

# 제어 및 모니터링 시스템 구현을 위한 ActiveX 기반의 애플리케이션 설계

송민규<sup>1\*</sup>

## Design of ActiveX based Application for the Implementation of Control & Monitoring System

Min-Gyu Song<sup>1\*</sup>

**요약** ActiveX는 컴포넌트 간 통신을 위하여 마이크로소프트사가 개발한 기술로서 COM 기반의 분산 애플리케이션 모델이 발전된 형태에 해당한다[1]. ActiveX는 애플리케이션 개발 과정에서 코드의 재사용 및 객체 링크를 지원하므로 개발자는 이를 통해 여러 객체를 하나의 애플리케이션으로 통합시키는 것이 가능하고 시스템 개발에 있어서 효율성을 얻을 수 있다. 뿐만 아니라 분산된 애플리케이션의 기능을 하나의 애플리케이션에서 구현할 수 있기 때문에 사용자 위주의 프로그램을 손쉽게 작성할 수 있다[2]. 그동안 Visual C++, Visual Basic 개발 환경에서 통신 애플리케이션을 개발하기 위한 도구로 사용되던 ActiveX는 프로그램 개발의 효율성으로 인하여 제어 및 계측 분야에서도 현재 널리 활용되고 있다. 본 논문에서는 이러한 ActiveX 기술을 활용하여 원격에서 시스템 제어 및 모니터링 기능을 수행하는 애플리케이션을 구현하고자 한다. 이를 위하여 웹 브라우저와 미디어 플레이어를 ActiveX 컨트롤의 형태로 구현하여 제어와 모니터링을 수행하도록 할 것이며 인스트루먼트 컨트롤을 위한 드라이버로서 VISA(Virtual Instrument Standard Architecture)를 활용할 것이다.

**Abstract** Microsoft's ActiveX corresponding to the advanced type of COM based distributed application model is made available for the use of component communication[1]. ActiveX supports reuse of code and object linking, so developers can integrate many objects into application and improve the efficiency of development. Also, Integration of separated application makes easy to develop customized program[2]. ActiveX, formerly used to develop communication application in Visual C++ or Visual Basic, has the efficiency of programming and is widely used in the M&C(Monitoring and Control) of instruments. In this paper, we will implement M&C application capable of remote operating, and besides, develop web browser and media player in the form of Activex control in order to control and monitor program remotely.

**Key words** : ActiveX, COM(Component Object Model), LabVIEW, ActiveX 컨테이너, ActiveX 컨트롤, VISA(Virtual Instrument Standard Architecture)

### 1. 서론

컴퓨터 기술 및 네트워크가 발전함에 따라 서로 다른 소프트웨어 애플리케이션 간의 통신 기술이 다양한 분야에서 폭넓게 사용되고 있다. 이는 현재 프로그램 개발 과정에서 매우 큰 역할을 하고 있는데, ActiveX의 활용을 통하여 현재 IT 업계가 얻는 개발의 효율성으로 그것이 차지하는 비중을 알 수 있다.

다른 소프트웨어의 객체를 삽입, 링크시키는 기술을 통하여 개발자는 소프트웨어를 보다 유연하게 통합할 수 있고 시스템 개발 시간을 획기적으로 단축시킬 수 있다. 이와 같은 애플리케이션 객체 간 통신을 구현할 수 있는 기술에는 여러 가지가 있는데 OLE/OCX, COM 그리고 ActiveX를 그 대표적인 예로 들 수 있다. ActiveX는 COM 기반으로 발전된 기술로서 기본적으로 윈도우 기반에서 동작하며 Microsoft사의 Visual C++, Visual Basic만으로 구현가능하다.

ActiveX 기술은 현재 단순한 소프트웨어 기술에서

<sup>1</sup>한국천문연구원 전파천문연구부

\*교신저자: 송민규(mksong@kasi.re.kr)

벗어나 제어 및 계측 솔루션에도 사용되고 있는데 작업 처리의 신속성 및 인스트루먼트 성능 면에 있어서 이전과는 비교할 수 없는 상당한 효율성을 제공하고 있다[1]. 이러한 특성에 힘입어 앞으로 시스템 제어 및 계측 분야에서 ActiveX가 차지하는 비중은 더욱 커질 것으로 전망되고 있다. 제어 및 계측 솔루션으로 전 세계적으로 널리 사용되는 소프트웨어로 National Instrument사의 LabVIEW를 들 수 있는데 이를 통하여 ActiveX 기술을 활용한 애플리케이션을 구현하는 것이 가능하다[3]. 프로그램 개발 과정에서 ActiveX 기술을 이용하게 되면 여러 이점을 얻을 수 있는데 영상 데이터, 스프레드시트 데이터 등의 처리를 위하여 윈도우 미디어 플레이어, 엑셀을 하나의 애플리케이션으로 통합시키는 것을 그 예로 들 수 있다. 따라서 프로그램의 이용측면에 있어서 더욱 사용자 친숙한 환경으로 바뀌게 되고 시스템 구축 시간을 단축시킬 수 있기 때문에 효율성 재고의 효과도 얻을 수 있다. 본 논문에서는 LaVIEW의 ActiveX 기술을 적용하여 웹 브라우저와 미디어 플레이어를 통합시킨 애플리케이션을 구현하고자 한다.

본 논문은 크게 5장으로 구성되며 1장 서론 이후는 다음의 순서에 따라 진행할 것이다. 2장에서는 ActiveX 컨테이너, ActiveX 컨트롤, ActiveX 도큐먼트 등의 ActiveX 관련 기술에 대해 살펴본 후 이를 LabVIEW상에서 구현할 수 있는 방안에 대해 논의할 것이며 3장에서 원격의 인스트루먼트 제어 및 모니터링을 위한 시스템 구성을 기술할 것이다. 그리고 4장에서 우리가 구현하고자 하는 애플리케이션을 ActiveX 기술을 기반으로 설계 것이고 5장에서 구현된 애플리케이션을 바탕으로 본 논문의 결론을 맺도록 한다.

## 2. LabVIEW 상의 ActiveX 기술

ActiveX는 Microsoft사가 개발한 애플리케이션 간 통신 기술의 하나로서 서로 다른 플랫폼에서 실행되는 애플리케이션 간의 통신을 서로 다른 프로그래밍 언어를 통하여 구현할 수 있는 기능을 제공한다[1]. 따라서 이를 통하여 개발자는 다른 애플리케이션의 코드를 재사용할 수 있고, 소프트웨어 객체를 링크시키는 것이 가능하다. COM(Component Object Model) 기술에 기반한 ActiveX는 이전기술에 해당하는 OLE(Object Linking and Embedding)의 확장판에 해당한다. ActiveX 기술을 통하여 여러 애플리케이션은 하나로 통합되는 것이 가능하기 때문에 컴포넌트는 각

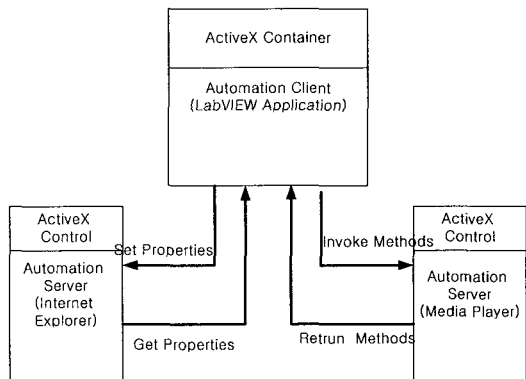
프로그램에서 재생성될 필요가 없다. ActiveX 기술은 크게 ActiveX 자동화, ActiveX 컨트롤, ActiveX 컨테이너로 분류될 수 있으며 이를 간략히 살펴보면 다음과 같다[4].

### 2.1.1 ActiveX 자동화

ActiveX/COM은 하나의 애플리케이션이 다른 애플리케이션을 제어하는 과정으로 그 기능을 나타낼 수 있다. ActiveX 자동화는 서버와 클라이언트에 해당하는 두 애플리케이션 사이의 통신을 위한 프로토콜에 해당하며 이를 통하여 독립적으로 존재하는 각 프로그램은 프로그램 정보를 공유하는 것이 가능하다. 이러한 프로그램 정보 공유를 위하여 자동화 클라이언트와 자동화 서버는 ActiveX 객체 간의 통신을 설정을 하는데 이 과정에서 서버 측의 객체는 자동화 클라이언트에서 접근 가능한 속성과 메소드를 가지게 된다. 속성은 다른 프로그램에서 설정 및 반환 가능한 객체의 특성에, 메소드는 객체에서 수행되는 함수에 해당하며 이것이 다른 프로그램으로부터 호출되어 프로그램 간 통신이 이루어지게 된다.

### 2.1.2 ActiveX 컨트롤 & 컨테이너

ActiveX 컨트롤은 ActiveX에서 가장 일반적으로 사용되는 기술로서 이는 ActiveX 컨테이너에 위치한 임베디드 컴포넌트에 해당한다. ActiveX 컨테이너로 동작하는 LabVIEW 애플리케이션에 이와 통신할 ActiveX 컨트롤을 삽입하여 이에 대한 제어 및 모니터링을 수행할 수 있다[5]. 본 논문에서는 웹 브라우저, 미디어 플레이어를 ActiveX 컨트롤로 활용하여 LabVIEW 애플리케이션을 구현하고자 하는데 먼저 이러한 관계를 다이어그램으로 나타내보면 아래와 같다.



【그림 1】 LabVIEW 상에서 ActiveX 컨테이너와 ActiveX 컨트롤의 조합

위 그림을 통해 알 수 있듯이 ActiveX 컨테이너와 ActiveX 컨트롤은 각각 네트워크에서의 클라이언트와 서버에 해당하며 각 객체는 속성과 메소드를 통하여 통신이 가능하다. 이처럼 Automation 클라이언트에 해당하는 ActiveX 컨테이너와 Automation 서버에 해당하는 ActiveX 컨트롤 간의 통신을 통하여 원격에서 제어 및 모니터링을 수행하는 애플리케이션을 구현하는 것이 가능하다.

### 3. ActiveX 기술을 사용한 LabVIEW 애플리케이션

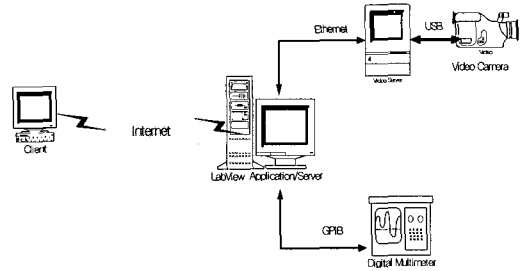
본 절에서는 원격에서 로컬 상의 인스트루먼트에 대한 제어 및 모니터링을 수행하는 애플리케이션을 설계할 것이다. 제어 및 모니터링 시스템을 구현하기 위한 소프트웨어로서 LabVIEW를 활용하였다. 애플리케이션의 전체적인 동작 메커니즘은 두 가지 통신 방식에 기반하여 이루어지며 ActiveX와 VISA(Virtual Instrument Software Architecture)가 이에 해당한다. LabVIEW와 웹브라우저, 미디어 플레이어 사이의 통신은 ActiveX를 통하여 이루어지고 LabVIEW 애플리케이션과 인스트루먼트 사이의 통신은 VISA를 사용하여 구현하였다. 본 절에서는 ActiveX를 기반으로 한 웹 브라우저, 미디어 플레이어의 설계에 대해 중점적으로 살펴볼 것이며 이를 위한 하드웨어, 소프트웨어의 구조에 대해 알아보도록 한다.

#### 3.1 원격 시스템을 위한 하드웨어 구성

원격에서 시스템을 제어 및 모니터링함에 있어서 여러 제약을 극복하기 위하여 몇 가지 특별한 요소가 수반되어야 할 것이다. 그 중에서 반드시 필요한 것으로 시스템과의 원활한 통신, 실험 진행 과정에 대한 영상 이미지 전송을 생각해 볼 수 있는데 ActiveX 기술을 통하여 이러한 요소들을 하나의 애플리케이션으로 통합하는 것이 가능하다. 본 논문에서는 원격 제어 및 모니터링을 위한 인스트루먼트로서 멀티미터를 사용할 것이며 로컬상의 실험 진행 과정을 지켜볼 수 있도록 PC 카메라를 LabVIEW 애플리케이션 서버에 설치하였다.

이에 대한 전체적인 시스템 구성을 먼저 살펴보면 다음과 같다. 먼저 멀티미터, PC 카메라는 시리얼 인터페이스로 서버에 연결되며 각각 VISA, USB 방식을 통하여 서버와 통신한다. 원격의 클라이언트는 네트워크를 통하여 서버에서 수신된 데이터 및 영상 이미지

에 접근하는 것이 가능하다. 이를 위한 대략적 시스템 구성을 도시해보면 아래와 같다.

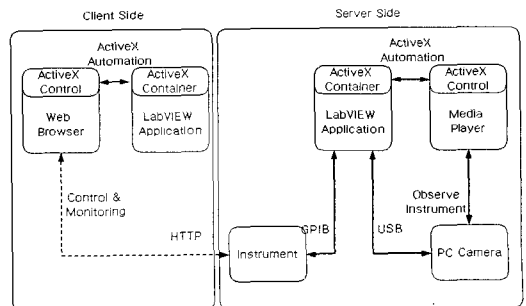


[그림 2] 원격 시스템의 하드웨어 구성

시스템은 클라이언트-서버 모델과 유사한 형태로 구성된다. 멀티미터, PC 카메라와 연결되는 컴퓨터는 서버로 동작하며 이는 멀티미터와 PC 카메라로부터 각각 수치 데이터와 영상 이미지를 획득한다. 반면, 네트워크 상의 컴퓨터는 원격에서 인스트루먼트에 대한 제어 및 모니터링을 수행하는 클라이언트에 해당한다.

#### 3.2 원격 시스템을 위한 소프트웨어 메커니즘

원격의 클라이언트가 서버에 접속하여 인스트루먼트에 대한 제어 및 모니터링을 성공적으로 수행하기 위해서는 이를 지원할 수 있는 적절한 애플리케이션이 구현되어야 할 것이다. 본 논문에서는 클라이언트와 서버 애플리케이션을 ActiveX 기술을 기반으로 LabVIEW로 작성하였다. 클라이언트는 원격의 서버에 접속하여 제어 및 모니터링을 수행하기 때문에 웹 브라우저가 구현되어야 하며 서버는 PC 카메라를 통하여 수신된 작업 영상을 나타낼 수 있도록 미디어 플레이어를 구현하고자 한다. 이러한 시스템 모델의 구현을 통하여 원격 제어 및 모니터링은 구현 가능하다. 각 시스템에서의 애플리케이션 구현에 대해 기술해보면 다음과 같다.



[그림 3] ActiveX 기반의 동작 메커니즘

먼저 서버 측에서 멀티미터에 대한 제어 및 모니터링을 구현하기 위하여 시리얼 인터페이스 상에서는 VISA를 활용하였다. 이와 동시에 LabVIEW 애플리케이션 내에 미디어 플레이어를 ActiveX 컨트롤로서 구현하였는데 이를 통하여 멀티미터의 동작 상태를 영상으로 확인하는 것이 가능하도록 하였다. 반면, 클라이언트 측에는 네트워크를 경유하여 서버 시스템에 접속할 수 있도록 웹 브라우저를 ActiveX 컨트롤로서 구현하였으며 이를 통하여 원격의 시스템에 대한 제어 및 모니터링을 수행하는 것이 가능하다. 이러한 기능을 수행하기 위한 대략적인 애플리케이션 구성을 ActiveX 기반으로 도시해보면 다음과 같다.

#### 4. 애플리케이션 구현

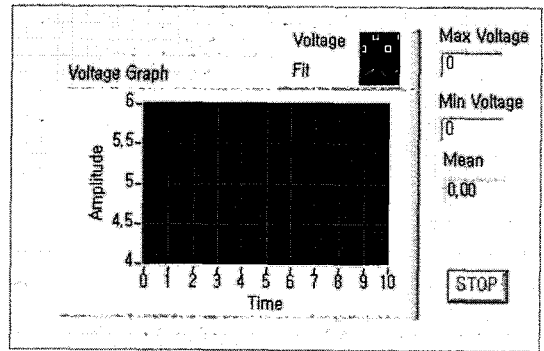
지금까지 논의된 하드웨어 및 소프트웨어 구성을 기반으로 원격에서 인스트루먼트에 대한 제어 및 모니터링을 수행하는 시스템을 실제로 구현해보고자 한다. 사용자는 네트워크가 연결된 곳 어디서든지 원격의 인스트루먼트에 대한 제어 및 모니터링을 수행할 수 있어야 하며 이를 위한 프로토콜로서 VISA, HTTP 등이 활용되어질 것이다.

M&C(Monitor & Control) 시스템에 ActiveX기술을 활용하면 하나의 애플리케이션 내에 시스템 제어 및 모니터링 관련 여러 기능을 통합시키는 것이 가능함에 따라 시스템 개발이나 유지·보수의 측면에 있어 크나큰 효율을 얻을 수 있는 장점이 있다. 즉 기능에 따라 제각각 분리된 애플리케이션을 하나의 애플리케이션으로 통합시키는 것이 가능하기 때문에 시스템 설계에 있어 유연성과 확장성을 제고할 수 있는데 본 논문에서 구현하려 하는 애플리케이션이 그에 대한 예가 될 수 있을 것이라 생각한다. 본 논문에서 우리는 임의의 전압 값을 측정하는 멀티미터를 외부에서 제어하고 그 결과를 데이터와 영상으로 모니터링하는 애플리케이션을 구현하고자 한다. 측정된 전압값이 설정된 상한치, 하한치를 넘어서면 애플리케이션 상의 램프가 점등되어 알리게 되며 매초마다 수신된 전압은 그래프에 그려지게 된다. 로컬상의 장비를 외부에서 제어 및 모니터링하기 위하여 웹 브라우저를 애플리케이션 내에 구현하였다. 뿐만 아니라 실험 진행 상태를 실제로 확인할 수 있도록 실험실에 PC 카메라를 설치하였는데 이를 확인할 수 있도록 ActiveX 기술을 통하여 애플리케이션 내에 미디어 플레이어를 구현하였다.

#### 4.1 애플리케이션을 구성하는 ActiveX 컨테이너, ActiveX 컨트롤 구현

ActiveX에 기반한 제어 및 모니터링 애플리케이션은 크게 세가지 모듈로 구성되며 이를 나열해보면 다음과 같다.

- LabVIEW 애플리케이션 - ActiveX 컨테이너
- 웹 브라우저 - ActiveX 컨트롤
- 미디어 플레이어 - ActiveX 컨트롤

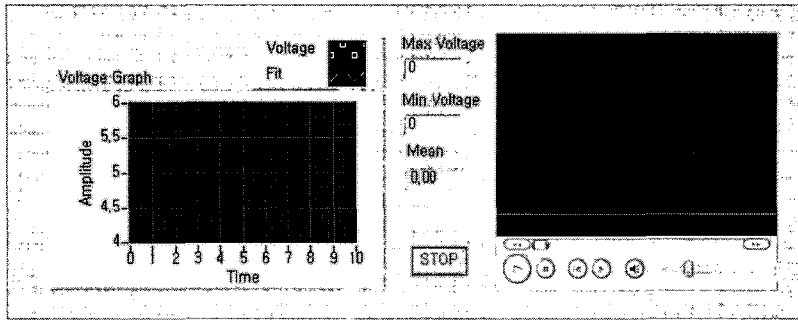


[그림 4] 서버 상의 제어 및 모니터링 프로그램

GPIB 인터페이스로 연결된 인스트루먼트를 제어 및 모니터링 하기 위한 프로그램은 서버 상에서 LabVIEW를 사용하여 구현되었으며 해당 프로그램의 GUI를 도시해보면 [그림 4]와 같다. 해당 그림에 도시된 프로그램을 통하여 인스트루먼트에 대한 제어 및 수치값에 대한 모니터링이 이루어진다. 수치값 이외에 인스트루먼트 운용 전반에 관련된 상황에 대한 거시적 모니터링을 위하여 미디어 플레이어를 ActiveX 컨트롤 형태로 LabVIEW 애플리케이션에 구현하였으며 이를 [그림 5]에 나타내었다.

ActiveX 컨트롤을 통하여 미디어 플레이어를 구현한 서버와 유사하게 클라이언트에도 원격의 서버와 통신을 수행하는 웹 브라우저가 ActiveX 컨트롤로서 구현되어야 할 것이다. 클라이언트는 이를 통하여 네트워크를 경유해 서버에 접속하게 된다. 사용자가 웹 브라우저를 통해 입력한 제어 정보는 서버 상의 LabVIEW 애플리케이션을 통하여 인스트루먼트로 전달되며 모니터링 정보는 미디어 플레이어나 웹 브라우저를 통해 사용자에게 전달된다.

이러한 프로그램 기능을 구현하기 위하여 ActiveX 자동화 서버에 해당하는 웹 브라우저, 미디어 플레이어의 기능을 LabVIEW 애플리케이션 내에서 ActiveX



[그림 5] ActiveX 컨트롤로서 구현된 미디어 플레이어

컨트롤로 지정하였으며 ActiveX 컨테이너에서는 해당 객체에 대한 속성 노드와 메소드 노드를 호출하도록 하였다.

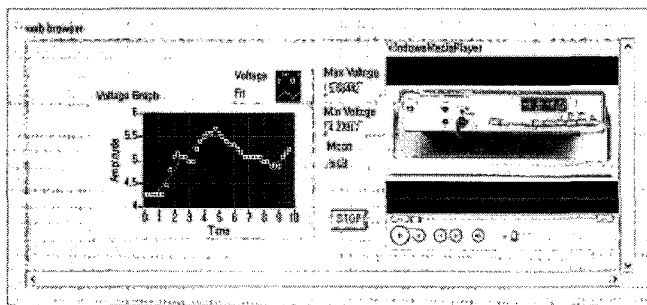
한 효과는 더욱 커질 수 있을 것이라 생각한다.

#### 4.3 ActiveX를 기반으로 통합된 사용자 인터페이스

지금까지 원격에서 인스트루먼트 제어 및 모니터링 수행을 위한 시스템 구성에 대해 살펴보았고 이를 위한 애플리케이션을 만들어 보았다. 제어와 모니터링 수행을 위하여 LabVIEW 애플리케이션 내에 ActiveX 컨트롤 형태로 웹 브라우저와 미디어 플레이어를 구현하였으며, 클라이언트에서 원격의 인스트루먼트에 대해 제어 및 모니터링 수행하는 모습을 [그림 6]을 도시하였다.

#### 4. 결론

ActiveX는 서로 다른 플랫폼에서 실행되는 애플리케이션 간의 통신을 서로 다른 프로그래밍 언어를 통하여 구현할 수 있는 일반적인 기능을 제공한다. 본 논문에서 우리는 원격에서 제어 및 모니터링을 수행하는 애플리케이션을 구현함에 있어서 ActiveX의 이러한 특징을 충실히 적용하였다. 제어 및 모니터링 시스템에 있어서 과연 ActiveX와 같은 애플리케이션 간 통신이 필요할까라고 의문을 제기하는 이들이 많을 것이라 생각한다. 하지만 제어 및 모니터링 시스템이



[그림 6] ActiveX 기술 기반의 원격 인스트루먼트 제어 및 모니터링

사용자는 웹 브라우저를 통하여 원격에서 실험실의 인스트루먼트에 대한 제어 및 모니터링을 수행 할 수 있다. 뿐만 아니라 우측의 PC 카메라를 통하여 실제 실험이 진행되는 상황을 관찰 할 수도 있다. 이처럼 제어 및 모니터링 관련 각각의 기능을 ActiveX를 통하여 하나의 애플리케이션으로 통합시킴으로서 보다 신속하고 효율적으로 프로그램을 개발할 수 있었는데, 구축하고자 하는 시스템의 기능 및 규모에 따라 이러

지원해야 하는 다양한 기능을 하나의 애플리케이션으로 통합시켜주는 ActiveX는 제어 및 모니터링 시스템을 구현함에 있어 촉매제 역할을 수행하며 이를 통하여 개발자는 프로그램 개발에 있어서 효율을 배가시킬 수 있다.

본 논문에서 우리는 웹 브라우저와 미디어 플레이어를 ActiveX 컨트롤로 구현하여 하나의 애플리케이션에 통합하였고 이를 통하여 원격 제어 및 모니터링

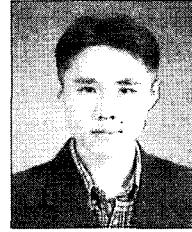
을 수행하였다. 어찌보면 서로 별개의 기능을 가지는 웹 브라우저와 미디어 플레이어이지만 이 두 애플리케이션은 ActiveX를 통하여 제어 및 모니터링 애플리케이션의 핵심 기능을 수행하는 컴포넌트로서 재탄생하였고 이에 비례하여 개발자는 보다 신속하고 효율적으로 애플리케이션을 개발할 수 있었다. 본 논문에서 구현한 ActiveX 기반의 제어 및 모니터링 시스템은 그 기능보다는 ActiveX 기술의 가능성을 예시하는 측면에 주안점을 두었기 때문에 다소 한정된 기능을 갖고 있음을 부인할 수는 없다. 하지만 제어 및 모니터링 시스템의 경우 기능 및 규모가 보다 세분화되고 방대해짐에 따라 애플리케이션 간 통신의 비중이 갈수록 큰 비중을 차지하고 있는 바, 본 논문에서 소개한 ActiveX 기반의 애플리케이션 개발의 중요성은 더욱 증가할 것으로 전망한다. 뿐만 아니라 ActiveX가 가진 모듈 형태의 프로그램 구조는 관리자의 시스템 유지보수에 있어서도 일목요연한 효율성을 제공할 것이라 생각한다.

## 참고문헌

- [1] Graf, Joseph J., "Microsoft, Exchange Messaging Using Mapi and Activex", John Wiley & Sons Inc, pp.50-51, 2005. 2.
- [2] Jason T. Roff, "ADO : ActiveX Data Objects", O'Reilly, pp.32-34, 2001. 6.
- [3] 광두영, "LabVIEW - 컴퓨터 기반의 제어와 계측 solution", Ohm사, pp.321-324, 2002. 11.
- [4] National Instrument, "LabVIEW Advanced: Performance & Communication", pp.64-69, 2003. 7.
- [5] National Instrument, "LabVIEW Advanced: Performance & Communication", pp.76-77, 2003. 7.

송민규(Min-Gyu Song)

[정회원]



- 2001년 2월 : 강원대학교 전기공학과 (공학사)
- 2003년 2월 : 강원대학교 전자공학과 (공학석사)
- 2002년 12월 ~ 현재 : 한국천문연구원 전파천문연구부 연구원

<관심분야>

BcN(Broadband Convergence Network), 분산 컴퓨팅, 리눅스 커널, 임베디드 시스템