

ASIC 설계용 통합 CAD Framework 개발

Development of Integrated CAD Framework for ASIC Design

엄 성 용* 신 혜 선** 이 규 원*** 박 선 화****
Seong-Young Ohm Hye-Sun Shin Kwe-Won Lee Sun-Hwa Park

요 약

ASIC 설계용으로 이미 개발된 여러 CAD 도구 또는 추후 개발될 CAD 도구들은 고유의 기능을 가지고 있으며, 서로 다른 환경에서 상이한 방식으로 운영되는 경우가 많다. 따라서 목표하는 최종적인 결과를 도출하기 위한 작업을 수행하기 위해서는 상호 밀접한 관계가 있는 여러 CAD 도구들을 하나의 시스템에 통합하여 운영할 수 있는 통합 CAD Framework의 사용이 효과적이다.

본 논문에서는 주로 Unix Shell 상에서 개발된 기존의 국내의 CAD 도구들을 Unix 운영체제의 X-windows라는 규격화된 그래픽 인터페이스상에서 통합하여 수행할 수 있게 하는 기법을 소개한다. 이를 위해 메뉴 구성 및 실행 환경 설정을 위한 각종 파일 양식과 메타 언어를 정의·개발하였으며, 성능 확인을 위해 기존에 개발되어 있던 두 가지 CAD 도구들을 제안된 CAD Framework상에 통합시킨 Prototype 시스템을 개발하고, 이를 토대로 테스트하였다.

Abstract

The CAD tools for ASIC design, which are already developed or will be developed in the future, have their own functions and different working environments in many cases. Therefore, it would be more effective in achieving the final design goal, if we have a system called CAD framework in which these CAD tools are systematically integrated.

In this paper, we introduce some novel techniques for integrating systematically such the CAD tools, which are usually developed under UNIX shell environments, into the CAD framework with the standard graphics interface such as X-windows. Some meta languages and script file formats are developed for flexible specification of the system MENU hierarchy and the data dependencies among executable programs. We integrated two existing CAD tools into our CAD framework using the techniques and find out the integrated prototype system is working well under the new system environments.

1. 서 론

반도체 산업 중 비메모리 영역의 대표적인 예인 ASIC(Application Specific IC) 설계의 경우, 아주 복잡하고 매우 다양한 설계 방법이 존재하기 때문에 하나의 도구를 사용하기보다는 여러 개의 CAD 도구들이 하나의 패키지로 통합되어 각 단계별로

반복 적용되는 특징이 있다[1]. 따라서 효과적인 ASIC 설계 환경 구축을 위해서는 기존에 개발되었거나 추후 개발될 국내의 개발 도구들을 하나의 통합된 체계에서 운영하는 것이 필요하다[1-6]. 이러한 통합 체계는 새로운 도구를 개발하려는 개발자들에게는 테스트 환경을 제공하여 결과 검증용 용이하게 하며, 기존 도구 개발자에게는 다른 도구와의 상호 연계를 가능하게 하여 시너지 효과를 얻을 수 있으며, 또한 각종 유사 도구들을 하나의 시스템으로 통합함으로써 여러 도구들의 성능 비교도 가능할 뿐 아니라, 이를 표준화된 양식으로 저장/출력 가능하게 함으로써 외부의 독립된 시스템과의 연계도 가능하다. 하지만, 기존의 Framework들은 기본적으로 개발 단계부터 그 Framework에

* 정회원 : 서울여자대학교 정보통신공학부 부교수

osy@swu.ac.kr

** 네띠앙 기술개발실

newsshin@hnc.net

*** 이랜드시스템 3G 연구소 연구원

leekw@eland.co.kr

**** 서울여자대학교 정보통신공학부 박사과정

bhan@korea.com

맞게 설계되고 개발된 CAD 도구들만을 통합 대상으로 하기 때문에, 기존에 Unix Shell상에서 동작하는 다른 CAD 도구들을 쉽게 통합하기 어려운 면이 있다.

본 연구에서는 국내외의 많은 기존 CAD 도구들이 Unix Shell상에서 개발된 점을 감안하여 일차적으로 이러한 각종 CAD 도구들을 Unix 운영체제의 X-windows라는 규격화된 그래픽 인터페이스상에서 통합하여 수행할 수 있게 하는 기법을 소개한다. 이를 위해 메뉴 구성 및 실행 환경 설정을 위한 각종 정의문과 메타 언어를 개발하였으며, 기존에 개발되어 있던 두 가지 CAD 도구들을 본 CAD Framework상에 통합시키는 작업을 통해 구현 가능성을 검증하였다.

2. 새로운 방식의 CAD Framework 통합 기법

ASIC 설계용으로 이미 개발된 여러 CAD 도구 또는 추후 개발될 CAD 도구들은 각각의 규칙과 기능을 가지고 서로 다른 환경에서 상이한 방식으로 운영된다. 따라서 목표하는 최종적인 결과를 도출하기 위한 작업을 수행하기 위해서는, 몇 가지 필요한 도구들을 잘 정의된 절차에 따라 순차적 또는 병렬적으로 수행해야 하는 불편이 제기되었다. 보통 이러한 도구들 사이에는 서로 밀접한 관계가 있어, 하나의 도구 사용만으로는 설계자가 만족스러운 결과를 얻을 수 없을 뿐 더러, 어떤 도구의 결과 값이 다른 도구의 입력으로 사용되는 것처럼 서로 긴밀한 연결 관계가 있기 마련이다. 따라서 본 연구에서는 밀접하게 관련되어 있는 여러 CAD 도구들을 하나로 묶어 연관된 작업을 하나의 통합된 환경에서 모두 수행할 수 있는 통합 CAD Framework를 제안한다.

Unix Shell상에서 동작하던 기존의 여러 CAD 도구들이 하나의 Framework상에 통합되어 운영되기 위해서는 다음의 두 가지 사항이 고려되어야 한다.

우선, 기존에 Unix Shell 명령어 방식으로 동작하던 여러 CAD 도구들을 표준 규격인 X-Windows 환경에서 동작되도록 구성해야 하며, 각 CAD 도구들을 일관성 있게 추가/변경/삭제 할 수 있는 기능을 제공하여야 한다. 또한 기존에 개발된 CAD 도구들의 기능은 물론 다양한 옵션 등이 그대로 유지되어야 한다. 이를 위해 본 연구에서는 동적 메뉴 구성 및 실행 환경 설정을 위한 새로운 방법을 고안하였다. 이 방법은 관련된 기능을 수행하는 여러 CAD 도구들이 손쉽게 기본 Framework에 통합될 수 있도록 지원한다.

2.1 동적인 메뉴 구성을 위한 스크립트 파일

통합 CAD Framework이 사용자의 목적에 따른 다양한 기능을 수행할 수 있도록 하기 위해서는 동적인 메뉴 구성 및 화면 조작과 같은 시스템 환경 설정이 필수적이다. 이를 위해, 간단한 조작만으로, 본 통합 CAD Framework에 통합될 각종 CAD 도구들을 일관성 있게 추가/변경/삭제할 수 있는 기능을 제공하는 메뉴구성 스크립트 파일을 제공한다. 이러한 동적 메뉴 구성 스크립트 파일은 작성 및 이해가 쉽고 또한 간단한 조작만으로 기존의 CAD 도구들을 위한 메뉴를 통합 CAD Framework상에 설정할 수 있도록 개발되었다. 통합 CAD Framework의 메뉴 구성에 대한 스크립트 파일은 동적인 메뉴 구성을 위해 CAD Framework를 수행하기 전에 스크립트 형식의 명령을 사용자가 직접 작성할 수 있도록 간단한 구문으로 정의된다.

.cnf라는 확장자를 갖는 메뉴 구성용 스크립트 파일은 통합 CAD Framework에서 기본으로 제공하는 메뉴 항목 외에 다른 기능을 추가/변경/삭제하기 위해 필요한 메뉴 구성을 정의하는 파일로서, 이 파일은 통합 CAD Framework 환경설정의 가장 기본이 되는 메뉴 항목들을 정의한다. 이 파일은 기능과 목적에 따라 CAD Framework의 기능

을 축소 또는 확장할 수 있는 유연성을 제공하며, 고정된 도구의 사용이 아닌 여러 도구들을 하나로 묶어 사용하기 위하여, 필요한 기능별로 직접 메뉴를 구성할 수 있는 기능을 제공한다.

그림 1은 메뉴 구성 스크립트 파일의 기본 형식을 보여준다. 그림에서 [MENU] 항목은 최상위 메뉴 항목, 즉 CAD Framework을 처음 실행했을 때 나타나는 각 메뉴 항목들을 구분하고 정의한다. [MENU] 항목 내에는 'NAME', 'DESCRIPTION', 'CNF_FILE' 등의 소항목들이 정의되는데, 'NAME' 항목은 메뉴바에 나타나는 각각의 메뉴 항목명 (title)을 지정하며, 'DESCRIPTION' 항목은 각 메뉴 항목의 기능에 대한 간단한 설명을 기술한다. 그리고 'CNF_FILE' 항목은 메뉴 선택시 수행할 기능에 따라 달리 설정되는데, 해당 메뉴 항목 선택시 하위 메뉴를 보여주고자 할 경우에는 그 하위 메뉴 구성을 정의하는 다른 스크립트 파일명을 지정하고, 메뉴 선택시 직접 실행할 명령어나 실행 파일이 있으면 이 실행 파일의 경로 및 파일명(execution file), 입출력 파일명, 그리고 실행 옵션 등이 정의된 실행 프로그램용 스크립트 파일을 각각 지정한다. 본 논문에서 제안한 통합 CAD Framework 시스템의 초기화면 메뉴 구성을 위해서는 MENU.cnf라는 스크립트 파일이 자동적으로 읽혀진다.

예를 들어, 초기 화면의 메뉴 바에 'FILE', 'EDIT', 'PROGRAM', 'HELP', 'OPTION' 등과 같은 메뉴 항목이 존재하도록 구성하고, 'FILE' 항목 아래는 'OPEN'과 'SAVE' 등의 하위 메뉴 항목들이 존재

```
[MENU]
NAME = ...
DESCRIPTION = ...
CNF_FILE = ...
[MENU]
NAME = ...
DESCRIPTION = ...
CNF_FILE = ...
```

(그림 1) 메뉴 구성 스크립트 파일의 기본 형식

```
[MENU]
NAME = FILE
DESCRIPTION = 파일 입출력과 관련된 메뉴항목임
CNF_FILE = SUBMENU.cnf
[MENU]
NAME = EDIT
DESCRIPTION = 편집기 vi를 실행함.
CNF_FILE = myedit.opt
[MENU]
NAME = PROGRAM
DESCRIPTION = 실행 파일과 관련된 메뉴항목임
[MENU]
NAME = HELP
DESCRIPTION = 설명과 관련된 메뉴항목임
[MENU]
NAME = OPTION
DESCRIPTION = 실행 옵션 정의 메뉴항목임
```

(그림 2) 메뉴 구성 스크립트 파일(MENU.cnf)의 예

한다. 'EDIT' 메뉴 선택시 문서편집기를 실행하고자 할 경우, 최상위 메뉴 구성 스크립트 파일 MENU.cnf는 다음의 그림 2와 같이 구성된다. 이 예에서 'FILE' 메뉴 항목에 대한 하위 메뉴를 설정하기 위해서 하위 메뉴 항목이 정의되어 있는 다른 스크립트 파일(이 예의 경우 SUBMENU.cnf)의 이름을 CNF_FILE 항목에 지정하였으며, 'EDIT' 항목에 대해서는 해당 실행 프로그램에 대한 정보가 기록된 스크립트 파일(이 예의 경우 myedit.opt)의 이름을 CNF_FILE 항목에 지정하였다. (.opt라는 확장자를 갖는 실행 프로그램 환경 설정용 스크립트 파일에 대해서는 다음절에서 설명한다.)

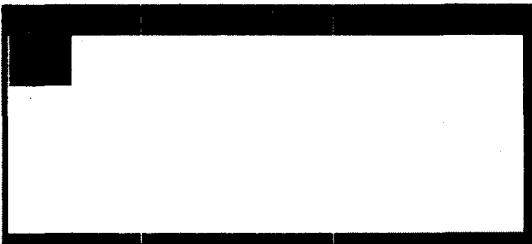
그림 3은 'FILE' 메뉴의 하위 메뉴를 정의하고 있는 SUBMENU.cnf의 내용을 보여준다. 만일 이 하위 메뉴에 또 다른 하위 메뉴가 있을 경우, 마찬가지로 하위 메뉴에 대한 다른 스크립트 파일을 CNF_FILE 항목에 설정해 주면 된다. 그림 3의 하위 메뉴 구성용 스크립트 파일은 그림 2의 상위 메뉴 설정용 스크립트 파일과 전체적인 형식에서는 유사하나, 그림 2와 달리, [SUBMENU]라는 항목을 사용하였다. 이는 [MENU] 항목이

```
[SUBMENU]
NAME = Input File
CNF_FILE = myopen.opt
DESCRIPTION = 파일 열기 명령 실행
[SUBMENU]
NAME = Output File
CNF_FILE = mysave.opt
DESCRIPTION = 파일 저장 명령 실행
[SUBMENU]
NAME = Base Name
CNF_FILE = mybase.opt
DESCRIPTION = Base Name 설정 명령 실행
[SUBMENU]
NAME = Exit
CNF_FILE = myexit.opt
DESCRIPTION = 프로그램 종료
```

(그림 3) 하위 메뉴 구성용 스크립트 파일(SUBMENU.cnf)의 예

시스템 초기 화면에서 메뉴 바로 구성되는데 반해, [SUBMENU]는 메뉴 선택시 풀다운 메뉴로 나옴을 구분하기 위함이다. 이 예에서는 'Input File'과 'Output File', 'Base Name', 'Exit'의 4가지 하위 메뉴 항목을 설정했는데, 각각 그 메뉴가 선택되었을 경우, 각각 myopen.opt, mysave.opt, mybase.opt, myexit.opt 라는 스크립트 파일에 정의된 바에 따라 특정 프로그램이 수행될 것임을 정의하고 있다.

그림 4는 앞의 두 스크립트 파일에 따라 동적으로 구성된 시스템 초기 화면을 보여준다. 메뉴 바에는 MENU.cnf에 정의된 바와 같이 5개의 항목. 즉 FILE, EDIT, PROGRAM, HELP, OPTION 등이 나타났으며, FILE 항목 선택시 SUBMENU.cnf에 정의된 4개의 하위 메뉴 항목이 의도한 대로 잘



(그림 4) 본 시스템의 초기화면 설정 예

나타났음을 확인할 수 있다.

2.2 실행 프로그램 환경 설정을 위한 스크립트 파일

.opt라는 확장자를 갖는 실행 프로그램 환경 설정용 스크립트 파일은 해당 실행 파일의 경로 및 파일명은 물론 실행 파일에 필요한 입출력 데이터 파일명과 각종 실행 옵션 등을 정의한다. 특히 입출력 파일의 지정은 사용자가 일일이 입출력 파일을 알 필요 없이 단지 수행할 명령어를 작업 순서에 맞게 수행하는 것만으로 원하는 결과를 얻을 수 있게 한다.

그림 5는 실행 프로그램용 스크립트 파일의 기본 형식을 보여준다.

설정 내용은 크게 4부분으로 나뉘어져 있다. 'EXEC'는 실행할 프로그램의 경로 및 파일명을 정의하며, 'INPUT'과 'OUTPUT'은 프로그램 수행 시 필요한 입력 파일명과 프로그램 수행 후 결과를 저장하는 출력 파일명을 각각 지정한다. 만일 입/출력 파일명을 임의로 지정하지 않은 경우에는 기본값으로 실행프로그램명.in과 실행프로그램명.

```
[EXEC]
FILE = .....
DESCRIPTION = 실행할 파일의 경로 및 파일명
[ INPUT ]
FILE = .in
DESCRIPTION = 입력 데이터 파일
[OUTPUT]
FILE = .out
DESCRIPTION = 출력 데이터 파일
[OPT ION]
TITLE = ...
SET = ..
DESCRIPTION = ...
SET = ...
DESCRIPTION = ...
DEFAULT = ...
BUTTON = ...
```

(그림 5) 실행 프로그램용 스크립트 파일의 기본 형식

out이라는 파일명을 각각 입/출력 파일명으로 지정한다. 또한 'OPTION'은 프로그램 실행시 선택 가능한 각종 선택사항들을 정의하는 부분이다. OPTION 정의는 별도의 옵션 선택 다이얼로그 창에서 선택할 수 있도록 구성하기 위해, 라디오 버튼(단일 항목 선택) 또는 체크 박스(복수 항목 선택) 기능을 제공한다.

그림 6은 실행 프로그램용 스크립트 파일의 작성 예를 보여준다. 그림에서 수행될 프로그램은 /usr/cad_frame/prg 디렉토리에 있는 lbe라는 실행 파일이며, 이 프로그램에 입력으로 들어가는 입력 파일은 DFG.in, 출력 파일은 lbe.out이라고 정의하고 있다. 여기서 유의할 점은 [OUTPUT] 항목으로는 단지 .out이라는 확장자만이 정의되어 있지만, 실제 출력 파일로는 프로그램명.out이 사용된다는 점이다. 또한 프로그램 실행시 열기 옵션은 '전체열기'와 '부분열기'가 있으나, 버튼 옵션으로 라디오(RADIO)가 지정되어 있으므로 둘 중 하나만을 선택할 수 있다. 그리고 이러한 선택이 별도로 이루어지지 않으면, '전체열기'가 자동 선택됨을 정의하고 있다. 만일 버튼 옵션으로 라디오 대신 체크 박스(CHECK)가 지정되어 있으면, 복수개의 항목을 동시에 선택할 수 있다.

```
[EXECUTION]
FILE = /usr/cad_frame/prg/lbe
[INPUT]
FILE = DFG.in
[OUTPUT]
FILE = .out
[OPTION]
TITLE = 열기옵션
SET = A
DESCRIPTION = 전체열기
SET = B
DESCRIPTION = 부분열기
DEFAULT = A
BUTTON = RADIO
```

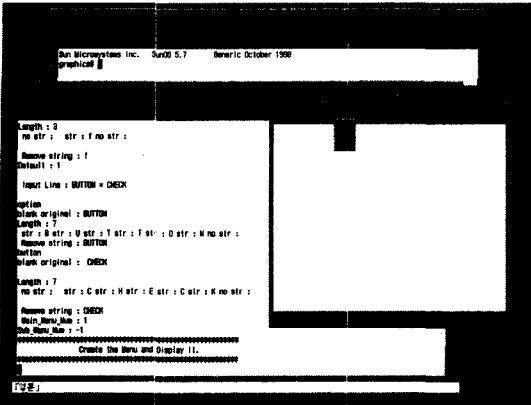
(그림 6) 실행 프로그램용 스크립트 파일의 예

3. 시스템 구현 및 실행 결과

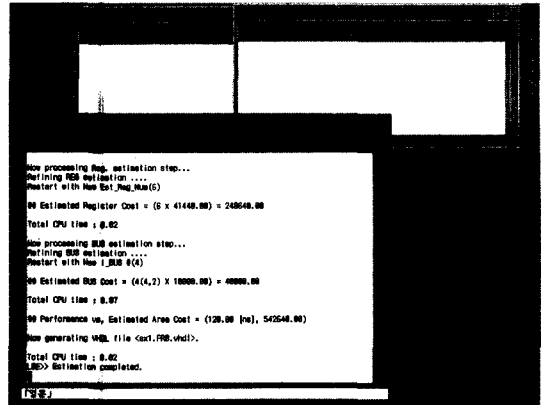
제안된 통합 CAD Framework 시스템은 SUN Sparc5 시스템의 Unix(Solaris 2.6+) 운영체제의 X-Windows(X11R5+) 환경에서 C++를 사용하여 구현되었다. 성능 검증을 위해 다양한 메뉴 구성 스크립트 파일을 테스트했으며, CAD 도구 통합 검증을 위해서는 기존에 개발되어 Unix Shell상에서 잘 동작되던 두 가지 CAD 도구(본 연구진이 개발하여 사용 중이던 CAD 프로그램)를 통합 대상으로 선정하여 모든 기능이 표현 가능하도록 구현하였다. 첫 번째 프로그램은 LBE[7]라는 CAD 도구로서 주어진 그래프 정보를 구현하는데 필요한 하한(下限)을 추정하는 도구이며, 두 번째는 LBS[8]라는 프로그램으로서, 이 역시 하한 추정 기법을 토대로 주어진 그래프로부터 최소한의 하드웨어만을 사용하되 최대한 빨리 동작하도록 그래프의 각 노드를 스케줄링[8]하는 상위 단계의 CAD 도구이다. 실험 결과, 선택 사항이 다양한 이 두 가지 프로그램을 X-Windows 환경 하에서 통합 운영하는 데 문제가 없었다. 다음의 그림들(그림 7-12)은 Unix Shell 상에서만 동작하던 이 두 가지 프로그램(도구)이 X-Windows 상에서 통합되어 운영되는 화면을 보여준다.

4. 결론

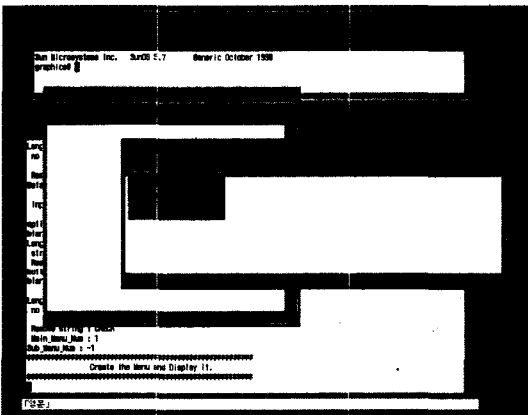
본 논문에서는 Unix Shell 상에서 동작하는 여러 CAD 도구들을 통합하여 운영할 수 있는 통합 CAD Framework 구현 방안을 제안하였다. 이 기법에서는 기 개발된 여러 도구를 통합하는 데 있어 공통성을 부여하고, 또한 상황에 따라 자유롭게 수정·추가·삭제가 용이한 두 가지 스크립트 언어를 사용하였는데, 첫째는 통합 CAD Framework의 환경 설정의 기본이 되는 메뉴를 동적으로 정의하는 X-Windows와 관련된 메뉴 구성용 스크립트 파일이며, 두 번째는 통합 CAD Framework에



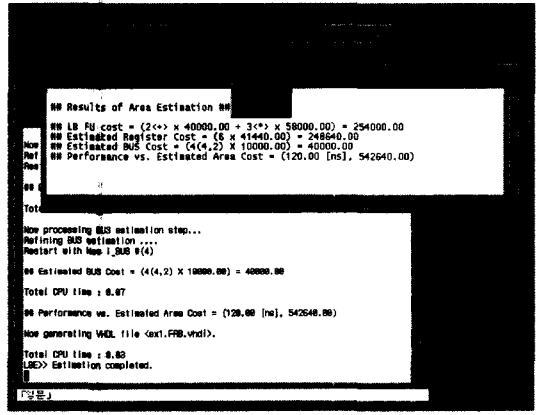
(그림 7) 통합 CAD Framework 실제 실행 화면(초기화면)



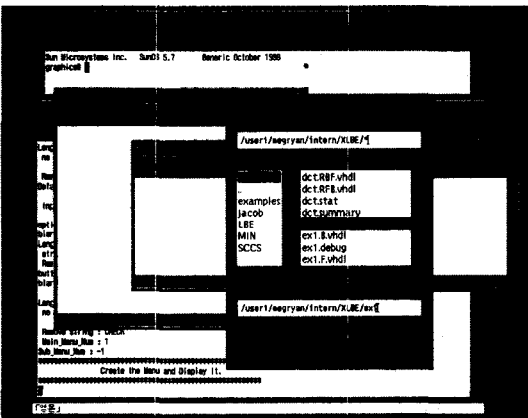
(그림 10) LBE 프로그램의 실행 결과 화면 1



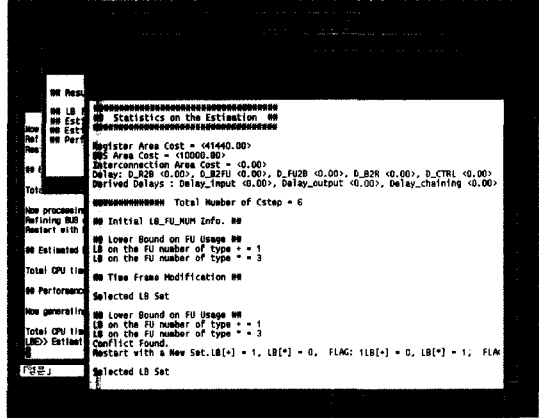
(그림 8) 통합된 LBE 프로그램의 실행 화면



(그림 11) LBE 프로그램의 실행 결과 화면 2



(그림 9) 입력 데이터 파일을 선택하는 화면



(그림 12) LBE 프로그램의 실행 결과 화면 3

포함되어 실행되어지는 각 CAD 도구의 실행 파일 및 입출력 파일, 그리고 다양한 실행 옵션을 정의하는 스크립트 파일이다. 이와 같이 기능별로 스크립트 파일은 통합 CAD Framework을 구성하는데 유연성을 제공하며, 목적에 따라 CAD Framework의 구성을 자유롭게 할 수 있다는 장점을 제공한다.

본 연구에서는 일차적으로 기존에 이미 개발되어 있던 두 가지 CAD 도구들을 본 CAD Framework 상에 통합시키는 작업을 통해 성능을 검증하였다. 하지만, 보다 다양하고 광범위한 CAD 도구들을 통합하기 위한 확장 노력이 필요하다. 또한 지금까지는 Unix Shell상에서 동작하던 기존의 CAD 도구들에 대한 통합에 초점을 맞추어 진행하였지만, 향후에는 Windows 상에서 개발된 여러 CAD 도구는 물론 추후 개발될 도구들을 통합하기에 유용하며, 또한 시스템 내에 각종 CAD 도구들간의 관련성(입출력 연결성 등)을 그래프 형태로 보여 줄 수 있는 CAD Framework을 개발하고자 한다.

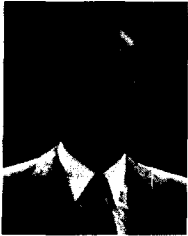
Acknowledgement

이 논문은 1998년 한국학술진흥재단의 학술연구비에 의하여 지원되었음.

참고 문헌

- [1] X. Hu, G. W. Greenwood, S. Ravichandran, and G. Quan, "A Framework for User Assisted Design Space Exploration", Proceedings of the 36th Design Automation Conference, pp. 414-419, June, 1999.
- [2] M. F. Jacome and S. W. Director, "Design Process Management for CAD Frameworks", Proceedings of the 29th Design Automation Conference, pp. 500-505, June 1992.
- [3] J. Altmeyer, B. Schurmann, and M. Schutze, "Generating ECAD Framework Code from Abstract Models", Proceedings of the 32th Design Automation Conference, pp. 88-93, June 1995.
- [4] J. B. Brockman and S. W. Director, "The Hercules CAD Task Management System", Proceedings of the IEEE International Conference on Computer-Aided Design, Nov. 1991.
- [5] J. T. Buck, S. Ha, E. A. Lee, and D. G. Messerschmitt, "Ptolemy: A framework for simulating and prototyping heterogeneous systems," International Journal of Computer Simulation, Vol. 4, pp. 155-182, Apr. 1994. Other related papers also can be found at <http://ptolemy.eecs.berkeley.edu/papers>.
- [6] S. Narayan, F. Vahid, and D. D. Gajski, "Spec-Charts: A language for system level specification and synthesis", presented at International Symposium on Computer Hardware Description Languages, Marseille, France, Apr. 1991.
- [7] Seong Y. Ohm, Fadi J. Kurdahi, and Nikil Dutt, "A Unified Lower Bound Estimation Technique for High Level Synthesis", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 16, No. 5, pp. 458-472, May 1997.
- [8] Seong Y. Ohm, Chu S. Jhon, and Fadi J. Kurdahi, "An Optimal Scheduling Approach using Lower Bound in High-Level Synthesis", IEICE Transactions on Information and Systems, Vol. E78-D, No. 3, pp. 231-236, March 1995.

◎ 저 자 소 개 ◎



엄 성 응

1985년 서울대학교 컴퓨터공학과 졸업(학사)
1987년 서울대학교 대학원 컴퓨터공학과 졸업(석사)
1992년 서울대학교 대학원 컴퓨터공학과 졸업(박사)
1992년~1993년 컴퓨터신기술공동연구소 특별연구원
1993년~1995년 University of California, Irvine에서 Post-Doc
1996년~현재 : 서울여자대학교 정보통신공학부 부교수
관심분야 : 컴퓨터그래픽스, CAD 소프트웨어, 홈네트워킹, IEEE1394, etc.
E-mail : osy@swu.ac.kr



신 혜 선

1997년 서울여자대학교 컴퓨터학과 졸업(학사)
1999년 서울여자대학교 컴퓨터학과 졸업(석사)
1999년 3월~9월 서울여자대학교 컴퓨터학과 인턴연구원
1999년 10월~현재 : 네띠앙 기술개발실(대리)
주요관심분야 : 컴퓨터그래픽스, 데이터베이스
E-mail : newsshin@hmc.net



이 규 원

1999년 서울여자대학교 컴퓨터학과 졸업(학사)
2001년 서울여자대학교 컴퓨터학과 졸업(석사)
2001년 1월~현재 : 이랜드시스템 3G 연구소 연구원
주요관심분야 : 암호화 알고리즘, PKI
E-mail : leekw@eland.co.kr



박 선 화

1998년 서울여자대학교 컴퓨터학과 졸업(학사)
2000년 서울여자대학교 컴퓨터학과 졸업(석사)
2000년~2001년 서울여자대학교 컴퓨터학과 인턴연구원
2001년~현재 : 서울여자대학교 정보통신공학부 박사과정
주요관심분야 : CAD 소프트웨어, 모바일 컴퓨팅, 임베디드 시스템
E-mail : bban@korea.com