Windows 환경의 발전소 실시간 시뮬레이터 개발 툴 소개

조병학, 박신열, 손기헌, 김동욱, 서한창, 이용관

(한국전력공사 전력연구원)

Introduction to Development Tool for Windows Based Real-Time Power Plant Simulators

Byung Hak Cho, Shin Yeol Park, Gi Hun Son, Dong Wook Kim, Han Chang Suh, Yong Kwan Lee

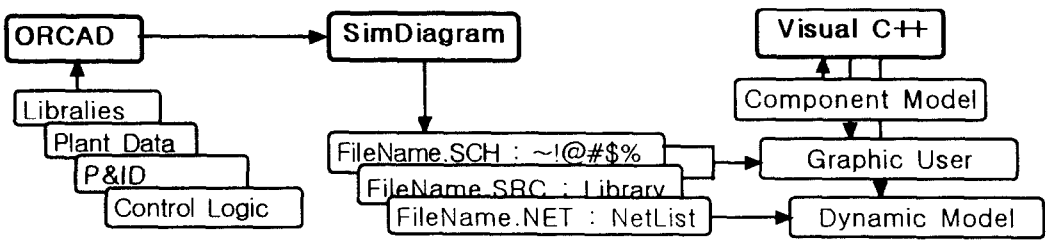
(Korea Electric Power Research Institute)

요 약

한국전력공사 전력연구원은 10년간의 시뮬레이터 개발 경험을 바탕으로 Windows(NT)환경의 시뮬레이터 개발 툴인 PowerSim을 독자적으로 개발하고 이를 이용하여 화력발전소 DCS(Distributed Control System) 검증용 시뮬레이터를 개발하고있다. PowerSim은 GMB(Graphic Model Builder)를 갖춘 국내 최초의 시뮬레이터 개발툴로 다양한 발전소 기기모델과 강사조작반기능을 갖추고 있다. PowerSim은 완벽한 GUI (Graphic User Interface)환경을 지원하여 User가 Icon Drag 방식으로 시뮬레이션 도면(SimDiagram)을 그리면 그래픽 에디터에서 출력된 각종 기기의 접속상태를 나타내는 Netlist를 변환기가 처리하여 기기의 연결상태를 정규화하고 Scheduler는 기기모델(일종의 Subroutine)을 Netlist에 맞게 Scheduling하여 Executive에서 실행 가능한 형태로 만드는 모든 과정이 자동화되어 있다. 따라서, 개발자는 발전소 P&ID(Pipe and Instrument Drawing)에 기초하여 Simdiagram을 그리고 발전소 데이터를 입력하는 것만으로 실시간 시뮬레이터를 구현할 수 있다. 본 논문에서는 PowerSim의 개요와 GMB(Graphic Model Builder) 및 강사조작반에 적용된 GUI 환경과 실시간 Executive에 대해 다룬다.

1. 개요

최근 들어 컴퓨터기술의 발달에 힘입어 발전소의 아날로그식 제어설비가 DCS 등의 첨단기술 설비로 대체되고 있는 추세여서 이에 대처하기 위한 발전소 운전원의 훈련은 물론 DCS 자체의 시험을 위해서도 시뮬레이터의 필요성이 한층 강조되고 있다. 시뮬레이터는 크게 발전소모델, 실시간 Executive, 데이터베이스, 강사조작반과 입출력제어장치로 구성된다. 이들 중 발전소모델은 개발을 위해 장기적인 시간이 필요한 특성을 가지고 있고, 한전은 10년간의 시뮬레이터 개발경험[1]을 통해 대부분의 모델을 자체기술로 개발할 수 있는 능력을 보유하고 있으며 또한 컴퓨터기술이 발달됨에 따라 발전소설계 및 검증용 코드를 시뮬레이터에 직접 적용할 수도 있는 상황이다. 따라서, 현시점에서 발전소모델은 시뮬레이터의 성능개선 차원에서 지속적으로 연구개발해야 할 필요성을 가지고는 있지만 시뮬레이터 개발에 있어 주요변수로 주목되지는 않는다. 그러나, 모델을 보유하고 있는 것과 시뮬레이터를 개발하는 것은 별개의 문제이다. 발전소의 수많은 계통을 수작업에 의존하여 모델링하기 위해서는 많은 시간이 소요되며 인적에러도 발생하기 마련이고 유지보수 면에서도 어려움이 많다. 이와 같은 이유로 발전소계통을 도식화하고 이에 데이터를 입력하는 것만으로 발전소모델과 데이터베이스를 생성하고 실행파일을 만드는 등의 작업을 자동으로 수행하는 시뮬레이터 개발툴의 중요성이 강조된다. 시뮬레이터 개발툴은 전술한 시뮬레이터 구성부분 이외에 GMB(Graphic Model Builder)를 추가로 가지고 있다. 한전은 GMB와 Database 및 실시간 Executive를 갖춘 시뮬레이터 개발툴 PowerSim을 자체기술로 개발하고 이를 호남화력발전소 보일러제어용 DCS를 검증하기 위한 시뮬레이터 개발에 적용하고 있다. 본 논문에서는 PowerSim의 GMB, 강사조작반 GUI 및 실시간 Executive의 개요에 대해 다루기로 한다.



<그림 1> OrCAD를 그래픽 에디터로 활용한 GMB의 구성

2. Graphic Model Builder

Graphic Model Builder는 크게 시뮬레이션 대상을 도면(SimDiagram)으로 표현하는 그래픽 에디터와 그래픽 에디터의 출력을 모델링에 사용할 수 있는 일련의 정보로 변환하는 Netlist변환기 및 각 기기모델(일종의 서브루틴)을 Scheduling 하여 Executive가 실행할 수 있는 형태로 만드는 Scheduler로 구성된다.

2.1 OrCAD와 Model Builder

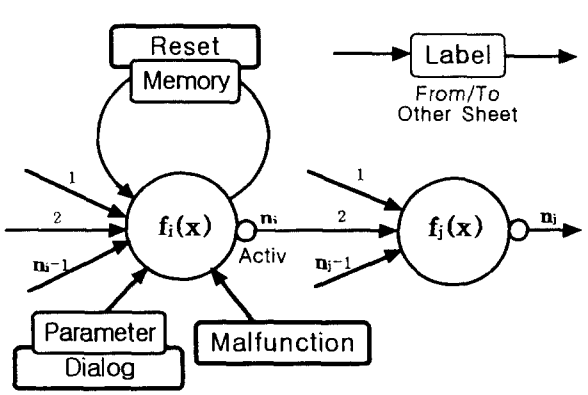
그래픽 에디터를 새로 개발하기 위해서는 많은 시간이 소요되므로 PowerSim에서는 범용 CAD를 사용하였다. 설계 단계에서 AUTOCAD, VISIO, OrCAD 등의 범용 CAD 툴을 고려하였으나 이들 중 OrCAD [2]는 기기심볼을 라이브러리로 관리할 수 있고 전자회로를 설계하는 데에 사용되는 특성상 각 심볼의 연결상태인 Netlist를 제공하는 등의 GMB용 그래픽 에디터로서의 필수요건을 갖추고 있어 이를 채택하였다. 그림 1은 OrCAD를 그래픽 에디터로 사용하여 강사조작반 GUI와 발전소의 다이내믹 모델을 자동으로 생성하는 과정을 보이고 있다. OrCAD를 GMB의 그래픽 에디터로 사용하기 위해서는 먼저 기기심볼의 그래픽 라이브러리를 후술하는 Netlist 변환기에서 규정한 핀 배열 규칙에 따라 만들고 이들과 발전소 데이터, P&ID(Pipe and Instrument Drawing) 및 제어로직도면 등을 이용하여 SimDiagram을 그려야 한다. OrCAD는 하나의 SimDiagram에 대해 Schematic 파일(.SCH), 그래픽 라이브러리 파일(.SRC) 및 Netlist 파일(.NET)를 제공한다. 이들 중 심볼, 선분, Text 등의 위치를 가지고 있는 Schematic 파일은 바이너리 파일이므로 해독이 필요하며, 라이브러리 파일은 심볼의 폭과 높이 및 심볼 내부의 그래픽 정보를 제공한다. ProTRAX[3], US3[4] 등의 GMB에서는 그래픽 에디터와 GUI가 공통으로 사용되는 반면 PowerSim은 그래픽 에디터가 그린 도면의 정보로부터 Visual C++에서 GUI를 재현한다. 따라서, PowerSim은 OrCAD가 제공하는 정보를 쉽게 변경하여 GUI 기능을 향상시킬 수 있는 반면 OrCAD가 제공하는 정보에 한계가 있다는 단점을 가지고 있다.

2.2 Netlist 변환기

SimDiagram으로 표현한 발전소계통의 다이내믹 모델을 자동으로 생성하고 SimDiagram 사이의 데이터 통신을 자유롭게 하기 위해 변환기는 OrCAD가 생성한 Netlist와 Schematic 및 라이브러리 파일을 활용하여 핀 배열에 의거 기기심볼을 정규화한 후 Sim Diagram에 그려져 있는 모든 기기심볼과 EP (External Parameter), BC(Boundary Condition) 등의 시뮬레이션 제어심볼의 연결상태를 ".CON"과 ".MAIL" 파일로 출력하고, 규정에 맞지 않는 연결상태가 발견되면 리포트파일 ".RPT"로 이를 User에게 알려준다.

A. 제어 SimDiagram의 변환

제어 SimDiagram에 사용되는 심볼의 경우 제일 마지막 핀이 출력에 해당하는 등의 규정을 정하였다. 그림 2는 변환기가 제어 SimDiagram을 정규화하는 과정을 보이고 있다.



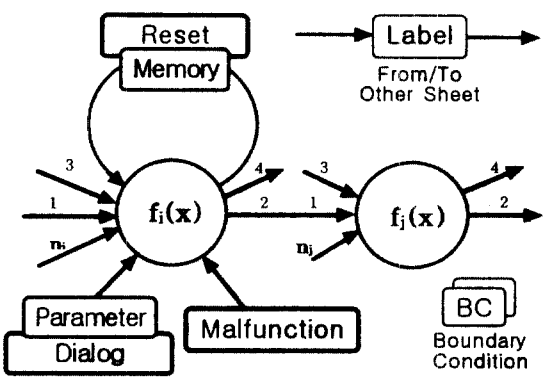
<그림 2> 제어 SimDiagram의 정규화

정규화가 끝나면 각 심볼사이의 연결상태를 처리한다. 계산순서는 심볼에 연결된 입력 핀이 모두 새로운 값을 가지고 있을 때 'Active' 신호를 출력하여 연결된 후속 심볼이 입력이 갱신되었음을 인식할 수 있도록 하여 뉴럴네트워크의 Feedforward 계산과 유사한 형

식으로 정한다. 그러나 내부에 Feedback 루프가 있는 경우에는 입력되는 핀의 값은 과거에 계산된 값을 사용한다. 서로 다른 SimDiagram 사이의 데이터 교류를 위한 Mail Address는 'Label'을 기준으로 처리한다. 그러나 같은 SimDiagram에서의 Mail은 Netlist가 자동으로 연결상태를 지정하므로 변환기에서 별도로 처리하지 않는다. 이외에 강사조작반 기능을 고려하여 Reset, Malfunction 및 Parameter 갱신을 위한 연결이 추가되는데 이들은 후술하는 Executive에서 직접 처리된다.

B. 공정(Process) SimDiagram의 변환

공정은 파이프, 펌프, 밸브, 열교환기 및 드럼 등의 심볼로 구성되는데 이들의 연결은 제어 SimDiagram에 적용된 규정보다 복잡하다. 유체의 흐름이 양분되어 있는 경우(예를 들면 열교환기)를 고려하여 1번과 2번 핀은 1차측 유로 3번과 4번 핀은 2차측 유로의 입력과 출력측 경로를 나타낸다. 이외에 다른 핀들은 각 심볼마다 정의하여 규정하였다.



<그림 3> 공정 SimDiagram의 정규화

특성상 공정의 시뮬레이션에서는 압력, 유량 등의 변수에 대한 경계조건(Boundary Condition)이 추가된다. 제어 SimDiagram의 변환기가 뉴럴네트워크처럼 계산 순서를 정하는 반면 공정 SimDiagram의 변환기에서는 각 심볼에 걸린 압력을 구하기 위해 어드미턴스 매트릭스를 만들어야 하는 등 구조적으로 복잡하여 다른 논문[5]에서 구체적으로 설명하였다. 어드미턴스 매트릭스는 Sparse 형태를 하고 있으며 이의 Inverse로부터 각 기기에 걸린 압력차이를 구하고 이로부터 유량을 구한 후 유량과 열의 입출력 등을 고려하여 유체의 온도를 얻는다.

2.3 Scheduler

Scheduler는 Netlist 변환기가 제공하는 기기들의 연결관계(".CON 파일")를 이용하여 SimDiagram에 그려진 계통모델을 Executive가 실행할 수 있도록 기기 모델을 서브루틴 형태로 Call할 수 있는 환경을 만들고 데이터베이스를 생성하는 한편, ".MAIL" 파일을 이용하여 서로 다른 SimDiagram 사이의 입출력 데이터를 지정한다. ProTRAX와 US3는 변환기에서 원시코드를 생성하고 컴파일/링크과정을 거쳐 실행파일을 생성하는 반면 PowerSim은 서브루틴형태의 기기모델을 Scheduling하여 실행 가능한 모델로 만든다는 점에서 차이가 있다.

3. GUI와 강사조작반

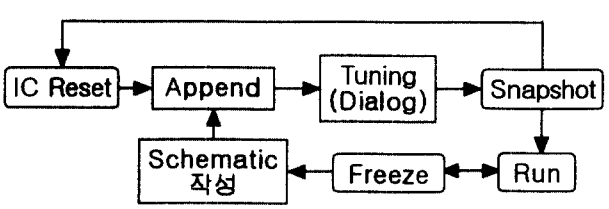
강사조작반은 시뮬레이터를 효율적으로 운영할 수 있도록 실행/정지, 초기조건의 Snapshot/Reset, Trend Display, Malfunction, Remote Function 등의 기능을 제공한다. 이외에 PowerSim 고유의 특징인 SimDiagram을 추가 또는 삭제하는 기능도 지원하므로 GMB와 강사조작반을 자연스럽게 통합할 수 있는 장점을 가지고 있다. 원시코드를 생성하는 GMB에서는 그래픽 에디터에서 기기의 파라미터를 입력하여야 하고 컴파일/링크 과정을 통해 실행파일을 만들지만 PowerSim의 경우 강사조작반에서 실행 도중 대화상자를 통해 입력이 가능하므로 모델의 온-라인 튜닝이 가능하여 시뮬레이터 개발기간을 현저히 단축할 수는 있는 잇점을 가지고 있다.

3.1 GUI(Graphic User Interface)

GUI의 기본정보는 OrCAD가 제공하는 Schematic과 라이브러리 파일이며 그래픽은 Visual C++ 가 제공하는 MFC(Microsoft Foundation Class)를 사용하여 MDI(Multiple Document Interface) 환경에서 구현하였다. OrCAD 라이브러리 파일은 각 심볼의 폭과 높이를 제공하므로 마우스를 통해 윈도우에 그려진 특정 심볼을 선정할 수 있다.

3.2 강사조작반 기능

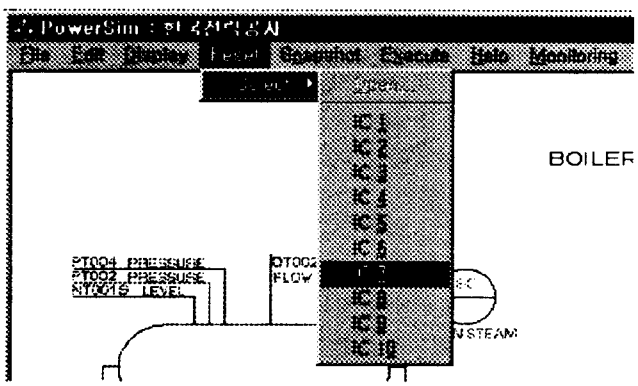
강사조작반 기능 사이의 연결관계를 그림으로 표현하면 아래의 그림 4와 같다.



<그림 4> 강사조작반 기능의 도식적 표현

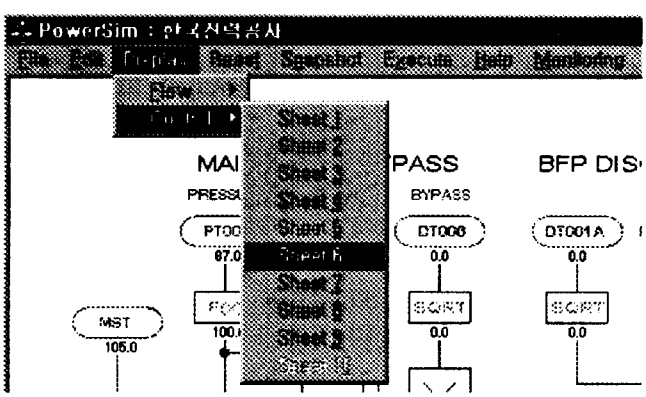
이들 기능은 메뉴와 대화상자(Dialog)에 의해 구현되었으며 시뮬레이터를 제어하는 기능이외에도 시뮬레이터에 발전소 계통 모델을 추가(Append)할 수 있는

기능도 포함하고 있다. 그림 5는 메뉴의 주요 부분을 보이고 있다.

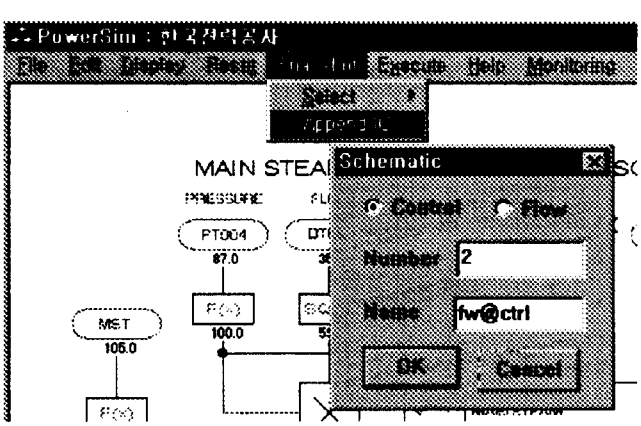


<그림 5.1> 강사조작반 메뉴 - Reset

<그림 5.2> 강사조작반 메뉴 - Execute

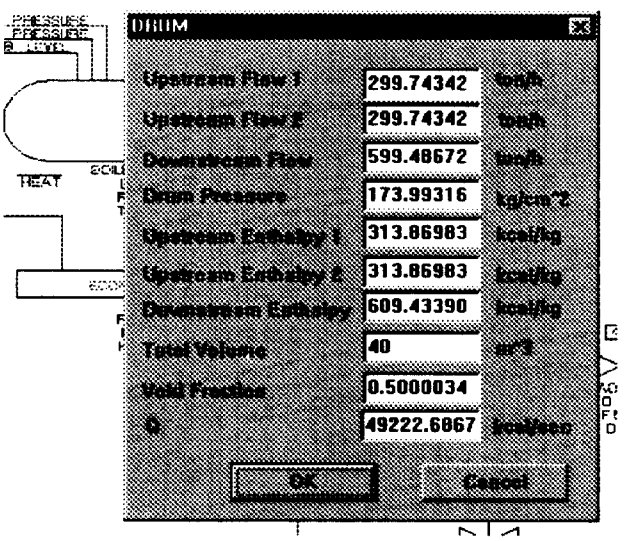


<그림 5.3> 강사조작반 메뉴 - Display

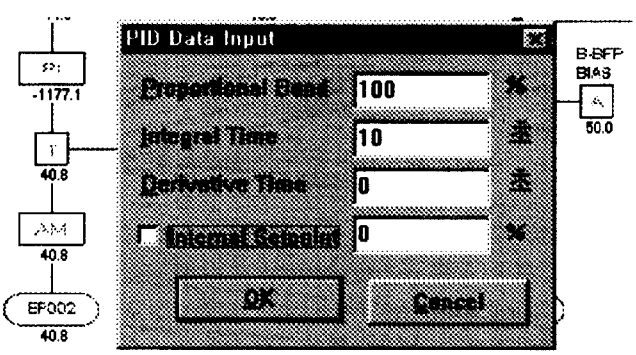


<그림 5.4> 강사조작반 메뉴 - Append IC

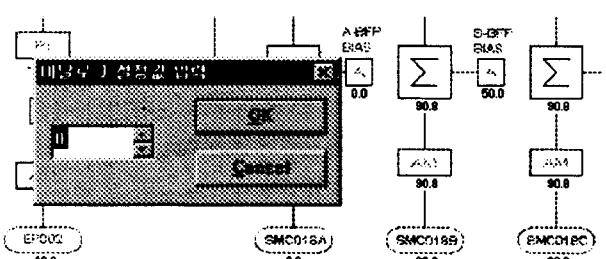
PowerSim에서는 주요변수(제어 SimDiagram의 경우 함수의 출력값, 공정 SimDiagram의 경우 유량, 온도 압력 등)를 심볼 아래의 여백에 표시하지만, 그 외의 변수는 대화상자를 통해 온-라인으로 확인이 가능하고 이의 변경도 가능하다. 특히 제어 SimDiagram의 경우 스위치의 전환, 설정값의 변경 등을 수행할 수 있어 PC 만으로도 시뮬레이터의 운전이 가능하며, 시뮬레이션 환경은 언제든지 메뉴의 Snapshot 기능으로 데이터베이스에 저장할 수 있다. PowerSim 대화상자의 일부를 보이면 그림 6과 같다.



<그림 6.1> 강사조작반 대화상자 - Drum



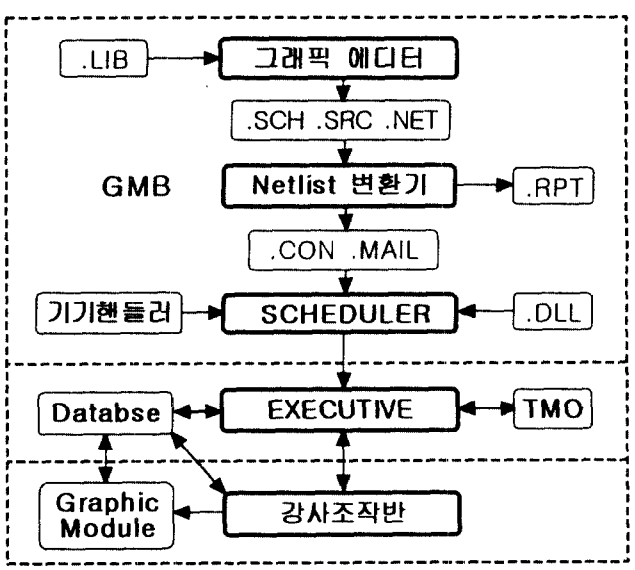
<그림 6.2> 강사조작반 대화상자 - PID 데이터 입력



<그림 6.3> 강사조작반 대화상자 - 설정값 입력

4. Executive

Executive는 Netlist 변환기와 Scheduler가 만든 다이내믹 모델을 실시간으로 실행한 후 계산결과를 Data base에 기록하고 User Interface를 처리하는 등의 일을 처리한다. Executive의 실시간 Scheduler (전술한 기기모델 Scheduler와는 다름)는 전체 시뮬레이터를 실시간으로 수행하는 중요한 일을 관장하는데 PowerSim에서는 TMO(Time-Triggered Message-Triggered Object System)를 적용하였다. TMO는 미국의 UCI(University of California at Irvine)에서 최초로 개발된 후 국내에서도 점차 산업에 적용되고 있는 실시간 Scheduler이다. TMO는 Time-Trigger에 의한 주기적인 Scheduling과 Message-Trigger에 의한 인터럽트식 Scheduling을 지원하는데 PowerSim에서는 Time-Triggered 방식만 사용한다. PowerSim의 구조를 GMB, Executive 및 강사조작반으로 크게 나누어 표현하면 아래의 그림 7과 같다.



<그림 7> PowerSim의 구성

Database는 C언어가 제공하는 Structure로 만들어져 Visual C++의 Document Class에 정의되어 있다. 강사조작반의 기능을 써서 시뮬레이션 환경을 Snapshot 할 때는 이들의 모든 내용이 하드디스크에 저장되고, 반대로 특정 초기조건으로 Reset 할 때는 하드디스크에 저장되어 있는 값으로 Reload된다.

5. 결론

PowerSim은 현재 버전 1.0이 개발된 상태이며 이를 활용하여 호남화력발전소 보일러제어용 DCS를 시험하기 위한 시뮬레이터를 제작 중이다. 보일러 수위제어계통의 시뮬레이션 결과 GMB에 의해 생성된 발전소 모델과 강사조작반 기능이 정상적으로 동작함을 확인 하였고, PowerSim의 특징인 강사조작반과 GMB가 통합된 구조가 모델 파라미터 변경 시마다 컴파일/링크를 하여야하는 기존툴에 비해 시뮬레이터의 튜닝이 쉽고 개발공정을 줄일 수 있음을 볼 수 있었다. PowerSim은 GMB에 범용 CAD인 OrCAD를 사용하여 툴자체의 개발비용이 저렴할 뿐 아니라 이를 이용하여 개발된 시뮬레이터는 PC의 Windows(NT) 환경에서 메뉴와 대화상자를 통해 기기와 제어기의 특성 파라미터를 쉽게 변경하고 이의 응답을 확인할 수 있는 등 전문가가 아니더라도 시뮬레이터 운전이 가능하여 발전소 운전원과 계측제어요원의 훈련에 활용시 친숙하고 경제적인 시뮬레이터로 활용될 전망이다. 향후 터빈제어용 DCS 검증용 시뮬레이터를 제작하면서 성능을 개선하면 PowerSim은 전범위 시뮬레이터를 만들기 위한 툴로서의 기본형태를 갖출 것으로 기대된다.

[참고문헌]

[1] 이용관 외 9명, "발전소 운전원 훈련용 모의제어반 국산화 개발적용 [최종보고서]", KEPRI Technical Report, 1998

[2] OrCAD CAPTURE for Windows, OrCAD Inc., 1995

[3] ProTRAX Analyst's Instruction Manual Volume 1 and 2, TRAX Corporation, 107 Vista Centre Drive, Forest, VA 24551, 1998

[4] US3 User Guide Release 5, S3 Technologies Columbia, Maryland, June 1994

[5] 손기헌 외 3명, "발전소 시뮬레이션 툴을 위한 열수력 모델 구현", 한국시뮬레이션학회 '98 추계 학술대회 발표예정, 1998