

## CAD 프레임워크의 개념과 기술 동향

張德浩, 李玄燦, 郭銘信, 李哲東  
韓國電子通信研究所

### I. 서 론

IC 설계에 컴퓨터를 활용하려는 기술은 도입된 지 20년이 넘어 이미 필수 불가결한 기술이 되었고, 컴퓨터 지원 설계 (computer aided design: CAD) 에서 자동 설계 (design automation) 로의 변화까지 가져올 만큼 상당한 발전을 해 왔다. 그러나 80년대 초기부터 근래까지의 워크스테이션에 기초한 CAD 기술은 설계자의 특정한 요구들을 만족시키는 개별 툴들의 개발에 초점을 맞추어 왔으므로, 툴들은 대개 제각기의 인터페이스와 데이터베이스, 수행 환경을 가지게 되었다. 이것은 설계 데이터베이스를 한 단계에서 다음단계로 옮겨 가는 것을 어렵게 하여 설계 공정에 일종의 병목현상이 되게 하였고, 설계자의 요구에 따라 새로운 툴을 기존 시스템에 신설하거나 교체하는 것이 어려운 폐쇄적인 CAD 시스템이 되게 하였다. 설계 규모의 증대에 따라 설계자 수, 필요한 툴의 수, 컴퓨터 종류등이 증가하게 되고 데이터양도 증가하게 되므로 전체 설계 환경의 복잡도를 시스템적으로 관리하는 것은 오늘날 CAD 분야의 주요 이슈가 되고 있다.

1979년 UC, Berkeley의 R. Newton 교수가 CAD 시스템 "프레임워크(Framework)"의 개념을 소개한 이후 지난 몇년간 이 분야의 연구가 활성화되어 왔다. 이 프레임워크는 다양한 툴들의 통합을 지원하고, 새로운 툴들의 용이한 수용을 위한 유연성을 제공하기 위한 수단이 될 것으로 기대되고 있다. 본고에서는 CAD 프레임워크의 출현 배경과 개념 및 요건을 살펴보고, 몇몇 기관들의 주요 개발 사례와 표준화 동향, 그리고 앞으로의 전망을 소개하고자 한다.

### II. CAD 프레임워크 출현배경

설계 환경은 아이디어로부터 시스템으로 변환시켜 나가는데 필요한 하드웨어, 소프트웨어, 설계 데이터의 총체적 집합으로서, 다음과 같은 요소로 구성되어 있다.

- 다양한 범용 혹은 특수용 하드웨어와 서로 다른 운영체제
  - 상업용 CAD 시스템과 개별 상용 툴, 내부 개발 툴들의 조합  
(한 설계자가 모든 것을 숙지하기에는 너무 종류가 많으며, 시스템의 원시 코드는 약 백만에서 천만라인 정도임)
  - CAD 데이터베이스에 있어서 온라인으로 요구되는 막대한 데이터 양  
(20만 논리 게이트를 갖는 첨단 IC의 경우 약 1 giga-byte의 데이터 양)
  - 설계 라이브러리들
  - 커뮤니케이션하고 일을 공유해야 하는 설계자들과 팀들
  - 설계 과정을 관리하기 위한 회사의 정책과 절차들
- 이러한 환경은 진보하는 공정기술과 툴 및 하드웨어의 진보에 맞추어 끊임없이 변화되고 있는데, 종전의 CAD 방식은 서로 다른 개발자들의 툴들을 합해야만 제대로 된 시스템이 구성되고, 특정화된 개별 툴들은 다른 툴과의 결합이 어려운 한계성을 가지고 있다. 기존 CAD 시스템은 다음과 같은 문제를 가지므로 약 30% 이상<sup>1)</sup>의 설계 생산성의 저하를 가지고 온다.

(1) 설계화일관리

하나의 설계를 완성하기까지는 많은 일시적 혹은 영구적 화일들이 생성되는데, 이들 화일들과 그 화일간의 관계성을 설계자가 관리해야 하고 공유되는 데이터에 관한 일치성을 유지시켜야 한다.

#### (2) 번역기의 생성 관리

툴들은 각기 고유의 데이터 화일을 가지고 있으므로 툴 데이터간에 많은 번역기가 필요하며 툴과 데이터의 변경에 따라 이들 번역기를 생성, 유지 관리하기가 어렵다. 또한 번역기와 그 사용 절차를 익히고 수행시켜야 하므로 설계 과정이 복잡해지고, 데이터 처리에 상당한 부수작업이 따른다.

#### (3) 설계 공정관리

설계 데이터간의 종속관계, 즉 한 툴의 입력과 출력 관계를 설계자가 일일이 추적할 수 있어야 한다. 예를 들면 설계 일부가 변경되었을 경우에 영향받는 다른 설계 데이터의 일관성 유지를 설계자가 직접해야 한다.

#### (4) 사용자 인터페이스

설계 툴들은 사용자 인터페이스, 툴 호출 명령어가 서로 다르다. 설계자는 각 툴마다 서로 다른 스타일의 사용자 인터페이스를 배워야 하며 툴 사용방법을 기억하고 있거나 매뉴얼을 참조해야 한다.

#### (5) 툴의 통합

특정 설계에 가장 적합한 툴이 사용중인 기존 시스템내에서 제대로 활용하기까지 많은 시간이 소모되는데, 새로운 시뮬레이터를 기존 시스템에 통합할 경우 약 6개월 정도<sup>[11]</sup>의 시간이 소요된다.

#### (6) 분산 환경

다른 시스템들에 동작하는 툴들이 어떠한 것이 있으며 이들 툴들의 원격실행 방법을 알고 있어야 한다.

#### (7) 버전 관리

설계는 대개 실험적인 과정을 거치므로 중간 혹은 최종 설계 비전들이 생성되는데, 이들 비전의 관리가 시스템에서 처리되지 못하고 있다.

CAD 시스템내의 툴들은 초기에 설계자들에게 유용함에도 불구하고 장기적으로는 제한적으로 활용되고, 일반적으로 개선된 툴이나 시스템에 의해 교체되곤 한다. 이러한 교체과정은 CAD 환경에 지속적인 불안정을 초래했다. 환경적 안정성과 기존 시스템의 역량을 증가하는 새로운 접근방식을 제공하기 위해, CAD 시스템의 새로운 구조적 개념이 정의되었다. 이 접근 방식은 “프레임워크”라고 하는 소프트웨어의 토대에 기초를 두고 있는데, 이것은 CAD 툴에 풍부한 동작(operation)환경을 제공해 준다.

### Ⅲ. CAD 프레임워크의 개념

다양한 툴들을 서로 묶는데 쓰였던 첫번째 시도는 개별 툴들 사이에 데이터 번역기를 각각 두는 방식인데, 툴이 추가됨에 따라 거미줄식으로 인터페이스 프로그램이 늘어나는 방식이었다. 이런 식으로 연결된 시스템은 번역기 자체의 생성, 유지, 관리에도 큰 부수 업무가 수반되었다.

다음으로 발전된 것은 electronic design interchange format(EDIF)과 같은 표준 데이터에 대해 각 툴에 양방향 인터페이스를 제공하는 것이었다. 툴의 갯수가 N개일 경우 앞의 방식이  $N(N-1)$ 개의 번역기를 가지는데 비하여, 이 방식은 2N개로 번역기의 수가 줄어드는 장점을 가진다. 그러나 이 방식은 성능 저하, 툴간의 느슨한 결합, 공동 사용자 인터페이스 부재, 데이터 및 프로세스 관리 서비스 부재 등의 단점을 안고 있다.

툴 통합 발전의 다음 단계는 프레임워크를 기반으로 하는 CAD 시스템의로의 진전이다. CAD 프레임워크를 통하여 표준 툴 인터페이스가 규정됨으로써 CAD툴은 특정 데이터베이스, 사용자 인터페이스, 그리고 CAD 환경으로부터 독립하게 된다. 요컨대 표준화된 “플러그(plug)” 세트가 각각의 툴에 규정되어 있어 프레임워크 인터페이스에 부착된 툴이라면 어느 것이나 마치 회로 보드를 표준화된 백플레인(back plane)에 접속시키는 것과 유사하게 이 인터페이스에 접속시킬 수 있다. 한가지 요건은 개개의 툴이 프레임워크 표준을 준수해야 한다는 점이다. 프레임워크 방식을 이용하면 사용자들은 그들 특유의 설계 방식, 기술적 요구사항, 그리고 성능의 필요성에 부합하는 가장 유용한 CAD 툴들을 구성할 수 있다.

CAD 프레임워크의 표준화를 위한 기구인 CFI(CAD framework initiative)에서는 CAD 프레임워크를 “CAD 툴을 위한 공통운영환경(common operating environment)을 제공하는 소프트웨어 하부구조”로 정의하고 있다.<sup>[12]</sup> CAD 프레임워크는 컴퓨터 운영체제와 더불어 CAD 툴이 개발, 통합, 운영되는 환경을 정의하는 소프트웨어 인터페이스와 프로그램들의 모음으로 볼 수 있다. 이 프레임워크는 오퍼레이션을 지원해 주는 하드웨어 및 소프트웨어의 상세한 것으로부터 툴들을 분리시켜 주고, 시스템의 큰 변경없이 툴의 교체가 가능하게 해 주며 데이터 관리, 사용자 인터페이스 서비스, 설계 과정의 자동화

를 지원하는 유틸리티를 제공해 준다. 여러가지 면에서 프레임워크는 컴퓨터 시스템의 운영체제 환경의 CAD 지향적인 확장으로 생각할 수 있다.

#### IV. CAD 프레임워크 요건

CAD 프레임워크의 핵심은 서로 이질적인 툴들이 함께 협동작업할 수 있게 하는 통합(integration)과 새로운 툴들이 앞으로 프레임워크에 얼마든지 포함될 수 있게 하는 개방성(openness)을 제공해 주는 것이다. 여기서 통합은 툴들이 하나로 합친다는 의미가 아니라 서로 다른 툴들이 함께 협동작업할 수 있는, 즉 툴들간의 정보 교환이 가능하고 일관된 방법(uniform manner)으로 툴들을 사용할 수 있는 것을 의미한다. 그리고 개방은 툴들이 고정되지 않고 얼마든지 교체될 수 있다는 것을 의미한다.<sup>[4]</sup> CAD 프레임워크의 대표적인 구성 예는 그림 1과 같다.<sup>[4]</sup>

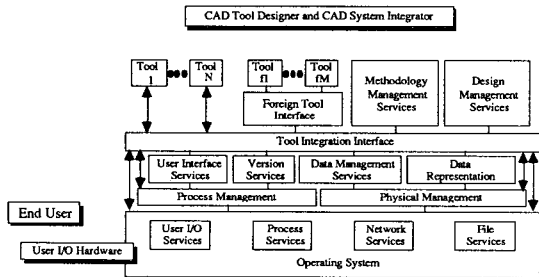


그림 1. 대표적인 CAD 프레임워크의 구성

물리적 데이터 관리는 호스트 컴퓨터상이나 네트워크상의 어딘가에 저장되어 있는 원시(raw) 데이터의 관리를 수행하며, 프로세스 관리는 CAD 툴의 수행에 필요한 설비를 말한다.

툴 통합을 구성하는 설비들은 사용자 인터페이스를 구축하는 서비스와 CAD 데이터를 관리하고 여러 툴들과 사용자들 사이에 데이터 접근(access)을 조정해 주는 데이터 관리 서비스, 설계의 진보와 역사를 관리하는 버전 서비스, 데이터의 구성과 다른 데이터 아이템과의 관계성을 정의하는 데이터 표현 서비스로 구성된다. 기존 툴들중에는 원시 프로그램이 제공되지 않는 외부 툴들이 많은데, 이 경우에는 외부 툴들을 encapsulation해 주어야 한다. 이 encapsulation을 담당하는 것이 외부 툴 인터페이스이다. 내부 툴들은 절차형 인터페이스(procedural interface)

를 통해 밀접하게 프레임워크에 통합된다. 툴의 “통합”과 “encapsulation”은 툴을 설계 시스템에 묶을 수 있는 두가지 메카니즘이다. 툴의 통합이란 용어는 분명하지 않은 의미로 통용되는 경우가 많다. 프레임워크의 관점에서 툴의 통합이라 함은 프레임워크가 제공하는 서비스를 직접 접근할 수 있도록 기존의 툴의 원시 코드를 부분 혹은 전체적으로 수정하거나 새로운 툴을 개발하는 것이다. 반면 encapsulation은 원시코드가 없는 툴들에 대해 부가적인 인터페이스 코드를 제공하여 툴을 포장하는 것이다. 이때 입·출력 데이터 정보가 시스템에 서술되어야 하는데, 대개의 경우 입력 데이터나 출력 데이터는 하나의 레코드로서 취급하게 된다. Encapsulation과 통합을 loose integration과 tight integration, black-box integration과 white-box integration, attached tool과 integrated tool로 부르기도 한다.

설계 관리와 방법론 관리는 특정 태스크를 수행하는데 필요한 툴의 조합과 수행순서등을 정의하게 하여, 태스크의 재수행시 이를 자동화 시켜주는 역할을 수행한다.

기능적인 면에서의 프레임워크 요건은 아래와 같이 요약될 수 있다.

##### (1) 정보 관리

설계 데이터의 복합 구조, 다중설계 표현과의 관계, 버전 관리, 설계 트랜잭션의 관리, 공용(released) 및 축적된 데이터의 관리 기능이 포함된다.

##### (2) 데이터 교환

서로 다른 툴들간과, 서로 다른 워크스테이션간에 데이터를 교환할 수 있고 데이터 양식이 다를 경우에는 번역할 수 있는 기능이 있어야 하는데, 이때 사용자는 번역에 참여하지 않아야 한다.

##### (3) 툴 통합

서로 다른 설계 태스크를 수행하는 다양한 툴들을 프레임워크에 포함할 수 있고 서로 이질적인 툴들간의 협동작업이 가능해야 한다.

##### (4) 프레임워크의 주문성(customization)

설계 응용마다 사용되는 설계 툴들 집합, 설계 관리 정책이 다를 수 있다. 여기서 설계관리 정책은 설계구성 관리, 버전 관리, 설계 과정 추적, 일관성 제약조건의 시행방법등을 의미한다. 따라서 각 설계 프로젝트마다 다른 설계환경을 구축하기 위해 프레임워크의 툴 통합과 사용자 인터페이스가 각각 다르게 설계되며 프로그래밍되는 것이 가능해야 한다.

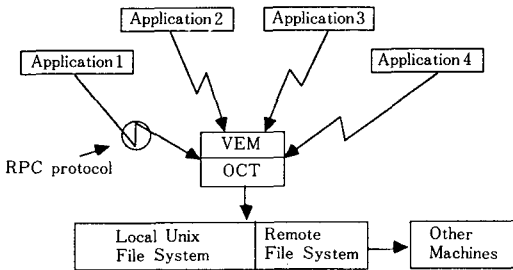
##### (5) 설계 관리

여러 툴들이 모여 하나의 설계 과정을 수행하는데 이들 설계 과정은 응용분야에 따라 다르게 정의될 수 있다. 따라서 설계 과정을 정의, 변경, 추적하는 기능이 있어야 한다. 이것은 특정 설계 과정을 수행하는데 필요한 툴들, 입력 데이터 파일들, 툴 사용법, 설계 과정간의 종속관계등을 정의하고 변경 추적할 수 있는 것을 의미한다.

### V. CAD 프레임워크 개발동향

#### 1. UC, Berkeley의 OCT/VEM/RPC

버클리 소재 캘리포니아 대학의 Semiconductor Research Corporation등의 지원을 받아 개발한 프레임워크가 OCT/VEM/RPC이다. 이 시스템의 개발은 미국 대학 연구 공동체간의 CAD 소프트웨어 교환을 활성화시키기 위해 1986년에 시작되었는데, 시스템의 구성은 그림 2와 같다.



- OCT: 설계 오브젝트를 저장하고 조작해 주는 오브젝트 중심 데이터 매니저.
- VEM: 툴들에 대한 공통 인터페이스를 제공하는 그래픽 에디터.
- RPC: 이질적이고 분산된 컴퓨터 환경에서의 설계를 지원해 주는 소프트웨어.

그림 2. OCT/VEM/RPC 시스템의 구성

이 시스템은 기존 시스템의 많은 개념을 도입하고 있으나 몇가지 새로운 개념이 추가되었다. 설계 정책이 구현 메카니즘과 분리되어 있어서, OCT 데이터 매니저는 설계 데이터를 다루는 기능들을 제공하지만, 그것이 어떤 정책과 관련되어 있지는 않다. 예를들면 셀(cell)이 가질 수 있는 뷰(view)는 무엇인지, 뷰 내부에 어떤 내용의 것이 있는지, 각 뷰들은

어떤 연관 관계가 있는지 등에 관한 것은 OCT에서 관여하지 않는다. 이러한 결정은 설계팀에 의해 결정되어야 할 설계 정책으로 남겨 두었다. 이러한 정책과 메카니즘의 분리는 정책이 변경됨에 따라 OCT를 변경해야 할 필요성을 없게 해 주는 것이다. 이 시스템에 앞선 시스템들과 또 하나는 UNIX, TCP/IP, X-window등과 같은 산업 표준을 목표로 한다는 것이다.<sup>[5]</sup>

이 시스템은 이 대학에서 개발한 IC 합성 시스템에 활용되고 있으며, register transfer에서 물리적 단계까지 적용가능한 것으로 알려져 있다. 약 80여 개의 툴들이 OCT/VEM/RPC에 통합되었고, 새로운 툴들이 계속 통합되고 있다. 최근에는 OCT에 설계 오브젝트에 관한 버전관리와 동시성 제어를 지원하는 기능이 추가되었다.<sup>[6]</sup>

#### 2. MCC

미국의 민간 공동 연구기관인 MCC가 수행하는 CAD 연구 개발 프로그램은 4개의 위성 연구와 의무적으로 참여해야 하는 1개의 핵심 기술로 세분되는데, CAD 시스템 프레임워크 개발이 이 핵심기술에 해당된다. 프레임워크 그룹에서 초점을 맞추고 있는 것은 설계 표현, 설계 관리, 데이터 관리, 사용자 인터페이스, 외부 툴 통합, 툴 관리와 지식 기반 편집, 그리고 설계 재활용성이다. 프레임워크는 개방형 운영체제와 산업 표준 하드웨어를 기초로 할 예정이다. 프레임워크는 입출력 데이터와 툴들, 그리고 분산 데이터와 네트워크의 관리를 제공하고, CAD프로그램에서 개발중인 진보된 시뮬레이션, 테스트, 합성, 물리적 레이아웃 툴들이 이 프레임워크내로 통합될 것이다. 프레임워크는 초기에 여러 개념들의 빠른 검증을 위해 Lisp 환경에 의해 시작되었으나, 1988년 Unix 환경에서의 구현으로 변경되었는데, 이유는 Unix에 기초한 대학연구와 상업용 CAD 소프트웨어, 좋은 조건의 Unix 워크스테이션을 활용할 수 있게 하기 위해서이다. 현재는 전술한 OCT 시스템을 기초로 하여 프레임워크를 진행중이며, UC, Berkeley와 더불어 분산 데이터, 설계 관리, 툴 관리, 외부 툴 인터페이스에 관한 새로운 기능을 추가하고 있다. 금년들어 CAD 시스템 공급 회사로는 처음으로 Cadence사가 이 프로그램에 참여하여 더욱 활발한 연구가 기대되고 있다. 프레임워크를 토대로 한 MCC의 CAD 시스템은 그림 3과 같다.<sup>[7]</sup>

MCC에서는 프레임워크의 표준화 업무를 지원하

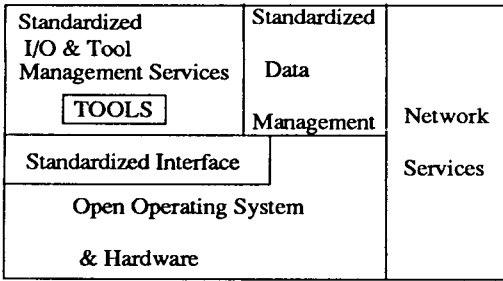


그림 3. 계획된 MCC CAD 시스템

기 위해 CAD Framework Laboratork(CFL)란 조직을 만들었다. 여기서는 CFI가 제안한 표준안을 검토하고 그것들이 구현될 수 있는지 테스트하기 위해 시제품화 해 보며 업체에서 그들의 툴들을 표준안에 대해 테스트해 볼 수 있는 검증 툴들을 개발하는 일을 맡고 있다.

### 3. 기 타

Mentor Graphics사에서는 최근 시스템 8.0 release와 함께 엔지니어링이 이루어지는 방식을 근본적으로 변화시키는 매우 혁신적인 계기를 맞았다고 발표하였다.<sup>18)</sup> 이 발표를 뒷받침하는 것은 이 회사가 3년간에 걸쳐 7,500만 달러의 연구개발비를 투입한 "concurrent design environment(CDE)"이다. 동시성 설계를 가능하게 하는 이 CDE의 기초를 이루는 것이 이 회사의 "Falcon" 프레임워크이다. 이 Falcon은 공통 사용자 인터페이스, 공통 데이터베이스 기술, 의사 결정지원 시스템등을 제공하는데, IC뿐 아니라 PCB, 기계 설계, 문서화등의 각종 제품에 핵심이 된다. 이 회사에서는 또한 "Open Door" 프로그램을 제공하여 상업용 프로그램 개발자나 고객들이 자신들의 툴들을 CDE의 환경에 통합시킬 수 있도록 문을 열어 놓고 있으며, 툴 개발 라이브러리와 각종 고객 서비스를 제공해 준다. Falcon은 역사는 짧으나 IC 설계를 포함한 넓은 범위를 지원하는 장점이 있다.

Cadence사의 프레임워크 기술은 비교적 성숙되어 있는데, 업계 최초로 1987년 이 회사의 전신인 SDA사에 의해 공급된 이래 약 2000개 이상의 복사본이 사용중에 있다.<sup>19),20)</sup> Cadence의 Opus 시스템은 Cadence의 "Design Framework"에 기초하고 있고, Cadence툴과 3자 제공 툴, 사용자가 개발한 툴들이 공동 환

경에서 일할 수 있다. Opus는 IC 설계의 모든 면을 커버하는 IC 설계 툴들의 집합이지만, 개별 사용자들에 의해 주문형 설계 환경으로 바뀔 수 있다. Cadence사에서는 오는 8월경 설계관리 기능등을 부가한 새로운 버전을 내놓을 계획이다.

유럽 공동의 정보기술 연구개발 프로그램인 ESP-RIT에서는 European CAD Integration Technology Project(ECIP)를 통해서 새로운 CAD 시스템 개발을 위한 프레임워크를 개발하고 있다. 여기에 참여하고 있는 회사는 Alcatel, Bull, ICL, Philips, SGS-Thomson, Siemens의 6개사이다. 이 프로젝트의 목적은 현재와 미래의 CAD 툴들에 대해 적용할 수 있는 표준을 정의하는 것이다. 이는 툴간의 회로 데이터 교환과 참여 회사간의 툴 교환을 용이하게 할 것이며, 궁극적으로 사용자 인터페이스의 표준화를 통해서 새 툴에 대한 설계자의 재교육을 최소화시킬 수 있을 것으로 기대되고 있다. 연결도, 레이아웃, 테스트 데이터들의 전달은 이미 표준방식으로 전달되고 있다.<sup>11)</sup>

## VI. CAD 프레임워크의 표준화 동향

### 1. CAD Framework Initiative(CFI)

CFI는 표준화된 프레임워크의 사양을 만들기 위해 CAD 공급자, 사용자, 그리고 ASIC 공급자로 구성된 비영리적인 회원간의 조직이다. CFI의 최근 조직은 그림 4와 같은데, 7개의 소위원회(subcommittee)는 주요 연구 영역을 나타내고 있다.

CFI는 Open Software Foundation, EDIF, EIS 프로그램등과 유사하다. CFI의 미션은 "다양한 툴들의 공존과 협력을 가능하게 하는 자동설계 프레임워크의 업계 수용가능한 수준의 가이드라인을 개발"하는데 있다.<sup>12)</sup> CFI에서는 이러한 분야들에 대한 확실한 연구 결과를 2~5년내에 만들 것을 목표로 하고 있다. CFI는 직접적인 후원으로 프레임워크 제품을 구현하거나 상업용 제품으로 만들지는 않을 계획으로 있는데, 그간의 표준화 작업이 90년 6월에 개최된 Design Automation Conference에서 발표되었다. 발표된 내용은 CFI의 설계 표현 그룹에서 정의한 연결도 데이터에 관한 모델링과 이에 대한 절차형 인터페이스(procedural interface)를 토대로 한 여러 공급자들의 툴들이 서로 밀접하게 결합될 수 있음을 보여 준 것인데, 성공적인 평가를 받은 것으로 알려졌다.

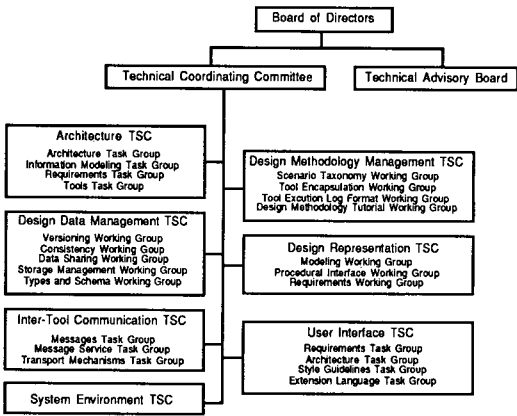


그림 4. CFI의 조직

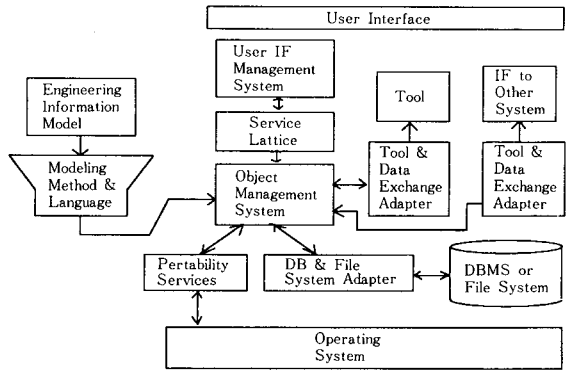


그림 5. EIS의 구조

## 2. Engineering Information System

미국방성(DoD)의 Very High Speed Integrated Circuits(VHSIC) 프로그램에서는 미래의 국방용 전자 시스템 설계와 기존 전자 시스템에 새로운 기술을 추가시키기 위해 통합된 설계 환경이 중요하다는 점을 인식하였다. Engineering Information System(EIS)은 툴의 통합과 설계 데이터의 관리를 위한 DoD의 표준명제이다. DoD에서는 1985년과 1986년 EIS의 필요성을 도출하고 1987년 Honeywell사를 대표로 하는 5개 기업(Honeywell, CCA, TRW Electronic Components Group, CAD Language Systems, Mac Donnell Douglas Astronautics Corp.)과 1개 대학(Arizona State Univ.)에 대하여 3개년 과제를 지원하였다. 이 계획에 의하면 '88년에 사양의 확정, '89년에 시제품 설계의 검토, '90년에 EIS 시제품의 데모로 예정되어 있다. EIS 시제품은 이질적인 하드웨어 시스템들, 운영체제들, DBMS들, CAD 툴들의 네트워크로 구성될 것이다. EIS의 기본 기능은 툴 통합, 데이터 교환, 엔지니어링 관리 및 제어, 정보 관리, EIS 관리에 있다.<sup>[13, 14]</sup>

EIS 프로그램의 목적은 이들 기능들에 대해 인더스트리가 사용가능한 사양안을 만드는 것과 개발된 사양이 사용자들의 요구 조건을 충족시킬 수 있음을 보이기 위해 시제품을 개발하는 것이다. EIS의 구조는 그림 5와 같다.

## VII. 기술 전망

앞으로 CAD/CAE 업체가 직면할 주된 도전 중의

하나는 툴의 통합 문제가 될 것으로 보이는데, 이는 곧 CAD 프레임워크의 목적이기도 하다. 향후 10년간은 시스템의 사양, 검증, 제조에 이르는 모든 면을 프레임워크를 통해 매끄럽게 연결시켜 주는 설계 환경에 중점을 둘 것으로 보이므로, 설계자들은 추상적이고 개념적인 단계에서 일할 수 있게 될 것이며, 구현과 검증, 제조에 따르는 문제들을 최소화하고 설계 질을 높일 수 있을 것이다. CAD 중에서도 시뮬레이션 분야는 프레임워크의 주요 응용 영역이 될 것으로 전망되고 있다.<sup>[15]</sup> 설계자들은 mixed-mode 시뮬레이션 환경을 매우 필요로 하고 있는데, 이 환경으로 시스템의 다른 부분들이 다른 추상적 레벨에서 시뮬레이션 가능하다. 시뮬레이션을 위한 프레임워크 기술의 출현은 mixed-mode 시뮬레이션 해결을 가져올 것으로 보인다. 설계자들은 좋은 VHDL 시뮬레이터, 회로 시뮬레이터, 논리 시뮬레이터를 가지고 mixed-mode, mixed-level 시뮬레이션을 위해 묶을 수 있는 것이다.

프레임워크 개발에 있어 예상되는 문제의 하나는 데이터 관리와 방법론에 관한 것으로, 이는 설계자가 칩의 설계에서 시스템 설계로 옮겨가는 추세와 관련이 깊다. 모든 요구 조건들을 만족시키기에는 설계 기술이 너무 빨리 변화하므로 완벽한 CAD 프레임워크는 존재하지 않으며 앞으로도 기대하기 힘들 것이다. 이것을 감안해 볼 때 CAD 프레임워크 자체가 유연성 있게 개발되어야 할 것이다. 또한 최근 연구가 활발한 오브젝트 중심 데이터 관리 시스템은 아직 결과가 만족스럽지는 않으나 앞으로 CAD 프레임워크의 하부구조로서의 역할을 충분히 해 낼 것

으로 기대되고 있다.


표준화 문제는 CFI쪽이 업체들의 직접적인 지원을 받고 있고, 그 영역도 비교적 한정시키고 있으므로 결과는 EIS보다 다소 빠를 것으로 예상된다. CFI와 EIS에 공동으로 참여하는 멤버들이 다수 있으므로, 두 프로그램은 상호간에 영향을 받을 것이며, 중복되고 경쟁적이기 보다는 상호 보완적인 방향으로 나갈 것이다. 한편 CFI와 같은 committee에서는 표준을 만들기보다, 비중있는 업체 몇몇이 모여서 표준안을 만들면, 곧 바로 표준화 될 수 있을 것이라는 의견도 있으나<sup>11)</sup> 이들 회사간에 공동 표준을 만들어 낼 만큼 별도로 협력하고 있지는 않고, 주요 업체들은 각기 그들 고유의 시스템을 만들 것으로 전망되며 외부 툴들이 자신들의 환경에 통합될 수 있도록 그들의 시스템을 개방시켜 나갈 것이다.

## Ⅷ. 결 론

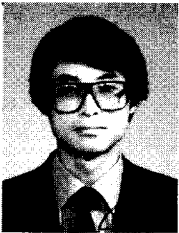
본고에서는 최근 기술 개발이 활성화되고 있는 CAD 프레임워크에 관한 개념과 요건, 주요 연구 개발 사례와 표준화 동향, 앞으로의 전망을 살펴 보았다. CAD 프레임워크는 치열한 CAD 시장 경쟁으로 인한, 근래의 업체간의 통합에서 발생하는 툴간의 통합 문제를 해결하기 위해 출현한 기술이고, 그 목적은 오로지 업체들의 상품판매에만 있다고 보는 견해들도 있으나, 사용자 측면에서도 CAD 시스템의 한 세대를 구획짓는 새로운 설계 환경이라는데 많은 사람들이 인식을 같이 하고 있다. 왜냐하면, 설계자들은 원하는 최적의 툴들을 이용해 가장 효율적인 CAD 시스템을 재단해 쓸 수 있고, 데이터 관리와 시스템 교육에 부담을 받지 않으므로 연구생산성도 훨씬 높아질 수 있기 때문이다. 툴 개발자의 입장에서는 인터페이스와 데이터 접근 구현에 종전에는 50% 정도의 시간을 투자하였으나 이를 보다 좋은 알고리즘의 개발에 투자할 수 있다.

이상적인 CAD 프레임워크의 출현은 가까운 시일 내에 기대하기 힘들 것으로 보이나, 빠른 시간내에 업계에서 수용할 수 있는 표준안이 빨리 만들어진다면 실용화는 가속화될 것이다. 표준화가 되지 않을 경우 사용자는 특정 프레임워크에 종속될 수 있기 때문에 CFI, EIS와 같은 표준화 활동은 그 귀추가 매우 주목된다 하겠다.

## 參 考 文 獻

- [1] K. Goldman, T. Stout, "A Design Automation Environment," VLSI Systems Design, June 1988.
- [2] "CAD Framework, Users, Goals, and Objectives," version 0.8, CAD Framework Initiative, Inc., May 1990.
- [3] 홍봉희, 장덕호, 광명신, 이철동, "VLSI/CAD 응용을 위한 데이터베이스 지원", 데이터베이스연구회지, 한국정보과학회, 제 6 권 1 호, 1990년 4 월.
- [4] D. Harrison, et al., "Electronic CAD Frameworks," *Proc. of the IEEE*, vol. 78, no. 2, Feb. 1990.
- [5] Ralph K. Cavin, Larry W. Sumney, Robert M. Burger, "The Semiconductor Research Corporation: Cooperative Research," *Proc. the IEEE*, vol. 77, no. 9, Sep. 1989.
- [6] Mario Silva, David Gedye, Randy Katz, Richard Newton, "Protection and Versioning for OCT," Proc. of 26th Design Automation Conference, June 1989.
- [7] Grant A. Dove, "Cooperative Research at MCC: A Focus on Semiconductor-Related Efforts," *Proc. of the IEEE*, vol. 77, no. 9, Sept. 1989.
- [8] "VISION," Mentor Graphics Corp., Apr. 1990.
- [9] "1st CADENCE Design Automation Conference," Cadence, Jan. 1990.
- [10] Bill Harding, "CFI support and user demand push open environments closer to reality," Computer Design, Nov. 1, 1989.
- [11] Giulio C. Grata, "Microelectronics Work in the ESPRIT Program," *Proc. of the IEEE*, vol. 77, no. 9, Sep. 1989.
- [12] "CFI News Line," vol. 1, Issue 2, CFI, June 1989.
- [13] John L Ebel, "The Engineering Information System Program," Proc. of CompCon88, 1988.
- [14] David Smith, "An Engineering Information System," VLSI Systems Design, Nov. 1988.
- [15] Lisa Gunn, "CAE in the 1990s: Will users want performance or integration?," *Electronic Design*, Jan. 11, 1990. 

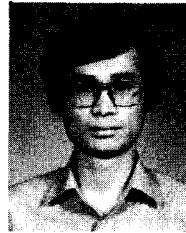
筆者紹介



張 德 浩

1958年 12月 8日生  
 1981年 2月 성균관대학교  
 전기공학과 졸업  
 1983年 2月 동대학(공학석사)

1983年 3月~1988年 2月 한국전자통신연구소  
 연구원  
 1988年 3月~현재 한국전자통신연구소 자동설계  
 S/W연구실 선임연구원



李 玄 燦

1956年 5月 5日生  
 1978年 2月 서울대학교  
 산업공학(학사)  
 1980年 2月 한국과학원  
 산업공학(석사)  
 1988年 4月 미시간대학교  
 산업공학(박사)

1980年 3月~1983年 6月 (주) 부산파이프 기획실  
 1988年 미시간대학교 강사  
 1988年 9月~현재 한국전자통신연구소 자동설계  
 S/W연구실 실장

郭 銘 信

1952年 1月 2日生  
 1977年 2月 인하대학교  
 전자공학과 졸업  
 1990年 2月 한양대학교 산업대  
 학원 졸업(전자계산전공)

1977年 2月 한국전자기술연구소 반도체설계부  
 연구원  
 1982年 3月 한국전자기술연구소 설계개발부  
 선임연구원  
 1985年 9月 한국전자통신연구소 자동설계응용  
 연구실 실장  
 1990年 4月~현재 한국전자통신연구소 자동설계  
 DB연구실 실장



李 哲 東

1952年 2月 12日生  
 1977年 2月 경북대학교 전자공학과  
 1989年 2月 한양대학교 산업  
 대학원 전자공학과(석사)

1977年 2月~1984年 1月 한국전자기술연구소 설계  
 개발실 선임연구원  
 1984年 2月~1985年 5月 한국전자기술연구소 설계  
 자동화연구실 실장  
 1985年 5月~1989年 10月 한국전자통신연구소 자동  
 설계기기연구실 실장  
 1989年 10月~현재 한국전자통신연구소 자동설계기술  
 개발부 연구위원  
 주관심분야: VLSI 설계 및 VLSI CAD 등임.