객체 기반의 임베디드 실물 및 가상 프로토타입 통합 기법

김 종 일[†] · 이 정 배[††] · 양 재 수[†††] · 이 영 란[††††] · 정 영 진[†††††]
한 강 우[†††††] · 강 신 관[†††††] · 김 대 응[††††††]

요 약

가상 프로토타이핑 환경에서는 다양한 컴포넌트와 라이브러리를 제공함으로써 정밀한 단위까지의 목표 모델의 시뮬레이션이 가능하다. 그리고 실물 프로토타이핑 환경에서는 목표 모델의 임베디드 구동 환경에 가까운 시뮬레이션 환경을 제공한다. 하지만, 가상 프로토타이핑 환경에서는 실제 목표 모델의 구동 환경 요소까지의 시뮬레이션이 어렵고, 실물 프로토타이핑 환경에서는 다양하고 정밀한 제품까지의 시뮬레이션이 어려운 실정이다. 이런 가상과 실물 프로토타이핑 환경의 단점을 보완하기 위한 통합 프로토타이핑 시스템을 위한 연동 기법이 개발되었으며, 본 논문에서는 기존의 실물/가상 프로토타이핑 연동 기법을 개선해서 객체 기반의 임베디드 시스템 통합 프로토타이핑 개발 환경을 구축하였다. 이러한 새로운 임베디드 시스템 개발 환경을 구축함으로써, 재사용성, 유연성, 확장성 그리고 편리성을 가질 수 있다.

키워드: 통합 프로토타이핑, 임베디드 시스템 , 통합인터페이스

The Integrated Mechanism based on Object between Virtual Prototyping and Physical prototyping

Jong-Il Kim[†] · Jeong-Bae Lee[††] · Jae-Soo Yang[†††] · Young-Ran Lee[††††] · Young-Jin Jung[†††††]
Kang-Woo Han[†††††] · Sin-Kwan Kang[†††††] · Dae-Eung Kim[††††††]

ABSTRACT

In the virtual prototyping environment, we can simulate details of a target model using many components and libraries provided in advance on a computer. The real prototyping environment provides the test simulation environment as a real product, but we can't do a detailed simulation of a real product in the virtual prototyping environment, and we can't do various simulations in the real prototyping environment. So, we made a integrated prototyping environment for linkage of the two types of prototyping environments. Also, in this paper, we developed the integrated prototyping environment based on objects. By this kind of embedded system development environment, we can have the advantages as reuse, flexibility, scalability, and more convenient of product making.

Key Words : Integrated Prototyping, Embedded System, Integrated Interface

1. 서 론

임베디드 시스템은 안정성이 매우 중요하고 개발 난이도가 높음에도 불구하고 상품의 개발 기간을 최소화해서 시장에 내 놓아야 하는 어려움이 있다. 이러한 점들을 고려해 볼 때, 제한된 시간과 비용, 인력으로 급 변화 하는 시장의 형태와, 시장의 요구에 적시에 부응해야 하는 경영적 요구를 적절히 반영할 수 있는 수단이 필요한데, 현재 널리 쓰이고 있는 방법이 프로토타이핑 방법론이다.

프로토타이핑 방법론에는 실물 프로토타이핑 방법론과 컴퓨터 기술을 이용해서 다양한 컴포넌트와 라이브러리를 제공하는 가상 프로토타이핑 방법론이 있다. 실물 프로토타이핑은 센서, 엑추에이터와 레고 블록을 이용하여, 외부와 입출력 가능한 실제 임베디드 제품을 제작하고 테스트한다[1]. 하지만, 이렇게 제작된 실물 프로토타이핑으로는 임베디드 시스템이 적용된 정보 가전제품의 다양한 컴포넌트나 세밀한 컨트롤을 위한 라이브러리를 지원하지 못한다. 그래서 새롭게 등장한 개발 방법론이 가상 프로토타이핑 방법론이

※ 본 연구는 정보통신부 및 정보통신 연구진흥원의 대학 IT 연구센터 육성 지원 사업의 연구 결과로 수행되었음
† 준 회 원 : 선문대학교 컴퓨터정보학과 박사과정(교신저자)
†† 종신회원 : 선문대학교 컴퓨터정보학과 교수
††† 정 회 원 : 경기도 도지사 정보화보좌관
†††† 종신회원 : 선문대학교 IT 교육원 교수
††††† 준 회 원 : 선문대학교 컴퓨터정보학과 박사과정
†††††† 정 회 원 : 선문대학교 컴퓨터정보학과 박사과정
논문접수 : 2007년 5월 1일, 심사완료 : 2007년 5월 16일

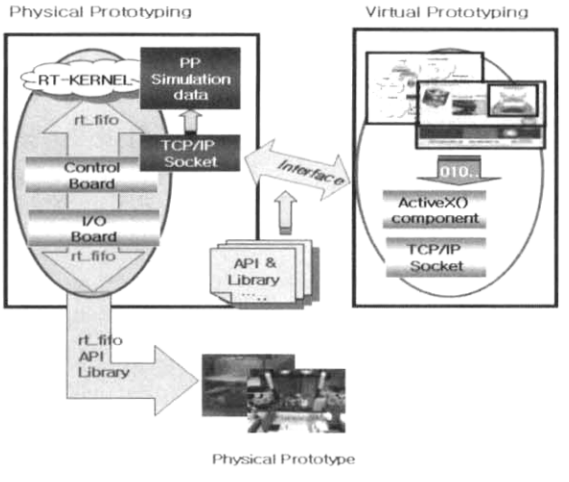
며, 가상 프로토타이핑은 임베디드 소프트웨어 개발 초기 단계에서 사용자 혹은 개발 의뢰자의 정확한 요구, 선호 및 취향 등 기능적, 비 기능적 요구들을 효과적으로 추출할 수 있도록 고안된 시스템 요구 및 제약조건 추출 방법론이다. 또한, 컴퓨터를 이용하여 다양한 컴포넌트와 라이브러리를 제공함으로 제품을 시각적인 3D로 모델링 할 수 있다. 컴퓨터 프로그램으로 프로토타입 모델을 작성하고 시연하기 때문에, 디자인 변경이 용이하고 세밀한 컨트롤을 할 수 있도록 한다[10]. 하지만, 가상 프로토타이핑은 컴퓨터 기술을 이용한 시각적인 가상 시뮬레이션이기 때문에 실제 제품을 구동시키기 위한 임베디드 시스템 환경에서의 정확한 시뮬레이션이라고 할 수 없는 불확실성이 있다.

이러한 문제점 들을 해결하기 위하여, 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 연동 기법이 개발되었다. 본 논문에서는 기존의 실물/가상 프로토타이핑 연동 기법을 개선해서 객체 기반의 임베디드 시스템 통합 프로토타이핑 개발 환경을 구축하였다. 일단 객체들이 개발, 축적되면 다른 임베디드 시스템을 만들 때에는 조립 과정만 거치면 되므로 기존의 방법에 비해 생산성을 대폭 향상 시킬 수 있다. 이러한 새로운 임베디드 시스템 개발 환경을 구축함으로써, 임베디드 시스템의 개발 기간 단축, 재사용성, 유연성, 확장성 그리고 편리성을 가질 수 있다.

2. 관련연구

프로토타입(Prototype)이란 제품 개발 과정 중 제품 생산 전 제작되어진 축소형 또는 실물 크기의 모델로, 제품 출시 전에 외형 및 기능 시험을 위한 테스트용 모델을 제작하는 것으로 제품 생산 과정의 일부로 볼 수 있다. 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[3][4][5].

임베디드시스템의 특성상 개발자들 간의 공동작업에서 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 즉시에 해결할 수 있다[2]. 그러나 실물 프로토타이핑이 실제 제품에 적용이 불가능하다고 가정 했을 때, 이와 유사한 환경에서 시뮬레이션 할 수 있는 프로토타입을 만들어 테스트를 할 수 있는 여건 조성이 필요하게 되었다. 그래서 기존의 물리적인 프로토타입을 이용하는 방법과는 달리 컴퓨터상에 가상으로 만들어진



(그림 1) 가상/실물 통합 프로토타이핑 시스템

프로토타입으로 실 제품과 같은 외형을 제공하고, 시스템 요구 사항이 복잡한 하드웨어 시뮬레이션을 소프트웨어로 가능하게 하는 가상 프로토타이핑(Virtual Prototyping)이 제공되었다[10]. 이는 시제품의 검증을 위한 실물 프로토타이핑 방법보다 전체 개발비용 절감과 생산 주기 등의 문제에 편리함을 제공하나, 실물 프로토타이핑에 비해 불확실한 면이 있다.

그래서 실물과 가상 프로토타이핑의 부족한 부분을 상호 보완 할 수 있는 통합 프로토타이핑 시스템을 위한 연동 기법이 개발되었다.

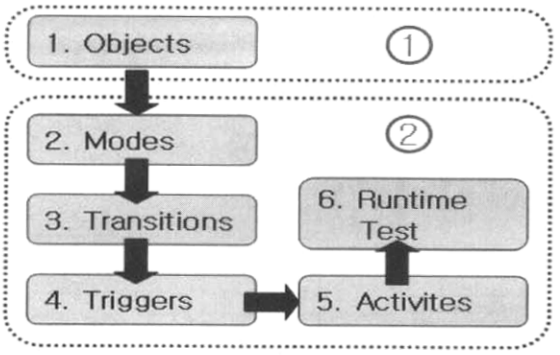
가상/실물 프로토타이핑 연동 기법을 위해 가상 프로토타이핑 환경에서는 다양한 컴포넌트 객체들을 지원하고 객체들을 위한 다양한 이벤트나 속성, 기능들을 지원하는 개발 도구인 RapidPLUS를 이용하여 시뮬레이션 인터페이스와 데이터 처리 인터페이스가 설계 및 구현되었다[6].

3. 시스템 설계

본 장의 1절과 2절에서는 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 3절에서는 정의된 데이터를 ActiveX 컴포넌트를 이용해서 가상 프로토타이핑 개발 도구에서 TCP/IP Socket 통신을 이용해서 데이터 연동 인터페이스로의 데이터 전송을 처리하는 인터페이스를 설계한다.

마지막으로 필요한 객체 모듈들을 만들어 기존의 통합 프로토타이핑 기법에 객체를 적용, 객체 기반의 개발환경을 통해서 가상 프로토타이핑 환경에서의 데이터를 실물 프로토타이핑 환경에서의 시뮬레이션을 위한 데이터로 사용하기 위한 데이터 연동 인터페이스를 설계한다.



(그림 2) 가상 프로토타이핑 설계 단계

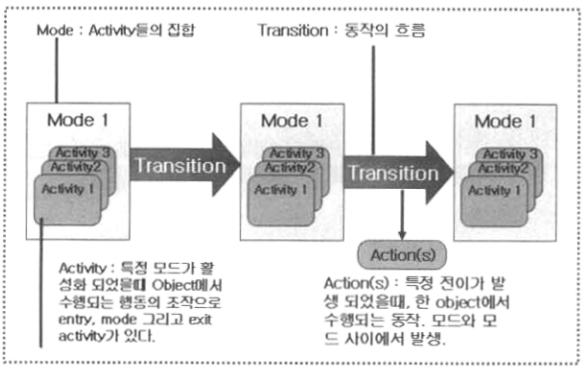
3.1 시뮬레이션 인터페이스 설계

가상 프로토타이핑 개발 도구인 RapidPLUS를 이용한 사용자 인터페이스를 설계하는 데는 아래 (그림 2)와 같이 시뮬레이션을 하는데 있어서 가시적인 객체 부분과 객체들의 이벤트나 기능, 행동을 정의하는 부분(Modes, Transitions, Triggers, Activities)으로 나눠서 설계한다.

3.1.1 가상프로토타이핑의 기능 정의

본 절에서는 RapidPLUS 제공된 객체들을 기반으로 객체들 간의 기능을 정의한다. 기능 정의는 다음 (그림 3)과 같은 논리 흐름을 따른다.

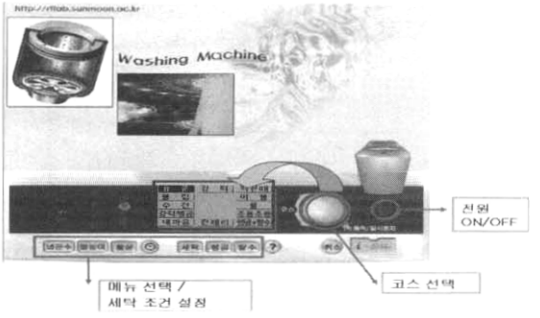
- Mode : activities의 집합
- Activity : 한 모드로부터 빠져나가거나 한 모드에 있거나, 한 모드로 진입 했을 때 일어나는 RapidPLUS 애플리케이션 동작
- Transition : 한 모드에서 다른 모드로 가는 경로
- Action : 모드의 전이시 일어나는 응용프로그램 동작



(그림 3) 기능 정의를 위한 논리 흐름

3.1.2 객체 기반 인터페이스 설계

시뮬레이션 인터페이스는 다음과 같은 역할을 하는 객체들을 가상 프로토타이핑 툴(RapidPlus)의 객체 디자인 창에 (그림 4)와 같이 구성한다.



(그림 4) 시뮬레이션 인터페이스

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

3.2 데이터 연동 명세 정의

가상 프로토타이핑 환경에서 시뮬레이션된 데이터를 실물 프로토타이핑 환경에서 시뮬레이션 하기 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]들과 같다.

〈표 1〉 코스 데이터 명세

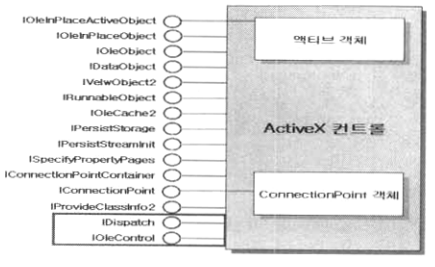
courseSlectStage			
setString	Label	courseIndex	name
01	표준	1	countStandard
02	강력	2	countPower
03	이불	4	countQuilt
04	울	5	countWool
05	조용	6	countQuiet
06	헹탈	7	countRince_Hydration
07	란제리	8	countRangeri
08	내마음	9	countMySelf
09	수건	11	countTowel
10	불림	12	countSoak1

〈표 2〉 물 높이 데이터 명세

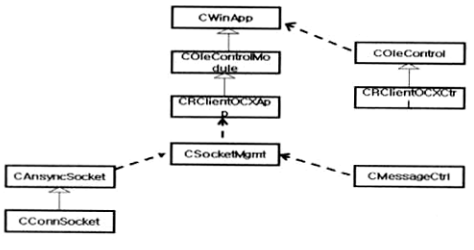
waterHeight			
setString	Label	int_waterHeight	name
01	고	1	level_1
02	중	3	level_3
03	저	5	level_5
04	소	7	level_7

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. 그리고 실물 프로토타이핑과 연동을 위한 객체를 설계한다[7].



(그림 5) ActiveX 컨트롤



(그림 6) CRClientOCXApp 클래스다이어그램

3.3.1 ActiveX 컴포넌트 설계

본 절에서는 ActiveX 컨트롤을 사용하여 가상 프로토타이핑과 실물 프로토타이핑의 통신 인터페이스를 만든다. ActiveX 컨트롤은 여러 개의 인터페이스를 제공하며, 본 시스템에서는 IDispatch와 IOlecontrol 인터페이스를 주로 사용한다.

본 시스템에서는 아래 (그림 6)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 비 동기 소켓 클래스인 CAnsycSocket, 데이터 송/수신 처리를 위한 CConnSocket 클래스등을 사용했다.

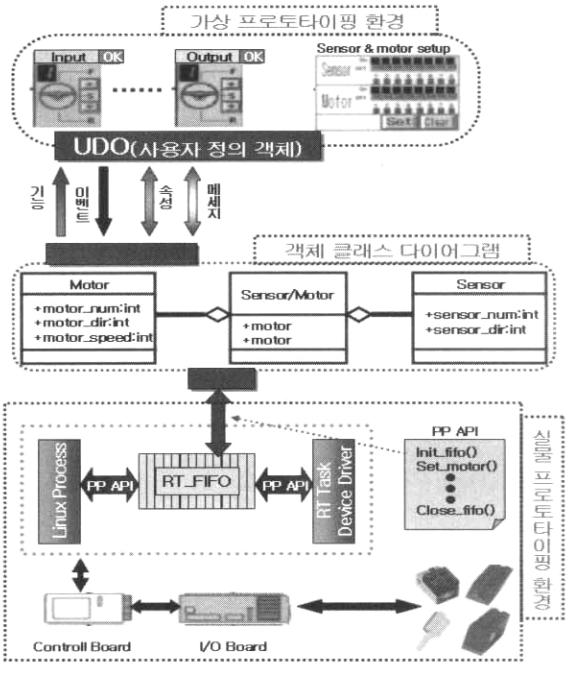
또한, 가상 프로토타이핑 환경에서 데이터 처리 인터페이스의 버튼 객체들의 이벤트에 적용하기 위해서 CWinApp 클래스의 COleControl 클래스를 상속받은 CRClientOCXCtrl 클래스를 사용해서 메소드를 외부에 노출시켜서 본 시스템에 적용시킨다.

3.3.2 객체 모듈 설계

(그림 1)에서 보여준 기존의 설계에서는 사용되는 모든 센서나 엑추에이터에 대해서 개별적으로 설계를 해야 했다. 하지만, 이 절에서는 가상 프로토타이핑과 실물 프로토타이핑에 연결되는 모든 센서와 엑추에이터에 대해 연결 가능한 객체 모듈을 설계한다. 아래 (그림 7)은 두 프로토타이핑 사이의 객체 모듈 설계를 보여준다.

엑추에이터(모터)와 센서를 위한 객체 모듈들은 실제 가상 프로토타이핑과 실물 프로토타이핑의 요소들을 사용해서 설계되었으며, 엑추에이터, 센서 그리고 센서/액추에이터로 구성이 된다. 각 객체 모듈의 기능은 아래와 같다.

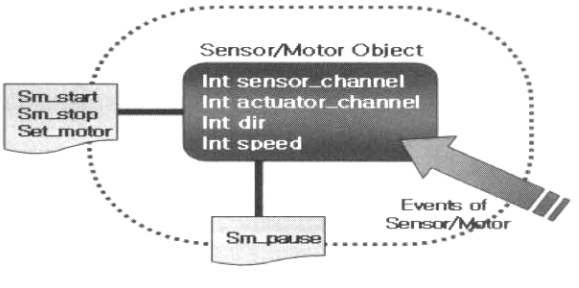
- 엑추에이터 모듈 : 실물 프로토타이핑의 엑추에이터와 통신을 위한 채널과 방향 조절 그리고 속도에 관한 속성을 가진다.
- 센서 모듈 : 센서와 통신을 위한 채널과 입력 처리에 관한 속성을 가진다.
- 센서/엑추에이터 모듈 : 센서와 액추에이터의 속성을 모두 가진다.



(그림 7) 가상/실물 프로토타이핑 사이의 객체 모듈 설계

아래 (그림 8)은 센서/엑추에이터 객체의 구조를 보여준다. 이 객체는 센서와 엑추에이터의 상태를 바꿀 수 있는 속성과 가상 프로토타이핑과 실물 프로토타이핑이 통신을 할 수 있는 채널로 구성이 되어있다. 또한 외부로부터 들어오는 센서나 엑추에이터의 이벤트를 처리할 수 있도록 구성이 되어있다.

객체 모듈들이 가상 프로토타이핑과 실물 프로토타이핑을 연결해서 통신을 하기 위해서는 데이터 명세가 필요하다. 이러한 데이터 명세는 <표 3>에서 보는 바와 같다.



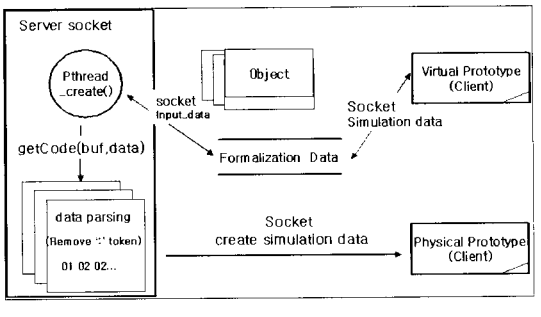
(그림 8) 센서/엑츄에이터 모듈의 구조

〈표 3〉 센서/엑추에이터 모듈의 데이터 명세

종류	요소	정의
객체	Actuator/ Sensor	Select Actuator/Sensor object
기능	Initialize	Initialize properties of Actuator and Sensor
	Start/Stop	Set up Start/Stop Function
	Order	Set up Auto/Manual Mode
속성	Num	Set up Actuator/Sensor's Channel
	Dir	Set up Actuator/Sensor's Direction
	Speed -	Set up Actuator's Speed

3.4 데이터 연동 인터페이스 설계

가상 프로토타이핑 환경에서 시뮬레이션을 통한 데이터를 TCP/IP 소켓 통신을 통해서 전송하면, 서버 소켓에서는 받은 데이터를 실물 프로토타이핑 환경의 시뮬레이션을 위한 데이터로 변환을 하기 위한 인터페이스 설계가 필요하다. 기존의 설계와는 달리 객체 기반의 설계이므로 아래 (그림 9)와 같이 데이터 명세를 위한 인터페이스를 쉽게 설계가능하다.



(그림 9) 데이터 명세를 위한 인터페이스

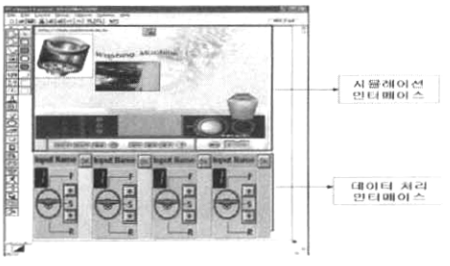
4. 구현

본 장에서는 가정용 전기 세탁기를 목표 모델로 실물과 가상 프로토타입을 만든다. 이미 설계된 가상 프로토타이핑 개발 도구인 RapidPLUS를 이용하여 가상 프로토타이핑 시뮬레이션, 레고 블록과 센서, 엑추에이터들로 이루어진 ESPS(Embedded System Prototyping Suit)를 이용한 실물 프로토타이핑 시뮬레이션을 만든다. 그리고 객체 기반의 실물 프로토타이핑과 가상 프로토타이핑을 연동하기 위한 데이터 인터페이스 모듈들을 구현한다[8][9][10].

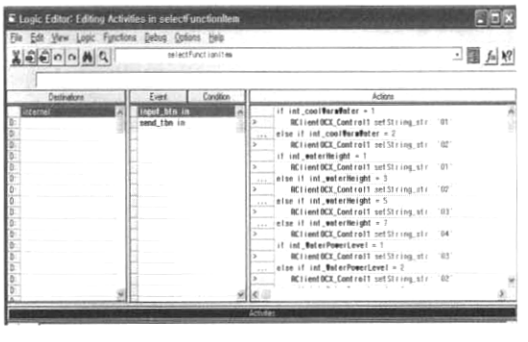
4.1 가상 프로토타이핑 인터페이스 모듈 구현

가상 프로토타이핑 개발 도구를 이용해서 가정용 전기 세탁기를 시뮬레이션 하기 위해서 아래 (그림 10)과 같이 구현된 사용자 인터페이스를 구현한다. 미리 정의 해 둔 센서와 엑추에이트 객체를 재 정의하여 쉽게 구현가능하다.

사용자 인터페이스에서는 가상 프로토타이핑 시뮬레이션을 위한 시뮬레이션 인터페이스와 시뮬레이션 인터페이스에서 시뮬레이션된 데이터를 실물 프로토타이핑 환경에서 시



(그림 10) 가상 프로토타이핑 인터페이스



(그림 11) 데이터 처리 모듈 구현

뮬레이션하기 위해서 전송하기 위한 데이터 처리 인터페이스 부분으로 나눠서 구현했다.

위 (그림 11)은 데이터 처리 인터페이스에서 일어나는 각종 이벤트나 액션 등의 기능을 Logic Editor와 Logic Palette를 이용해서 구현했으며, 입력 버튼 이벤트 발생시 데이터 처리 인터페이스로 시뮬레이션된 데이터 입력을 보여준다.

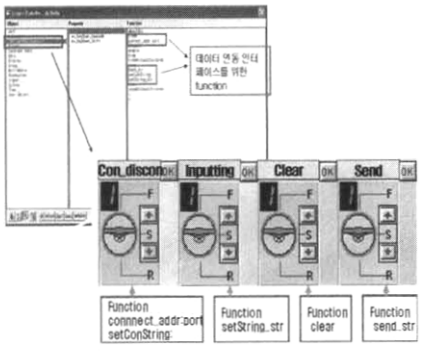
4.2 ActiveX 컴포넌트 모듈 구현

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.

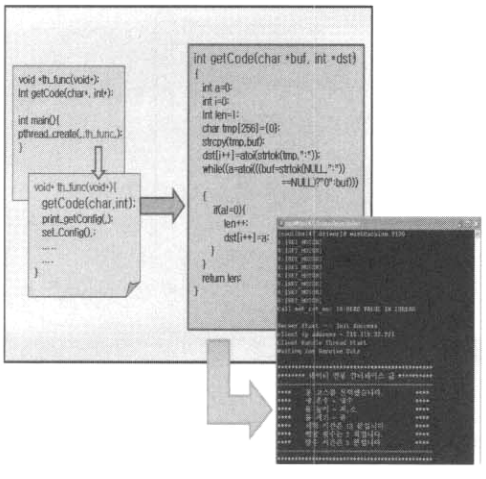
ActiveX컴포넌트를 사용하기 위해서 레지스터에 등록 (regsvr32"c:\RClientOCX.ocx")

가상 프로토타이핑 개발도구에 ActiveX컴포넌트 추가

ActiveX컴포넌트 기술을 이용한 데이터 처리 인터페이스 구현



(그림 12) 데이터 처리 인터페이스 구현



(그림 13) 데이터 연동 인터페이스

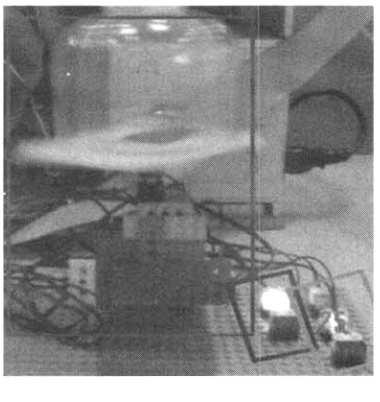
4.3 데이터 연동 인터페이스 모듈 구현

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 13)과 같은 형태로 데이터를 변환시켜주는 데이터 연동 인터페이스를 구현하였다.

가상 프로토타이핑에서 실물 프로토타이핑으로 전달되는 데이터를 getCode()에 입력하여 실물 프로토타이핑을 위한 시뮬레이션 데이터로 변환한다. 이 시뮬레이션 데이터를 소켓을 통해 스트림으로 실물 프로토타이핑에 전송하여 시뮬레이션 할 수 있게 한다.

4.4 실물 프로토타이핑 구현

가전 모듈인 세탁기의 표준, 강력, 이불 등과 같은 코스 메뉴와 세탁, 탈수, 등의 기능 설정 정보, 수온, 수위 물의 세기 등과 같은 세탁 조건들을 위한 기능을 레고 블록, 센서 그리고 모터로 이루어진 ESPS를 이용하여 아래 (그림 14)와 같이 구현한다.



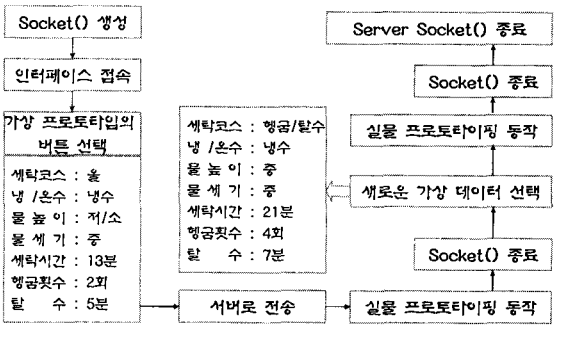
(그림 14) ESPS를 이용한 실물 프로토타이핑

5. 구현결과 평가

본 논문에서 구현된 통합 프로토타이핑 시스템의 인터페이스 검증을 위하여 가전 시스템의 세탁기를 시뮬레이션 한다. 구현 결과 평가를 위하여 정보 가전 시뮬레이션을 위한 환경을 마련하고, 현실감 있는 그래픽 기반의 시뮬레이션을 수행한다.

5.1평가를 위한 시뮬레이션 시나리오

구현 결과의 평가를 위하여 정보 가전의 적용 사례로 세탁기 모델의 시나리오를 구성한다. (그림 15)는 구현 결과 평가를 위한 통합 프로토타이핑 시스템을 위한 시뮬레이션 시나리오를 도식화하였다.



(그림 15) 구현 결과 평가를 위한 시나리오

5.2 구현결과 평가 방법

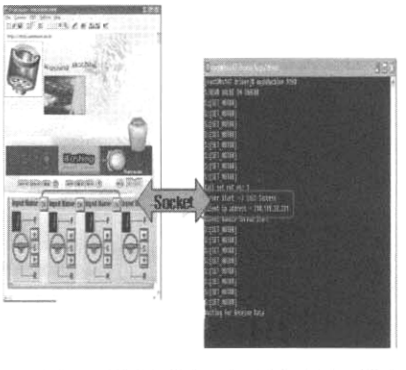
가상 프로토타이핑에서 실행 결과를 인터페이스를 통해서 실물 프로토타이핑에서 동작하는 것을 확인한다.

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

② 데이터 처리 인터페이스에 입력된 데이터를 데이터 연동 인터페이스를 통하여 실물 프로토타이핑으로 전송한다.

③ 데이터 연동 인터페이스 모듈을 통한 데이터를 입력으로 한 실물 프로토타이핑 시뮬레이션이 실행된다.

④ 세탁 코스 변경 시 소켓이 다시 open() 되면서 새로운 세탁 코스가 실행된다.



(그림 16) 가상 프로토타입 소켓 연결

5.3 시뮬레이션을 통한 평가

본 논문에서 구현된 통합 프로토타이핑 시스템의 인터페이스 검증을 위하여 정보 가전 시스템 중 세탁기를 시뮬레이션 한다.

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

아래 (그림 17)은 구현 결과 평가를 위한 통합 프로토타이핑 시스템의 실시간 시뮬레이션 과정을 도식화 하였다.

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

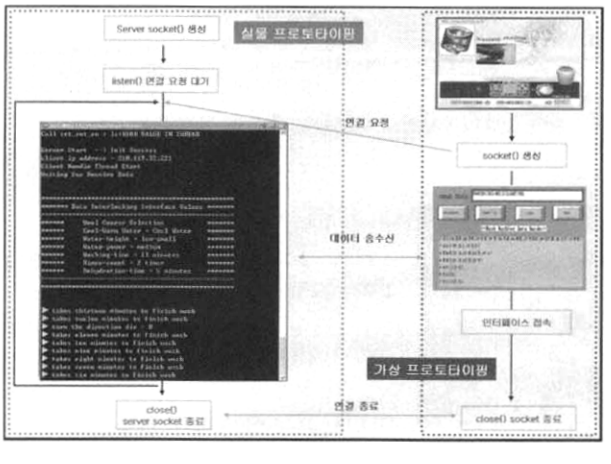
② 연결 후 가상 프로토타입의 시뮬레이션 데이터를 데이터 처리 인터페이스에 입력한다.

③ 가상 프로토타이핑 환경에서 데이터 처리 인터페이스에 입력된 데이터를 연동 인터페이스를 통하여 실물 프로토타이핑 환경으로 전송한다.

④ 데이터 변화 인터페이스 모듈을 이용해서 추출된 데이터를 실물 프로토타이의 시뮬레이션을 위한 입력 데이터로 사용한다.

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

가상 프로토타이핑에서 시뮬레이션한 결과 데이터를 연동 인터페이스를 이용해서 실물 프로토타이핑에서 시뮬레이션 되는 것을 확인하였다. 이러한 통합 프로토타이핑 구현 결과를 통하여, 가상 프로토타이핑의 세탁기 버튼들을 시뮬레이션 하였으면 똑같이 실물 프로토타이핑에서도 동작 하는 것을 확인하였다. 또한, 가상 프로토타이핑의 데이터 값이 실시간으로 실물 프로토타입에 전송되어 (그림 17)에서 보는 바와 같이 시간대별로 소켓이 오픈되어 데이터가 전달되어 가상 프로토타이핑 시스템과 실물 프로토타이핑이 동기화되어 동작되는 것을 확인 하였다. 그러므로 본 연구에서 설계 및 구현한 실물 및 가상 임베디드 통합 환경이 성공적으로 구축된 결과를 얻을 수 있다.



(그림 17) 실시간 시뮬레이션 과정

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 객체 기반으로 설계 및 구현하였다.

기존에는 가상 프로토타이핑 개발 도구를 이용해서는 가상 프로토타이핑 시뮬레이션을 하고 실물 프로토타이핑으로는 실물 모형을 시뮬레이션을 하였다. 이를 통해 가상 프로토타이핑으로는 다양한 컴포넌트와 라이브러리로

세세한 부분까지의 가상 시뮬레이션이 가능하였지만 실제 목표 모델의 임베디드 운영체제나 하드웨어, 소프트웨어적인 환경까지의 지원이 힘들었다. 또한, 실물 프로토타이핑 환경에서는 공장 자동화나 산업분야의 실물 목표 모델에 적용하기는 적합하지만 정보가전분야의 소형 제품의 다양한 컴포넌트와 라이브러리를 지원하지 못하는 부분이 있었다.

하지만 본 논문에서는 가상과 실물 프로토타이핑을 동시에 시뮬레이션 할수 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 보완하였다.

그리고 임베디드 시스템 분야에 폭넓게 적용시키기 위한 객체 기반의 통합 프로토타이핑 인터페이스를 설계 및 구현하기 위해서 현재 시스템에서 사용되고 있는 센서와 엑추에이터들을 미리 모듈로 정의 했다. 이를 통해 시스템 개발 기간 단축, 재사용성, 유연성을 가진 환경을 구축하였다.

앞으로의 과제는 좀 더 보완된 인터페이스를 설계 및 구현하고, 센서와 엑추에이터를 이용해서 외부와 입출력을 하는 임베디드 시스템의 동작을 모니터링 가능한 환경을 구축하는 것이라 할 수 있겠다.

참 고 문 헌

[1] 강순주, 이정배, 박종진, “리눅스 기반 임베디드 시스템”, 대영사, 2004.

[2] 정기훈, 채화영, 김정길, 이재선, 강순주 “실물프로토타이핑 기법을 적용한 임베디드 실시간 시스템 소프트웨어 방법론”, 전자공학회지, 제31권11호, 2004.11

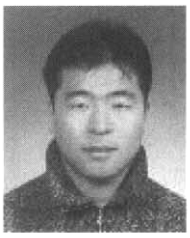
[3] 원종혁, 한상용, “사용자 인터페이스 향상을 위한 3차원 VP 시뮬레이터 설계”, HCI2000 학술발표대회 논문집, pp.417~422, 2000.

[4] Brown, T.(IDEO), Computer Human Interaction 2004 Closing Plenary Speech.

[5] Lawson, B. *How Designers Think*. Architectural Press, 1997.

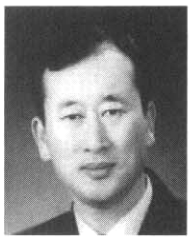
[6] (주)파이널이펙트, “RapidPlus 교재”, 2005.

[7] 윤성우, "TCP/IP 소켓 프로그래밍", 프리렉, 2003.
[8] http://www.lego.com
[9] http://mindstorms.lego.com
[10] http://www.e-sim.com



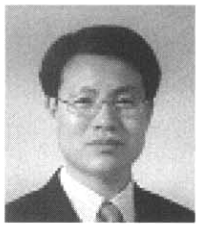
김 종 일

e-mail : rumpet0@sunmoon.ac.kr
2002년 2월 부산외국어대학교 컴퓨터공학과(공학사)
2004년 2월 선문대학교 전자계산학과(이학석사)
2004년 3월~현재 선문대학교 컴퓨터정보학과 박사과정
관심분야 : 임베디드 시스템, 임베디드 프로토타이핑



이 정 배

e-mail : jblee@sunmoon.ac.kr
1981년 2월 경북대학교 전자공학과 전산전공(공학사)
1983년 2월 경북대학교 대학권 전산전공(공학석사)
1995년 2월 한양대학교 대학원 전자공학과(공학박사)
1982년~1991년 한국전자통신연구원 선임연구원
1991년~2002년 부산외국어대학교 컴퓨터·전자공학부 부교수
1996년~1997년 U.C.Irvine 객원교수
2002년~2005년 선문대학교 컴퓨터정보학과 부교수
2006년~현 재 선문대학교 컴퓨터정보학과 교수
관심분야 : 임베디드 시스템, 실시간 시스템, 실시간통신 프로토콜

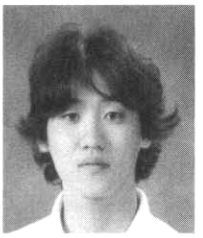


양 재 수

e-mail : jyang1@paran.com
1981년 2월 한국항공대학교통신공학과 학사 졸업
1985년 8월 건국대학교 전자공학과 석사 졸업
1993년 1월 미 NJIT전기 및 컴퓨터공학 박사 졸업
1991년 7월 서울대 MBA 수료
1981년 3월~1981년 12월 MIC 통신사무관
1982년 1월~2006년 1월 KT 인터넷시설부장, 인터넷사업국장, 전화상품팀장, 월드컵통신팀장, 수도권강북본부 사업지원국장, KT 중앙지사장
2006년 3월 광운대 산학협력단 전담교수
2007년 2월~현 재 경기도 도지사 정보화보좌관
관심분야 : 정보통신 정책 및 정보화 추진, RFID/USN, 홈 네트워킹, 차세대이동통신 등

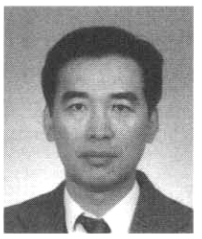
이 영 란

e-mail : yrlee@sunmoon.ac.kr
2002년 2월 부산외국어대학교컴퓨터·전자공학(공학석사)
2001년~2002년 (주)이지뉴텍 연구소 연구원
2002년~현 재 선문대학교 IT교육원 전임강사
2006년 2월 선문대학교 컴퓨터정보학과(이학박사)
관심분야 : 임베디드 시스템, SoC, 통합프로토타이핑, 실시간 시스템



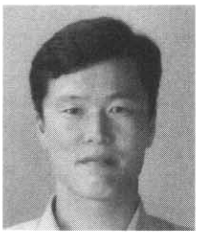
정 영 진

e-mail : yjjung.kr@gmail.com
2004년 2월 선문대학교 컴퓨터정보학부(이학사)
2006년 2월 선문대학교 전자계산학과(이학석사)
2004년~현 재 선문대학교 차세대 임베디드소프트웨어 개발환경 연구센터 연구원
2006년~현 재 선문대학교 컴퓨터정보학과 박사과정
관심분야 : 실시간 시스템, 임베디드 시스템



한 강 우

e-mail : ttico@korea.com
1979년 2월 숭실대학교 전자공학과(공학사)
2000년 2월 숭실대학교 전자공학과(공학석사)
2004년~현 재 선문대학교 컴퓨터정보학과 박사과정
관심분야 : 임베디드 시스템, 실시간 시스템



강 신 관

e-mail : kangsk@kopo.ac.kr
2002년 8월 호서대학교 전자계산 전공(이학석사)
2003년 3월~현재 선문대학교 컴퓨터정보학과 박사과정
2001년 8월~현재 한국 폴리텍IV 아산대학 멀티미디어과 근무
관심분야 : 멀티미디어, 임베디드 시스템

김 대 응

e-mail : kde6767@dreamwiz.com
1992년 2월 한밭대학교 전자공학과 졸업(공학사)
1997년 2월 한밭대학교 대학원 전자공학과(공학석사)
2007년 2월 선문대학교 대학원 컴퓨터정보학과(박사수료)
관심분야 : 임베디드 시스템, 실시간 시스템