

# MOTIF 을 이용한 그래픽 설계 도구의 구현

이 해 동, 이 상 민, 김 용 연

한국전자통신연구원 입출력시스템연구팀

Tel. 042-860-4877, Fax. 042-860-6699, E-mail: leehd@puddle.etri.re.kr

## Implementation of Graphic Design Entry using MOTIF Toolkit

Hae-Dong Lee, Sang-Min Lee, Yong-Yeun Kim

I/O System R&D Team, Electronics & Telecommunications Research Institute

**Abstract** – This paper describes implementation of a high-level graphic design entry tool operating on X Window system. The proposed design entry tool includes visual schematic entry, hierarchical modeling ability and VHDL source code generation. Experimental results show the efficiency of the proposed design system.

하기도 한다. 소형 chip 에서부터 대형 시스템의 설계에 이르기까지 이러한 CAD 환경의 구축에 대한 연구는 필수적이며, 나아가 보다 효과적인 설계 환경을 위하여 설계자의 design concept 을 신속히 문서화하고 design cycle 을 단축시키는 역할을 한다.

본 논문에서는 점차 다양화되어가는 CAD Tool 의 일반적인 설계 흐름 (design flow) 상에서 논리 수준 및 RTL 수준 합성기와 연계되어 동작할 수 있는 schematic entry 에 대하여 기술한다. 2 장에서는 제안된 설계 도구가 동작하는 CADPIA 환경에 대한 개요를 기술하며, 설계 도구의 구체적인 기능 및 동작 그리고 제안된 시스템의 효율성을 보이기 위한 실험결과를 3 장에서 기술하고 결론을 5 장에서 제시한다.

### 1. 서론

최근 전자 시스템 및 VLSI (Very Large Scale Integrated circuit) 설계 기술의 급속한 발전으로 인하여, 보다 다양하고 복잡한 기능을 가지는 회로의 집적화 및 다기능화 현상을 보이고 있다. 이러한 추세를 능동적으로 대처하기 위하여 반도체 설계에 있어서 설계 주기 (design cycle) 혹은 제품 주기 (product cycle)를 최소화하기 위한 CAD tool 에 대한 연구가 활발히 진행되고 있으며, 1990 년대 후반부터 이러한 개념에 위한 상용화 제품도 발표되고 있다 [1-4]. 일반적으로 chip 의 설계는 주어진 사양으로부터 최종 단계인 layout synthesis 에 이르기까지 다양한 단계를 거쳐 진행되며, 각 단계를 처리하기 위한 CAD tool 이 보편화될 정도로 설계 흐름은 세분화 및 전문화되어 있다. 이러한 설계 자동화 tool 의 가장 큰 목적은 설계 주기의 단축에 있으며, 또한 반도체에 전문적인 지식이 없는 사용자로 하여금 쉽게 chip 설계에 적용할 수 있는 환경을 제공

### 2. CADPIA 환경

고속병렬 컴퓨터 SPAX 시스템의 개발을 위한 하드웨어 설계 환경 CADPIA [5]는 top-down 설계 방식과 시스템 레벨의 시뮬레이션을 지원하기 위한 하드웨어 설계 환경으로 구축되었으며, CADPIA 의 전체적인 구성도를 그림 1 에 보였다. CADPIA 의 하드웨어 구성은 네트워크 과부하 및 서버 시스템으로의 부하 집중에 의한 성능 저하 등의 문제점들을 동시에 해결하기 위하여, 양 방향 서버 시스템 및 팀별로 서버 서버 개념을 도입하였다. CADPIA 하드웨어 구성의 특징으로서

대용량 멀티 프로세서인 두 대의 서버 시스템을 양 방향으로 설치하여 동일한 소프트웨어를 분리, 탑재함으로써 한 대의 서버 시스템으로 집중되었던 과부하를 분산시키며, 아울러 한대의 서버 시스템 고장 시 전체 시스템으로의 파급 영향을 방지할 수 있다. 그리고, 대용량의 CAD 소프트웨어 및 설계 데이터를 탑재시킬 수 있도록 디스크 어레이를 장착하여 하드웨어 서버 시스템 통합 시뮬레이션에 대비하였다. 또한 각 서브 네트워크 내에 또 다른 서브 네트를 설치, 운용함으로써 가능한 한 메인 서버 시스템의 과부하가 분산되도록 시스템을 구성하였다. 이러한 구성은 대규모의 CAD 작업을 여러 명이 동시에 수행할 때 발생하는 성능 저하를 최대한 방지할 수 있으며, 시스템 고장에도 대비할 수 있는 하드웨어 구성이다.

### 3. Graphic Design Entry

제안된 그래픽 설계 도구 GODS (Graphic Oriented Design Environment)는 X Window 환경 상에서 MOTIF tool kit 을 사용하여 구현되었으며, 그림 2 에 주윈도우 화면을 보였다.

주메뉴는 파일 관련 메뉴, 라이브러리 관련 메뉴, 심볼, 그리기 관련 메뉴, entity, 입출력 포트 설정 메뉴, 확대/축소 등의 보기 메뉴 그리고 계층 구조 간의 이동을 위한 메뉴 등으로 구성된다.

#### 3.1 각 메뉴의 기능 정의

GODS 의 입력으로서 VHDL 행위 및 구조 기술 형태를 지원하며, 반대로 스케매틱 데이터를 VHDL 로 저장할 수 있다. 파일 메뉴는 이러한 HDL 관련 함수를 제공하며 자체적으로 VHDL 언어 분석기를 제공한다. Library 와 Symbol 메뉴는 각각 기정의 설계 데이터와 심볼 라이브러리의 관리 기능을 제공하며, 신규 라이브러리의 등록을 위한 별도의 그래픽 에디터를 제공한다. Draw 메뉴는 physical module 을 블록 형태로 그릴 수 있는 다양한 부메뉴를 제공하며, 생성된 모듈의 포트 정보의 정의는 Port 메뉴로 구동된다. 각 블록이 수행하는 기능의 정의는 GODS 에서 제공되는 텍스트 에디터를 통하여 기술할 수 있다. 그림 2 상에서 Draw 메뉴를 하드웨어 블록을 선택하면 팝업 메뉴의 형태로 행위기술을 입력할 수 있는 환경을 제공한다.

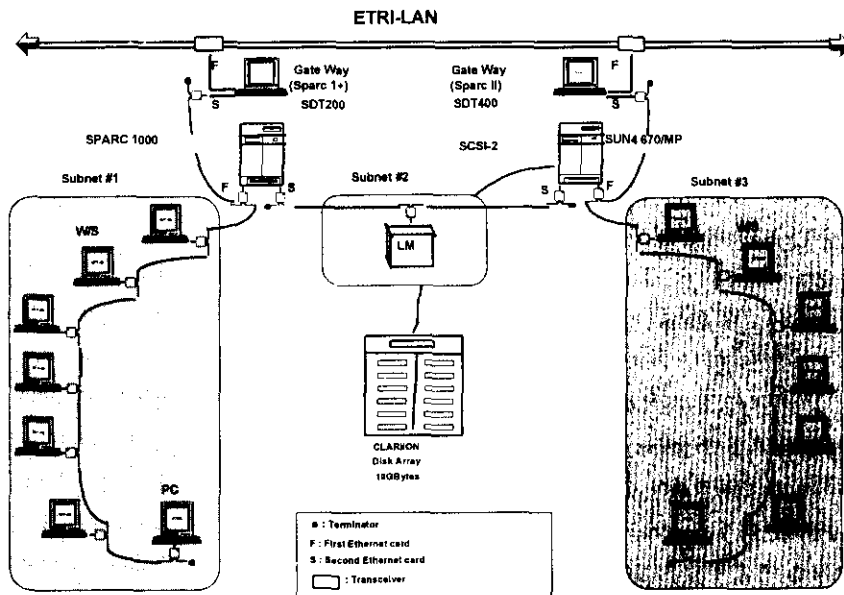


그림 1. CADPIA 설계 환경

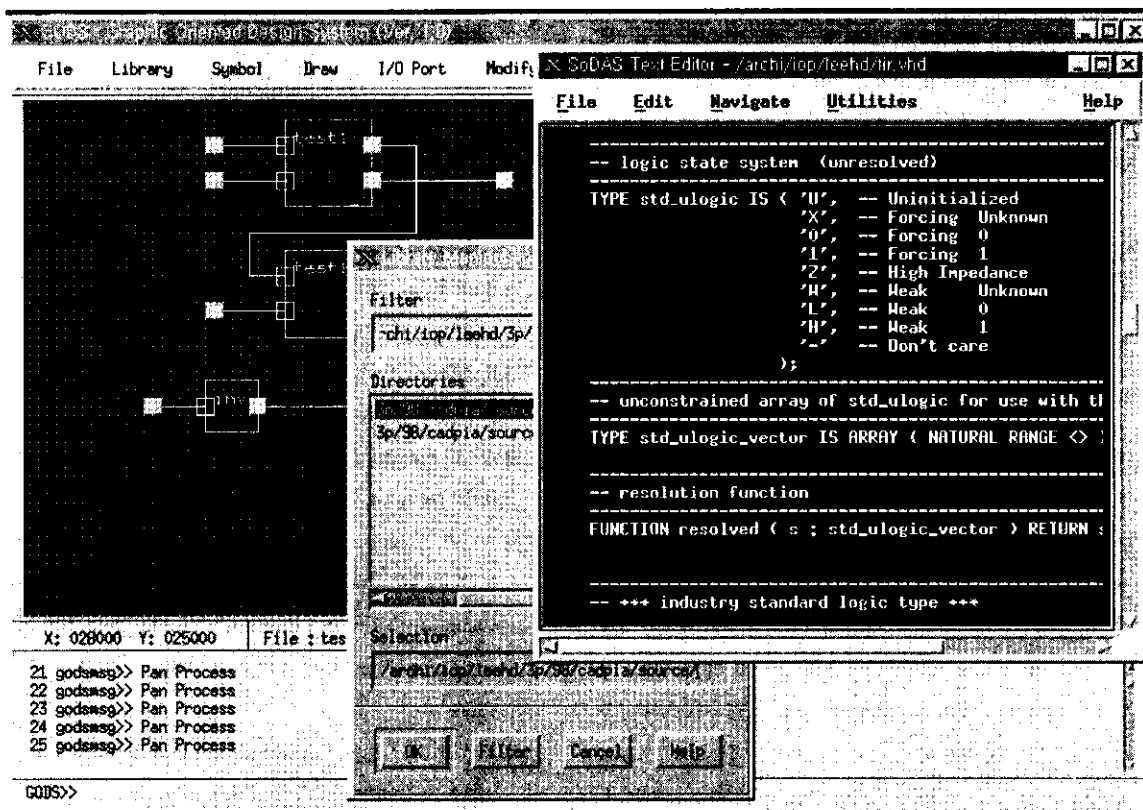


그림 2. GODS 그래픽 설계 환경

Modify 메뉴는 스케메틱의 편집 기능을 제공하며, zoom-in, zoom-out 등의 보기 기능은 Display 메뉴 상에서 제공된다. 선택값, 두께 등의 윈도우 상에서 출력되는 스케메틱 데이터의 속성의 변경은 Option 메뉴에서 가능하며, 계층적 설계 데이터 간의 이동을 위하여 Interview 메뉴를 제공한다.

### 3.2 GODS 설계 흐름

GODS 시스템은 설계자의 design concept 을 모델링하기 위한 도구로서 비주얼 입력 환경을 제공하며, 그림 3에 GODS 설계 환경 상에서의 design flow 를 보였다. Top-down 설계 방식은 시스템 설계에서 기존의 스케메틱 캡춰 방식이 가지는 한계를 극복하기 위한 텍스트 기반 행위 기술의 모델링 과정을 거쳐 RTL (Register Transfer Level) 수준, 게이트 수준에서 칩면적

과 속도 그리고 전력소모 등의 다양한 요소 함수를 최적화하는 단계를 거쳐 최종 레이아웃을 생성하는 compilation 의 개념에서 시작되었다. 이러한 top-down 설계 방식이 가지는 다양한 장점에도 불구하고 시스템 칩의 설계에 있어서 다수의 설계자가 참여할 경우 상호 설계 데이터의 전달에 있어서 문서화의 중요성이 점차 커지고 있다. 하드웨어 설계에 있어서 이미 표준화되어 사용되고 있는 VHDL 과 Verilog-HDL [6]로 인하여 설계 데이터의 호환 문제는 어느 정도 해결되었으나, 인식도 측면에서는 텍스트 기반 데이터가 가지는 단점을 그대로 드러내고 있다.

그림 3에서 볼 수 있듯이 GODS 시스템은 top-down 설계 흐름상에서 최상위 도구로서 위치한다. 설계자의 design concept 은 GODS 그래픽 에디터 상에서 HDL 에디터를 비롯한 다양한 입력 도구를 사용하여 모델링되어 VHDL 로 자동 생성된다. VHDL 의 자동 생

성시 모델링된 스케메틱 상에서 발생 가능한 에러 체크 과정을 지원함으로써 추후 발생할 수 있는 에러의 가능성을 최소화하였다. 생성된 설계 데이터는 Verilog-XL 시뮬레이터와 연계되어 그 동작을 검증하게 되고, 검증된 데이터는 상용 RTL 및 논리 합성 도구의 입력으로 주어져 최적화 과정을 거치게 된다.

4. 결론

본 논문에서는 하드웨어 설계 주기 (design cycle)의 최소화 및 설계 데이터의 문서화에 효과적으로 적용할 수 있는 그래픽 기반 설계 환경 구현에 대하여 기술하였다. System-On-Chip 을 지향하는 최근 반도체 업계의 급격한 발전에 적극적으로 대처하기 위해서는 하드웨어 설계 기간의 단축이 필수적으로 요구되며, 시스템의 단일 칩 구현을 위해서는 보다 효과적인 문서화 도구가 요구된다. 제안된 그래픽 설계 환경 시스템은 설계주기 단축과 문서화 도구의 기능을 제공할 수 있도록 구현되었으며, 나아가 설계자의 design concept 을 구현하기 위한 visual entry 를 제공함으로써 효과적인 설계 환경을 구축하였다.

추후 과제로서 상태천이도와 알고리즘 입력기와 같은 다양한 design entry 의 지원 및 Verilog-HDL 지원이 있다.

5. 참고문헌

[1] B. T. Egan, "ASIC designers turn to VHDL tools despite obstacles," Computer Design, pp. 55-64, Feb., 1992.  
 [2] E. Meyer, "VHDL opens the road to top-down design," Computer Design, pp. 57-62, Feb. 1989.  
 [3] A. S. Gilman, "VHDL - The Designer Environment," IEEE Design & Test, pp. 42-47, Apr., 1986.  
 [4] A. P. Gupta, "A Hierarchical Problem Solving Architecture for Design Synthesis of Single Board Computers," Research Report CMUCAD-89-15, Electrical and Comp. Eng. Dept., Carnegie Mellon University, Feb., 1986.  
 [5] Y. Y. Kim, "Approach to System Level Simulation in the SPAX System." ICEIC. pp. 28-31, Aug., 1995.  
 [6] Cadence Co., Verilog-XL User's Guide, pp. 1-78, Dec. 1994.

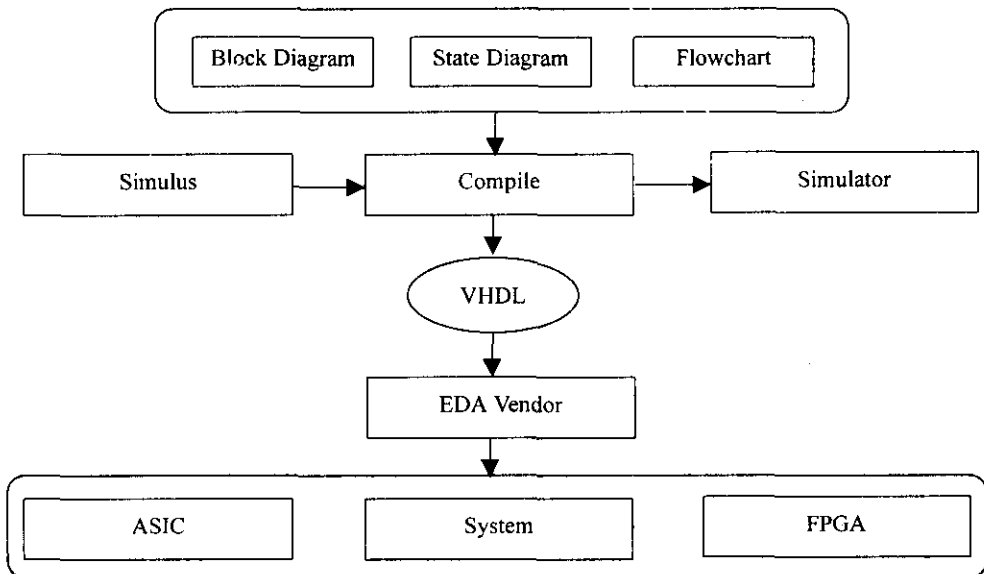


그림 3. GODS 설계 흐름도