출력}, 1 \leq p \leq P \}.$$

RTL 회로에서 모든 기능모듈들의 입력과 출력의 집합은

$$F_i = \bigcup_{a=1}^A f_{a,i}, F_o = \bigcup_{a=1}^A f_{a,o} \text{로 나타낼 수 있다.}$$

그리고 내부 연결선을 제외한 RTL 회로 전체를 집합  $U$ 라 하고 할 때 집합  $U$ 는 다음과 같이 나타낼 수 있다.

$$U = \{ u_j \mid u_j \in I \cup O \cup F_i \cup F_o \cup M \cup X, 1 \leq j \leq (N + O + \sum_{a=1}^A |f_{a,i}| + \sum_{a=1}^A |f_{a,o}| + B + C) \}.$$

그림 1은 타 논문들에서 벤치마크 회로로 많이 사용되고 있는 RTL 회로인 paulin을 보여주고 있다[3]. Paulin 회로의 주입력, 주출력, 기능모듈, 메모리모듈 그리고 멀티플렉서는 각각 집합

$$I = \{ i_1, i_2 \} = \{ INPORT1, INPORT2 \},$$

$$O = \{ o_1, o_2 \} = \{ OUTPORT1, OUTPORT2 \},$$

$$F = \{ f_1, f_2, f_3, f_4 \} = \{ ADDER, MLT1, MLT2, SUB \},$$

$$M = \{ m_1, m_2, m_3, m_4, m_5, m_6, m_7 \} = \{ REG1, REG2, REG3, REG4, REG5, LATCH1, LATCH2 \},$$

$X = \{ x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11} \}$ 로 나타낼 수 있다. 11개의 멀티플렉서에 필요한 제어신호인 ml1-ml11과 메모리모듈을 위한 제어신호인 load1-load5는 콘트롤러에서부터 데이터패스로 인가되는 신호들이다. 본 논문에서 이들 제어 신호들은 주입력과 마찬가지로 완벽하게 제어할 수 있다고 가정하였다

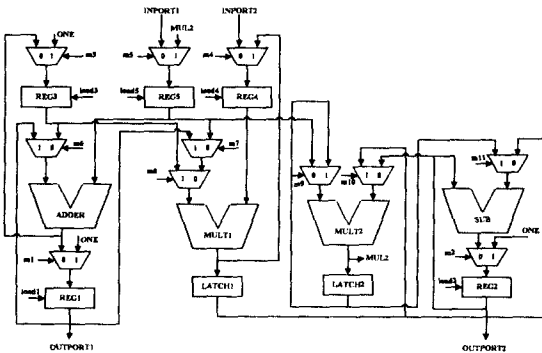


그림 1 Paulin

### 3. 테스트용 멀티플렉서의 삽입 방식

본 논문에서 제안한 그림 2의 테스트용 멀티플렉서의 삽입 방식은 기능모듈에 대해서만 제어도와 관측도를 분석하며 메모리모듈과 멀티플렉서는 기능모듈에 비해 로직이 매우 간단하기 때문에 제어도와 관측도 분석을 하지 않는다. 그러므로 제안한 방식을 적용하기 위해서는 테스트 용이도를 분석하고자 하는 RTL 회로의 데이터패스와 함께 어떠한 모듈이 기능모듈인지를 우선 알고 있어야 한다.

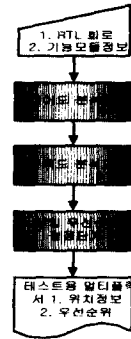


그림 2 테스트용 멀티플렉서 삽입 방식

기능모듈  $f_a$ 의 제어도와 관측도 분석은 RTL 회로를 구성하는 모듈  $u_j$ 들 간의 경로를 이용하며 RTL 회로에서 모듈  $u_j$ 들 간의 경로는 순서가 있는  $n$ 짜리(ordered  $n$ -tuple)으로 나타낼 수 있다. 모듈  $u_j$ 들 간의 경로는 기능모듈  $f_a$ 에 대한 제어경로와 관측경로를 형성한다.

정의 3.1: 기능모듈  $f_a$ 에 대한 제어경로(controllable path)는 집합  $C_{f_a}$ 이다.

정의 3.2: 기능모듈  $f_a$ 에 대한 관측경로(observable path)는 집합  $O_{f_a}$ 이다.

첫 번째 과정인 제어도 분석단계에서는 집합  $C_{f_a}$ 의 원소인  $n$ 짜리들의 마지막 원소들의 집합을  $L_{C_{f_a}}$ , 제어도가 낮은 기능모듈들의 집합을  $B_f$ 로 정의하여 기능모듈  $f_a (\in B_f)$ 의 제어도를 높이기 위한 테스트용 멀티플렉서의 삽입위치를 결정한다. 제어도가 낮은 기능모듈의 입력단을 위해서 테스트용 멀티플렉서를 삽입하면서 다른 모듈에 대한 제어도 역시 향상시킬 수 있는 위치에 테스트용 멀티플렉서를 위치시켜야 한다.

두 번째 과정인 관측도 분석단계에서는 집합  $O_{f_a}$ 의 원소인  $n$ 짜리들에서 첫 번째 원소들의 집합을  $L_{O_{f_a}}$ , 관측도가 낮은 기능모듈들의 집합을  $B'_f$ 로 정의하여 기능모듈  $f_a (\in B'_f)$ 의 관측도를 높이기 위한 테스트용 멀티플렉서의 삽입위치를 결정한다.

삽입 우선순위 결정단계에서는 제어도 분석단계와 관측도 분석단계에서 제어도와 관측도 향상을 위해 추가하기로 결정한 테스트용 멀티플렉서들의 위치 중 RTL 회로 내의 다른 모듈들에 대해서도 제어도 또는 관측도를 부가적으로 향상시키는 순서로 삽입위치의 우선순위를 결정하게 된다. 제어도와 관측도 분석단계에서 삽입 위치가 결정된 테스트용 멀티플렉서들의 집합을  $T$ 라 할 때, 하나의 테스트용 멀티플렉서  $tx_h (\in T, 1 \leq h \leq |T|, \text{단 } |T| \neq 0)$ 의 추가로 새롭게 형성될 제어경로와 관측경로를 나타내는  $n$ 짜리들의 원소들의 집합을  $N_{tx_h}$ 라 하면  $tx_h$ 의 삽입 우선순위는 다음과 같이 정의할 수 있다.

정의 3.3: 테스트용 멀티플렉서  $tx_h$ 의 삽입 우선순위  $P_{tx_h}$ 는  $P_{tx_h} = |N_{tx_h}|$ 이다.

우선순위가 높은 테스트용 멀티플렉서를 삽입하면 면적증가율을 줄이면서 높은 고장검출율을 기대할 수 있다. 테스트용

멀티플렉서의 추가 이외에 RTL 회로에는 테스트 모드 입력 단 하나와 테스트용 멀티플렉서에 필요한 제어신호를 만들기 위한 간단한 로직이 추가된다.

4. RTL회로의 데이터패스에 대한 테스트 용이도 분석

그림 1의 paulin 테스트 용이도 향상을 위해 추가될 테스트용 멀티플렉서들의 위치를 그림 3에서 보여주고 있다.

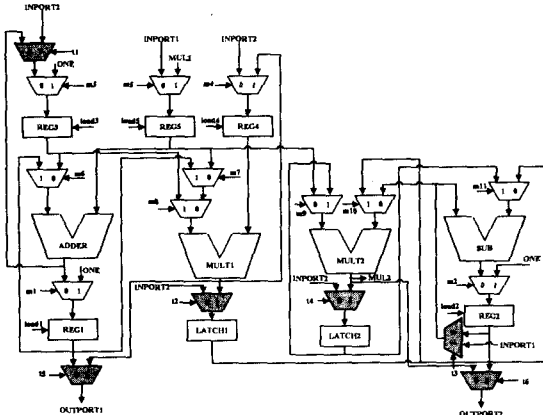


그림 3 Paulin에서 추가될 테스트용 멀티플렉서(어두운 부분)

그림 3에서 보여주고 있는 paulin의 테스트 용이도 향상을 위해 삽입할 수 있는 테스트용 멀티플렉서  $T = \{tx_1, tx_2, tx_3, tx_4, tx_5, tx_6\}$ 의 우선순위는 각각  $P_{tx_1}=8, P_{tx_2}=7, P_{tx_3}=5, P_{tx_4}=7, P_{tx_5}=3, P_{tx_6}=3$ 이며  $tx_1$ 가 가장 높은 삽입 우선순위를 갖는다.

5. 실험결과

표 1은 본 논문에서 제안한 테스트용 멀티플렉서 삽입방식의 효과를 실험해보기 위해 선택한 RTL 회로인 paulin과 tseng에 대한 특성을 나타내고 있다.

표1 실험 RTL 회로의 특성

Circuit	Characteristics		
	Bit-width	# FU	# Test mux
Paulin	16	4	1(6)
Tseng	32	5	1(4)

DFT 방식을 적용하지 않았을 때의 고장검출율은 표 2의 두 번째 열인 Original에 나타나 있으며 세 번째 열은 본 논문에서 제안한 방식을 적용했을 때의 고장검출율을 나타내고 있다. Paulin의 경우 총 여섯 개의 테스트용 멀티플렉서들 중 우선순위가 가장 높은 하나의 테스트용 멀티플렉서를 삽입했을 때 100%의 고장검출율을 얻을 수 있었다.

표2 고장검출율(Fault coverage)

Circuit	Fault Coverage(%)	
	Original	Proposed
Paulin	98.59	100(1/6)
Tseng	90.17	95.64(1/4)

표 3은 벤치마크 회로에 어떠한 DFT 방식도 적용하지 않았을 때와 주사사슬(scan chain)을 삽입했을 때의 면적증가율을 비교하여 보여주고 있다. 실험에서는 kg75 라이브러리를 사용하여 Synopsys로 합성한 후 면적을 구했다.

표3 면적증가율(Area overhead)

Circuit	Total cell area			
	Original	Partial scan	Full scan	Proposed
Paulin	3215	3375(4.98%)	3430(6.69%)	3269(1.64%)(1/6)
Tseng	6651	6972(4.82%)	7036(5.78%)	6754(1.55%)(1/4)

표 4는 paulin과 tseng에 대해서 Syntest를 이용하여 ATPG를 했을 때의 cpu time을 초단위로 나타내고 있다.

표4 cpu time

Circuit	cpu time(sec)	
	Original	Proposed
Paulin	1082	52(1/6)
Tseng	1743	2138(1/4)

6. 결론

본 논문에서는 RTL 수준 회로의 콘트롤러에 대한 정보없이 데이터패스에 대한 정보만을 갖고 테스트용이도를 분석하여 테스트용이도가 낮은 기능모듈을 위해 테스트용 멀티플렉서를 삽입해야 할 위치와 우선순위를 결정하는 방식을 제시하였다. 실험결과를 통해 RTL 수준 회로의 데이터패스 정보만을 이용한 테스트 용이도 분석을 통해서도 높은 고장검출율과 짧은 테스트패턴 생성시간을 얻을 수 있음을 확인하였다.

7. 참고 문헌

- [1] I. Ghosh, N. K. Jha, S. Bhawmik, "A BIST Scheme for RTL Circuits Based on Symbolic Testability Analysis," *IEEE Trans. on CAD*, vol. 19, no. 1, Jan. 2000.
- [2] S. Bhattacharya, F. Brglez and S. Dey, "Transformations and resynthesis for testability of RTL control-data path specifications," *IEEE Trans. VLSI Syst.*, vol. 1, Sept. 1993.
- [3] I. Ghosh, A. Raghunathan, N. K. Jha, "A Design-for-Testability Technique for Register-Transfer Level Circuits Using Control/Data Flow Extraction," *IEEE Trans. on CAD*, vol. 17, Aug. 1998.
- [4] K. Takabatake, M. Inoue, T. Masuzawa and H. Fujiwara, "Non-scan design for testable data paths using thru operation," in *Proc. of Asia and South Pacific Design Automation Conference*, 1997.
- [5] H. Wada, T. Masuzawa, K. K. Saluja, H. Fujiwara, "Design for Strong Testability of RTL Data Paths to Provide Complete Fault Efficiency," *13th International conference on VLSI Design*, Jan. 2000.
- [6] S. Dey, M. Potkonjak, "Non-Scan Design-For-Testability of RT-Level Data Paths," in *Proc. Int. Conf. CAD*, Nov. 1994.
- [7] Y. Makris, A. Orailoglu, "DFT Guidance through RTL Test Justification and Propagation Analysis," *Proceedings of the International Test Conference*, 1998.
- [8] S. Ravi, G. Lakshminaravana, N.K. Jha, "TAO: Regular Expression based High-Level Testability Analysis and Optimization," *Proceedings of the International Test Conference*, 1998.
- [9] I. Ghosh, A. Raghunathan, N.K. Jha, "Design for Hierarchical Testability of RTL Circuits Obtained by Behavioral Synthesis," *IEEE Trans. on CAD*, vol. 16, no. 9, 1997,