

FPGA 설계 기술 및 응용

金基鉉, 鄭正和
漢陽大學校 工科大学 電子工學科

I. 서론

FPGA가 출현함에 따라 기존의 주문형 반도체인 gate array나 standard cell시장에 변화가 일어나고 있다. 첫째는 엔드유저에 의한 off-the-shelf 제품설계생산이 가능해짐으로서 칩부품의 생산제조 시간을 대폭 단축시켰다. 둘째는 반도체 제조사와의 복잡한 상담을 하지않아도 되며 수량이 적은 칩 부품의 경우 초기 투자 비용을 절약할 수 있어 제품 단가를 낮출 수 있다. 셋째, 간단한 툴 사용법만 익히면 설계의 노하우 축적이 가능해져 설계의 대중성을 확보할 수 있다.

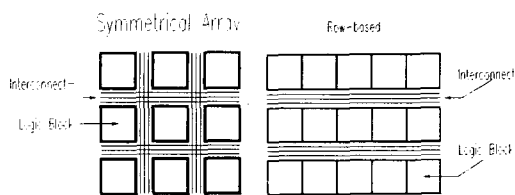
이러한 장점들을 갖고 앞으로 90년대 주문형 반도체로서 성장이 대단히 기대되는 소자이다. 그러나 현재 시판되는 FPGA의 단점으로는 동작 속도의 한계점을 갖고 있다는 점이다. 이러한 단점을 극복하기 위한 방법들이 계속 발표되고 있으나 아직 괄목할만한 성과는 없는 것 같으며 소위 중저속 중규모 집적 회로에 사용될 전망이다.

특히 FPGA가 갖는 장점인 재프로그램 가능성을 유지하면서 단점인 저속의 동작 속도를 이용한 하드웨어 검증 시스템이 시판되고 있다. 하드웨어 검증 시스템은 소프트웨어 검증(시뮬레이션)의 한계인 실제 데이터의 흐름을 확인할 수 있다는 장점이 있으나 일정 규모 이상의 회로에 대해서는 실시간 처리가 불가능하다는 단점이 있다.

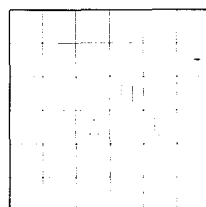
본 고에서는 FPGA를 간단히 언급한 후 시판되고 있는 하드웨어 검증 시스템을 예로 들어 시스템 구성과 성능을 평가한다.

II. FPGA 기술 현황

현재 FPGA의 아키텍처는 크게 2개의 방향(Antifuse 방식과 SRAM 방식)으로 이루어지고 있다. Antifuse 방식의 경우 Actel사의 diffusion-topolysilicon antifuse가 발표된 이후 많은 연구가 진행되어 현재는 metal-to-metal antifuse를 쓰고 있다. Antifuse 방식의 FPGA를 개발한 회사는 Actel, Quicklogic, Crosspoint등이 있으며 현재 국내에서는 한국전자통신연구소가 DSP에 이용할 수 있는 FPGA를 antifuse 방식으로 개발하여 실험중에 있다. 대부분의 antifuse FPGA는 cell-based 방식으로 기본적인 함수를 구현하는 셀이 배열되어



(a) 대칭적인 어레이 구조 (b) Cell-based 구조



(c) Sea-of-gates 구조

그림 1. FPGA의 구조

있고 배선 영역으로 부터 입력부분에 대한 antifuse를 fusing함으로써 회로를 구현하고 있다. (그림 1 (b)) 그러나 Crosspoint 사의 FPGA는 전형적인 gate array형태로 구성되어 있다. SRAM 방식은 1985년 Xilinx사에서 발표한 이래로 Algotronix, Concurrent, AT&T등에서 개발하였으며 국내에서는 (주)서두 로직에서 개발중에 있다. Xilinx와 AT&T에서 개발한 FPGA는 그림 1 (a)와 같은 대칭적인 어레이 구조이고 Algotronix와 Concurrent는 sea-of-gates 형태(그림 1(c))로 되어 있다.

Antifuse 방식의 경우, 기본 소자는 특정한 형태로 결정되어 있고 배선 영역에 있는 antifuse를 이용하여 소자들의 입력을 연결하기 때문에 fuse의 저항, 기본 소자들의 지연시간등이 동작 시간에 영향을 주기 때문에 fuse의 저항을 최소화 하여야 한다. 그러나 SRAM 방식의 FPGA에 비해 동작 속도가 빠르다는 특징을 들 수 있다. 그러나 SRAM 방식의 경우 pass tr.에 의해 논리 함수가 구현되고 논리 블럭간의 배선이 이루어지기 때문에 전반적으로 동작 시간이 느리지만 반복적으로 사용할 수 있다는 장점이 있으며, 최근에는 배선에 의한 지연 시간을 줄이기 위해 다양한 길이의 금속선을 배열하는 구조를 AT&T에서 발표하였다. 표 1은 현재 발표된 각 FPGA의 특성을 요약한 것이다.

표 1. 프로그램 방법에 따른 특성 비교

프로그램 방법	회사명	저항(Ω)	용량(ff)
SRAM cells		1-2K	10-20
PLICE Anti-fuse	ACTEL	~500	3-5
ViaLink Anti-fuse	Auicklogic	50-80	1.3

Ⅲ. SRAM 방식 FPGA의 응용

SRAM 방식의 FPGA의 선두 주자인 Xilinx사의 X3000 또는 X4000시리즈의 FPGA는 아키텍처가 갖는 고유의 결점인 신호의 전달 시간이 느리다 점과 FPGA에 프로그램하는 데이터의 복제가 가능하다는 단점을 가지고 있으나 반복적으로 프로그램이 가능하

다는 장점에 의해 새로운 응용분야가 생성되었다. 즉 Field programmable circuit board 개념에 의한 ASIC emulator가 그 예이다. ASIC emulator는 ASIC이나 custom IC등을 제조하기 전에 전체적인 시스템의 prototype을 구현하자는 목적으로 Quick-turn에서 1990년대 초에 발표한 RPM이 발표됨으로서 주목을 받기 시작하여 영국의 InCA사와 미국의 PIE사에서 ASIC 에뮬레이터를 상품화 함으로써 새로운 시장을 형성하고 있다. ASIC 에뮬레이터는 하드웨어와 소프트웨어로 구분할 수 있다. 하드웨어는 Xilinx사의 FPGA를 사용하여 PCB를 설계하는 것인데 FPGA의 용도는 다음과 같이 크게 3가지로 볼 수 있다.

- Logic 구현용 FPGA : 사용자가 설계한 회로가 구현되는 FPGA
- 신호 연결용 FPGA : Logic 구현용 FPGA간의 연결 또는 Logic 구현용 FPGA와 입출력용 FPGA의 연결만을 하기위한 FPGA로 crossbar라 불리기도 한다.
- I/O용 FPGA : ASIC 에뮬레이터로 구현된 시스템의 prototype을 검증하기 위해 외부 시스템과의 연결을 사용자가 임의로 설정할 수 있도록 제공되는 FPGA(선택적으로 사용)

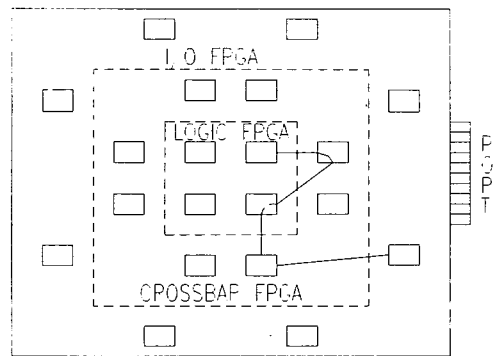


그림 2. ASIC 에뮬레이터 하드웨어 모듈의 일반적인 형태

그림 2는 ASIC 에뮬레이터의 일반적인 형태를 보

여주고 있다. 또한 ASIC 에뮬레이터를 지원하는 CAD툴의 일반적인 구성은 다음과 같다.

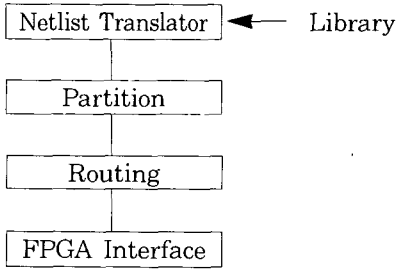


그림 3. ASIC emulator의 CAD 툴 구성도

1. Netlist translator

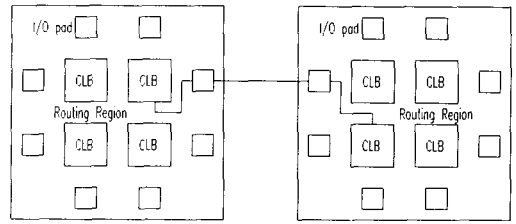
전자 회로의 설계는 다양한 schematic capture에 의해 이루어 지고 또한 ASIC vendor의 라이브러리도 다양하기 때문에 공통적으로 이용되는 중간 형태인 EDIF로 기술된 netlist를 입력으로 받아들인다. 이때 ASIC vendor의 라이브러리를 FPGA에서 지원할 수 있도록 라이브러리 매칭을 하게 된다. AND, OR 또는 플립플롭과 같은 기본적인 논리 셀에 대한 매칭은 단순하게 이루어 질 수 있으나 매크로 셀은 FPGA에서 구현될 수 있는 형태로 변형시켜야 한다. 또한 FPGA의 종류에 따라 지원이 되지 않는 셀들이 존재한다. Xilinx사의 XC3000 family에서는 D F/F를 지원하는 데 이때 reset은 지원하지만 preset은 지원하지 않고 있다. 따라서 FPGA에서 지원하지 않는 셀들에 대한 매칭 방법, 또는 사용자 메시지의 방법등을 고려하여야 한다. 현재까지 발표된 ASIC 에뮬레이터는 EDIF, NDL 또는 TDL등의 netlist를 Xilinx 사의 netlist 형태인 XNF로 변환한다.

2. 분할

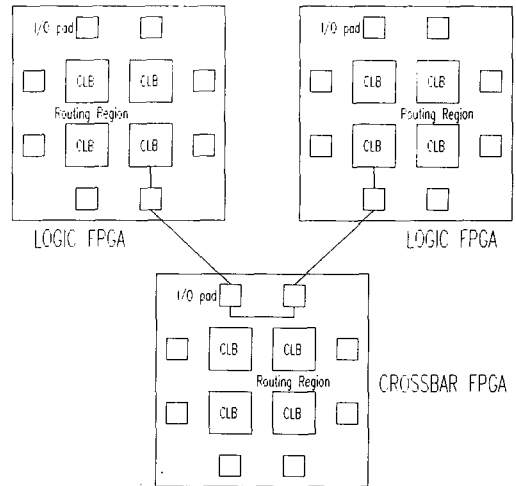
회로를 분할하기 위해서는 먼저 한 개의 FPGA에서 구현할 수 있는 게이트 수를 제한하여야 한다. 기존의 시스템은 20 ~ 30 %정도의 FPGA 칩 이용율을 보이고 있다. 한개의 FPGA 칩에서 구현할 수 있는 게이트 수가 제한 됨에 따라 회로를 분할하게 된다. 이때 중요한 것은 timing과 실제적인 FPGA 이용율이다. 입력이 게이트 단위의 netlist이고 실제 FPGA에 구현되는 것은 게이트 단위가 아니고 FPGA내에 있는 논리 블록단위로 구현되기 때문에 하나의 논리 블록에 들어가는 게이트의 수는 주로 입

력 수에 비례한다. 예를 들면 Xilinx 사의 X3000 family에 속하는 칩들은 5개의 변수(입력)으로 구현되는 모든 함수를 게이트 수에 관계없이 하나의 논리 블록에 실현할 수 있다. 따라서 같은 게이트 수에 의해 분할될 지라도 실제 FPGA내에서 논리 블록에 구현되는 것은 차이가 있다. 따라서 이러한 점 때문에 실제적인 이용율이 낮아질 수 있기 때문에 FPGA의 기술 매핑을 먼저 수행하고 기술 매핑된 결과를 이용하여 분할하는 방법도 생각해볼 수 있다.

또한 설계된 회로가 하나의 FPGA로 구현된다면 배선에 의한 신호 지연은 FPGA의 배선기에 영향을 받게되지만 2개 이상의 FPGA에 의해 구현될 경우는 그림 4와 같이 FPGA의 배선 영역을 거쳐서 출력 PAD를 통과한 후 다른 FPGA의 입력 PAD를 통과하여 배선영역을 거쳐서 연결되기 때문에 하나의 FPGA에 의해 구현되는 경우보다 전달 지연 시간이 길어진다. 따라서 경로에 대한 분석을 수행하여 전달 지연시간이 짧아야하는 소자들을 같은 FPGA에 할당하도록 회로를 분할하여야 한다.



(a) 2개의 FPGA간의 직접적인 연결



(b) Crossbar를 이용한 FPGA 연결

그림 4. 2개의 FPGA를 이용한 경우의 신호선 연결

분할에서 고려하여야 할 부분은 하드웨어 모듈이 확장 되는 형태이다. 단순히 하나의 하드웨어 모듈에 탑재 되는 FPGA 수가 증가하는 경우(PIE사의 경우)와 하드웨어 모듈의 수가 증가하는 경우(InCA사의 경우)이다. 하나의 모듈에 탑재되는 FPGA 수가 증가 하는 경우보다 하드웨어 모듈의 수가 증가하는 경우는 하드웨어 모듈간의 연결 형태를 고려하여야 한다. 또한 하드웨어 모듈의 확장 구조에 따라 하드웨어 모듈간의 전달시간 지연이 발생하기 때문에 분할 방법은 기존의 분할 방법보다는 하드웨어의 아키텍처에 따른 보완된 분할 방법을 적용하여야 할 것이다.

3. 배선(Routing)

배선은 일반적인 IC나 PCB의 배선과는 다른 개념이다. 즉, 그림 4 (b)에서와 같이 FPGA A와 B를 연결하고자 할 때 crossbar 역할을 하는 FPGA를 매개로 하여 연결된다. 이때 세 칩의 pin간 연결은 PCB상에 금속선에 의해 이미 연결이 되어있기 때문에 FPGA A와 B의 연결 경로는 pin의 위치를 결정 하는 것이다.

이때 다른 FPGA의 연결을 고려하여야 한다. 또한 crossbar내에서의 연결되는 pad위치도 중요하다. 한 예로 crossbar를 Xilinx 사의 XC2064로 사용한다고

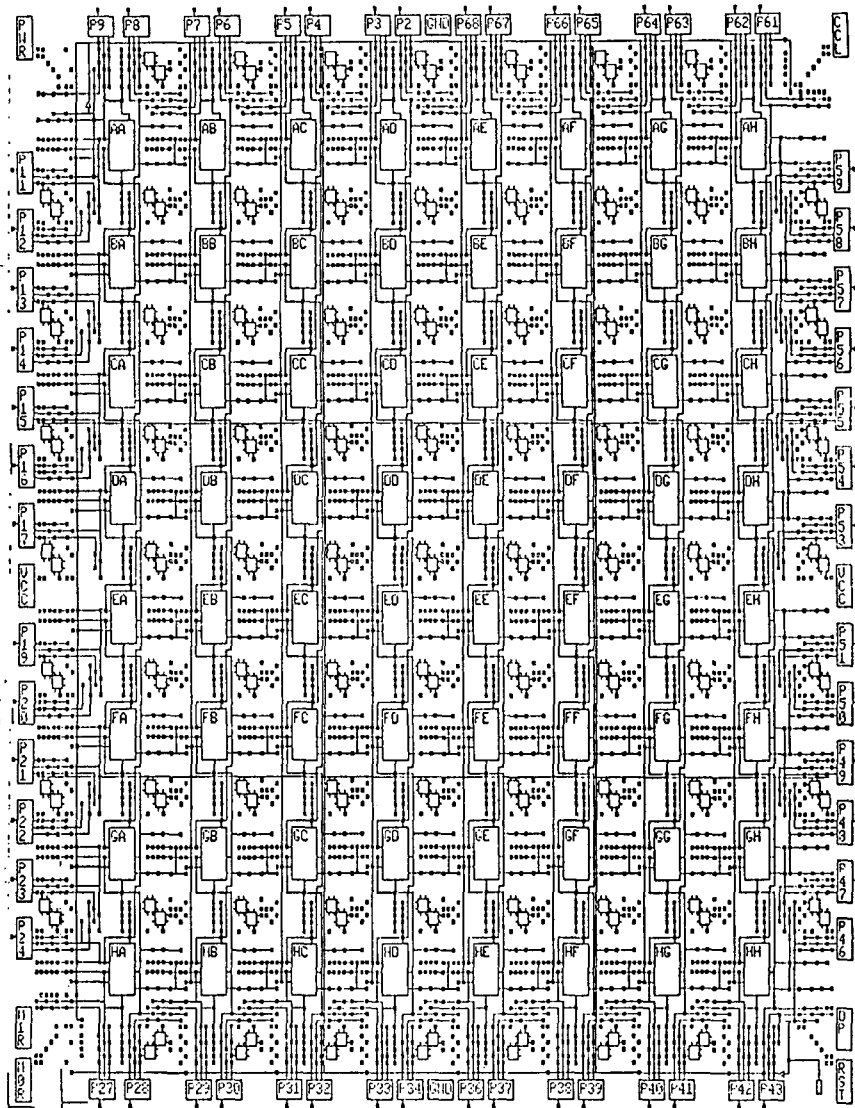


그림 5. XC2064의 내부 구성도

가정하면 XC2064의 내부 구성은 그림 4와 같다. 그림 4 (b)와 같은 연결을 위해 XC2064의 내부 연결을 고려하면 pad의 사용이 중요하게 된다. 예로서 그림 5의 좌하변에 있는 P27 과 P28을 연결하면 가장 근접한 위치의 pad간 연결이므로 연결을 위한 지연 시간은 작고 대부분 pad를 통과하는 지연 시간만을 가질 것이다. 그러나 최악의 경우로 그림 5의 좌하변에 있는 P27과 우상변에 있는 P58을 연결한다고 하면 pad통과 시간에 비해 무시할 수 없는 내부 배선에 의한 신호 지연이 발생할 수 있다. 따라서 이러한 점까지 고려하여 각 FPGA의 연결을 위한 배선 경로를 결정하여야 한다.

4. FPGA Interface

FPGA interface는 위에서 설명한 분할 및 배선 결과에 따라 하드웨어 모듈내에 있는 FPGA를 프로그램하기 위해 FPGA 틀에 필요한 정보를 발생시키고 FPGA 틀을 사용자가 관여하지 않도록 FPGA 프로그램에 관련된 부분을 macro 명령으로 처리한다.

IV. 기존의 ASIC 에뮬레이터

현재까지 발표된 ASIC 에뮬레이터는 각각 특징을 가지고 있다. InCA의 경우 빠른 동작 속도와 확장성이 특징이며 최근에는 VA- II를 발표하면서 DSP 시스템에서의 사용이 가능해 졌다. PIE사의 MARS- II는 1000핀 정도의 probing 기능과 대규모 회로의 구현이 가능하다는 특징을 가지고 있으며 또한 에뮬레이션 결과에 대한 전기적 신호의 분석을 가능하게 한다. 이러한 특징에도 불구하고 현재의 시스템에서 갖는 문제점은 신호 연결용 FPGA를 논리 구현을 위한 FPGA를 이용함으로써 신호 연결용 FPGA를 통과하는 신호의 전달 지연 시간이 크고 연결용 FPGA가 Logic 구현용 FPGA간의 연결을 위해 완전한 crossbar가 아닌 부분적인 crossbar 역할만을 하기 때문에 100% 연결을 보장할 수 없다는 단점을 가지고 있다. 현재의 ASIC 에뮬레이터가 갖는 단점인 동작 시간의 제한을 극복하기 위해 미국의 Aptix사에서는 신호 연결용 FPGA를 대신하여 interconnectuon 전용 칩인 FPIC(Field Programmable Interconnection

Chip)을 개발하였다. FPIC는 936개의 pin을 서로 연결할 수 있으며 신호 전달 시간이 연결용 FPGA보다 약 1/2 정도이고 pin간의 거리에 관계없이 거의 비슷한 전달 지연 시간을 갖는 특징을 가지고 있기 때문에 ASIC 에뮬레이터의 발전에 영향을 미칠 것이다.

1. InCA의 ASIC 에뮬레이터

InCA의 ASIC 에뮬레이터는 VA(Virtual ASIC)와 CS(Concept Silicon)로 나누어 진다. CS는 VA에 설계된 회로를 입력하기 위해 개발된 CAD 툴로서 2장에서 설명한 내용을 담고 있다. 따라서 ASIC 에뮬레이터의 하드웨어 부분에 대해 설명하고자 한다. VA는 CAT6과 CAT12이라는 2개의 Emulation module(본문의 하드웨어 모듈과 같은 의미임)을 갖는다. CAT6 Emulation module은 6개의 XC3090으로 구성되어 있다. CAT12 Emulation module은 12개의 XC3090으로 구성되어 있으며 그림 6과 같은 구조로 되어 있다.

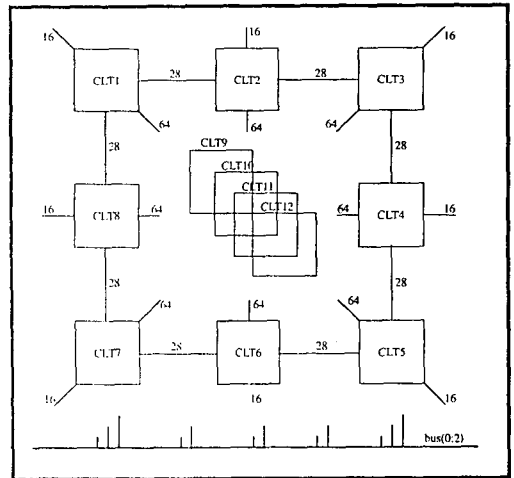


그림 6. CAT12_128 의 내부 아키텍처

그림 6에서 내부에 있는 CLT9 - 12까지의 FPGA는 crossbar의 역할을 한다. 또한 CLT1 - 8까지는 논리 회로를 구현하는 FPGA이다. 이 구조의 특징은 외부 시스템과 연결되는 I/O 신호를 crossbar를 이용하지 않고 직접 논리를 구현하는 FPGA로 연결하고 논리를 구현하는 FPGA중 인접된 FPGA간에는 직접적인 연결이 가능하도록 하여 crossbar를 통과한

으로서 생기는 전달지연시간을 단축하려고 하였다. 따라서 분할된 회로를 Emulation module에 탑재된 FPGA에 할당할 때 인접된 FPGA간의 연결 형태를 고려하여야 한다. 또한 VA는 Emulation module를 8개 까지 확장할 수 있기 때문에 Emulation module간의 연결은 외부 port를 이용하여 연결한다. 이러한 구조에 의해 InCA에서는 20MHz까지 에뮬레이션이 가능하다고 발표하고 있다. 또한 InCA에서 1993년 DAC 쇼에서 VA-II를 발표하였다. VA-II의 특징은 그림 7에서 보듯이 DSP, μ -P, RAM 및 ROM등을 daughter board 개념으로 VA내에 삽입함으로써 별도의 connection 없이 연결이 가능하게 하여 응용 범위를 확장하였다.

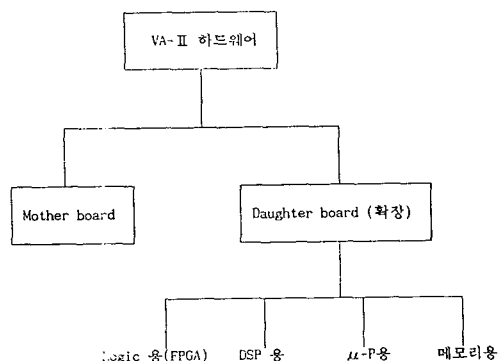


그림 7. VA-의 board 구성

또한 VA의 단점인 probing 기능을 추가하여 probing pin을 지원하는 84pin XC3042를 사용하고 있다.

2. PIE사의 ASIC 에뮬레이터

PIE사의 ASIC 에뮬레이터인 MARS-II는 MP (Modular Packaging)과 IP(Integrated Packaging)로 구분되며 MP는 LBM(Logic Block Module)에서 구현할 수 있는 게이트의 크기에 따라 3종류로 구분된다. 즉, 25,000 게이트를 구현할 수 있는 LBM2, 50,000게이트를 구현할 수 있는 LBM3-S와 100,000 게이트를 구현할 수 있는 LBM3-D로 나뉜다. 설계한 회로의 크기가 증가하면 InCA사의 ASIC 에뮬레이터는 slot 확장 개념으로 Emulation Module을 추가시키지만 PIE사의 ASIC 에뮬레이터는 LBM을 교체하여야 한다. 만일 LBM을 교체하지 않고 몇 개의 LBM으로 설계한 회로를 구현하여 에

뮬레이션 할려면 connector를 통하여 연결을 하여야 한다. MARS-II의 특징은 그림 7과 같이 임의의 외부 port에서 회로 내부의 신호를 probing 할 수 있다는 것이다. VA-II (InCA)의 경우는 84 pin XC3042를 이용하여 제한된 pin의 probing을 할 수 있지만 MARS-II (PIE)의 경우는 하나의 LBM에서 1000 pin정도의 probing이 가능하다. 또한 논리 회로를 구현하는 회로 중 clock을 발생시키는 회로들을 클럭 경로의 분석 후 하나의 클럭 전용 FPGA로 할당한다. 한편 Xilinx사의 X4000 family가 갖는 메모리 모델링 기능을 이용하여 메모리의 모델링이 가능하다.

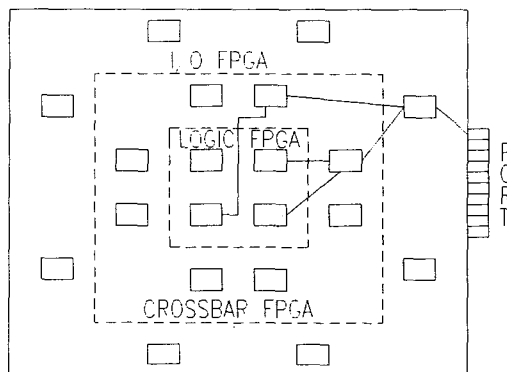


그림 8. MARS-II에서 probing 경로 설정 예

현재 LBM2의 경우 FPGA는 Xilinx사의 XC4005를 사용하며 그 구성은 다음과 같이 예측된다. 28개 논리 구현용 FPGA는 38개의 crossbar와 I/O용 FPGA으로 구성되어 있는 것으로 추정된다. MARS-II의 LBM 내부 구조를 보면 그림 9와 같은 구조로 되어 있다.

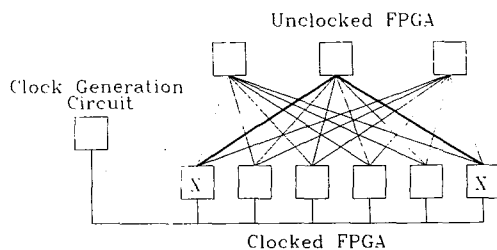


그림 9. MARS-II의 내부 구성

그림 9에서 보면 Clocked FPGA는 논리를 구현하기 위한 FPGA이고 Unlocked FPGA는 crossbar

역활을 하는 FPGA이다. Unclocked FPGA가 crossbar로 사용되지 않을 때는 논리 구현용 FPGA 역활을 하게 된다.

V. 결 론

회로 설계에 FPGA의 이용이 증가하고 일부 제품에서 FPGA를 적용함에 따라 FPGA에 대한 연구도 활발히 진행되어 다양한 형태의 아키텍처를 갖는 FPGA가 발표되었다. FPGA는 구현할 수 있는 회로의 크기 및 동작 속도가 제한됨에 따라 응용 범위가 제한되는 단점을 가지는 반면 회로 설계 시간이 단축되고 가격이 저렴하다는 점에 의해 소량 생산될 회로의 제작에 많이 이용되고 있다. 또한 대규모 회로에 대한 검증을 위해 FPGA를 이용한 ASIC 에뮬레이터가 개발되어 회로 검증에 이용되고 있다.

FPGA나 ASIC 에뮬레이터는 각 특징에 따라 응용 범위가 제한되는 경우가 많다. 일반적으로 판매 회사에서는 가능한 응용 범위로 분리하였을 지라도 실제적으로는 가능하지 않은 경우가 많다. 따라서 FPGA나 ASIC 에뮬레이터의 기능이 사용하고자 하는 용도에 적합한 지를 확인하여야 하며 향후 설계 가능성이 있는 회로도 고려하여야 한다.

參 考 文 獻

- [1] The Programmable Gate Array Data Book, Xilinx Co, 1992.
- [2] A. El Gamal , "An Architecture for Electrically Configurable gate arrays," IEEE JSSC vol.24, no.2, April, 1989, pp.394-398.
- [3] ACTTM Family Field Programmable Gate Array DATA BOOK , Actel Corporation, March, 1991.
- [4] S. D. Brown, R. J. Francis, J. Rose and Z. G. Vranesic, FIELD-PROGRAMMABLE GATE-ARRAYS, Kluwer Academic, 1992.
- [5] S. Walters, "Computer-Aided Prototyping for ASIC-Based Systems," IEEE Design & Test Of Computers, vol. 8, no. 2, June, 1991, pp.4-10.
- [6] VERY HIGH SPEED FPGAS DATA BOOK, Quicklogic Corporation, 1992.
- [7] Achitecture design of base array and CAD tool development for PGA, technical report, 한양대학교 CAD 및 통신 회로 연구실, 1992. ㉸

筆者紹介



鄭 正 和

1950年 3月 10日生

1975年 2月 한양대학교(공학사)

1977年 2月 한양대학교(공학석사)

1981年 3月 와세다대학교(공학박사)

1979年 3月 ~ 1980年 12月 NEC 중앙연구소 위탁연구원

1981年 3月 ~ 현재 한양대학교 전자공학과 교수

1986年 9月 ~ 1987年 9月 미국 Berkeley 대학교 교환교수

1993年 1月 ~ 현재 대한전자공학회 CAD 및 VLSI 연구회 위원장

주관심 분야 : VLSI CAD, FPGA 및 H/W emulator, Video compression H/W 및 S/W



金 基 鉉

1960年 2月 2日生

1982年 2月 한양대학교(공학사)

1986年 2月 한양대학교(공학석사)

1990年 3月 ~ 현재 한양대학교 박사과정

1986年 ~ 현재 한국전자통신연구소 연구원

주관심 분야 : ASIC 설계, VLSI CAD