

통합된 FPGA 개발 방법 및 환경

曹漢鎮, 嚴洛雄, 裴英煥,
白瑛錫, 朴永秀, 朴仁學, 金景洙
韓國電子通信研究所

요 약

본 논문은 원판과 전용 CAD 툴로 구성되는 FPGA 시스템을 개발하는데 있어서 서로 다른 요소 기술들의 관계와 이들 요소 기술들과 시스템 성능의 관계를 모델하여 시스템 사양을 만족하기 위하여 가장 효율적인 방법을 찾게하는 방법에 관한 것이다. 본 논문에서는 실제로 개발된 시스템을 예로 하여 FPGA 시스템 개발에서 고려해야 할 점들을 고찰하였다. 새로운 FPGA 시스템의 개발 순서는 먼저 개발할 FPGA의 응용 분야를 결정하고, 그 응용 분야에 필요한 시스템 사양에 맞게 개발한 요소 기술들과 그 기술들의 범위를 정한다. 개발 흐름도를 이용하여 이 요소 기술들의 연관 관계를 수직적으로는 시스템 성능에 미치는 영향을 모델링하고 수평적으로는 요소 기술간의 서로 미치는 영향을 모델링 하여 시스템 사양을 만족하기 위한 최적의 해를 구한다. 이때 최종적인 FPGA 시스템을 평가하고 검증할 수 있는 방법을 결정한다. 요소 기술들이 개발 됨에 따라 좀 더 구체적이고 정확한 모델에 의해 전체 시스템의 성능은 평가되고 검증될 수 있다. 이러한 방법과 환경은 FPGA 시스템을 빠르고 효율적으로 개발할 수 있게 한다.

I. 서 론

FPGA(Field Programmable Gate Array)는 전자회로 설계자가 현장에서 직접 집적회로를 구현할 수 있는 새로운 형태의 반주문형 반도체(ASIC: Application Specific Integrated Circuit)이다. 범용 회로의 구현이 어렵고 용량이 적어 응용 범위가 제한된 PLD(Programmable Logic Devices)의 단점과 제작 시간과 비용(Non-Recurring Engineering Cost)이 많이 드는 ASIC의 단점을 해결한 것이 FPGA이다. 반면, FPGA는 프로그램을 위한 복잡한 배선 구조로 말미암아 동작 속도나 집적도가 주문형 반도체에 비해 떨어

지는 단점이 있다^[1]. 현재 FPGA가 주로 사용되고 있는 응용 분야는 칩의 주기(life-cycle)가 짧은 ASIC의 대체용이나, 논리 에뮬레이터(logic emulator) 분야와 논리 가속기(logic accelerator) 등이다. FPGA의 단점을 극복하기 위해서는 회로 구조의 효율성의 향상과 더불어 응용 분야에 적합한 특수 FPGA를 개발하는 것이 필요하다. 효율적인 회로 구조와 특수 응용 분야의 FPGA 개발은 단순한 회로 설계에서 얻어지는 것이 아니라 FPGA 시스템 안의 요소 기술들이 서로 효율적이고 유기적으로 연결된 FPGA 시스템의 개발로부터 가능하다.

새로운 FPGA 시스템의 개발은 공정, 원판 설계, 전용 CAD 툴, 하드웨어 프로그래머 설계 등의 기술로 구성된다. 이들은 서로 밀접한 연관성을 갖고 있으므로 각 기술의 단독적인 성능뿐만 아니라 서로의 관련성을 반영한 개발이 수행 되어야 한다. 따라서 개발 과정을 전체적인 면에서 관찰할 수 있는 통합된 시스템 개발 환경이 필요하다. 예를 들면 공정 기술 분야에서 개발되는 스위치 소자의 성능은 원판 설계분야에 영향을 미치며, 원판 설계 분야의 프로그램 회로는 하드웨어 프로그래머의 개발과 관계가 있다. 그리고 원판 설계 분야의 논리모듈 구조와 배선 구조는 전용 CAD 툴 분야의 기술 매핑기(technology mapper)와 레이아웃 툴의 개발과 병행하여 이루어져야 한다. FPGA 칩의 성능을 결정하는 하드웨어적인 요소로서 크게 스위치 소자, 논리 모듈, 배선 및 프로그래밍 구조 등이며, 성능이 우수한 FPGA를 개발하기 위해서는 각 요소 기술의 장단점을 개발 초기나 수행 과정에서 파악하여 단점을 보완하는 노력이 필요하다. 통합된 FPGA 개발 환경에서 하드웨어적인 요소와 CAD 툴을 분석하고 서로간의 영향을 종합적으로 평가하여 개선함으로써 FPGA 성능 최적화를 도모할 수 있다.

본 논문은 원판과 전용 CAD 툴로 구성되는 FPGA 시스템을 개발하는데 있어서 서로 다른 요소 기술들의 관계와 이들 요소 기술들과 시스템 성능의 관계를 모델링하여 시스템 사양을 만족하는 가장 효율적인 방법을 찾게하는 방법에 관한

것이다. 본 논문에서는 실제로 개발된 시스템^[2] 예로 하여 FPGA 시스템 개발에서 고려해야 할 점들을 고찰하였다. 2 장은 FPGA 응용 분야의 결정에 대한 것으로, 이 응용 분야의 결정은 앞으로 개발될 FPGA 시스템의 사양 결정에 중요한 요소이다. 3 장과 4장은 각각 개발될 요소 기술을 정하고 개발 흐름도를 이용하여 그 요소 기술의 범위를 결정하는 것에 관한 것으로, 현재의 기술과 개발 목표 사이에 개발이 필요한 요소 기술과 그 기술들의 성능을 정하는 것에 관한 것이다. 5 장은 요소 기술들의 개발 방법과 성능 비교 방법, 검증 방법 등을 정하고 개발된 FPGA 시스템의 성능을 평가 하는 것에 관한 것이며, 6장에서 결론을 맺는다.

II. 응용분야의 결정

FPGA의 응용 분야에 따라 개발의 범위와 사양이 크게 달라지므로, 새로운 FPGA를 개발하기 위해서는 응용 분야를 결정하는 것이 필요하다. 응용 분야에 따라 달라지는 변수로서는 속도, 집적도, 재사용성(reusability) 등이 있다. Emulator 용으로 사용할 경우에 중요도의 순서는 재사용성, 집적도, 속도일 것이며, 범용 칩으로 활용할 경우 속도, 집적도, 재사용성의 순서이다. 여기서 재사용성 요소는 전체 개발 환경에 큰 영향을 미치므로 우선적으로 결정되어야 한다. 기존의 재사용 가능한 FPGA의 스위치 구조는 트랜지스터의 게이트 입력에 메모리 소자를 연결하여 메모리 소자의 값에 따라 트랜지스터 동작을 제어한다. 메모리 소자로는 SRAM(Static Read Memory)^[3]이나 E(E) PROM^[4] 등을 사용하고 있다. 안티 퓨즈 형태의 스위치 구조는 절연 막이 전도체 사이에 삽입된 형태로 되어 있다^[5]. 평소에는 절연 막에 의해 부도체 상태에 있다가 양단의 전도체에 일정 전압이 인가되면 절연 막이 파괴되어 도체 상태로 된다. 일반적으로 메모리 소자를 이용한 스위치는 집적도나 속도 면에서 안티 퓨즈 형태의 스위치에 뒤떨어진다. 그러나 안티 퓨즈의 경우 표준 CMOS

공정에 2-3개의 마스크(mask layer)가 추가되고, 스위치 소자의 공정 후 테스트가 어려워 공정 가격이 증가한다. 따라서 재사용성 요소의 결정에 따라 스위치의 구조가 결정되는 것이 대부분이다. 메모리 소자를 이용할 경우 기존 제품의 특허를 피하기 쉽지 않으므로, 원판 구조 설계에 많은 시간이 투자되어야 할 것이다. FPGA 개발에 예상되는 비용과 회수할 수 있는 응용 분야의 시장 규모와 절충하여 응용 분야를 결정해야 한다.

III. 개발의 범위

새로운 FPGA 시스템에서 개발되어야 하는 요소 기술로는 스위치 소자, 원판 설계와 전용 CAD 툴 등이 있다. 응용 분야의 결정 단계에서 스위치 소자의 구조를 정하였으면 그 스위치 소자가 타 소자에 비해 경쟁력이 있는지를 파악해야 한다. 또 확보된 원판 설계 기술과 CAD 툴을 바탕으로 각 요소 기술들의 개발되어야 할 범위를 정하고 범위 각각의 목표가 정해져야 한다.

표 1은 ETRI의 스위치 소자에 대한 성능을 기존의 스위치 소자와 비교한 표이다. 이 예에서는 스위치 소자를 안티 퓨즈로 정하였으므로, 안티퓨즈 형태의 스위치 소자들만을 비교하였다. 표 1에

〈표 1〉 안티퓨즈 스위치의 전기적 특성비교

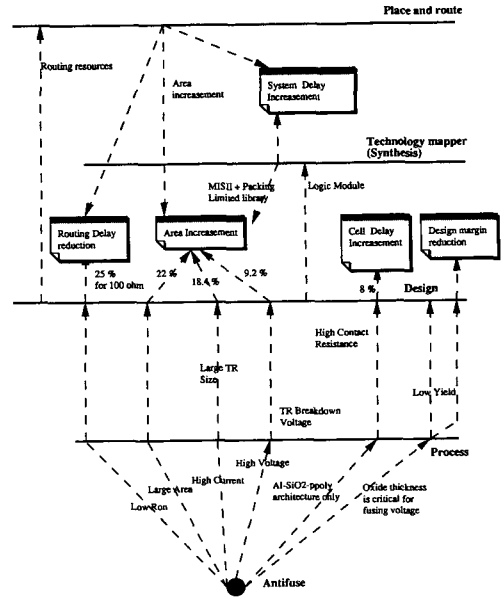
서 볼 수 있듯이 ETRI 스위치 소자는 기존의 소자에 비해 도통 저항이 작은 장점이 있으나 프로그램 전류나 전압이 높다. 도통 저항 값이 적으므로 고속의 FPGA를 구현할 수 있지만, 높은 프로그램 전류로 인해 프로그램 트랜지스터의 크기가 커져서 집적도가 떨어질 수 있다. 또한 높은 프로그램 전압은 프로그램 트랜지스터의 파괴 전압과 비슷한 수준이므로 이를 막기 위한 방법이 필요하다. 따라서 원판 설계에서는 스위치 소자의 장점을 살리면서 높은 프로그램 전류와 전압의 문제를 해결해야 한다. 프로그램 트랜지스터의 크기가 증가함으로써 발생하는 집적도의 손실은 논리 모듈의 효율성을 높여서 보상할 수 있고, 파괴 전압 보다 높은 프로그램 전압을 수용하기 위해서는 특별한 프로그램 회로가 설계되어야 한다. 효율성이 높은 논리 모듈의 설계는 원판 설계 기술만으로는 할 수 없다. 기술 매핑기가 효율적으로 회로를 논리 모듈에 할당 해야 하므로 논리 모듈의 설계와 기술 매핑기의 개발이 병행되어야 한다. 또 떨어진 집적도를 보상하고 고속의 장점을 살릴 수 있는 배선 구조의 개발과 배치 및 배선 툴의 개발이 필요하다. 따라서 ETRI에서는 스위치 소자 분야에서는 높은 프로그램 전류와 전압을 낮추는 것을, 원판 설계 분야에서는 효율적인 논리 모듈과 프로그램 트랜지스터의 파괴를 막는 프로그램 회로 개발 및 효율적인 배선 구조의 개발을, 그리고 전용 CAD 툴

	Actel	QuickLogic	ETRI
이름	FLICE	ViaLink	Fusing-link
전극	n ⁺ diffusioin n ⁺ polysilicon	metal 1 metal 2	metal 1 polysilicon
절연체	O-N-O	a-Si	SiO ₂
절연체 두께	9nm	150 nm	8.5nm
도통 저항	300-500 ohm	50-80 ohm	~30 ohm
절연 용량	> 100 M ohm	> 1 G ohm	> 1 G ohm
정전 용량	3-5fF	1.3fF	<= 3fF
크기	0.8×0.8um	1.0×1.0um	1.2×1.2um
공정	double-metal standard CMOS	double-metal standard CMOS	double-metal standard CMOS
추가 mask	3	2(1)	2
Prog. 전압	18 V	10-11 V	10-12V
Prog. 전류	< 10mA	< 1 mA	< 15 mA

분야에서는 기술 매핑기와 배치 및 배선 툴의 개발로 개발의 범위를 정하였다. 효율적인 논리모듈의 개발을 위해서는 원판 설계와 기술 매핑기가, 효율적인 배선 구조를 위해서는 원판 설계와 배치 및 배선 툴이 서로 연관되어 개발되어야 한다.

IV. FPGA 개발 흐름도

현재 또는 앞으로 개발될 기술을 종합적인 면에서 고찰하여 새로운 FPGA 개발의 문제점을 초기에 파악하는 것이 중요하다. 그림 1은 ETRI의 개발 초기에 요소 기술들이 FPGA 시스템의 성능에 미치는 영향을 도식화 한 것으로 FPGA 개발 흐름도라 한다. 이 개발 흐름도는 개발 요소(스위치 소자, 원판 설계, 전용 CAD 툴)들이 근간이 되어 각 요소 기술의 장단점이 다른 요소 기술에 미치는 영향과 시스템 성능에 미치는 영향을 모델링하고 정량화 한다. 이러한 흐름도로부터 각 개발 요소간의 영향을 평가하여 개발될 시스템 성능을 추정하고, 현재 보유하고 있는 개발 요소들의 장단점과 향후 개발되어야 할 기술들의 상대적 중요성을 파악한다. 따라서 이 개발 흐름도를 이용해 시스템 성능을 가장 효과적으로 향상시키기 위해서 중점적으로 개발되어야 할 기술을 알아 낼 수 있다. 그림 1에서 우선 스위치 소자를 보면 장점으로 적은 도통 저항 값으로 시스템의 속도 성능을 30% 이상 향상시킬 수 있다. 도통 저항 값과 시스템의 속도 성능의 관계를 알기 위해서 여러 가지 형태의 네트를 구성하여 도통된 스위치는 도통 저항으로, 도통이 되지 않은 스위치와 배선은 커패시터로 모델하여 배선 지연 시간을 계산하고, 배선 지연 시간이 시스템 속도 성능에 미치는 영향을 모델하였다. 단점으로 지적된 비교적 큰 스위치 소자의 면적, 높은 프로그램 전류와 전압이 시스템의 집적도에 각각 22%, 18.4%, 9.2%의 손실을 초래한다. 여기에 계산된 숫자는 설계 초기에는 간단한 모델로부터 계산되나 설계가 진행될 수록 점점 더 정확하게 계산된다. 또 ETRI 스위치 소자의 경



〈그림 1〉 ETRI FPGA 시스템의 개발 흐름도

우 스위치 구조로서 알루미늄 contact을 사용해야 하므로 별도의 마스크를 사용하지 않기 위해서는 소자도 알루미늄 contact을 사용해야 하고, 프로그램 전압이 스위치 소자의 절연막 두께에 민감하다. 전자의 경우 contact 저항이 커지므로 셀의 지연에 8% 기여하고 후자는 공정 수율에 영향을 주어 설계시 프로그램 회로의 설계 여유가 줄어든다. 따라서 스위치 소자 기술이 시스템의 속도 향상에는 큰 장점이 되는 반면 집적도나 설계 여유에는 단점이 되는 것을 알 수 있다. 특히 스위치 소자의 크기와 높은 프로그램 전류가 집적도의 손실에 가장 큰 원인이 되는 것을 알 수 있다. 이러한 집적도의 손실은 개발 흐름도에서 볼 수 있듯이 원판 설계 기술의 논리 모듈 설계와 전용 프로그램의 기술 매핑 기술이 보상할 수 있다. 개발 당시 ETRI가 보유하고 있는 기술 매핑기로는 MIS-pga를 사용하고 있었는데 이 기술은 다양한 형태의 논리 모듈의 특성을 효율적으로 이용하지 못하므로 새로운 기술 매퍼의 개발이 필요하다. 즉, MIS-pga는 멀티플렉서를 기본으로 한 논리 모듈에 적합하나 멀티플렉서의 제어 게이트를 적절하게 사용하지 못한다. 또 스위치 소자의 크기가 커

서 발생하는 면적의 손해는 논리 모듈의 입출력 수와 배선 구조를 최적화 함으로 보상할 수 있다. 배선 구조는 배치 및 배선 툴과 밀접하게 관계가 있으므로 병행하여 개발해야 한다.

V. 평가 및 검증

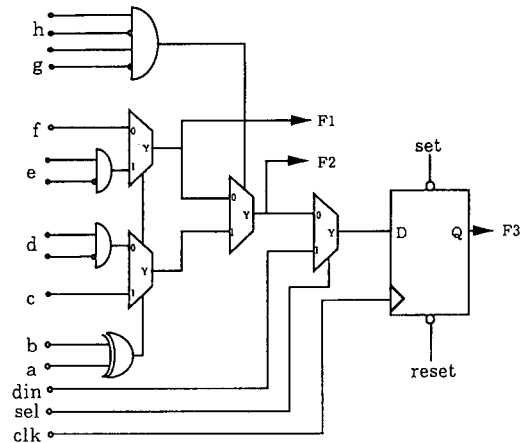
앞에 설명된 장들에서 개발되어야 할 기술 분야와 범위들을 정하였다. 이 장에서는 각 분야에서 새로 개발되는 기술들이 시스템에서 요구하는 만큼 기여하는지를 평가하고 그 기술들로 구성된 전체적인 시스템의 동작을 검증하는 방법에 대해 논의한다. 먼저 개발된 기술들은 각 분야 별로 벤치마크를 통해 기존의 기술들과 비교하고 평가한다. 그리고 FPGA 시스템의 종합적인 평가는 회사별로 다른 회로들을 벤치 마크로 사용하는데 근간에는 PREP(PRogrammable Electronic Performance) 벤치 마크를 많이 이용하고있다^[6]. FPGA 시스템의 검증은 원판 회로와 전용 CAD 툴의 동작을 점검할 수 있어야 하고, 더 나아가서는 하드웨어 프로그래머와의 동작도 같이 검증하여야 한다. 이 장에서는 ETRI에서 FPGA를 개발할 때 사용된 방법을 소개한다. 우선 각 기술의 방법과 결과를 소개하고 이 기술들의 개발로 전체 FPGA의 성능을 PREP 벤치 마크의 결과로 나타낸다. 여기서 FPGA 시스템의 검증은 원판을 VHDL을 이용해 모델한 후, 스위치 소자가 프로그램 되는 과정과 프로그램 된 후의 논리 동작을 분석해 이루어진다.

1. 논리 모듈의 개발

FPGA 원판은 동일한 형태의 논리 모듈이 이차원으로 배열된 구조를 갖는다. FPGA의 회로 구현은 이러한 논리 모듈들을 스위치 소자로 프로그램 하여 이루어진다. 따라서 FPGA의 논리의 효율성은 논리 모듈의 설계에 따라 크게 달라진다. 논리 모듈은 일반적으로 스위치 소자의 형태에 따라 달라진다. 메모리를 스위치 소자로 사용하는 경우에는 LUT(Look Up Table) 방식을 사용하는데^[4],

스위치 소자의 지연 시간이 다른 스위치 소자에 비해 상대적으로 크므로 많은 기능을 수용할 수 있도록 논리 모듈을 크게 설계하는 것이 일반적이다. 반면 안티퓨즈를 사용하는 경우에는 안티퓨즈의 지연 시간이 적으므로, 기능은 적지만 범용성이 높은 논리 모듈을 사용한다. 기존의 형태로는 게이트 어레이처럼 NAND 게이트를 사용하기도 하고^[7], 또는 멀티플렉서를 이용하기도 한다^[8].

논리 모듈의 효율성은 크게 3가지로 나누어진다. 프로그램으로 형성될 수 있는 논리의 종류, 논리 모듈의 지연 시간 그리고 입출력 수 등이다. 이들 서로는 한 쪽이 유리하면 다른 쪽이 불리하므로 응용 분야에 따라 중요성을 결정해야 한다. ETRI에서는 응용 분야로서 통신 및 정보 분야에 주로 쓰이는 데이터 패스(datapath) 회로로 정하였다. 기존의 통신 및 정보 분야의 회로를 분석한 결과 덧셈기와 비교기가 많이 사용된다고 판단하였다. 따라서 논리 모듈의 설계 목표를 한 비트 덧셈기와 두 비트 비교기를 각각 한 논리 모듈로 구현 가능하도록 하는데 두었다. FPGA의 논리 모듈은 일반 논리 구현의 효율성도 높아야 하므로, 위의 목표를 만족하는 몇 개의 논리 모듈 중에서 논리 효율성이 가장 높은 것으로 결정하였다. 논리 모듈의 효율성을 평가하기 위해서는 그 논리 모듈이 구현할 수 있는 논리 라이브러리를 자동 생성하여^[9] 논리 매핑의 벤치 마크를 이용하였다(표 2). 결정



(그림 2) ETRI 논리모듈의 논리회로도

〈표 2〉 ETRI 논리 모듈의 MCNC 벤치 마크^[10] 결과

CIRCUIT	논리 모듈의 수			CIRCUIT	논리 모듈의 수		
	MIS-pgs	ours	ratio(%)		MIS-pgs	ours	ratio(%)
5xp1	39	30	-30.0	misex1	27	22	-22.7
9sym	27	18	-50.0	misex2	47	40	-17.5
alu4	156	113	-38.0	rd53	10	8	-25.0
apex1	590	439	-34.4	sqrt8	30	19	-57.9
apex2	91	66	-37.9	squar5	30	24	-25.0
apex3	617	551	-12.0	xor5	5	2	-15.0
apex4	1027	1047	+0.02	vg2	49	34	-44.0
apex5	434	98	-343.0	sao2	81	56	-44.6
b12	42	30	-40.0	seq	509	383	-33.0
bw	84	69	-21.7	ex1010	1391	1048	-32.7
clip	51	39	-30.8	spla	180	142	-27.0
con1	14	9	-55.5	rd73	17	12	-41.7
cordic	29	15	-93.3	rd84	23	18	-27.8
cps	416	334	-82	table3	498	374	-33.1
duke2	194	145	-33.8	table5	427	324	-31.8
e64	95	95	0.0	misex3	247	189	-30.7
ex4p	281	201	-39.8	misex3c	271	193	-40.0
ex5p	156	127	-22.8	t481	17	11	-54.5
inc	59	47	-25.5	pdc	184	137	-47.0

된 논리 모듈은 그림 2의 구조를 갖는다. 이 논리 모듈은 2단의 멀티플렉서와 EX-OR와 AND 게이트를 제어 게이트로 갖는 조합 논리 부분과 순서 논리를 위한 플립플롭으로 이루어진다. 입력의 효율성을 높이기 위해 반전신호를 제공할 수 있는 4개의 입력을 포함하여 12개의 입력과 3개의 출력을 갖는다. 출력단자가 여러개인 것은 서로 독립적인 작은 논리를 하나의 논리모듈에 구현하여 면적을 효율적으로 사용할 수 있도록 하기 위함이다. 이 논리모듈은 3개의 변수로 표현되는 모든 논리로부터 12개의 변수로 표현되는 일부 논리까지도 구현할 수 있다.

2. 배선 구조의 개발

배선 지연 시간은 FPGA 칩의 동작 성능에 큰 영향을 끼친다. 일반적으로 배선 구조는 채널이 수평 및 수직으로 구성된 대칭형 구조와 한 방향으

로만 채널이 구성된 채널형 구조로 나누어진다. 배선 구조는 스위치 소자와 논리 모듈의 입출력 수에 따라 한 채널당 트랙의 수가 달라진다. ETRI FPGA는 고속 응용 분야에 목표를 두었으므로 논리 모듈의 입력 신호는 수직 채널을 통해서만 연결되고 다른 행간의 논리 모듈의 배선은 수평 채널을 이용하도록 하여 대칭형 구조와 채널형 구조의 혼합적인 형태를 택하였다. 이때 수평 트랙의 수는 전체 논리 모듈의 수에 따라 결정되고, 한 채널의 트랙들에 다양한 길이의 세그먼트를 두며 필요에 따라 연결될 수 있도록 하였다.

세그먼트는 인접한 두 논리모듈간 거리의 1, 2, 4, 8, 16배의 길이를 갖는 금속의 배선자원이다. 그리고 수평과 수직의 채널이 교차하는 스위치 블록 내부의 트랙을 선분 화하여 서로 다른 두 네트가 한 트랙을 공유할 수 있도록 하였다. 배선 장이 큰 신호 선을 위해 칩을 좌우나 상하로 가로지르

는 배선자원을 두어 지연 시간을 줄이도록 하였다. 수평과 수직의 세그먼트들이 인접한 곳과 교차한 각 지점에 안티 퓨즈를 배치하여 개략배선에서 결정한 배선경로가 배선 자원의 부족으로 인해 상세 배선 될 수 없는 경우를 배제하였다. 각 채널에서 세그먼트의 종류와 수 그리고 트랙 수는 벤치 마크를 통해 평가되고 결정되었다.

클럭의 skew 문제에 유연성 있게 대처하기 위해 클럭 신호가 I/O 핀뿐만 아니라 논리모듈로부터도 발생할 수 있도록 설계하였다. 논리모듈이 제공하는 4개의 독립적인 클럭 배선자원은 power-on enable이나 reset과 같은 global 신호에 사용될 수 있다. 또한 클럭 버퍼의 위치나 구동 능력은 delay 와 skew가 최적화 되도록 설계하였다.

일반적인 배치배선기와 같이 논리모듈의 배치, 개략배선 및 상세배선의 순서로 진행된다. 논리모듈의 배치는 배선장 최소화 및 배선밀도 완화를 목적함수로 하는 simulated annealing 알고리즘을 이용하여 결정한다^[11]. 개략배선은 배선 밀집 도를 평가하여 미결 선이 발생하지 않도록 배선경로를 할당한다. 배선의 관점에서 ETRI FPGA의 회로구조는 채널형에 가까우므로 상세배선은 세그먼트 채널배선 기법을 이용하였다^[12]. 이에 따라 배선은 matrix row matching 문제로 변형된다. 채널 세그먼트 양단의 좌표에 따라 채널을 여러 개의 zone 으로 분리한 후, 신호 선과 세그먼트가 zone을 포함하는 정도에 따라 그 정보를 각각 matrix로 표현한다. 상세배선은 이들 matrix의 row 들이 효과적으로 matching되는 경우를 찾는 것이다. 이 과정에서 안티퓨즈 사용, 세그먼트 사용, 배선에 사용된 세그먼트 중에서 활용되지 않는 부분 최소화를 시도한다. 상세배선이 끝나면 배선에 사용된 칩 내의 안티퓨즈를 프로그램하기 위한 정보를 bit-stream으로 생성한다.

3. 프로그램 구조 및 하드웨어 프로그래머의 개발

프로그래밍 회로는 addressing 회로, decoding 회로 그리고 전원공급 회로로 이루어진다. Addressing 회로는 외부의 하드웨어 프로그래머가

제공하는 300 비트의 이진 코드를 1 비트의 입력 단자를 통해 받아 연속적인 shift 동작으로 플립플롭에 저장한다. Decoding 회로는 프로그래밍 할 안티퓨즈를 가진 논리모듈을 선택하는 회로와 논리 모듈 내의 안티퓨즈를 선택하는 회로로 구성된다. 전원공급회로는 안티퓨즈 프로그래밍에 필요한 전압과 전류를 공급한다.

원하는 안티퓨즈를 프로그램 하려면 프로그램 하려는 안티퓨즈 양단간에는 프로그램 전압 12 V를 인가하고 나머지 안티퓨즈들에는 프로그램이 되지 않는 전압(6V이하)이 인가되어야 한다. 프로그램 회로에서 해결해야 할 과제는 프로그램 트랜지스터가 파괴되지 않고서 높은 프로그램 전압과 전류를 스위치 소자에 전달하는 것이다. 프로그래밍 전압 12 V는 0.8 um 공정에 의해 제조된 트랜지스터의 파괴(breakdown) 전압보다 크다. 프로그래밍 트랜지스터를 위하여 특별한 공정을 거치는 방법은 제조 공정이 복잡하여 수율을 악화시킨다. 따라서 프로그래밍 트랜지스터는 6 V의 전압 차로 동작하면서 안티퓨즈에 12 V를 유기할 수 있는 특수 회로를 설계하였다^[13]. 안티퓨즈의 연결저항 값은 프로그래밍 하는 전류 크기에 반비례하는 특성을 지니고 있어서 전류를 많이 흘릴수록 안티퓨즈의 연결저항 값은 줄어든다. FPGA 원판은 안티퓨즈의 연결저항을 10 ohm 이하로 감소시키는 25 mA의 전류를 허용하도록 전압공급회로를 설계하였다.

하드웨어 프로그래머는 프로그램 회로와 밀접하게 관계가 있으므로 프로그램 회로와 병행하여 개발되어야 한다. ETRI FPGA 원판은 4 가지 동작 모드를 가지고 있다. 첫째로 프로그램 모드는 안티퓨즈를 프로그램 하는 모드이고, 두 번째로 탐침 모드(Probe mode)는 프로그램된 안티퓨즈의 프로그램 여부를 확인하는 모드이고, 세 번째로 스캔 모드(Scan mode)는 순서적으로 연결되어 있는 원판 내의 플립플롭들의 작동상태를 점검하는 모드이고, 마지막으로 구현된 회로로 작동하는 보통 모드(Normal mode)가 있다. 표 3은 각 모드에서 원판의 프로그램용 핀들의 상태를 나타낸 것으로 원판의 모드를 정하는 핀은 MODE 핀과 TE 핀이다.

〈표 3〉 여러 가지 모드에서 FPGA 원판의 프로그램용 핀들의 상태

	DCLOCK	MODE	SDI	CYCLE	TE	STO	VPP	VP	VGA	VGB	VG	PROB	GLOBAL
Program mode	CLOCK	1	address	0	0	0	VPP	VP	VGA	VGB	VG	-	VDD
Scan mode	CLOCK	1	test data	0	1	0	VDD	VDD	GND	VDD	VDD	-	-
Probe mode	CLOCK	1	address	0	0	0	VDD	VDD	GND	VDD	VDD	0	-
Normal mode	I	0	-	-	-	-	VDD	VDD	GND	VDD	VDD	-	-
Direction	입력	입력	입력	출력	입력	출력	입력	입력	입력	입력	입력	출력	입력

〈표 4〉 PREP benchmark 비교 결과표

Benchmark 회로	Actel (ACT1425)	QuickLogic (QL12×16)	ETRI FPGA
Datapath	10	12	22
Timer/Counter	6	5	8
Simple FSM	10	10	14
Complex FSM	4	5	5
Arithmetic	4	4	5
Accumulator	3	4	12
16-bit Counter	6	8	7
16-bit prescaled-counter	4	8	7
Memory map	14	17	17

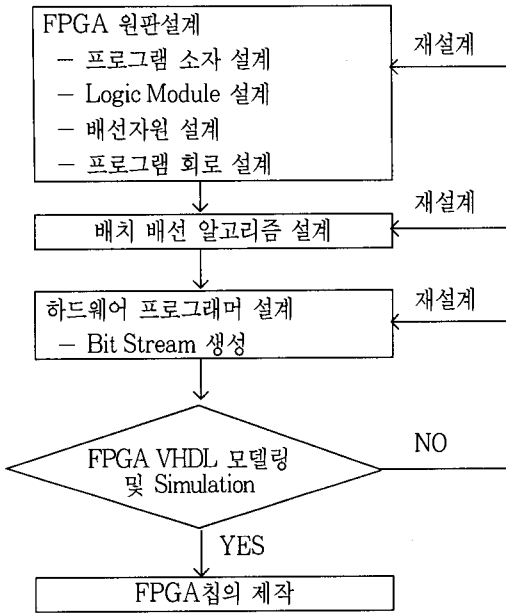
즉 원판은 (MODE, TE)가 (1,1)이면 프로그램모드나 탐침모드로, (1,1)이면 스캔 모드로 작동하고 MODE가 '0'이면 보통모드로 작동한다. 프로그램 모드와 탐침모드는 입력되는 데이터와 전압 원의 전압 수준에 의해 결정된다.

4. System 성능 평가 및 동작 검증

ETRI가 설계한 논리모듈의 성능을 분석하기 위해 FPGA 성능평가용 PREP 벤치마크를 이용해 평가하였다. 이 벤치마크는 9 가지 분야의 회로들로 구성되어 있다. 하나의 FPGA 칩 내에 각 회로를 반복하여 구현할 수 있는 최대 수와 최종 동작 속도를 비교한다. 표 4는 안티퓨즈를 이용하면서 2,500 게이트 급의 Actel(ACT1425 모델) 및 QuickLogic(QL12X16 모델) FPGA, 그리고 동일 크기로 조정된 ETRI FPGA에 PREP benchmark

를 적용한 결과를 나타낸다. 조합논리 구현에는 큰 차이가 없으나 datapath와 accumulator 분야에서는 월등히 우수한 결과를 나타낸다.

종합적으로 FPGA 시스템을 검증하기 위해서 하드웨어 프로그래머와 원판을 VHDL로 모델한 후, 전용 CAD 툴에서 출력된 프로그램 정보를 인가하여 회로의 동작을 에뮬레이션 하는 환경을 구성하였다^[4]. 이 환경은 실제로 FPGA가 프로그램 되고 동작하는 것을 검증하는 것으로 시스템 전체 흐름의 문제점을 정확하게 파악할 수 있다. 원판 VHDL 모델은 원판의 프로그램 회로, 논리모듈, 배선선분, 안티퓨즈 및 입출력 패드 등을 표현한다. 또 회로 구현후의 배선경로의 지연시간은 프로그램된 안티퓨즈들의 저항과 프로그램되지 않은 안티퓨즈들과 배선선분의 정전용량에 의해 발생하는데, 복잡한 배선경로의 경우 배선 지연 시간은



〈그림 3〉 FPGA 시스템의 검증 흐름도

논리모듈의 지연시간의 최대 수배에 달하여 무시할 수 없다. 배선경로의 지연시간을 추출하기 위해, C 언어를 이용하여 프로그램된 안티퓨즈의 위치정보로부터 배선경로를 재구성한다. 재구성된 배선경로로부터 지연시간을 추출하여 VHDL 시뮬레이션에 피드백한다. 지연시간 모델에 따른 영향 평가를 위해 SPICE, Elmore 및 Horowitz 등의 방식을 구현하여 실험하였다. 본 검증 기법은 ETRI FPGA 시스템에 구현되어 원판과 회로의 논리 및 타이밍검증을 수행하며, 또한 임계경로 탐색을 통해 구현된 회로의 성능을 예측하는 수단을 제공한다.

VI. 결 론

본 논문에서는 실제로 개발된 FPGA를 예로 하여 FPGA 시스템 개발에 대해 설명하였다. 먼저 개발할 FPGA의 응용 분야를 결정하고, 그 응용 분야에 필요한 시스템 사양에 맞게 개발한 요소 기술들과 그 기술들의 범위를 정한다. 개발 흐름

도를 이용하여 이 요소 기술들의 연관 관계를 수직적으로는 시스템 성능에 미치는 영향을 모델하고 수평적으로는 요소 기술간의 서로 미치는 영향을 모델하여 시스템 사양을 만족하기 위한 최적의 해를 구한다. 이때 최종적인 FPGA 시스템을 평가하고 검증할 수 있는 방법을 결정한다. 요소 기술들이 개발됨에 따라 좀 더 구체적이고 정확한 모델에 의해 전체 시스템의 성능은 평가되고 검증될 수 있다.

FPGA는 공정, 회로 설계, 전용 CAD 등의 기술을 복합적으로 요구한다. 즉, FPGA 개발이라는 것은 단순히 원판 설계를 의미하는 것이 아니라 공정 및 전용 CAD 툴을 포함하는 하나의 시스템을 개발하는 것이다. 앞에서 살펴본 바와 같이 이들 기술은 밀접한 연관 관계에 있으므로 개발 초기 및 수행 과정에서 항상 각 요소들의 장단점을 파악하여 부족한 부분에 대한 보완 및 개선이 이루어져야 한다. 이러한 개발 환경을 갖추는 데는 추가의 노력이 필요하므로 과제 초기에는 어느 정도 부담이 있지만 개발이 진행되면서 각 부분에 대한 검증이 개발과 동시에 이루어지므로 FPGA의 전체 개발 기간은 이러한 환경을 사용하지 않는 것에 비해 크게 감소될 수 있다. 또한, 타 요소를 고려한 성능 보완도 이루어지므로 전체 성능 향상도 기대할 수 있다.

참 고 문 헌

- [1] Stephen. D. Brown, R. J. Francis, J. Rose, Z. G. Vranesic, "Field- Programmable Gate Arrays", Kluwer Academic Publisher, 1992.
- [2] Hanjin Cho, et al., "A New Field Programmable Gate Array : Architecture and Implementation," ETRI Journal, vol. 17, no. 2, pp. 21-30, July 1995.
- [3] Hung-cheng Hsieh, et al., "Third-generation architecture boosts speed and

- density of field programmable gate arrays," IEEE 1990 Custom Integrated Circuits Conference, Boston, MA, May 12-16, 1990, pp. 31.2.1-31.2.7.
- [4] Sau C. Wong, et al., "A 5000-gate CMOS EPLD with multiple logic and interconnect arrays," IEEE 1989 Custom Integrated Circuits Conference, San Diego, CA, May 15-18, 1989, pp. 5.8.1-5.8.4.
- [5] Esmat Hamdy et al., "Dielectric Based Antifuse for Logic and Memory ICs", IEDM Tech. Digest, pp.786-789, 1988.
- [6] Dennis McCarty, et al., "PREP benchmark for programmable logic devices," IEEE 1993 Custom Integrated Circuits Conference, San Diego, CA, May 9-12, 1993, pp. 7.7.1-7.7.6.
- [7] David Marple, "An MPGA-like FPGA," IEEE Design & Test of Computers, pp. 51-60, Dec. 1992.
- [8] Abbas El Gamal, et al., "An architecture for electrically configurable gate arrays," IEEE J. of Solid-State Circuits, vol. 24, no. 3, pp. 394-398, 1989.
- [9] 조한진, 배영환, 박인학, "멀티플렉서에 기초한 논리모듈의 Library 생성방법", 대한전자공학회논문지 제32권, 제 10호, pp. 1411-1419, 1995
- [10] S. Yang, "Logic synthesis and optimization benchmarks user guide - version 3.0," Microelectronic Center of North Carolina, Jan. 1991
- [11] S. Kirkpatrick, C. D. Gelatt, Jr., M.P. Vecchi, "Optimization by Simulated Annealing", Science, Vol. 220, No. 4598, pp. 671-680, 1983.
- [12] K. Roy, "A Bounded Search Algorithm for Segmented Channel Routing for FPGA.s and Associated Channel Architecture Issues", IEEE Trans. on CAD, Vol. 12, No. 11, pp. 1695-1705, 1993.
- [13] 조한진, 박영수, 박인학, "안티퓨즈를 기초로 한 현장 가공형 반도체의 새로운 프로그래밍 회로 구조", 대한전자공학회논문지, 제 32권, 제 11호, pp. 1491-1498, 1995
- [14] 백영석, 조한진, 박인학, 김경수, "안티퓨즈 FPGA의 배선지연시간을 고려한 VHDL 모델링", 대한전자공학회 논문지, 제 33권, 제 5호, pp. 908-915, 1996

저자 소개



曹 漢 鎭

1960年 7月 8日生

1982年 2月 한양대학교 전자공학과 학사

1987年 5月 New Jersey Inst. of Tech. 전기공학과 석사

1992年 5月 University of Florida 전기공학과 박사

1992年 11月~현재 한국전자통신연구소 반도체단 자동설계연구실

주관심 분야: FPGA 설계, 회로설계, 소자 모델링

嚴 洛 雄

전자공학회 논문지 제33권 A편 제4호 참조
현재 한국전자통신연구소 선임연구원

裴 英 煥

전자공학회 논문지 제32권 A편 제10호 참조
현재 한국전자통신연구소 선임연구원

白 瑛 錫

전자공학회 논문지 제33권 A편 제5호 참조
현재 한국전자통신연구소 선임연구원

朴 永 秀

전자공학회 논문지 제 32권 A편 제11호 참조
현재 한국전자통신연구소 연구원

朴 仁 學

전자공학회 논문지 제 32권 A편 제11호 참조
현재 한국전자통신연구소 연구원

金 景 洙

전자공학회 논문지 제 33권 A편 제1호 참조
현재 한국전자통신연구소 책임연구원