

임베디드 OPC 기반의 통합형 HMI 시스템 개발에 관한 연구

여훈구*, 임성락**, 한광록***

A Study on Development of an Integrated HMI System based on Embedded OPC Server

Hoon-Koo Yeo *, Seong-Rak Rim **, Kwang-Rok Han ***

요 약

본 논문은 다양한 환경에서 자동화장비를 제어하고 감시하기 위하여 OPC 기반의 통합형 HMI 시스템 개발에 대하여 논한다. 다양한 프로세스 제어 분야의 통신 표준 메커니즘을 위해 제안된 OPC 표준에 따라 임베디드 OPC 서버를 구현하고, OPC 서버의 데이터를 로컬 장비와 인트라넷 클라이언트, Web 페이지 등 다양한 HMI 환경에 접속할 수 있도록 인터페이스 방법을 제안한다.

본 논문의 방법을 다양한 제어 시스템에 적용함으로써 이기종간의 연결 및 데이터 상호교환 뿐만 아니라 네트워크 상의 원격 제어 및 웹 모니터링도 가능하기 때문에 유비쿼터스 환경의 HMI 시스템 구현에 기여할 것으로 기대된다.

Abstract

In this paper, we discuss the development of an integrated HMI system based on OPC(OLE for Process Control) to control and monitor an automation equipment in the diverse environment. We implement an embedded OPC server according to the OPC standard which is proposed as the communication standard mechanism in the process control field, and design the interface module in which the output data of the OPC server can be connected to various HMI environment such as the local application, intranet client and Web page.

This paper will be contributed in the HMI system implementation for the ubiquitous environment because it is possible to make remote control on the network and Web monitoring as well as the connection and data exchange of heterogeneous equipments by applying to various control systems.

▶ Keyword : OPC, Embedded, HMI, Monitoring

• 제1저자 : 여훈구

• 접수일 : 2008. 5. 27, 심사일 : 2008. 9. 1, 심사완료일 : 2008. 9. 25.

* (주) IT 에비뉴 주임연구원 ** 호서대학교 컴퓨터공학부 교수 *** 교신저자, 호서대학교 컴퓨터공학부 교수

I. 서론

일반적으로 산업용 컴퓨터나 패널 PC에 탑재되어 하위의 PLC(Programmable Logic Controller)와 같은 산업용 장비나 컨트롤러를 통합 감시하는 역할을 하는 것을 HMI(Human Machine Interface) 소프트웨어라 하고, 하드웨어와 저장도구 등과 통합하여 HMI 시스템이라 한다[1]. 사람과 기계간의 인터페이스를 해주는 HMI 시스템을 구축하기 위해 반드시 필요한 부분이 산업용 장비나 컨트롤러와의 통신 소프트웨어 인터페이스이다[2].

산업용 장비의 제어, 측정, 유지보수, 모니터링 등 시스템 전반에 걸친 정보와 데이터의 통합 작업은 많은 비용을 유발하는 요인이 된다. 산업 표준이 없이 시스템 상호간의 데이터의 공유 및 전송을 이루어 낸다는 것을 그만큼 힘든 작업이 될 것이다.

이와 같은 필요에 따라 OPC(OLE for Process Control) 개념의 데이터 접근방법이 제안되었다. OPC는 마이크로소프트의 OLE (Object Linking and Embedding)/COM (Component Object Model) 표준에 그 기반을 두고 있으며 필드버스 네트워크, PLC, 분산 제어 시스템, 상태 모니터링 시스템 등의 이기종간 연결 및 데이터의 상호 교환이 가능한 산업 표준으로 나타났다[3].

OPC 표준은 자동화 장비의 데이터 접근과 클라이언트와 서버 사이의 통신 인터페이스 방식을 규정한 OPC-DA(OPC Data Access)이다[4]. OPC-DA 호환 클라이언트는 모든 종류의 OPC-DA 호환 서버들로부터 데이터를 읽고 쓸 수가 있다.

OPC-DA는 DCOM(Distributed COM) 기술에 바탕을 두고 산업의 표준으로 자리 잡아 가고 있으나 네트워크에 방화벽이 존재하면 이 기술은 Web 서비스를 원활하게 제공할 수 없는 단점이 있다. 이러한 문제로 인하여 OPC 협회에서는 OPC-XMLDA(OPC for XML Data Access)에 관한 기술 그룹을 만들어 DCOM 기반의 OPC-DA의 규격과 같지만 인터넷 기술과 환경에서 어플리케이션 상호 간에 데이터 접근과 제어가 가능한 새로운 규격을 정의하기 시작하였다[4-7].

그러나 아직까지 산업용 자동화 장비의 프로세스 데이터에 대한 접근과 제어 및 감시를 위해 필요한 HMI 시스템과 통신 인터페이스의 연구 및 개발은 산업용 기기에 종속적인 통신 프로토콜을 이용하는 기존의 레거시 방식과 기기에 독립적인 OPC-DA 방식이 주를 이루고 있으며 저장도구 등 사용자 편리성을 증대시키는 소프트웨어에 대한 연구는 거의 이루어지지 않고 있다. 또한 장비의 모니터링 환경도 기존의 전용

HMI 시스템뿐만 아니라 인트라넷이나 Web에서도 제어 및 감시가 가능한 통합된 환경의 HMI 시스템이 요구되고 있다.

따라서 본 논문에서는 공장 자동화를 위한 장비와 환경이 산업체마다 다르기 때문에 인트라넷이나 Web 등의 다양한 환경에서도 장비의 프로세스 데이터에 접근하여 제어와 감시가 가능하도록 하는 임베디드 OPC 기반의 통합형 HMI 시스템을 제안한다. 이를 위하여 임베디드 형태로 OPC 서버를 구현하고, COM/DCOM과 Web 서버 기술 등을 적용하여 접속환경에 맞는 인터페이스 모듈을 제공한다. 본 논문의 실험은 교육용 PLC의 접점들을 현장의 자동화기기의 접점들과 가상 맵핑하여 시뮬레이션하였다. 이와 같은 통합형 HMI를 다양한 제어 시스템 및 환경에 적용함으로써 이기종간의 장비 연결 및 데이터 상호교환 뿐만 아니라 네트워크 상의 원격 제어 및 Web 모니터링도 가능하기 때문에 유비쿼터스 환경의 HMI 시스템 구현에 기여할 것으로 기대된다.

II. 관련 연구

현재의 자동화 장비를 제어하고 감시하기 위한 프로세스 데이터의 통신 인터페이스는 대부분 기존의 레거시 인터페이스와 인트라넷 상의 OPC-DA에 집중되어 있고, Web 기반의 OPC-XMLDA 같은 기술 개발은 거의 이루어지지 않고 있다. 또한 OPC-DA 개발을 편리하게 해주는 저장도구 역시 기술적 개발이나 상품화가 이루어지지 않고 있다[14].

국내의 OPC 관련 기술은 서버와 클라이언트만을 제공하고 있는데 만일 OPC와 함께 저장도구를 제공한다면 사용자는 이것을 이용하여 환경에 맞는 HMI를 편집하여 원하는 산업용 장비의 프로세스 데이터에 접근할 수 있을 것이다. 반면 OPC만을 제공하는 경우에는 산업용 장비의 프로세스 데이터에 접근한 후에 장비의 프로세스 데이터를 가져오기 위한 추가 작업이 필요하게 된다.

OPC 클라이언트는 대부분 HMI 시스템 제조사에 의해 제공된다. Wonderware사의 InTouch HMI 소프트웨어는 산업 자동화, 프로세스 제어 및 감시 모니터링을 제공하는 그래픽 HMI이다. 전세계적으로 가장 많이 사용하는 HMI 소프트웨어이며, 제공되는 OPC 클라이언트는 서버의 기능을 할 수도 있다. 그러나 InTouch 경우 OPC 표준을 따르기는 하지만 적극적으로 OPC의 표준시양을 채택하고 있다[8,9,10].

국내 HMI 시스템 업체인 M2I는 다양한 제품으로 응용분야별 선택의 폭을 넓히고, 전 모델을 32bit RISC CPU를 적용하여 연산과 그래픽 속도를 크게 향상 시켰다. 또한 국내의

의 거의 모든 PLC 혹은 컨트롤러와의 통신을 가능케 함으로써 어떠한 시스템으로의 적용을 용이하게 하였다. 기존의 PC급 HMI 소프트웨어 시스템에서 가능했던 Recipe, Data Logging, Trending, Alarm Summary 등의 고급 기능을 실현하여 제어기기의 추가 부담을 경감시키고, 빠른 시간 안에 시스템을 완성 할 수 있도록 각종 지원기능을 향상시켰다[11].

국내의 여러 회사에서 OPC 서버 및 클라이언트를 지원하는 HMI 제조사가 세계적인 업체들과 경쟁을 하고 있으나 주로 OPC-DA에만 치중하고 있으며, Web 기반 모니터링이나 사용자 편의를 제공하는 저작도구 기술에 대한 연구가 미비한 상태이다.

Intouch와 M2I의 HMI 시스템을 비교해 보면 국내의 기술수준이 아직 세계의 수준을 넘어서지 못하고 있다. Advantech의 Advantech Studio와 같은 HMI 통합 개발 환경 역시 국내의 기술이 아닌 다국적 기업의 소프트웨어이며, 국내 기술로 개발된 범용의 HMI 시스템은 아직 없다고 봐야 할 것이다. 따라서 본 논문에서는 이와 같은 필요성을 만족시키기 위하여 임베디드 형태의 로컬 어플리케이션용 HMI와 인터넷 상의 클라이언트용 HMI 그리고 OPC 데이터를 Web에서도 접근할 수 있는 Web 페이지용 HMI를 모두 지원할 수 있는 통합형 시스템을 설계하고 구현하였다. 또한 임베디드 시스템에서 사용할 수 있는 저작도구 프로토 타입에 관한 내용도 기술하였다.

III. 임베디드 OPC 서버를 이용한 통합형 HMI 시스템

3.1 OPC 기반의 데이터 접근 방법

현재 산업용 장비의 제어를 위해서 각 장비마다 개별적인 HMI 시스템을 개발하고, 통신 프로토콜 환경을 구성해야 한다. 이러한 레거시 통신 인터페이스 환경은 새로운 장비의 적용이나 통신 프로토콜의 변화에 따른 대처에 많은 비용이 소모된다.

OPC-DA는 이러한 문제를 해결하기 위해 그림 1과 같은 소프트웨어 버스 개념을 도입하였다. HMI 시스템과 같은 응용프로그램의 개발자는 OPC-DA의 클라이언트의 규격사항을 준수하여 프로그램을 개발하면 OPC 서버를 지원하는 모든 산업용 장비에 응용이 가능하다. OPC 서버와 클라이언트는 OLE를 통해 통신을 하며, OPC 서버는 기존 HMI 시스템의 통신 인터페이스를 통해서도 각 산업용 장비와 통신을 수행하게 된다.

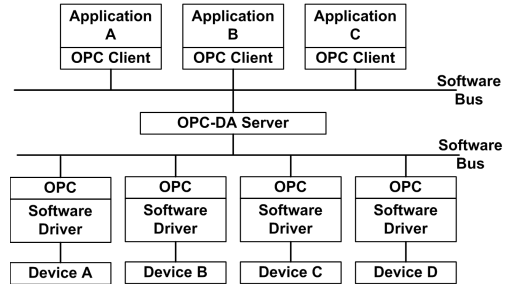


그림 1. OPC-DA 소프트웨어 버스
Fig 1. S/W bus of OPC-DA

3.2 통합형 HMI 시스템의 구성

본 논문에서 제시하는 통합형 HMI 시스템의 구성은 그림 2와 같다. 통합형 HMI 시스템은 OPC 서버와 Web 서버가 내장되어 있는 임베디드형 서버로 구성된다.

PLC로부터 데이터를 받은 OPC 서버는 이 데이터를 OPC 표준형 데이터로 변환하여 Web HMI, 인트라넷 HMI, 내장형 HMI 등의 사용 환경에 맞게 데이터를 공급한다.

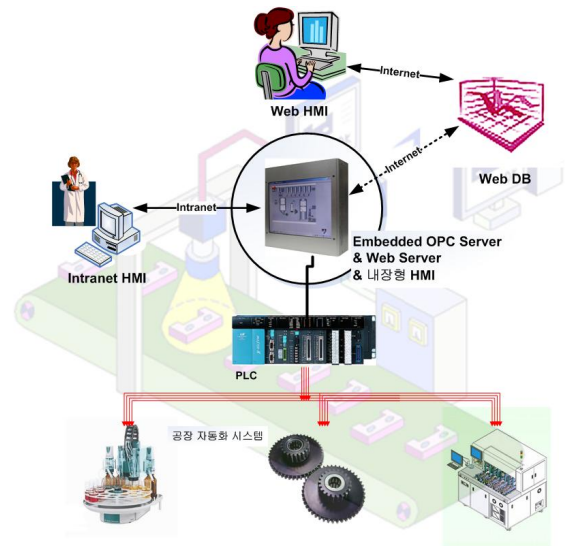


그림 2. 통합형 HMI 시스템 구성
Fig 2. Construction of Integrated HMI system

그림 3은 본 논문이 구현한 시스템의 각 모듈간의 연결 관계와 모듈에 연동되어 구동하는 클라이언트들의 상태를 보여준다. 통합형 HMI 시스템은 OPC 서버와 Web 서버, 공유 메모리 기반의 내장형 HMI, 그리고 경보 관리 모듈로 구성된다.

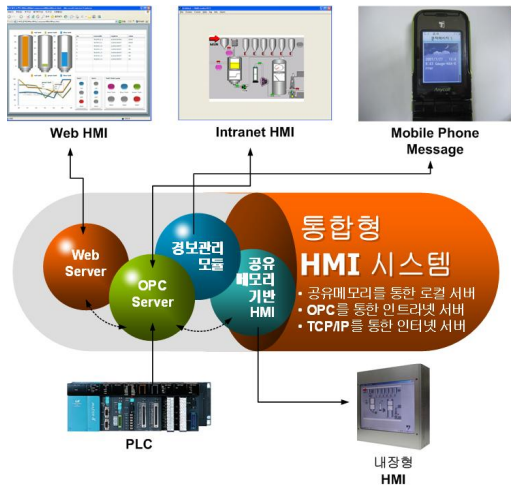


그림 3. 통합형 HMI 시스템 모듈
Fig 3. Modules of Integrated HMI system

OPC 서버는 PLC로부터 데이터를 받으며, 해당 데이터를 Web 서버와 공유 메모리에 전달한다. 또한 인트라넷 HMI를 통하여 PLC 데이터를 표현한다. Web 서버는 OPC 서버로부터 전달받은 데이터를 Web HMI를 통하여, 공유 메모리는 내장형 HMI를 통하여 표현한다. 경보 관리 모듈은 PLC 데이터 중 시스템에서 정의한 오류가 발생하면 해당 메시지를 관리자에게 전송하는 모듈이다.

3.3 임베디드형 OPC 서버의 구현

기존의 HMI 시스템은 공장 자동화 환경에서 통신 기능이 약하고, 통합 시스템을 구성함에 있어 어려움이 있다. 이러한 기존의 자동화 장비를 본 논문에서는 표준 데이터 형식을 지원하는 임베디드 OPC 서버를 이용하여 통합하였다. 임베디드 OPC 서버의 구조는 그림 4와 같다.

실제 PLC 데이터는 OPC 서버의 HMI 통신 드라이버를 거쳐 데이터 관리자(Data Manager)로 전달되고, 데이터 관리자는 이 데이터를 표준화된 OPC 형태로 변환시킨다. 변환된 데이터는 모니터링 환경에 따라 Web HMI와 인트라넷 HMI 또는 공유 메모리를 이용한 내장형 HMI 등에 전달되도록 한다.

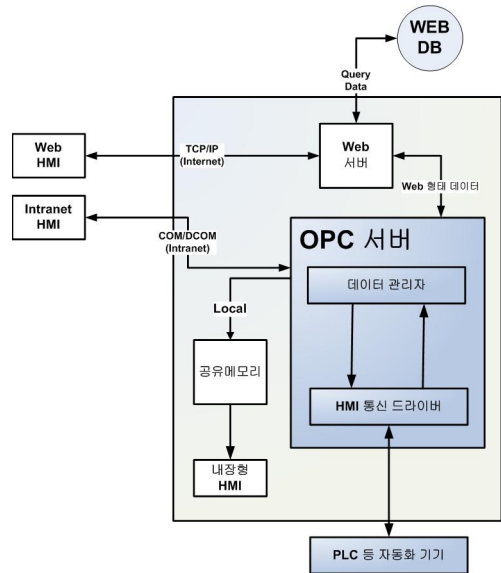


그림 4. 임베디드 OPC 서버 모듈
Fig 4. Modules of embedded OPC server

3.4 로컬 어플리케이션을 위한 임베디드형 HMI 시스템 설계

(1) 임베디드형 HMI 시스템

그림 5는 로컬에서 자동화 장비를 모니터링하기 위한 OPC 기반의 내장형 HMI 시스템의 전체적인 구성을 나타낸다. 이 시스템은 HMI 프로그램과 함께 OPC 서버가 임베디드 형태로 구성이 된다. 실제 PLC와 내장된 OPC 서버는 시리얼 통신으로 데이터를 전송한다.

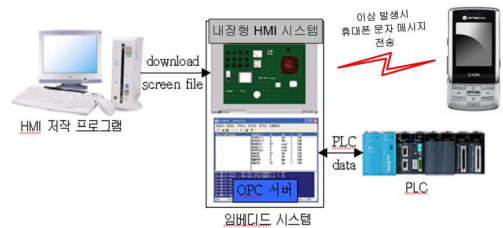


그림 5. OPC 기반의 임베디드 HMI 시스템 구성도
Fig 5. Construction of embedded HMI based on OPC

OPC 서버 내의 데이터 관리자는 PLC로부터 받은 데이터를 시스템의 공유 메모리에 저장한다.

그림 6의 내장형 HMI 시스템은 외부 PLC의 HMI 제작 프로그램에서 만들어진 HMI 스크린 파일을 로딩한 후에 공유 메모리에 저장된 PLC 데이터를 읽어들이어 HMI의 스크린에 디스플레이함으로써 관리자가 장비의 동작 및 이상 유무 등을 모니터링할 수 있게 한다.

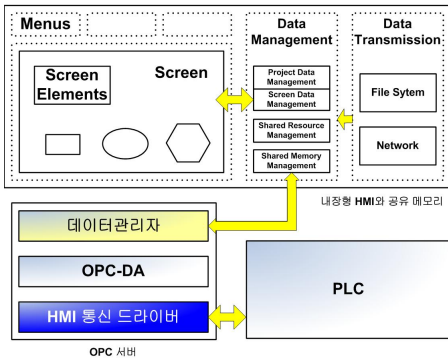


그림 6. OPC 기반의 임베디드 HMI 시스템 동작 과정
Fig 6. Process flow of embedded HMI based on OPC

(2) 경보 관리

본 논문의 HMI 시스템은 그림 5에 나타난 것과 같이 장비의 모니터링 중에 긴급 상황 및 경보 상태를 신속히 알리고 처리하기 위하여 관리자의 휴대폰에 문자 메시지(SMS:Short Message System)를 전송할 수 있도록 경보 관리 모듈을 설계하고 구현하였다.

문자 메시지 전송 시스템은 그림 7과 같이 동작한다. 내장형 HMI 경보 관리 모듈에서 정의한 시스템의 오류 조건에 상응하는 오류가 발생하면 경보 관리 모듈은 SMS 서버에 접속을 한 후 시스템의 오류 내용을 SMS 서버에 전송하고, SMS 서버는 오류 내용을 관리자에게 전송한다.

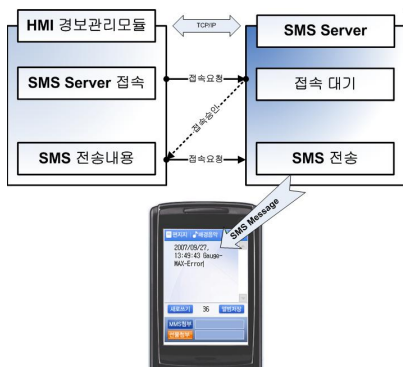


그림 7. HMI 시스템에서 SMS 동작
Fig 7. SMS operation on HMI system

(3) 공유메모리에 의한 OPC 서버와 임베디드 HMI 시스템의 통신

공유메모리는 여러 개의 프로세서가 하나의 메모리를 공유으로 사용하는 방법으로 HMI 시스템의 운용 프로그램에서는 PLC와의 통신횟수를 줄이기 위해 각 태그의 PLC 접근 주소를 별도로 관리하여 이것에 따라 PLC에서 데이터를 읽고 공유메모리에 쓰고, 각각의 태그들은 PLC 접근 주소에 해당하는 데이터를 공유메모리에서 읽어온다.

공유메모리의 구조는 그림 8과 같이 PLC의 메모리 구조와 동일하게 생성하여 사용하기 때문에 HMI의 스크린에서는 통신모듈과는 별도로 PLC에 직접 접근하는 것과 같은 방식으로 공유메모리에 접근하여 데이터를 읽는다.

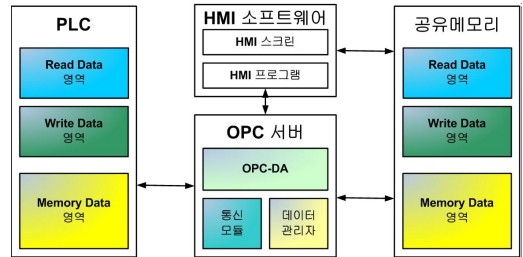


그림 8. 공유 메모리를 이용한 PLC 데이터 접근
Fig 8. PLC data access using shared memory

OPC 서버의 주소 및 데이터 관리자는 HMI 스크린의 태그에 접속되는 PLC를 정확하게 구분하여 선택하여야 하고, 해당 PLC 기기의 메모리 구조를 정확하게 파악하도록 공유메모리를 설계한다. 메모리는 여러 가지 데이터 형이 접근하기 쉽도록 바이트, 워드, 더블워드 형을 공용체로 정의하여 사용한다.

3.5 인트라넷 HMI 시스템 설계

본 논문의 인트라넷 HMI 시스템은 OPC 표준의 COM/DCOM을 사용하였다. COM은 공장 자동화, 프로세스 데이터 제어 및 상태 모니터링 어플리케이션의 주소 체계를 지원하기 위한 표준객체이고, DCOM은 분산 이더넷 네트워크에서의 서버/클라이언트 어플리케이션의 지원을 위한 객체다.

그림 9는 본 논문에서 구현한 OPC 서버의 객체 모델을 보여준다. OPC 서버는 서버, 그룹, 아이템으로 구분되고, OPC 서버 객체는 서버와 OPC 그룹 객체의 주요 정보를 갖는다. OPC 그룹 객체는 자기 정보와 OPC 아이템의 논리적이고 구조적인 정보를 가진다. 그리고 OPC 서버가 제공하는 아이템 집합을 관리하는 역할을 담당한다.

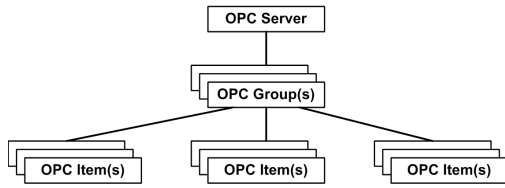


그림 9. OPC 객체 모델
Fig 9. Model of OPC object

또한 그룹 객체는 그룹의 아이টে를 담당하는 객체로 실제 인터넷 HMI의 요청을 직접 관리하는 많은 함수를 가지고 있어 인터넷 HMI의 데이터 구성을 지원한다. 각각의 함수는 아이টে를 관리하고, 아이টে의 정보를 갱신하는 역할을 담당하게 된다. 모든 인터넷 HMI 그룹은 하나 이상의 OPC 아이টে를 정의하고, OPC 아이টে는 데이터 소스를 서버에 전달하고 값, 품질, 시간 정보를 갖는다.

그림 10은 OPC 서버와 인터넷 HMI의 통신 구조를 나타내고 있다. 인터넷 HMI는 서버가 정의해 놓은 COM/DCOM 인터페이스를 통해 서버에 접근하고, 서버의 클래스를 구동하여 원하는 정보를 얻는다. 그리고 OPC 서버에 연결 요청을 함으로써 서버와 연결이 되고 그룹을 등록하게 된다. 또한 등록된 그룹에 아이টে를 추가하여 서버와 연결된 장비의 정보를 얻을 수 있게 된다. 추가된 아이টে는 클라이언트에 의해 검색이 가능하고, 읽기/쓰기 명령을 통하여 직접 제어를 할 수 있다.

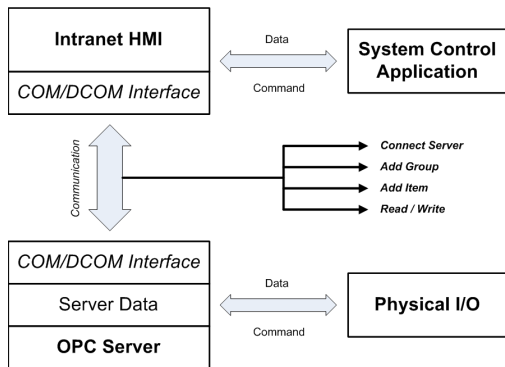


그림 10. OPC 서버와 인터넷 HMI의 통신 구조
Fig 10. Communication structure of HMI between OPC server and Intranet

3.6 Web HMI 시스템 설계

본 논문의 Web HMI 시스템은 3장의 그림 2와 3에 나타난 것과 같이 관리자가 Web으로도 공장의 장비 상태를 모니터

링 할 수 있도록 하기 위하여 OPC 서버와 함께 Web 서버를 내장형으로 구축한다. 그리고 Web 서버는 OPC 서버로부터 받은 데이터를 Web HMI에 전달함으로써 모니터링이 가능하게 하였다. 여기서 Web HMI가 Web 서버로부터 받은 데이터를 디스플레이하기 위한 도구로서 본 시스템에서는 Web 2.0 기반의 Flex를 사용하였다. Flex는 XML 태그로 된 스크립트를 작성해서 컴파일하면 플래시로 된 화면을 만들 수 있는 리치 인터넷 어플리케이션(RIA) 개발 도구이다. Flex는 사용자가 편리한 유저 인터페이스를 제공 받을 수 있고 개발자가 빠른 시간 내에 코딩할 수 있으며 실행 속도가 빠르기 때문에 실시간으로 장비를 모니터링하기에 적합하다.

(1) Web 서버

본 논문에서는 기존의 Web 페이지와 Web 2.0 기반의 Flex 컴포넌트를 사용하는 Web 페이지 모두 서비스 가능하도록 Web 서버를 구현하였다.

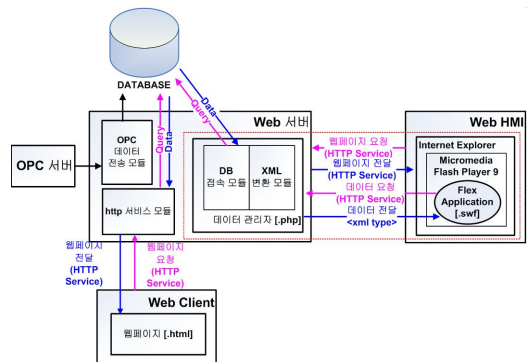


그림 11. OPC Web 서버와 Web HMI 구조
Fig 11. Structure of OPC Web server and Web HMI structure

그림 11은 Web 서버의 구조와 Web HMI 동작을 보여준다. Web 서버는 OPC 데이터 전송 모듈과 데이터관리자 그리고 http 서비스 모듈로 구성되어 있다. OPC 데이터 전송 모듈은 OPC 서버로부터 전달 받은 PLC 데이터를 데이터베이스에 저장한다. Web HMI로부터 Web 페이지와 데이터 요청을 받고 전달해주는 데이터관리자는 DB 접속 모듈과 XML 변환 모듈로 구성되어 있다. DB 접속 모듈은 Web HMI로부터 데이터 요청이 들어오면 데이터베이스로부터 해당 데이터를 받아 XML 변환 모듈로 전달한다. XML 변환 모듈은 이 데이터를 XML 형태로 변환시키고 데이터관리자가 변환된 데이터를 다시 Web HMI로 전달한다.

(2) Web HMI

본 시스템에서 구현한 Web HMI의 XML 데이터 이동 구조는 그림 12와 같다.

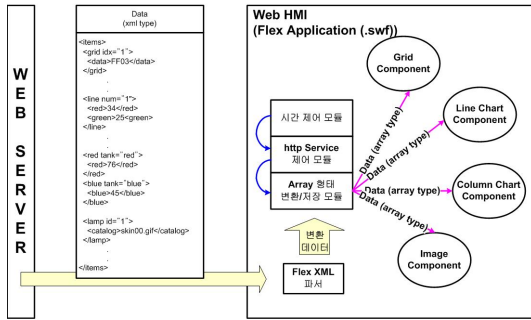


그림 12. Web HMI의 XML 데이터 이동
Fig 12. XML data flow of Web HMI

Web HMI의 Flex가 실행되면 주기적으로 http 서비스 제어모듈을 호출하는 시간 제어 모듈이 동작한다. http 서비스 제어모듈은 호출을 받을 때마다 Web 서버에 데이터를 요청하여 해당 데이터를 Array 변환/저장 모듈에 전달한다. Array 변환/저장 모듈은 Flex에 내장되어 있는 파서를 이용하여 전달받은 데이터를 각 컴포넌트들에 맞는 Array 형태로 변환시키고 저장한다. 이 데이터는 컴포넌트들의 속성에 맞춰 플래시로 표현된다[12].

그림 13은 본 논문에서 구현한 통합형 시스템의 실험 환경을 나타낸다. 이 실험을 위하여 ㉠임베디드형 OPC 서버와 ㉡PLC를 시리얼 통신으로 연결하였으며 공장 자동화 장비와 연결할 수 없는 관계로 PLC에서 직접 데이터를 입력하는 방식으로 시뮬레이션하였다.

IV. 실험

4.1 시스템 통합 실험

본 논문에서 제시한 방법들을 실험하기 위하여 현장의 자동화장비와 연결하여 하지만 실험 환경 상 어려움이 있기 때문에 본 논문의 실험은 교육용 PLC장비를 활용하여 장비의 각 스위치 및 램프와 점점들을 실제 장비의 제어 컴포넌트로 가정하고 이에 대응하는 PLC 번지를 할당함으로써 화면 HMI와 인터페이스하는 방법으로 시뮬레이션하였다. 따라서

PLC의 스위치 및 점점들의 값을 변경하거나 ON, OFF하면서 대응하는 HMI의 값들이 변경되는지를 확인하는 방법으로 실험하였다.

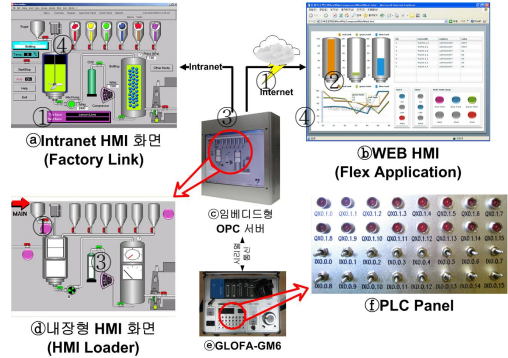


그림 13. 실험 환경
Fig 13. The environment of experiment

IX0.0.0의 스위치를 On시키면 ㉠인트라넷 HMI와 ㉡Web HMI, 그리고 ㉢내장형 HMI의 ①Mix Tank가 선택되어 IX0.0.0~IX0.0.10까지의 입력에 따라 Mix Tank의 그래프 게이지가 변하는 것을 확인할 수 있다. IX0.0.1의 스위치를 On시키면 ㉡WEB HMI의 ②표가 선택되어 IX0.0.0~X0.0.10까지의 입력에 따라 바뀌는 값의 변화를 숫자로 확인할 수 있다. IX0.0.2의 스위치를 On시키면 ㉡WEB HMI의 ③이 선택되어 값의 변화를 꺾은선 그래프로 확인할 수 있고, ㉢내장형 HMI의 ③이 선택되어 두 개의 게이지로 값이 변화를 확인할 수 있다. IX0.0.3의 스위치를 On시키면 ㉠인트라넷 HMI의 ③(6개의 Tank)와 ㉡WEB HMI의 ④(6개의 Lamp)가 선택이 되어 IX0.0.4~IX0.0.9까지의 스위치 버튼의 변화에 따라 Tank와 Lamp가 선택됨을 확인할 수 있다.

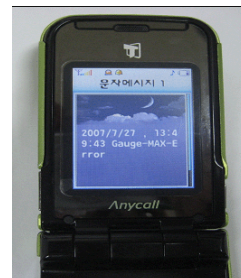


그림 14. SMS를 이용한 오류 메시지 전송
Fig 14. Error message using SMS

또한 Tank의 최소값을 0, 최대값을 100으로 설정하여 PLC로부터 입력되는 값이 100이 넘어갈 경우 경보관리모듈에 의하여 관리자에게 오류 메시지를 보내도록 하였다. 해당 결과는 그림 14에서 확인할 수 있다.

4.2 임베디드 HMI 실행환경 측정

본 논문에서 구성한 HMI 운용 시스템은 ActiveSync를 통해 화면편집 데이터를 다운로드 받아 사용자 인터페이스로 사용하면서 산업용 장비의 감시를 수행하게 된다. 논문에서 구성한 임베디드 환경의 평가를 위해 같은 환경에서 동작하는 HMI 시스템을 찾을 수 없는 점을 고려하여 실행환경 측정 결과만을 나타낸다.

그림15는 임베디드 시스템의 HMI 운용프로그램의 시작을 전후하여 측정된 메모리 량을 나타낸다.

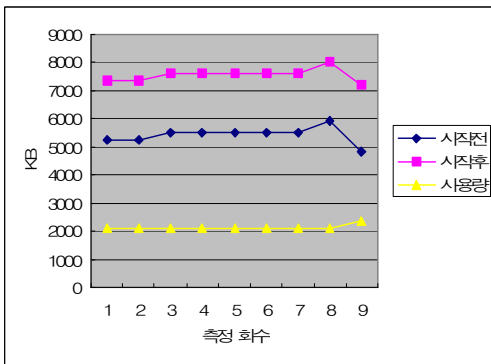


그림15. HMI 동작 전후의 메모리 사용량(KB)
Fig 15. Memory share before and after HMI Operation

그림15는 사용자 인터페이스 데이터를 불러오기 전의 메모리 사용량이며, 시스템은 비교적 안정적으로 동작하는 것으로 나타난다. 표 1은 HMI 운용프로그램의 시작 이후 사용자 인터페이스 데이터로서 각각의 객체를 10개씩 화면에 나타내고 PLC와의 통신을 통해 데이터를 측정할 때의 메모리 사용량을 나타낸다.

표 1의 실험 결과 전반적으로 임베디드 시스템에서 안정적인 메모리 사용량을 나타냈으나 막대 그래프, 라인 그래프, 게이지 객체, 타임 객체의 경우 비정상적으로 많은 메모리의 소모가 있음을 실험을 통해 알 수 있었다. 분석을 통해 그래프의 화면 표현을 위한 클래스 구조에 문제가 있음을 알게 되었으며, 클래스 구조의 수정 후 다시 측정된 결과가 표 2에 나타난다.

표1. HMI 시스템 화면로딩 전후의 메모리 사용량(KB)
Table 1. Memory share before and after screen load in HMI system(KB)

생성 객체	화면 로딩 전	화면 로딩 후 운용	차이
숫자	7376	7460	84
텍스트	7460	7520	60
문자열	7520	7580	60
버튼	7596	7608	12
램프	7592	7604	12
막대그래프	7596	9200	1604
라인그래프	7596	9384	1788
게이지	7184	8980	1796
비트맵	7608	7616	8
모터	9384	9400	16
타임	8008	8384	376

표2. 클래스 수정후 메모리 사용량(KB)
Table 2. Memory share after class modification

생성 객체	화면 로딩 전	화면 로딩 후 운용	차이
숫자	7371	7453	82
텍스트	7457	7518	61
문자열	7519	7572	58
버튼	7598	7612	14
램프	7592	7603	11
막대그래프	7590	7715	125
라인그래프	7592	7714	122
게이지	7187	7304	117
비트맵	7605	7612	7
모터	9381	9399	18
타임	8010	8076	66

표 2의 결과는 막대 그래프, 라인그래프, 게이지와 타임 객체의 비정상적 메모리의 사용이 해결 되었음을 보여준다. 그래프와 게이지의 태그는 램프 등의 간단한 태그의 표현에 비해 내부적으로 많은 데이터의 처리과정이 필요하므로 다른 태그에 비하여 많은 메모리를 사용 하는 것을 보여준다. 향후 데이터 처리 부분과 화면의 갱신 부분의 알고리즘을 개선하여

보다 적은 자원을 사용하도록 수정할 계획이다. 타임 객체의 경우 시스템의 클럭을 얻어와 시간을 계산하는 방법으로 변경함으로써 376KB에서 66KB로 현저하게 사용량이 감소함을 보여준다. 임베디드 시스템에서 사용하는 시간객체를 사용하지 않고 필요시마다 시스템의 클럭을 얻어와 시간 계산을 할 경우 훨씬 효율적으로 메모리의 사용량을 줄일 수 있음을 보여준다.

V. 결론

본 논문에서는 공장 자동화 장비를 인트라넷이나 Web 등의 다양한 환경에서도 장비의 제어와 감시가 가능하도록 임베디드 OPC 기반의 통합형 HMI 시스템을 제안하고 구현하였다. 이를 위하여 임베디드 형태로 OPC 서버를 구현하였고, COM/DCOM과 Web 서버 기술 등을 활용하여 각기 다른 접속 환경에 맞는 인트라넷 HMI와 Web HMI, 내장형 HMI와 같은 다양한 클라이언트 환경에서 활용할 수 있게 하였다. 그리고 경보관리 모듈을 도입하여 관리자의 휴대폰으로 문자 메시지를 전송함으로써 시스템 오류에 대한 빠른 조치를 기대할 수 있게 하였다. 또한 기기종간의 장비 연결 및 데이터 상호교환 뿐만 아니라 네트워크상의 원격 제어 및 Web 모니터링의 가능성을 보여줌으로써 유비쿼터스 환경의 통합형 HMI 시스템 구현이 가능할 것으로 기대된다.

참고문헌

- [1] 박지환 "HMI 소프트웨어의 기술 동향", 월간 제어계측 제 16권 1호, pp. 28-32, 2003.
- [2] Rockwell 삼성오토메이션팀, "공장 제어 시스템의 향상 및 진화", 제어계측, 34-35쪽, 2005.
- [3] 이광민, 선복근, 류제, 이지현, 임성락, 한광록, "임베디드 시스템 기반의 자동화 시스템 제어와 웹 서비스 제어 기술 구현", 대한전자공학회, 제 29권 제 2호, pp. 1023-1024, 2006.
- [4] OPC Foundation, "OPC Data Access Automation Specification", February 1999.
- [5] OPC Task Force, "OPC(OLE for Process Control) Overview" October 27, 1998.
- [6] Advosol Inc., "XMLDA.NET White Paper", <http://www.advosol.com>, 2004.

- [7] <http://www.wonderware.com>, "InTouch 9.0 Visualization Software", 2004년 11월
- [8] <http://www.wonderware.com>, "Thin Client HMI Changes the Face & Economics of Plant Floor Control Systems, Devices & Workstations", 2001년 9월
- [9] <http://www.advosol.com>, "XMLDA.NET White Paper", 2004년
- [10] Frank Iwanitz, "XML-DA opens windows beyond the firewall", The Industrial Ethernet Book, pp 10-12, May 2004
- [11] <http://www.m2i.co.kr>, "Human Machine Interface White Paper", 2005년
- [12] Adobe Flex Developer Center, <http://www.adobe.com>.
- [13] 엘지산전 교육부, "LG Programmable logic controller 프로그래밍 Glofa GM 시리즈", 1998.
- [14] 선복근, "OPC 기반의 자동화 장비의 제어 및 감시를 위한 임베디드 HMI 시스템에 관한 연구", 호서대학교 박사학위 논문, 2005.

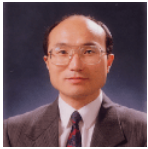
저 자 소 개



여훈구

2006년 2월 : 호서대학교 공학사
2008년 2월 : 호서대학교 공학석사
2008년 3월~ (주)IT에비뉴 주임연구원

관심분야: 멀티미디어, HCI, 임베디드 시스템



임성락

1992년 8월 : 서울대학교 컴퓨터공학박사
1993년 ~현재 : 호서대학교 컴퓨터공학부 교수

관심분야: 시스템 소프트웨어, 임베디드 시스템



한광록

1989년 8월 : 인하대대학교 정보공학박사
1991년 ~현재 : 호서대학교 컴퓨터공학부 교수

관심분야: 정보검색, 멀티미디어, 시멘틱웹, 임베디드 시스템