

정보가전용 임베디드 프로토타이핑 시스템 설계*

이정배** · 이영란** · 정영진** · 김종일** · 김남동** · 신현철***

요 약

임베디드 제품의 개발 비용과 경쟁력 강화를 위한 개발 방법론으로 널리 이용되고 있는 것이 프로토타이핑이다. 프로토타이핑에는 컴퓨터 기술을 이용한 가상 프로토타이핑과 레고 블록을 이용한 실물 프로토타이핑이 있다. 가상 프로토타이핑 환경에서는 다양한 컴포넌트와 라이브러리를 제공함으로써 정밀한 단위까지의 타겟 모델의 시뮬레이션이 가능하고, 실물 프로토타이핑 환경에서는 타겟 모델의 임베디드 구동 환경에 가까운 시뮬레이션 환경을 제공한다. 하지만, 가상 프로토타이핑 환경에서는 실제 타겟 모델의 구동 환경 요소까지의 시뮬레이션이 어렵고, 실물 프로토타이핑 환경에서는 다양하고 정밀한 정보가전분야의 소형 제품까지의 시뮬레이션이 어려운 실정이다. 본 논문에서는 이런 가상과 실물 프로토타이핑 환경의 단점을 보완하기 위한 통합 프로토타이핑 시스템을 위한 인터페이스를 ActiveX 컴포넌트 기술과 TCP/IP 소켓 통신, 데이터 연동 인터페이스를 설계 및 구현하였다.

A Design of Embedded Prototyping System for Home Appliance*

Jeong-Bae Lee** · Young-Ran Lee** · Young-Jin Jung**
Jong-Il Kim** · Nam-Dong Kim** · Hyun-Cheul Shin***

ABSTRACT

It seems necessary to have a new methodology to develop the complex embedded system in a short time with a small amount of money and secure the reliability of products. For this, in this study, the investigator presents the state-of-the-art technology with which the limitation of physical prototyping and virtual prototyping can be overcome. Also as a developing methodology of the embedded system in which the new technology is applied, the investigator suggests a new methodology of developing process for the integrated design, integrated simulation, and debugging of the physical prototyping and the virtual prototyping.

Key words : Virtual Prototype, Real Prototype, Embedded System, Rapid PLUS, ESPS

* 본 연구는 산학협동재단 2005년 학술연구과제(05-B-II)로 수행되었습니다.

** 선문대학교 컴퓨터정보학과

*** 백석문화대학 컴퓨터정보학부

1. 서 론

현재 널리 이용되고 있는 프로토타이핑 방법론에는 실물 프로토타이핑 방법론과 컴퓨터 기술을 이용해서 다양한 컴포넌트와 라이브러리를 제공하는 가상 프로토타이핑 방법론이 있다.

실물 프로토타이핑은 공장 자동화 시스템 모형이나 자동판매기 같은 실제 임베디드 제품을 제작하고 테스트하는데, 레고 블록을 이용한다. 하지만, 실물 프로토타이핑으로는 임베디드 시스템이 적용된 정보 가전제품의 다양한 컴포넌트나 세밀한 컨트롤을 위한 라이브러리를 지원하지 못한다. 그래서 새롭게 등장한 개발 방법론이 가상 프로토타이핑 방법론이다.

가상 프로토타이핑 방법론은 컴퓨터를 이용하여 다양한 컴포넌트와 라이브러리를 제공함으로써 제품을 시각적인 3D로 모델링 할 수 있다. 3D로 모델링된 제품은 디자인 변경이 용이하고 세밀한 컨트롤을 할 수 있도록 한다. 하지만, 가상 프로토타이핑은 컴퓨터 기술을 이용한 시각적인 가상 시뮬레이션이기 때문에 실제 제품을 구동시키기 위한 임베디드시스템 환경에서의 정확한 시뮬레이션이라고 할 수 없는 불확실성이 있다.

이러한 문제점들을 해결하기 위하여, 본 논문에서는 실물 프로토타이핑의 실제 제품의 임베디드시스템 구동 환경과 컴퓨터 기술을 이용한 시각적인 3D 모델링 제품의 가상 프로토타이핑 시뮬레이션 환경을 동시에 지원할 수 있는 통합 프로토타이핑 시스템을 위한 인터페이스를 설계 및 구현 할 것이다. 이러한 통합 프로토타이핑 제품 개발 방법은 임베디드 제품 설계에서 발생하는 오류 수정, 기능의 보완 등을 즉각적으로 제품개발에 반영할 수 있고, 기존의 가상 및 실물 프로토타입에서 개발되어 있는 프로토타입들을 재사용 가능한 환경을 제공한다.

2. 관련 연구

현재, 프로토타이핑은 금속·조형술 분야, 외형

설계에 중심을 둔 건축·설계, 디자인 분야 그리고 의료 장비 분야의 설계자에게 3차원 객체의 형상을 시각화하고 정의하는 수단으로 제공이 된다. 하지만, 대체로 제품의 외형 설계 구현에 사용되고 있다. 또한 컴퓨터 기술을 이용한 공학적 관점에서의 프로토타이핑 방법이 활발히 연구 중이다[2-4].

기존의 프로토타이핑은 시뮬레이션을 의미했다. 하지만 최근에는 복잡하고 고난도의 개발기술을 요구하는 임베디드시스템 분야의 신속한 제품 개발과 최고의 신뢰성을 갖춘 제품 생산을 위하여 프로토타이핑의 연구가 활발하게 진행되고 있다.

임베디드시스템의 특성상 개발자들 간의 공동 작업에서 다양한 전공분야의 개발자들의 이해가 필수적이다. 따라서, 보다 효과적인 문제 해결 방법을 고안하기 위한 방법으로 실물 프로토타이핑을 이용하여 프로젝트를 진행하면, 개발 과정에서 하드웨어와 소프트웨어 개발자의 동시 작업으로 개발 시 당면하게 되는 문제점을 즉시 해결할 수 있다. 그러나 실물 프로토타이핑이 실제 제품에 적용이 불가능하다고 가정 했을 때, 이와 유사한 환경에서 시뮬레이션 할 수 있는 모형을 만들어 테스트를 할 수 있는 여건 조성이 필요하게 되었다.

그래서 기존의 물리적인 모의제품(Prototype)을 이용하는 방법과는 달리 컴퓨터상에서 가상으로 만들어진 프로토타입으로 실제 제품과 같은 외형을 제공하고, 시스템 요구 사항이 복잡한 하드웨어 시뮬레이션을 소프트웨어로 가능하게 하는 가상 프로토타이핑(Virtual Prototyping)이 제공되었다. 이는 임베디드시스템의 개발 초기 단계에서 사용자의 요구를 정확히 반영하고, 기능적, 비 기능적 요구를 효과적으로 구현하기 위하여 고안된 시스템 제약 조건 추출방법으로 컴퓨터 기술을 이용하여 제품의 프로토타입을 제공한다.

그리고 가상 프로토타이핑은 인터페이스가 복잡한 제품 설계 및 실제 제품으로 테스트를 하

는 것과 같은 현실감을 제공하기 위해서 컴퓨터 상의 데이터를 이용한다. 이는 시제품의 검증을 위한 실물 프로토타이핑 방법보다 전체 개발비용 절감과 생산 주기 등의 문제에 편리함을 제공한다.

이러한 가상 프로토타이핑 개발 도구는 e-sim사의 RapidPlus, WorkShop, Virtio사의 Virtio, i-Logix의 Rapsody, Matworks사의 MTLAB 등과 같은 GUI 환경의 개발 도구가 상용화되어 있으며 이러한 가상 프로토타이핑은 제품 출시 전 하드웨어나 소프트웨어 분야의 완벽한 시뮬레이션을 위하여 컴퓨터상의 데이터 형태로, 실물 프로토타입을 대체 할 수 있는 기술이다.

실물 프로토타이핑은 제품의 외형 변경 및 기능 추가시 새로운 프로토타입을 만들어야 하지만, 가상 프로토타이핑은 개발과 시뮬레이션을 동시에 실행하면서 설계상의 오류 및 성능을 보완한 개발과정에서 즉시 개발자들에게 제공한다. 시제품을 제작한 후 성능평가의 과정을 거치는 실물 프로토타이핑에 비하여 개발과정의 시간적, 경제적 손실을 최소화함으로써 제품의 경쟁력을 극대화 하는 효과가 있다. 가상 프로토타이핑은 현재 가장 활발히 연구가 진행되고 있는 분야로 컴퓨터 기술을 이용한 다양한 컴포넌트와 플랫폼 라이브러리를 제공함으로써 실물 프로토타이핑 방법 보다 하드웨어와 소프트웨어라는 이질적인 시스템의 통합 개발 및 통합 디버깅과 현실감 있는 통합 시뮬레이션을 제공한다.

그리고 본 논문에서는 위에서 서술한 실물과 가상 프로토타이핑의 부족한 부분을 상호보완 할 수 있는 통합 프로토타이핑 시스템을 위한 인터페이스를 설계 및 구현한다. 그러기 위해서 먼저 가상 프로토타이핑 환경에서는 다양한 컴포넌트 Object들을 지원하고 Object들을 위한 다양한 이벤트나 속성, 기능들을 지원하는 개발 도구인 RapidPLUS를 이용하여 시뮬레이션 인터페이스와 데이터 처리 인터페이스를 설계 및 구현한다.

3. 인터페이스 설계

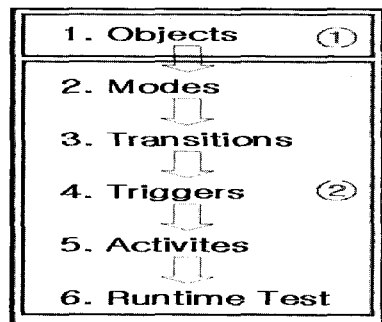
본 장에서는 가상 프로토타이핑 환경에서 인터페이스를 설계하기 위해서 e-sim사의 가상 프로토타이핑 개발 도구인 RapidPLUS를 이용하여 가정용 전기 세탁기 프로토타입을 위한 시뮬레이션 인터페이스를 설계하고, 시뮬레이션된 가상 프로토타이핑 데이터를 실물 프로토타이핑과 연동하기 위한 데이터 명세 언어를 정의한다.

그리고 정의된 데이터를 ActiveX 컴포넌트를 이용해서 가상 프로토타이핑 개발 도구에서 TCP/IP Socket 통신을 이용해서 데이터 연동 인터페이스로의 데이터 전송을 처리하는 데이터 처리 인터페이스를 설계한다.

마지막으로 가상 프로토타이핑 환경에서의 데이터를 실물 프로토타이핑 환경에서의 시뮬레이션을 위한 데이터로 사용하기 위한 데이터 연동 인터페이스를 설계한다.

3.1 시뮬레이션 인터페이스 설계

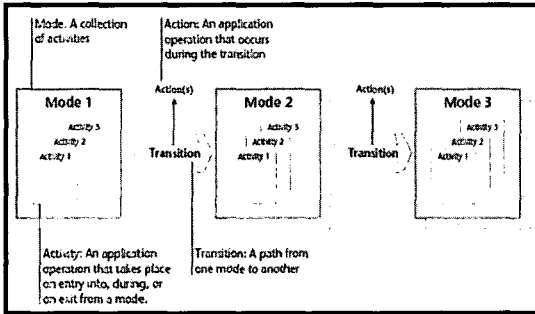
가상 프로토타이핑 개발 도구인 RapidPLUS를 이용한 사용자 인터페이스를 설계하는 데는 아래 (그림 1)과 같이 시뮬레이션을 하는데 있어서 그래픽한 Object부분과 Object들의 이벤트나 기능, 행동을 정의하는 부분(Modes, Transitions, Triggers, Activities)으로 나뉘어서 설계한다.



(그림 1) 가상 프로토타이핑 설계 단계

3.1.1 Object 기능 정의

본 절에서는 구성된 Object들을 기반으로 Object들 간의 기능을 정의한다. RapidPLUS에서의 기능 정의는 다음 (그림 2)와 같은 논리 흐름을 따른다.

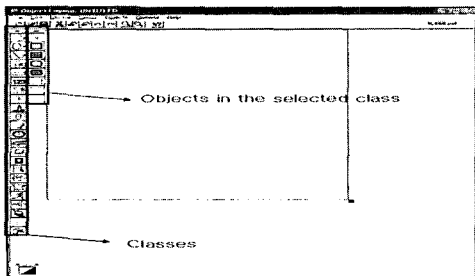


(그림 2) 기능 정의를 위한 논리 흐름

- Mode : activities의 집합
- Activity : 한 모드로부터 빠져나가거나 한 모드에 있거나, 한 모드로 진입 했을 때 일어나는 RapidPLUS 애플리케이션 동작
- Transition : 한 모드에서 다른 모드로 가는 경로
- Action : 트랜지션 동안에 일어나는 RapidPLUS 애플리케이션 동작

3.1.2 Object 인터페이스 설계

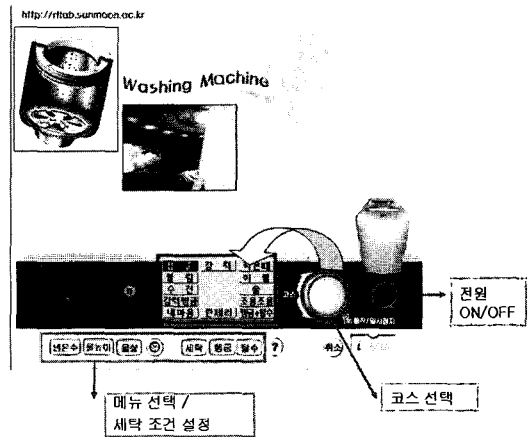
아래 (그림 3)과 같이 개발 도구에서 지원해주는 Object Layout에서 Class의 Object들을 이용해서 가상 프로토타이핑 인터페이스를 설계할 수 있다.



(그림 3) Object Layout

시물레이션 인터페이스는 다음과 같은 역할을 하는 Object들을 Object Layout에 (그림 4)와 같이 구성한다.

- ① 가정용 전기 세탁기의 전원을 켜다.
- ② 코스를 선택한다(표준, 강력, 행굼 등).
- ③ 냉·온수, 물높이, 물살의 특성을 선택한다.
- ④ 세탁, 행굼, 탈수의 조건을 선택한다.



(그림 4) 시물레이션 인터페이스

위와 같이 Object들의 구성 후 Object들간의 정의를 위한 논리 설계는 RapidPLUS에서 제공하는 Logic Editor와 Logic Palette를 이용해서 Object들간의 이벤트, 행동, 기능의 설정이 가능하다.

3.2 데이터 연동 명세 정의

가상 프로토타이핑 환경에서 시물레이션된 데이터를 실물 프로토타이핑 환경에서 시물레이션하기 위해서는 가상과 실물 프로토타이핑 환경간의 연동을 위한 데이터 명세를 정의해야 한다.

본 절에서는 가상 프로토타이핑 환경에서의 데이터 명세를 정의하였다. 데이터 처리를 위한 데이터 명세의 예는 다음 <표 1>, <표 2>와 같다.

〈표 1〉 코스 데이터 명세

courseSlectStage			
setString	Label	courseIndex	name
01	표준	1	countStandard
02	강력	2	countPower
03	이불	4	countQuilt
04	울	5	countWool
05	조용	6	countQuiet
06	헹탈	7	countRince_Hydration
07	란제리	8	countRangeri
08	내마음	9	countMySelf
09	수건	11	countTowel
10	불림	12	countSoak1

〈표 2〉 물 높이 데이터 명세

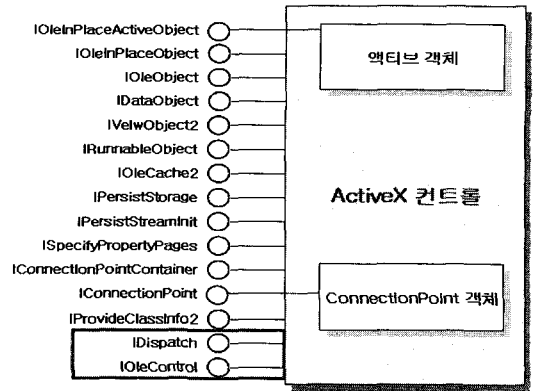
waterHeight			
setString	Label	int_waterHeight	name
01	고	1	level_1
02	중	3	level_3
03	저	5	level_5
04	소	7	level_7

3.3 데이터 처리 인터페이스 설계

본 절에서는 가상 프로토타이핑 환경에서 실물 프로토타이핑 시뮬레이션과의 연동을 위해서 ActiveX 컴포넌트 기술과 TCP/IP 소켓 통신을 사용해서 데이터 처리 인터페이스를 설계한다.

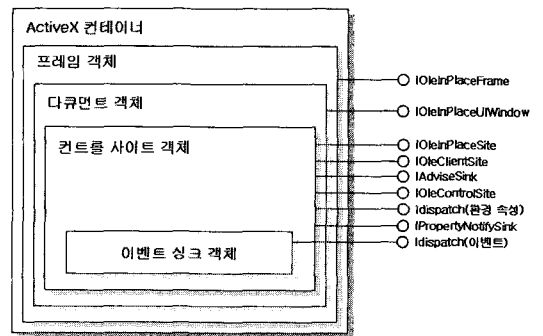
3.3.1 ActiveX 컴포넌트 설계

본 절에서는 ActiveX 컨트롤을 사용한 컴포넌트 기반의 크기가 작고 인터넷 환경에서 컨테이너와 효율적으로 상호 작용하기 위하여 여러 가지 새로운 인터페이스를 사용한다. ActiveX 컨트롤은 아래 (그림 5)와 같이 여러 개의 인터페이스를 제공한다. 그리고 본 시스템에서는 IDispatch와 IOlecontrol 인터페이스를 주로 사용한다.



(그림 5) ActiveX 컨트롤

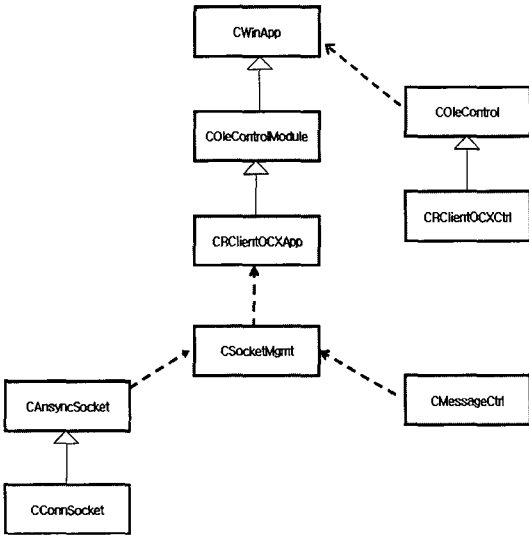
ActiveX 컨테이너는 (그림 6)과 같이 일종의 윈도우로서 하나의 윈도우의 자식 윈도우로 부모 윈도우에 부착되어 사용되며, 주로 사용자 인터페이스에 관련한 작업을 수행하며, 부모 윈도우와 메시지를 주고 받음으로서 통신을 수행하게 된다.



(그림 6) ActiveX 컨트롤 컨테이너

본 시스템에서는 아래 (그림 7)에서 보듯이 여러 클래스로 구성되어 있으며, 서로 상속과 포함관계를 이루고 있다. 메인 클래스는 CRClientOCXApp 클래스이고 그리고 ActiveX 컨트롤 인터페이스를 이용해서 데이터 연동 클래스와 연결을 하기 위해서 소켓 통신 컨트롤인 SCocketMgmt 클래스와 비동기 소켓 클래스인 CAnsyncSocket, 데이터 송/수신 처리를 위한 CConnSocket 클래스등을 사용했다.

또한, 가상 프로토타이핑 환경에서 데이터 처리 인터페이스의 버튼 Object들의 이벤트에 적용하기 위해서 CWinApp 클래스의 COleControl 클래스를 상속받은 CRClientOCXCtrl 클래스를 사용해서 메소드를 외부에 노출시켜서 본 시스템에 적용시킨다.



(그림 7) CRClientOCXApp 클래스 다이어그램

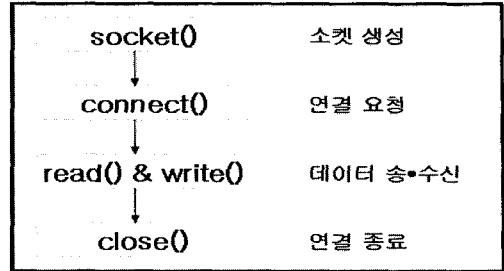
3.3.2 TCP/IP 클라이언트 소켓 설계

본 시스템에서는 서로 멀리 떨어져 있는 호스트들이 서로 데이터를 주고 받을 수 있도록 소프트웨어 차원에서 호스트들간에 연결을 해주는 추상적인 장치 역할을 하는 소켓을 사용한다.

소켓에서는 클라이언트/서버라고 불리는 통신 당사자가 있다. 클라이언트 프로그램은 통신을 시작하고, 서버 프로그램은 수동적으로 기다리면서 자신에게 접근한 클라이언트들의 요구에 응답한다.

본 절에서는 데이터 연동을 위한 클라이언트 소켓을 설계한다.

클라이언트는 다음 (그림 8)과 같이 4단계를 수행한다.



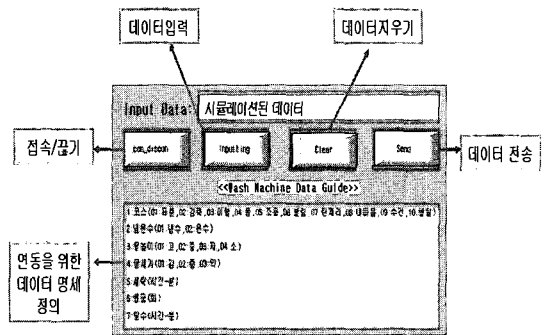
(그림 8) 클라이언트 함수 호출

- ① socket()을 사용하여 TCP 소켓을 생성
- ② connect()를 사용하여 서버와 연결을 설정
- ③ send()와 recv()를 사용하여 통신
- ④ close로 연결을 종료

3.3.3 데이터 처리 인터페이스 설계

데이터 처리 인터페이스가 하는 역할을 보면 다음과 같다.

- ① 시물레이션 된 데이터를 연동 인터페이스 모듈로 보내기 위해 서버로 접속
- ② 데이터 입력
- ③ 데이터 전송



(그림 9) 데이터 처리 인터페이스

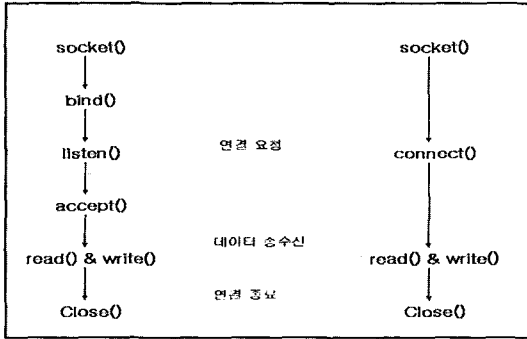
3.4 데이터 연동 인터페이스 설계

3.4.1 TCP/IP 서버 소켓 설계

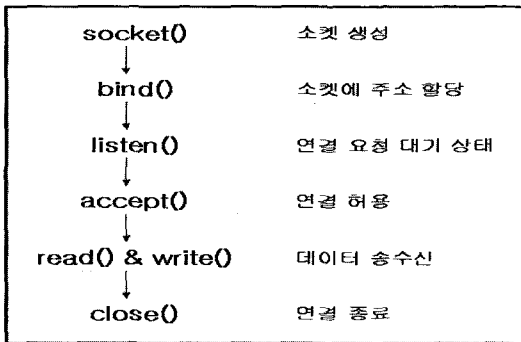
본 절에서는 가상 프로토타이핑 모듈에서 넘어

오는 데이터를 소켓 통신을 이용해서 받기 위한 서버를 설계한다.

소켓 통신에서의 클라이언트와 서버와의 함수 호출에 대해서 아래 (그림 10)과 (그림 11)과 같은 관계를 갖는다.



(그림 10) 클라이언트/서버 함수 호출



(그림 11) 서버 함수 호출

서버 통신을 위해서는 다음과 같은 단계가 필요하다.

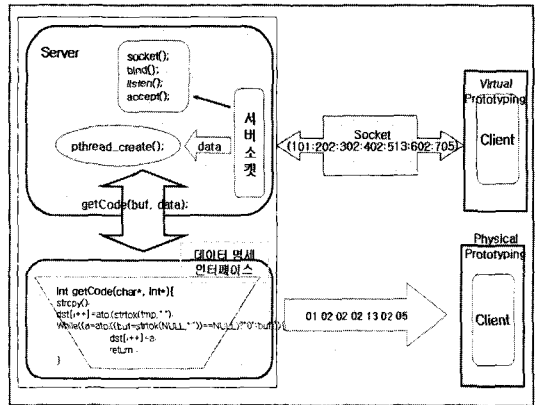
- ① socket()을 사용하여 TCP 소켓 생성
- ② bind()로 소켓에 한 포트번호를 부여
- ③ 시스템에 listen()을 그 포트로 연결 요구를 허락하도록 알림
- ④ 다음과 같이 과정을 반복
 - 각 클라이언트 연결에 대해 새로운 소켓을 얻기 위해 accept() 호출

- read()와 write()를 사용 그 새로운 소켓을 통해 클라이언트와 통신
- close()를 사용하여 해당 클라이언트와의 연결 종료

3.4.2 데이터 명세 인터페이스 설계

가상 프로토타입 환경에서 시뮬레이션을 통한 데이터를 TCP/IP 소켓 통신을 통해서 전송하면, 서버 소켓에서는 받은 데이터를 실물 프로토타입 환경의 시뮬레이션을 위한 데이터로 변환을 하기 위한 인터페이스 설계가 필요하다.

본 절에서는 이런 데이터 명세를 위한 인터페이스를 다음 (그림 12)와 같이 설계한다.



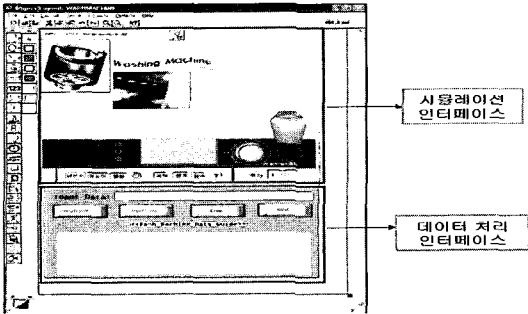
(그림 12) 데이터 명세 인터페이스

4. 구 현

본 장에서는 가정용 전기 세탁기를 타겟 모델로 실물과 가상 프로토타입을 만든다. 이미 설계된 가상 프로토타입 개발 도구인 RapidPLUS를 이용하여 가상 프로토타입 시뮬레이션, ESPS를 이용한 실물 프로토타입 시뮬레이션 그리고 실물 프로토타입과 가상 프로토타입을 연동하기 위한 데이터 인터페이스 모듈들을 구현한다.

4.1 가상 프로토타이핑 인터페이스 모듈 구현

가상 프로토타이핑 개발 도구를 이용해서 가정용 전기 세탁기를 시뮬레이션 하기 위해서 아래 (그림 13)과 같이 구현된 사용자 인터페이스를 구현한다.

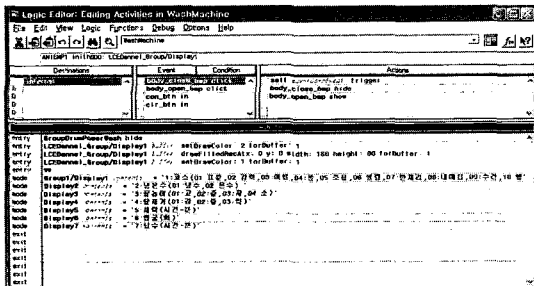


(그림 13) 가상 프로토타이핑 인터페이스

사용자 인터페이스에서는 가상 프로토타이핑 시뮬레이션을 위한 시뮬레이션 인터페이스와 시뮬레이션 인터페이스에서 시뮬레이션된 데이터를 실패 프로토타이핑 환경에서 시뮬레이션하기 위해서 전송하기 위한 데이터 처리 인터페이스 부분으로 나눠서 구현했다.

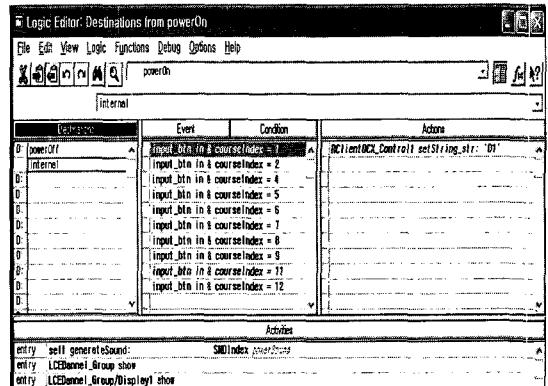
다음은 데이터 처리 인터페이스에서 일어나는 각종 이벤트나 액션 등의 기능을 Logic Editor와 Logic Palette를 이용해서 구현했다.

Object Layout에서 설계된 여러 Object들의 속성이나 기능을 적용하기 쉽도록 제공한 것이다. 다음 (그림 14)는 데이터 처리 인터페이스의 연동을 위한 데이터 명세 정의를 구현한 것이다.

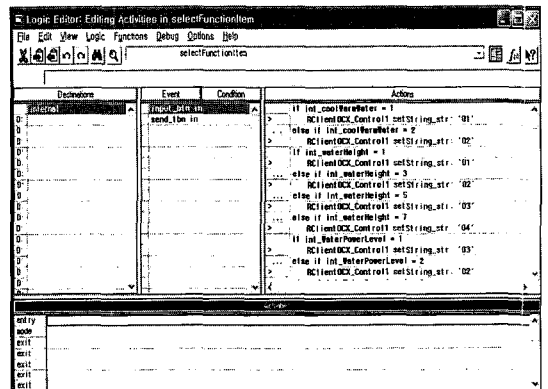


(그림 14) 연동을 위한 데이터 명세 정의

또 아래 (그림 15)는 시뮬레이션 인터페이스에서 코스를 선택했을 때의 데이터 처리를 위해 구현을 보여준다. (그림 16)은 입력 버튼 이벤트 발생시 데이터 처리 인터페이스로 시뮬레이션된 데이터 입력을 보여준다.



(그림 15) 코스 모듈 구현

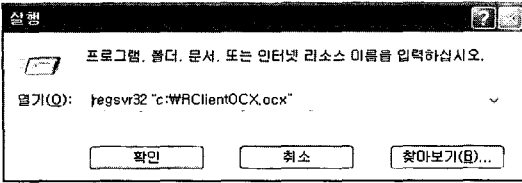


(그림 16) 데이터 처리 모듈 구현

4.2 ActiveX 컴포넌트 모듈 구현

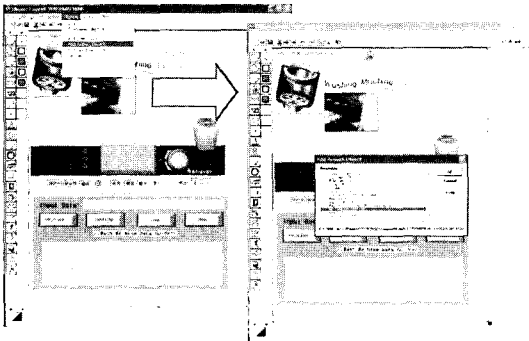
본 절에서는 앞 장에서 언급한 ActiveX 컴포넌트 설계를 기반으로 구현하였다. 구현은 Visual C++를 이용해서 구현하였고, 구현된 파일을 가상 프로토타이핑 개발 도구에서 사용하기 위해서는 다음과 같은 절차로 적용시킨다.

- ① ActiveX 컴포넌트를 사용하기 위해서 레지스터에 등록



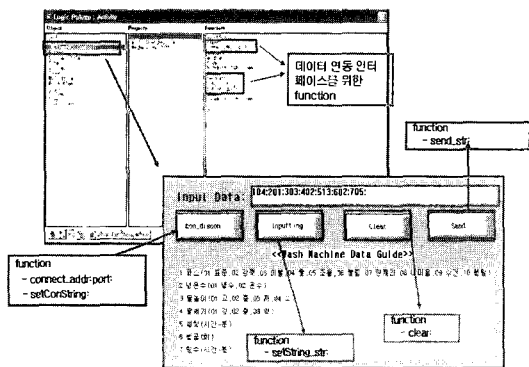
(그림 17) OCX 레지스터 등록

- ② 가상 프로토타이핑 개발 도구에 ActiveX 컴포넌트 추가



(그림 18) RapidPLUS를 이용한 ActiveX 컴포넌트 추가

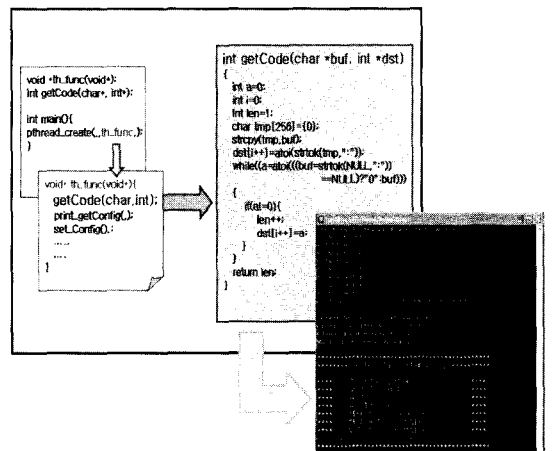
- ③ ActiveX 컴포넌트 기술을 이용한 데이터 처리 인터페이스



(그림 19) 데이터 처리 인터페이스

4.3 데이터 연동 인터페이스 모듈 구현

소켓 통신을 통한 가정용 전기 세탁기 가상 프로토타이핑 시뮬레이션 데이터를 실물 프로토타이핑 시뮬레이션의 데이터로 사용하기 위해서는 데이터를 변환시켜주는 인터페이스가 필요하다. 그래서 본 절에서는 아래 (그림 20)과 같은 형태로 데이터를 변환시켜주는 데이터 연동 인터페이스를 구현하였다.



(그림 20) 데이터 연동 인터페이스

5. 구현결과 평가

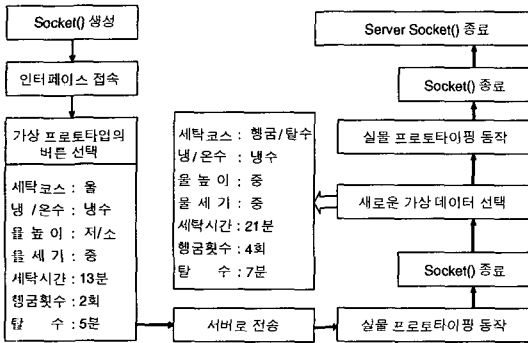
본 논문에서 구현된 통합 프로토타이핑 시스템의 인터페이스 검증을 위하여 가전 시스템의 세탁기 시뮬레이션을 프로토타이퍼(Prototyper)로 본다.

구현 결과 평가를 위하여 정보 가전 시뮬레이션을 위한 환경을 마련하고, 현실감 있는 그래픽 기반의 시뮬레이션을 수행한다.

5.1 평가를 위한 시뮬레이션 시나리오

구현 결과의 평가를 위하여 정보 가전의 적용

사례로 세탁기 모델의 시나리오를 구성한다. (그림 21)은 구현 결과 평가를 위한 통합 프로토타이핑 시스템을 위한 시뮬레이션 시나리오를 도식화하였다.

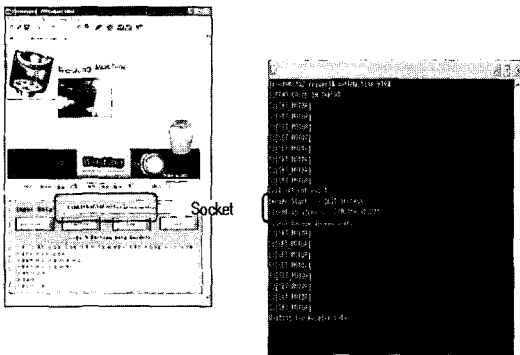


(그림 21) 구현 결과 평가를 위한 시나리오

5.2 구현결과 평가 방법

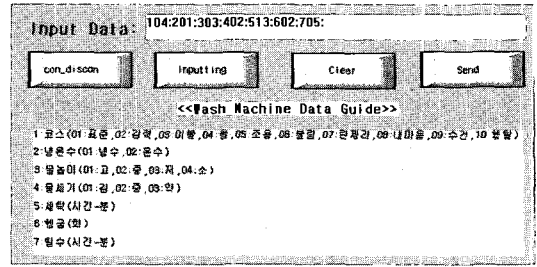
가상 프로토타이핑에서 실행 결과를 인터페이스를 통해서 실물 프로토타이핑에서 동작하는 것을 확인한다.

- ① 서버와 가상 프로토타입 개발 도구는 소켓 통신을 통하여 직접 연결된다.



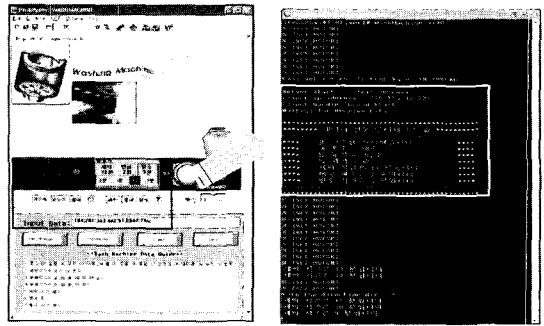
(그림 22) 가상 프로토타입 소켓 연결

- ② 연결후 가상 프로토타입의 데이터 입력 대기 상태에 있음을 (그림 23)을 통하여 확인할 수 있다.



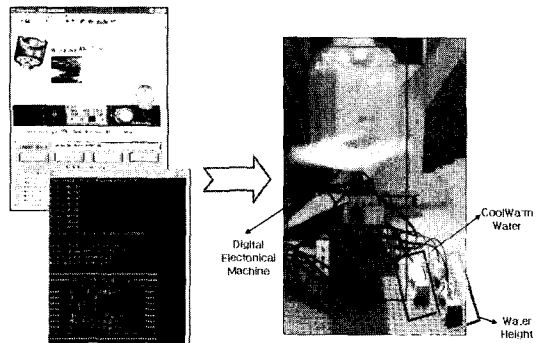
(그림 23) 입력 데이터

- ③ 데이터 처리 인터페이스에 입력된 데이터를 데이터 연동 인터페이스를 통하여 실물 프로토타이핑으로 전송한다.



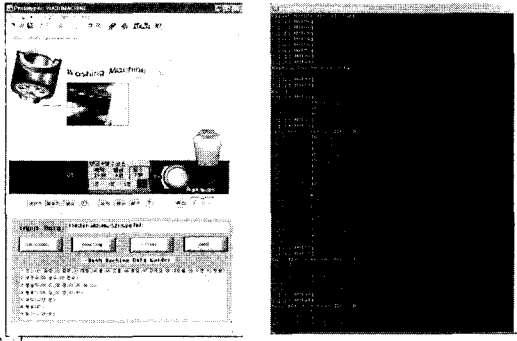
(그림 24) 입력 데이터 전송

- ④ 데이터 연동 인터페이스 모듈을 통한 데이터를 입력으로 한 실물 프로토타이핑 시뮬레이션이 실행된다.



(그림 25) 데이터 연동을 통한 통합 시뮬레이션

- ⑤ 세탁 코스 변경 시 소켓이 다시 open() 되면서 새로운 세탁 코스가 실행된다.

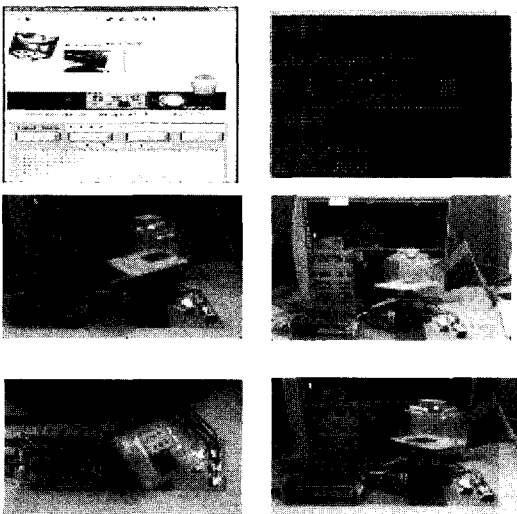


(그림 26) 새로운 설정 후 세탁 결과

5.3 시뮬레이션을 통한 평가

통합 프로토타이핑 구현 결과를 평가하기 위하여, 가상 프로토타이핑의 데이터 값이 실물 프로토타이핑에 전송되어 동작하는 것을 확인 할 수 있으며, 논문에서 구현하고자 하는 목표 시스템의 성공적인 구현 결과를 알 수 있다.

(그림 27)은 시뮬레이션 시나리오에 따라 시뮬레이션 되는 그림이다.



(그림 27) 시뮬레이션 결과

구현 결과 시뮬레이션을 통하여 가상 프로토타이핑의 세탁기 버튼들을 시뮬레이션 하였으며 똑같이 실물 프로토타이핑에서도 동작하는 것을 확인하였다.

6. 결 론

본 논문에서는 가정용 전기 세탁기를 모델로 가상 프로토타이핑 개발 도구인 RapidPLUS를 이용한 시뮬레이션과 가상 프로토타이핑에서 시뮬레이션된 데이터를 레고 블록을 이용한 실물 프로토타이핑 시뮬레이션을 하기위한 통합 프로토타이핑 시스템을 위한 인터페이스를 설계 및 구현하였다.

기존에는 가상 프로토타이핑 개발 도구를 이용해서는 가상 프로토타이핑 시뮬레이션을 하고 실물 프로토타이핑으로는 실물 모형을 시뮬레이션을 하였다. 그럼으로써 가상 프로토타이핑으로는 다양한 컴포넌트와 라이브러리로 세세한 부분까지의 가상 시뮬레이션이 가능하였지만 실제 타겟 모델의 임베디드 운영체제나 하드웨어, 소프트웨어적인 환경까지의 지원이 힘들었다. 또한, 실물 프로토타이핑 환경에서는 공장 자동화나 산업분야의 실물 타겟에 적용하기는 적합하지만 정보가전분야의 소형 제품의 다양한 컴포넌트와 라이브러리를 지원하지 못하는 부분이 있었다.

하지만 본 논문에서는 가상과 실물 프로토타이핑을 동시에 시뮬레이션 할 수 있는 환경을 지원하는 통합 프로토타이핑 시스템을 위한 인터페이스를 설계 및 구현함으로써 가상 프로토타이핑에서의 임베디드 시스템 구현 환경과 실물 프로토타이핑의 다양한 컴포넌트와 라이브러리를 지원하지 못하는 부분을 보완하는데 목적을 두었다.

인터페이스를 설계 및 구현하기 위해서 가상 프로토타이핑 환경에서는 ActiveX 컴포넌트 기술을 이용해서 TCP/IP 소켓 통신을 할 수 있게 설계 및 구현하였고 실물 프로토타이핑 환경에서는

서버 소켓을 만들어서 가상 프로토타이핑 시뮬레이션 데이터를 받아서 데이터 연동 인터페이스를 통해서 시뮬레이션 할 수 있게 하였다.

앞으로의 과제는 좀 더 보완된 인터페이스를 설계 및 구현하고 이 데이터 연동 인터페이스를 폭 넓게 적용시키기 위해서 다른 가상 프로토타이핑 지원 개발 도구와 실물 프로토타이핑과의 연동에 대해서도 연구하는 게 과제라 할 수 있겠다.

참 고 문 헌

- [1] Hennessy and Patterson, Computer Architecture 2nd edition, 1997.
- [2] 원종혁, 한상용, “사용자 인터페이스 향상을 위한 3차원 VP 시뮬레이터 설계”, HCI2000 학술발표대회 논문집, pp. 417-422, 2000.
- [3] T. Brown(IDEO), Computer Human Interaction 2004 Closing Plenary Speech.
- [4] B. Lawson, *How Designers Think*, Architectural Press, 1997.
- [5] 김남규, 박홍진, 김영찬, “가상 프로토타이핑을 위한 Statechart 도구”, 정보과학회 추계 학술대회 논문집, 제27권, 제2호 pp. 0012-0014, 2000.
- [6] 김철웅, 한상용, 최진영, 이정아, “VIP/Sim : Statecharts에 기반을 둔 가상 프로토타이핑 시뮬레이터 설계 및 구현”, 정보처리학회 논문지 A, 제7권, 제3호, pp. 0892-0900, 2000.
- [7] 신숙경, 이종국, 김수동, “객체지향 설계에서 정형명세를 이용한 컴포넌트 설계로의 변환 기법”, 정보과학회논문지 : 소프트웨어 및 응용, 제31권, 제7호, pp. 883-900, 2004.
- [8] 김홍남, “임베디드 S/W 최신 기술 동향”, 전자공학회지, 제31권, 11호, pp. 1385-1395.
- [9] 김동건, 이건표, “제품 디자인을 위한 개체지향적 Virtual Prototyping에 관한 연구”, 대한 인간공학회, 학술대회 논문집, 제1권, pp. 70-75, 1998.
- [10] 박준기, 김소영, “Inside Secrets Visual C++6”, 삼각형프레스, 1999.
- [11] Ivor Horton, 정동원 역, “Beginning Visual C++6”, 정보문화사, 2000.
- [12] Michael J. Donahoo, Kenneth L. Calvert, 박준철 역, “TCP/IP 소켓 프로그래밍”, 사이텍 미디어, 2001.
- [13] 윤성우, “TCP/IP 소켓 프로그래밍”, 프리렉, 2003.
- [14] <http://fsmllabs.com>
- [15] <http://www.lego.com>
- [16] <http://mindstorms.lego.com>
- [17] <http://www.emultex.com>
- [18] <http://www.livecomus.com>
- [19] <http://kldp.org/KoreaDoc>
- [20] <http://www.e-sim.com>
- [21] <http://www.rtlinux.co.kr>
- [22] <http://www.rtai.org>
- [23] Alessandro Rubini, *LINUX Device Drivers*, O'Reilly, 1998.
- [24] Hassan Gomaa, *Software Design Methods for Concurrent and Real-Time Systems*, Addison Wesley, 1996.
- [25] 정기훈, 김도훈, 박성호, 강순주, “임베디드 실시간 시스템 개발 교육 과정”, 정보처리학회지, 제9권, 1호, 2002.
- [26] 강순주, 이정배, 박종진, “리눅스 기반 임베디드 시스템”, 대영사, 2004.
- [27] 김동건, 이건표, “제품 디자인을 위한 개체지향적 Virtual Prototyping에 관한 연구”, 디자인학연구, No. 18, pp. 137-150, 1996.
- [28] 이희웅, 최진영, “Virtual Prototyping 이란”, 전자공학회지, 제25권, 제2호, pp. 191-198, 1998.



이 정 배

경북대학교 전자공학과
전자계산기 전공(공학사)
경북대학교 전자공학과
전자계산기 전공(공학석사)
한양대학교 전자공학과(공학박사)
현재 신문대학교 컴퓨터정보학부
교수

관심분야 : 임베디드 시스템, 임베디드 프로토타이핑,
실시간 운영체제



김 종 일

부산외대 컴퓨터공학과(공학사)
신문대학교 대학원 전자계산학과
(이학석사)
현재 신문대학교 대학원 컴퓨터정
보학과 박사과정 재학중

관심분야 : 임베디드 시스템, 임베디드 프로토타이핑,
실시간 운영체제



이 영 란

부산외대 전자컴퓨터공학과
(공학석사)
신문대학교 컴퓨터정보학과
(이학박사)
현재 신문대학교 IT교육원
강의전담교수

관심분야 : 임베디드 시스템, SoC, 통합프로토타이핑,
실시간 시스템



김 남 동

신문대학교 컴퓨터정보학부(이학사)
현재 신문대학교 대학원 전자계산
학과 재학중

관심분야 : 임베디드 시스템, 통합
프로토타이핑, 실시간
시스템, RFID



정 영 진

신문대학교 컴퓨터정보학부
(이학사)
신문대학교 대학원 전자계산학과
(이학석사)
현재 신문대학교 대학원 컴퓨터
정보학과 박사과정 재학중

관심분야 : 임베디드 시스템, 통합프로토타이핑, 실시간
운영체제



신 현 철

원광대학교 컴퓨터공학과(공학박사)
현재 백석문화대학 컴퓨터정보
학부 교수

관심분야 : 임베디드 시스템, 정보
통신, 무선통신, 이동성
관리