



A Study on the Design and Implementation of ODSP in Different PLC of Production Process

Dal-Joong Kim¹, Pyoung-Jung Kim², Soo-Cheol Ha¹

¹*Department of Computer Engineering, Daejeon University*

²*Department of Computer Intergrated Engineering, Chungbuk Provincial College*

A B S T R A C T

In this paper, we designed and implemented by ODSP(OPC Data Sharing Platform) to share data in real time the production facilities of different PLC. Also We made a ODSP that a management of manufacturing process integrated data from real-time. Interface of the production relies on providing DLL by the manufacturer for communication. We solved problem that Interface module programed in every DLL communication for different PLC with gathering server logic. In this paper, ODSP was implemented by system that solved providing real-time information about the End User for several different PLC data. ODSP services in real time from the different PLC by the OPC data sharing platform and stores the data in real-time, it is implemented to take advantage of a variety of information through the connection with other systems. In addition, automation equipment can be applied to production test reports submitted proof of origin for imports through the FTA partners in the export business operators. ODSP can be the self diagnosis of the real-time manufacturing graph, production management monitoring, facility management and production process. This shows middle-and-small companies can expand production management, process management, and PLC data sharing to real-time production operation management by the level and sector, and can control production facility by the real-time base.

© 2015 KKITS All rights reserved

KEYWORDS : OPC server, Factory automation, MES, Automated diagnosis, PLC, Databases, Realtime data sharing

ARTICLE INFO: Received 11 November 2015, Revised 11 December 2015, Accepted 11 December 2015.

*Corresponding author is with the Department of Computer Engineering, Daejeon University, 65 Daehak-ro,

Dong-gu, Daejeon, 300-716, KOREA.
E-mail address: djkim@sayinfo.co.kr

1. 서론

빠르게 변화하는 IT 환경에서 글로벌 생산제조 환경을 사용하는 공정에서 다양한 시장 요구에 대하여 능동적으로 대처할 수 있는 정보시스템 요구 사항이 빠르게 증대되고 있다. 생산관리시스템 또는 제조실행시스템 MES(Manufacturing Execution System)은 생산계획·작업지시·자재소요·생산추적·설비관리·생산성과분석 등을 맡아 생산관리의 효율성을 높이는 시스템이다. 현재는 제조 기업에 적용되는 통합생산관리시스템으로 변화되어 제조기업의 핵심 영역의 가용성, 효율성 및 안정성을 향상시키기 위한 다양한 접근방법과 솔루션이 제공되며, 비즈니스 분석과 빅 데이터를 통한 유통 및 물류 업계의 불확실성 제거 및 미래 예측 방안에 대해 공유하는 시스템으로 발전되고 있다. MES는 생산공정 현장의 정보를 상위의 ERP(Enterprise Resource Planning) 시스템과 연계시켜 현장에 대한 통제와 계획을 수행하는 시스템이다. MESA(Manufacturing Enterprise Solutions Association)에서 MES는 생산현장에서 발생하는 다양한 정보를 효율적으로 수집할 수 있는 데이터수집시스템을 기반으로 구성되어야 한다고 정의한다[1,2,3].

생산관리시스템 기본은 생산에 필요한 스케줄링과 이를 위한 계획 및 자재수급으로 파악하고자 하는 방식으로 시스템의 기능을 살펴보면, 공정진행 정보의 조회 및 제어, 설비제어 및 조회, 품질정보색출 및 제어, 실적정보 집계, 창고운영 관리, 재고품 관리, 자재투입 관리, 인력 관리, 공무관리 등 생산 현장에서 발생할 수 있는 모든 정보를 통합 관리하고 있다. 전 제조 공정에 걸쳐 비용 낭비와 재고 관리 등을 도와주는 온라인 시스템이다. 수집된 데이터를 기반으로 불량품 비율을 줄이고 공정에서 문제가 발생한 지점을 점검하여 관리하게 된다[2,4].

제품 생산을 위한 것은 작업자, 설비, 도구 등을 이용하여 각 공정에서 시간, 효율, 설비상태, 작업자 정보 등 각종 실시간 데이터가 발생되어 DB에 저장되고 활용된다. 이러한 시점에 제품 수요처에서는 생산 공정 현장에서 제품을 생산하는 경우 설비에서 자동 생성되는 데이터의 요구(검사 Data, 장비설정 Data, 생산 Recipe Data 등)나 부품의 원산지 증명을 위한 데이터와 품질 검사성적서 등을 요구하여 제품에 대한 이력추적을 원하고 있으며, 수작업으로 기록된 Data는 변조의 위험에 노출되고 신뢰성이 떨어져, 자동화공정에서 생산설비로부터 실시간 추출되는 데이터를 제품과 함께 요청하고 있다.

생산제조업에서 PLC (Program Logic Control) Device는 공장자동화 작업의 공정 범위에서 많이 사용된다. 생산 공정 정보시스템은 생산현장의 일차 데이터들을 기반으로 가공하여 생산현황을 파악한다. 또한 생산 계획 수립 및 품질 관리 등의 기업경영에 활용할 수 있도록 데이터를 수집한다. 이런 정보시스템의 데이터 신뢰성을 높이기 위해서는 생산 현장 및 설비의 실시간 현황 정보 데이터를 기반으로 정보시스템을 수집 및 운영하여야 한다. 생산설비 인터페이스는 PLC 업체에서 제공하는 통신 DLL을 사용하여 각각의 장비마다 인터페이스(interface) 모듈을 개발하고, 이렇게 개발되어 사용되는 DLL은 1:1 통신방식으로만 가능하다. PLC 생산업체가 다를 경우 PLC마다 통신을 할 수 있는 모든 DLL을 구별하여 구축해야 한다. 이것은 생산설비와 설비 간 원격지 데이터 인터페이스에 대한 구현의 어려움이 발생하는 문제가 있다[4,5,6].

본 논문에서는 표준 설비 인터페이스 OPC(OLE Process Control)를 사용하여 해결하고자 한다. 서로 다른 제품의 생산설비 PLC를 OPC 인터페이스로 공유하고, PLC 데이터를 데이터 수집서버로 실시간 공유 하게 함으로써 해결하고자 한다. 우리는

생산설비에서의 서로 다른 PLC와 데이터 수집서버 간 OPC 데이터 공유를 위한 플랫폼(ODSP, OPC Data Sharing Platform)을 설계하고 구현한다. 이를 통해 서로 다른 PLC 업체에서 제공하는 통신 DLL을 사용하여 각각의 장비마다 인터페이스 모듈을 개발해야 하는 문제점을 해결할 수 있음을 보여주고 있다.

2. 관련 연구

2.1 공장자동화(MES)

MES는 제품 주문에서 출고 단계까지의 생산최적화를 위한 시스템으로서, 제조자원계획 (MRP II: Manufacturing Resource Planning)이나 ERP와 달리 생산계획을 제조현장에 지시하고 자재투입 상황을 통제하여 실적을 집계하고, 설비 및 품질현황정보를 수집하여 적절한 조치를 취하는 등의 통합생산 공정 관리 기능을 수행할 수 있다[7].

제조업의 환경은 제품개발에서부터 생산, 정보공유에 이르기까지 실시간 기업(RTE: Real-Time Enterprise) 모형을 따르는 활동이 더욱 강조되는 방향으로 흐르고 있다. 이러한 환경 변화에는 생산과 물류의 글로벌화, 제품개발 간격과 생산 Lead Time 단축, 주문에 대한 정확하고 신속한 대응 증대, 품질규정 준수 및 공정추적에 대한 요구 증대, 고객 및 파트너 간 정보 공유의 필요성 증대 등이 있으며, 이에 따라 MES와 ERP, SCM(Supply Chain Management), CRM (Customer Relationship Management), PLM(Product Life cycle Management), MIS(Management Information System) 등과의 유기적인 협업은 한층 더 강조되어 가고 있다.

ERP, SCM, CRM과 같은 전사시스템은 솔루션 도입에 의한 구축이 일반화되어 있지만, MES 솔루션

은 생산방식이 기업체별로 다양하고 산업별로 특화되어 있으며 종류 또한 다양하기 때문에 많은 기업들이 적절한 솔루션 선택에 어려움을 겪고 있다. 이러한 특성으로 인하여 상당수 기업이 패키지 MES 솔루션의 도입보다는 자체 개발을 통하여 MES를 구축하는 방법을 택하고 있다.

생산 공정 특성에 따른 특정 품질데이터 수집시스템이 필요하다. 이에 따라 작업자의 간섭 없이 설비와 직접 인터페이스 하기위한 방법과 설비상태 모니터링에 관한 많은 연구가 수행되고 있으며, 국내에서는 진동, 음향, 전류 등의 멀티센서를 이용하여 절삭가공 중 공구의 마모와 상태를 모니터링 할 수 있는 시스템에 대한 연구가 수행되었고, XML을 활용하여 설비 상태 정보를 원활히 처리하고 모니터링 할 수 있는 미들웨어 모니터링 시스템에 대한 연구가 진행되어 왔다[3,6].

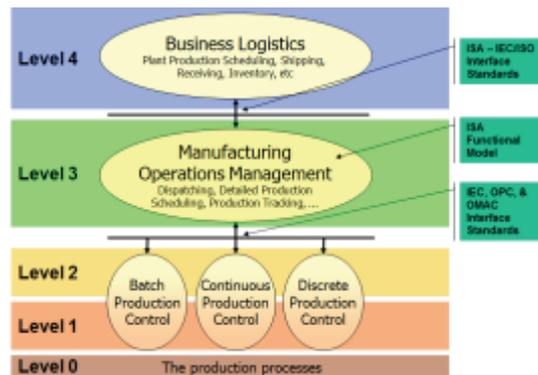


그림 1. ISA-95 제어 계층 구조
Figure 1. ISO-95 Control Hierarchy Levels

<그림 1>은 ISA(International Society of Automation)-95에서 제정한 Enterprise domain hierarchy 기반의 설비정보 수집시스템 모델이다. 위 그림은 제조현장(Level 0)에서부터 ERP와 같은 기업시스템(Level 4)까지의 정보 흐름을 보여주고 있다. MES는 Level 0에서 Level 3까지의 영역에 해

당하며, 설비 인터페이스 부분은 Level 0에서 Level 2까지의 영역이다[3]. 여기에는 제조현장에 있는 실제 설비 및 장비들과 모니터링 및 제어 기능이 포함되어 있는 device communication(Level 2)을 통해 수집되는 실제 설비의 생산정보가 매우 중요하다. ERP와 같은 상위시스템은 제조현장에 대한 단순한 형태의 작업지시 명령만을 전달하고 제조현장에서는 작업지시에 따른 실적만을 상위시스템에 보고하기 때문에 작업지시부터 실적보고 사이의 중간과정에 대한 추적, 감시, 제어가 불가능하다. 반면 MES는 bottom up 방식으로 생산공정 스케줄링에 기반을 두어 제조현장에 작업지시를 하고 지시된 작업에 대해 개별공정에서 발생하는 다양한 생산공정 데이터들을 실시간으로 수집하여 가공된 정보를 데이터베이스에 저장하며 이를 제공한다. 제조업의 특성에 따라 제조현장의 생산공정 특성은 상이하므로 이에 적합한 데이터수집시스템의 개발이 필요하다.

표 1. 제조현장 발생정보 분류
Table 1. Data Classification of Manufacturing Business

생산관리	원가관리	설비관리	품질관리
- 생산능력, 설비수	- 가동시간	- 가동상태	- 운전조건
- 생산 진척 정보	- 재료사용량	- 비가동/고장 시간	- 복구시간
- 재고수	- 에너지사용량	- 사용/가공이력	- 이력(특성, 성능, 불량 원인 등)
- 공정 위치	- 재고(원자재, 재공품, 제품 등)	- 비가동내역 (준비, 공정 대기, 불량 발생, 고장 등)	- 양품&불량 수
- 생산량, 작업시간	- 작업자공수		- 불량 내용
- 작업내용	- 관리비		
- 시작 W/O 번호			
- 시작/종료 시간			

<표 1>에 생산 제조현장에서 발생된 정보와 이들 정보의 사용 항목을 분류하였다. 생산공정현장에서 발생하는 정보는 동시에 발생되며 이들 정보를 이용하여 생산관리, 원가관리, 설비관리, 품질관

리 등에 적용되도록 데이터베이스에 실시간으로 저장한다. 이들 정보는 전압, 전류, RPM 등 설비에서 직접 수집한 정보인 1차 생산공정정보와 1차 정보에서 가공되는 2차 정보로 분류할 수 있다. 2차 정보는 가동시간, 고장시간, 불량여부, 공정 전/후 대기시간 등이 있다. 3차 정보로는 1,2차 정보와 설비정보, 제품정보, 작업자 정보 등과 같은 기준 정보와의 mapping을 통해 산출되는 정보로 설비별 일일 생산품 수, 작업 실적, 작업평균시간 등이 있다[7].

2.2 PLC 기반의 설비 인터페이스

CNC(Computerized Numerical Control)를 장착한 공작기계는 제조현장에서 생산성이 높은 장비로 자리 잡고 있다. 그러나 제조현장의 대부분 CNC 공작기계들은 폐쇄형 구조라서 외부와의 인터페이스가 어렵게 설계되어 있다. 주로 제조사에서 제공하는 고가의 API를 사용하여 설비의 상태감시와 제어를 수행할 수 있다. 대부분의 CNC 공작기계는 PLC 혹은 PMC(Programmable Machine Controller) 입/출력 접점을 가지고 있다. 이러한 PLC를 이용하여 자동화기계가 발생하는 실시간 데이터를 인터페이스하여 CNC와 제어부의 입/출력 접점과 외부 접점을 연결하여 기계의 상태를 추출하는 방법이다[7].

<그림 2>에서 HMI(Human machine interface)를 이용한 CNC와 PLC의 데이터 처리 흐름을 나타내었다. CNC 출력 접점을 PLC 외부 입력단자에 연결하여 CNC 출력접점에서 출력된 신호를 PLC 논리 프로그램을 이용하여 처리한다. CNC 접점에서 출력된 신호를 PLC 외부입력단자와 연결한다. CNC에서 출력되는 신호는 기종마다 다르며 각 접점의 조합에 의해 상태가 결정되는 경우도 있다. Digital signal input module을 통해 CNC 접점의 출

력 신호를 받아 양변환 검출을 거치며, 양변환 검출을 통해 4개의 Move Function (Digital signal forward features)을 이용하여 내부 메모리에 저장한다. 저장된 신호는 bool 연산 (SUM)을 이용한 신호합성을 거쳐 ASCII code 값으로 변환한 후 통신 함수를 이용해 외부기기(PC)로 전송하게 된다. 통신 방법은 PLC 통신모듈(Cnet I/F)과 PC(HMI DAS client)의 RS232 또는 LAN으로 TCP/IP 연결을 통해 실제 생산설비정보 전송이 이루어진다. PLC 인터페이스에서는 송신용 function block을 작성하여 통신용 프레임을 구성하고 데이터 처리와 통신을 위한 ladder diagram을 작성한다.

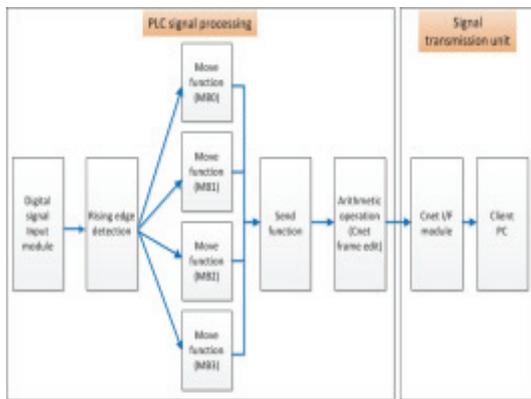


그림 2. CNC와 PLC의 데이터 프로세스
Figure 2. Data Process of CNC and PLC

PLC 프로그램은 제조업의 자동화 수준을 결정한다. 그러나 PLC의 프로그래밍 시스템 언어는 상대적으로 융통성이 없기 때문에 오류 검출 및 장비 진단의 능력이 제한된다. PLC Device와 제어 시스템 자체 오류 발견 및 진단을 위한 고유한 모듈이 없기 때문에 추가적인 자동화 모듈이 필요하다. PLC 제어 생산 시스템 오류 검증 및 진단 업무는 PLC 컨트롤러의 내부 데이터를 기반으로 한다. 생산 시스템에서 PLC는 시스템의 동작을 제어하는데 사용된다. 이러한 시스템의 작업과 운영 작업의 시

퀀스는 PLC 제어 프로그램으로 PLC 제작 회사마다 다르게 사전 작성된다. 제어 프로그램은 생산설비 시스템 일련의 운영 작업을 계획하고, PLC는 계획된 작업에 따라 생산설비 시스템을 제어하도록 한다[8,9,10].

2.3 공정자동 표준 OPC

OPC는 산업자동화 태스크 포스에 의해 1996년 개발된 표준이다. 이러한 표준은 서로 다른 제조업체의 컨트롤 디바이스 사이에서 실시간 플랜트 데이터 통신을 명시하고 있다. 표준은 OPC Foundation에 의해 관리되며, OPC Data Access Standard로 명명하였다. 현재 OPC Data Access의 스펙은 3.0이 최신이며 Windows 기반 어플리케이션을 연결하고 컨트롤 하드웨어 및 소프트웨어 어플리케이션을 처리하기 위해 고안된 것으로 공정 설비로부터 데이터 접근을 위한 연속적인 방법을 허용하는 개방표준이다.

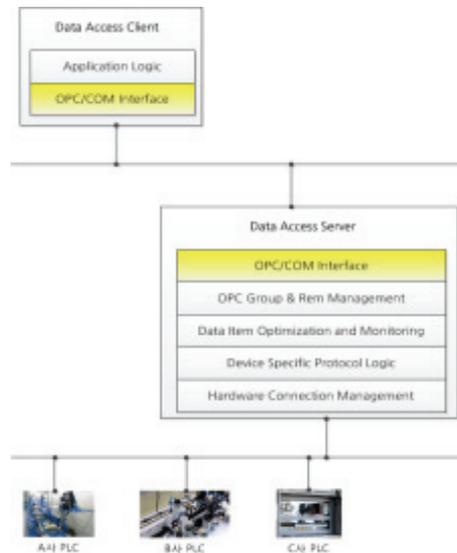


그림 3. OPC 데이터 액세스 구조
Figure 3. OPC Data Access Architecture

OPC의 목적은 데이터에 접근하기 위해 소프트웨어 패키지가 필요할 때마다 작성되어야 했으며, 한번 작성된 이후 SCADA(Superbisory Control And Data Acquisition), HMI, 맞춤형 소프트웨어 패키지에 의해 재사용될 수 있는 일반적인 인터페이스를 정의하는 것이다[5].

3. ODSP 설계 및 구현

3.1 ODSP 전체 구조

본 연구는 생산공정 현장에서 생산설비가 가진 서로 다른 PLC 데이터를 실시간으로 수집하여 운전상태 및 제어, 진단 등의 정보를 추출하고 저장하도록 하는 ODSP 플랫폼 설계하고 구현하는 것이다. 그러므로 ODSP는 생산설비의 서로 다른 PLC와 데이터 수집서버 간 실시간 OPC 데이터 공유 플랫폼(ODSP, OPC Data Sharing Platform)이다. 본 연구를 통하여 서로 다른 PLC에서 공유되는 통신 인터페이스 플랫폼을 개발하여 첫째, 생산공정 현장의 서로 다른 PLC 설비 운영 상황을 통합 모니터링 할 수 있고, 둘째, 수기로 작성되던 검사결과에 대한 수치를 설비에서 제공되는 데이터로 자동 입력되고, 셋째, 제품 생산 시 투입되는 원자재 정보를 자동으로 식별할 수 있도록 설계 및 구현한다. 최근 들어 고객사에서는 납품 회사에게 제품 납품 시 제품에 대한 생산품질 데이터를 요구하고 있고, 제품 생산 시 설비에서 생산되는 원산지 추적성 데이터, 품질 데이터, 검사 데이터, 원자재 데이터를 포함하여 제품과 함께 납품을 요구하는 실정이다. 중견 제조기업에서는 MES를 도입하여 운영하고 있으나 포괄적인 패키지 형태의 시스템으로 구매하여 사용하고 있으며, 설비와 직접 통신을 하는 서로 다른 PLC의 공유 데이터에 대한 인터페이스 기능이 없는 경우가 많다. 이러한 추세에 맞

추어 ODSP는 생산설비 요구 사항에 부합하여 설비와 통신 인터페이스 할 수 있도록 설계/개발한다. 생산공정 자동화 시스템 도입에는 비용과 시간적인 문제 대두되고 있으며, 신규 시스템 개발은 실시간 서로 다른 PLC의 생산공정설비 데이터를 공유 플랫폼으로 가져와 생산공정 데이터를 수집하고 저장하여 이용할 수 있도록 데이터베이스를 구축한다.

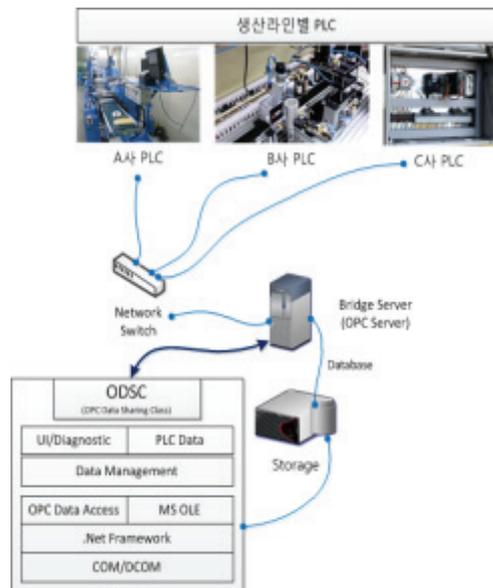


그림 4. OPC Data Sharing Platform 구조
Figure 4. ODSP Architecture

<그림 3>은 실시간 생산설비 PLC 데이터 공유 플랫폼 구조이다. 이 플랫폼은 첫째, 생산라인의 제조사별 PLC와 표준 OPC 인터페이스 기술을 확보하고, 둘째, OPC 서버와 ODSP 간 DCOM 통신을 이용한 서로 다른 PLC 데이터의 실시간 공유 기술을 구현하며, 셋째, 생산라인의 제조사별 PLC 실시간 데이터 수집모듈을 통하여 플랫폼을 구성한다.

공정생산설비 PLC와 데이터수집서버 간 통신은 이더넷 10/100과 시리얼 RS-485를 사용한다. 하나의 생산공정에서는 제조사가 다른 여러 개의 생산

라인을 가지고 있을 수 있다. 각 생산라인의 제조 사별 PLC가 서로 다를 수 있고, PLC당 윈도우 SW Driver를 통하여 접속을 제어한다. 다행히도 OPC Foundation에 의해 관리되고 있는 국제 표준 프로토콜 OPC가 있고, 거의 모든 PLC 제조사가 OPC Foundation에 가입되어 OPC 표준 (<http://www.opcfoundation.org>)을 준수하고 있다.

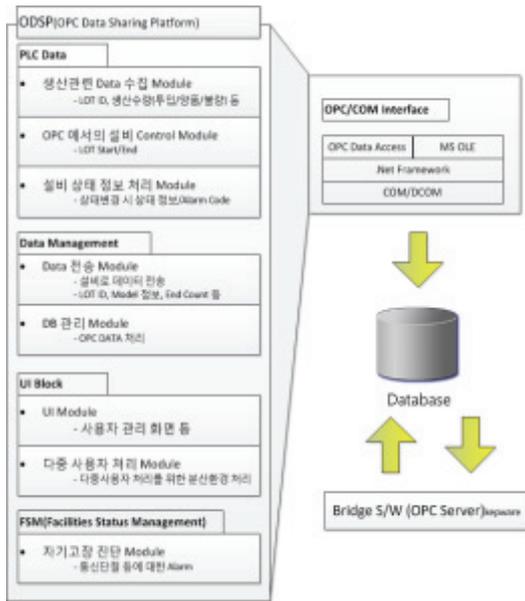


그림 5. ODSP 구조
Figure 5. ODSP Architecture

따라서 생산라인의 제조사별 PLC와 표준 OPC 인터페이스 기술을 이용하여 설계한다. 생산공정 데이터수집서버에 Bridge S/W와 ODSP 간 통신으로 DCOM (Distributed Component Object Model) 통신을 이용하여 설계한다. 서로 다른 PLC 데이터로는 생산관련 Data 수집 (Lot_ID, 생산수량 (투입/양품/ 불량 등), 설비 Control (Lot_Start/End 등), 설비로 Data 전송(Lot_ID, Model 정보, End Count 등), 설비 상태 정보(상태 변경 시 상태 정보/Alarm_Code 등)이고, 이를 통하여 실시간 생산 OPC 데이터 공

유를 제공하는 데이터 수집서버를 설계 및 구현한다.

ODSP은 생산관련 Data 수집 Module, OPC에서의 설비 Control Module, 설비 상태 정보 처리 Module, Data 전송 Module, DB 관리 Module, UI Module, 다중 사용자 처리 Module, 자기 고장 진단 Module로 구성한다. <그림 5>는 ODSP의 데이터 수집 구조이다.

3.2 ODSP 설계

ODSP를 구현하기 위해서는 앞 장에서 구조를 설명 하였듯이 ODSP는 세 가지 Block을 기준으로 구현한다. PLC Data Block, Database Management Block, UI/Diagnostic Block 등으로 구별한 ODSP의 상세 내용은 <표 2>와 같다.

표 2. ODSP Block 구현
Table 2. ODSP Block Implementation

Block	구조	내용
PLC Data Block	생산관련 Data 수집 Module	Lot_ID를 기준으로 관리 Lot_ID를 기준으로 생산수량 등을 실시간으로 처리 생산 수량 중 투입/양품/불량에 대한 정보 처리
	OPC에서의 설비 Control Module	LOT Start 시점과 END 시점을 제어
	설비 상태 정보처리 Module	설비의 상태 변경 시 상태 정보를 처리 설비의 상태 정비에 따라 Alarm Code를 처리
Database Management Block	Data 전송 Module	설비로 LOT ID, Model 정보, End Count 등 Data 전송처리를 수행 DCOM 통신을 이용한 외부 시스템 연계
	DB 관리 모듈	OPC Data를 처리하고 DB 저장 관리 UI Module에서 사용자 화면에서 분석자료 요청시 DB Access 수행
	ODSP Library Module	OPC Data Access 참조 모듈화 OPC Automation Wrapper 참조 모듈화 Component화 된 Class
UI/Diagnostic Block	UI Module	모니터링 용 사용자 관리 UI Data 수집용 사용자 관리 UI 공정 현장에 맞는 UI
	MultiUser Process Module	분산환경 처리 기능 Real Time Process FIFO Process Data Sharing Manager
	Diagnostic Module	자기고장 진단 관리

PLC Data Block은 생산관련 Data 수집 모듈을 포함하여 Lot_ID를 기준으로 관리하도록 구축하며, 생산수량의 실시간 관리, 투입/양품/불량에 대한 정보를 처리하도록 구현한다. OPC에서의 설비 제어 모듈을 구축하며, Lot Start 시점, End 시점을 제어하게 된다. 설비 상태 정보처리 모듈에서는 설비의 상태변경 정보처리와 상태에 따른 Alarm Code를 처리하게 된다.

Database Management Block은 Data 전송모듈, DB 관리 모듈, ODSP Library 모듈로 구현된다. Lot ID, Model 정보, End Count 등 데이터 전송처리를 수행하는 Data 전송 모듈이 있다. OPC Data를 처리하고 DB 저장을 관리하며 사용자 분석자료를 위한 DB Access를 담당하는 DB 관리 모듈이 있으며, OPC Data Access 참조 모듈화 또는 컴포넌트 클래스인 ODSP Library 모듈을 구축한다.

UI/Diagnostic Block은 UI 모듈, MultiUser Process 모듈, Diagnostic 모듈 등으로 나뉘어 구현한다. 모니터링용 사용자 UI와 Data 수집용 사용자관리 UI, 그리고 공정에 맞는 UI를 가지고 있는 UI모듈이 있다. 분산환경 처리, RTP(Real Time Process), FIFO, Data Sharing을 위한 MultiUser Process 모듈과 자가 고장 진단관리를 위한 Diagnostic 모듈을 구현한다.

자가진단을 위한 로직은 생산설비 PLC에서 ODSP 시스템이 고장을 진단하고 확인하는 것이며 다음과 같은 절차를 따르도록 구현한다.

- ① ODSP 프로그램 시작과 동시에 자가진단을 위한 Comm Check 타이머를 작동 시켜 매 초마다 증가시킴
- ② 증가된 Count와 설정해둔 시간과 같은 값인 경우 고장진단을 위한 Bit 정보를 true 값인 1로 하여 ODSP를 통해 생산설비 PLC에 전송함.
- ③ 생산설비 PLC는 ODSP를 통해 받은 Bit가 true 값인 1인지를 지속적으로 확인함과 동시에 ODSP와의 통신에 이상이 없다는 것을 확인

④ 생산설비 PLC에서 확인된 Bit를 false 값인 0으로 변경해 두고 다시 정해진 시간 동안 기다림

⑤ 1,2,3,4번을 반복 수행하므로 생산설비 PLC는 자가 진단을 계속 실행

⑥ 2번 실행에서 고장진단을 위한 Bit 정보를 ODSP를 통해 보내지 못하게 되면 생산설비 PLC는 계속 기다리다가 정의된 시간간격 이상의 시간이 지나면 ODSP와의 통신 이상 발생으로 판단

⑦ 생산설비 PLC에서 알람이 발생하여 작업자가 ODSP와 연결이 끊어진 것을 인지하고 고장으로 판단함.



그림 6. ODSP 진단 모듈
Figure 6. ODSP Diagnostic Modul

공정에 따른 프로세스는 작업지시로부터 시작되며 LOT요청과 발번으로 LOT를 생산설비로 보내고 착공지시에 따라 생산이 이루어지며 착공완료 보고를 OPC 서버에게 전송한다. 이때 Lot ID 값을 생산 Data로 DB에 저장하고 완공 지시를 발생한 다. 이에 따라 완공 완료보고를 OPC 서버에게 전달되어 작업완료처리를 DB에 저장하여 공정 프로세스를 완료한다. 이때 설비에 대한 상태모니터링은 사용자 정의 단위 값으로 정하여 시간 내 응답이 없는 경우는 설비의 이상으로 판단하고 알람을 울려 경고한다. 생산실적에 대한 모니터링도 사용자 정의 단위 값을 사용하여 생산실적을 실시간 모니터링 한다. 이와 같이 생산설비의 작업지시에서부터 작업 완료처리에 이르는 부분을 도식화 하면 <그림 8>과 같다.

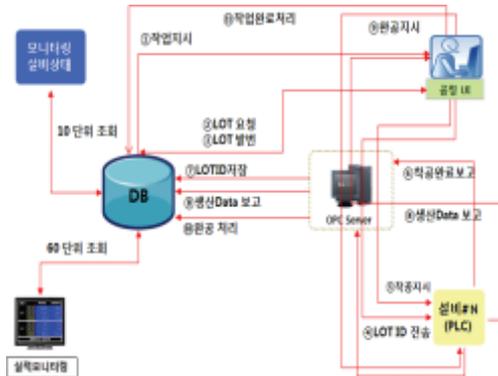


그림 7. 데이터 수집 프로세스
Figure 7. Data Gathering Process

생산설비와의 인터페이스를 위해 생산설비 상태 현황 테이블을 구현하고 이를 기반으로 실시간 생산설비 실적 현황을 수집하여 저장하며 생산설비에 대한 이상여러 코드, Lot 정보 등을 실시간으로 수집하도록 구현한다. 이러한 구현은 <그림 9>와 같이 OPC 서버를 이용하여 ODBC를 통한 Data Gathering을 수행하도록 구성한다.

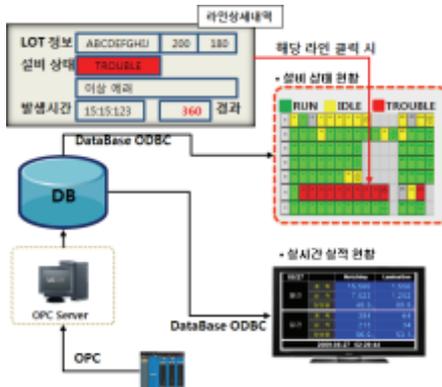


그림 8. 실시간 데이터 수집
Figure 8. Realtime Data Gathering

3.3. ODSP 구현

ODSP의 UI 구현은 <그림 10>과 같이 설비모니

터링, Log, 환경설정, 그래프 모니터링으로 4개의 탭윈도우로 구성한다. 설비 모니터링은 공정코드, 설비코드, 타이머의 리스트가 있으며 생산설비 PLC와의 연결 정보를 보여준다.

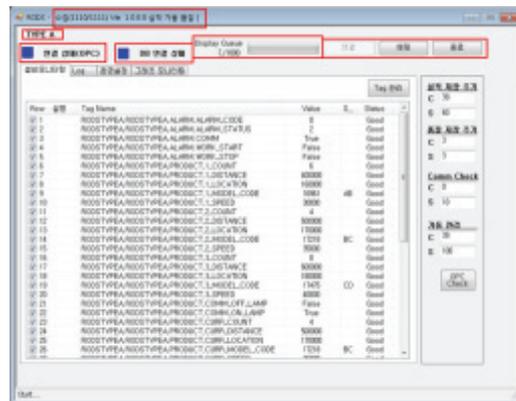


그림 9. 설비 모니터링
Figure 9. Device Monitoring

Log Tab은 현재 설비 PLC의 상태들에 대한 로그를 저장하고 화면에 보여 주기 위한 UI이며 현재 상태에 대한 모든 이슈가 저장되고 윈도우에 표시된다. 그리고 설비 PLC의 움직임이나 Lot의 변경과 정지, 시작 등의 로그 데이터를 화면에 보여준다.

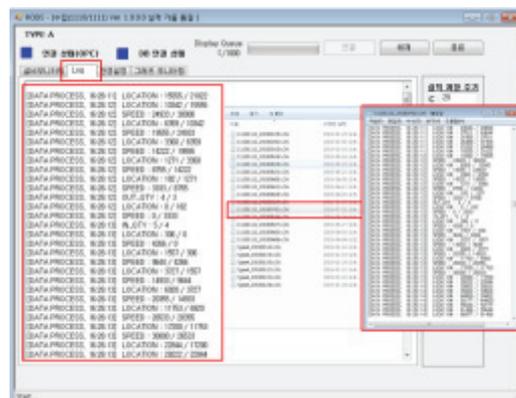


그림 10. 로그 분석기
Figure 10. Log Analyzer

환경설정 Tab은 현재 연결되어 있는 DB 정보와 Display 개수 설정, OPC의 설정, 포트 설정과 타이머를 설정하는 화면이다.

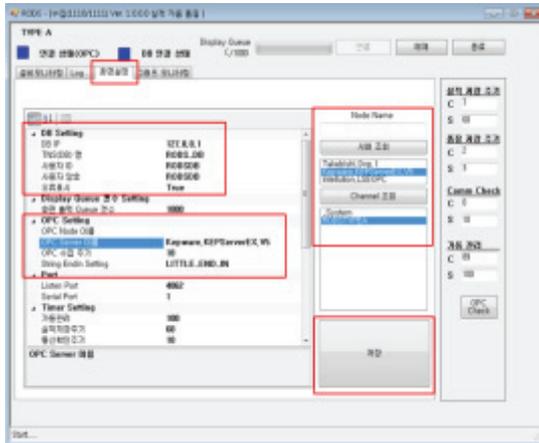


그림 11. PLC 환경 설정
Figure 11. Environment Setting PLC

그래프 모니터링 UI는 <그림 13>과 같이 설정된 시간단위로 PLC의 센서데이터 위치를 화면에 실시간으로 표시되도록 구현되었으며 샘플 PLC의 움직임 거리를 화면에 표시한다.



그림 12. 실시간 그래프 모니터링
Figure 12. Realtime Graph Monitoring

ODSP의 서버 구현은 다음과 같이 서버 리스트, 서버의 아이템을 획득하고, 속성값을 가져오도록 구현하며, 속성값을 읽어 공유하도록 구현하였다. 서로 다른 PLC 데이터를 읽기위해 형식을 찾는 기능과 연결정보, 그리고 서버와의 통신 종료로 프로그램을 구현하였다.

```

Public Function GetServerList(Optional ByVal sNodeName As String = "") As Object
    Dim AnOPCServers As New OPCAutomation.OPCServer

    Me.m_NodeName = sNodeName

    Try
        Return AllOPCServers
    Catch ex As Exception
        RaiseEvent Exception(ex.Message, ex.StackTrace)
        Return Nothing
    End Try
End Function

Public Function GetRootChannelList() As Array

    Try

    Catch ex As Exception
        RaiseEvent Exception(ex.Message, ex.StackTrace)
        Return Nothing
    End Try

End Function

Public Function GetServerItem(ByVal sChannel As String()) As Array

    Try

    Catch ex As Exception
        RaiseEvent Exception(ex.Message, ex.StackTrace)
        Return Nothing
    End Try

End Function

Public Function GetItemProperty(ByVal asItems As String) As Array

    Try

    Return retAry

    Catch
    Catch ex As Exception
        RaiseEvent Exception(ex.Message, ex.StackTrace)
        Return Nothing
    
```

```

    End Try
    End Function

    Public Function GetItemValue(ByVal asItems As
Array) As Array
        Try
            Catch ex As Exception
                RaiseEvent Exception(ex.Message,
ex.StackTrace)
                Return Nothing
            End Try
        End Function

        Public Function GetItemValue(ByVal asItems As
String) As csItem
            Try
                Dim item = New csItem

                Catch ex As Exception
                    RaiseEvent Exception(ex.Message,
ex.StackTrace)
                    Return Nothing
                End Try
            End Function

            Public Function TypeMatch(ByVal aiType As
Integer) As Type
                Select Case aiType

                End Select

                Return GefType(Object)
            End Function

            Public Function Connection(ByVal sServerName As
String, ByVal sNodeName As String) As Boolean

                Return Me.Connection()
            End Function

            Public Function Connection() As Boolean
                ConnectedOPCServer = New
OPCAutomation.OPCServerClass

                Try
                    ConnectedOPCServer.Connect(Me.m_ServerName,
Me.m_NodeName)
                Catch ex As Exception

                End Try

            End Function

            Public Function Disconnect() As Boolean
                Try
                    ConnectedOPCServer.Disconnect()
                    Return True
                Catch ex As Exception

                End Try
            End Function
    
```

4. ODSP 성능 및 실험

ODSP의 개발결과에 따른 성능 시험은 크게 2가지 영역으로 구분한다. 첫째 설비 이기중 PLC 응답시간 처리율, 둘째 다중 Node 처리율이다.

실험 환경 설정을 위해 서로 다른 PLC를 장착한 생산공정 프로토타입을 구성하였으며 <그림 14>와 같이 제작되었다.



그림 13. 서로 다른 PLC 공정
Figure 13. Different PLC Flow

① 이기중 PLC 응답시간 처리율

실험대상으로는 Mitsubishi PLC의 위치 데이터를 기준으로 하고 실험 조건으로는 PLC의 작동으로 생성되는 위치 데이터를 Database에 저장하는 것이다. 그리고 처리되는 데이터는 0.5초 안에 데이터베이스에 저장되도록 구축하여 신뢰성과 신속성이 확보되었다.

표 3. 미쓰비시 PLC 위치 Database
Table 3. PLC Location Database of Mitsubishi

순번	위치 값(mm)	등록 시간
1	359	2015-07-02 16:25:48.423
2	1688	2015-07-02 16:25:48.520
3	4021	2015-07-02 16:25:48.637
4	7349	2015-07-02 16:25:48.750
5	11557	2015-07-02 16:25:48.853
6	16666	2015-07-02 16:25:48.943
7	22088	2015-07-02 16:25:49.057
8	27511	2015-07-02 16:25:49.167
9	32888	2015-07-02 16:25:49.273
10	38266	2015-07-02 16:25:49.383

표 4. L기업 PLC 온도 Database
Table 4. Temperature Database of PLC

순번	온도 값(°C)	등록 시간
1	47	2015-07-02 17:01:46.730
2	48	2015-07-02 17:01:55.873
3	49	2015-07-02 17:02:06.997
4	50	2015-07-02 17:02:21.223
5	51	2015-07-02 17:02:42.500
6	52	2015-07-02 17:03:09.890
7	51	2015-07-02 17:05:57.213
8	52	2015-07-02 17:06:02.267
9	51	2015-07-02 17:06:07.353
10	52	2015-07-02 17:06:09.383

<표 3>의 미쓰비시 PLC 위치 데이터를 기준으로 PLC의 동작 위치 로그파일은 다음과 같다. 그러므로 PLC의 응답시간은 0.3초의 시간 간격이 있었으며 위치데이터 처리율은 100%로 처리되었다.

[DATA PROCESS, 16:25:48]	LOCATION : 359 / 0
[DATA PROCESS, 16:25:48]	LOCATION : 1688 / 359
[DATA PROCESS, 16:25:48]	LOCATION : 4021 / 1688
[DATA PROCESS, 16:25:48]	LOCATION : 7349 / 4021
[DATA PROCESS, 16:25:48]	LOCATION : 11557 / 7349
[DATA PROCESS, 16:25:48]	LOCATION : 16666 / 11557
[DATA PROCESS, 16:25:49]	LOCATION : 22088 / 16666
[DATA PROCESS, 16:25:49]	LOCATION : 27511 / 22088
[DATA PROCESS, 16:25:49]	LOCATION : 32888 / 27511
[DATA PROCESS, 16:25:49]	LOCATION : 38266 / 32888

<표 4>의 L기업 PLC의 데이터 수집에서도 ODSP는 실시간으로 정확하게 수집하여 데이터베이스에

넣는 것을 확인하였다. 그러므로 ODSP는 서로 다른 PLC의 생산공정 실시간 데이터를 추출 후 데이터베이스에 넣는데 성공적이었다.

[DATA PROCESS, 17:01:46]	CURRENT_TEMP : 47 / 46
[DATA PROCESS, 17:01:55]	CURRENT_TEMP : 48 / 47
[DATA PROCESS, 17:02:06]	CURRENT_TEMP : 49 / 48
[DATA PROCESS, 17:02:21]	CURRENT_TEMP : 50 / 49
[DATA PROCESS, 17:02:42]	CURRENT_TEMP : 51 / 50
[DATA PROCESS, 17:03:09]	CURRENT_TEMP : 52 / 51
[DATA PROCESS, 17:05:57]	CURRENT_TEMP : 51 / 52
[DATA PROCESS, 17:06:02]	CURRENT_TEMP : 52 / 51
[DATA PROCESS, 17:06:07]	CURRENT_TEMP : 51 / 52
[DATA PROCESS, 17:06:09]	CURRENT_TEMP : 52 / 51

② 다중 Node 처리율

실험환경과 조건은 다중 Node 처리를 위해 총 4개의 노드를 생성시키고 이에 대하여 ODSP에서 4개의 Node를 동시에 처리되는 것을 확인한다. 확인한 결과는 <그림 15>와 같이 100% 처리율을 확인하였다.

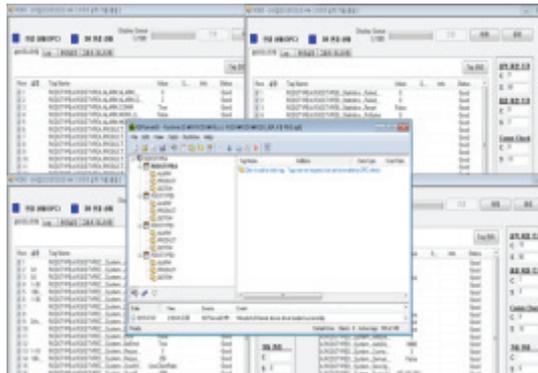


그림 14. 다중 Node 처리
Figure 14. Multi Node Process

5. 결 론

본 논문은 서로 다른 PLC의 생산설비 데이터를 실시간으로 공유하는 ODSP로 설계 및 구현하여 생산공정 데이터를 통합 관리할 수 있도록 하였다.

현재 대부분의 생산 자동화 시스템은 PLC 업체에서 제공하는 DLL을 활용한 하드코딩에 의존하여 설비와 통신 인터페이스 모듈을 개발하고, DLL 종속으로 인해 PLC와 PC간 1:1 통신으로만 가능하게 설계되었다. 본 논문에서 구현한 ODSP는 타 시스템 또는 여러 명의 End User에게 서로 다른 PLC 데이터에 대한 실시간 정보 제공이 불가능한 문제를 해결하였다. 즉 실시간 서로 다른 PLC의 OPC 데이터 공유 플랫폼을 통해 데이터를 실시간으로 수집 및 저장하며, 타 시스템과의 연계를 통해 다양한 정보로 활용할 수 있도록 구현하였다. 추가적으로 자동화 설비를 보유하고 운영 중인 수출 기업에서 FTA 체결을 통해 수입 상대국의 원산지 증명을 위한 생산 시험성적서 제출에 적용할 수 있다. ODSP는 생산설비의 OPC 서버가 가지고 있는 실시간 PLC 데이터를 수집하는 라이브러리를 활용하도록 구현하였으며, 서로 다른 생산공정의 MES, POP에서 생성되는 다양한 실시간 PLC 데이터를 수집하고 활용하여 실시간 모니터링 그래프, 생산관리 모니터링 및 설비 관리, 자가진단을 가능케 한다. 따라서 ODSP는 생산관리, 공정관리, PLC 데이터공유 등에 필요한 수준별·분야별로 실시간 제조운영관리 시스템의 확대 보급이 가능하고, 다양한 생산설비의 실시간 제어가 가능함을 보여주고 있다.

References

- [1] Jai-Kyung Lee, Seung Woo Lee, So Jeong Nam, and Jong Kweon Park, *Acquisition of data of equipments on shop floor using interface between various equipments*, The Journal of Korean Society of Mechanical Engineers, Vol. A, No. 35-2, pp. 149-156, 2011.
- [2] Hwa Sub Song, Hyun Rak Jung, and Seung Ho Jeong, *A review of MES implementation methodology and its representative cases*, SAMSUNG SDS Consultin Review, No. 1, pp. 103-121, 2007.
- [3] MESA, *MES functionalities & MRP to MES data flow possibilities*, Mesa International Whitepaper, No. 2, pp. 74~103, 1997.
- [4] Phyoung-Jung Kim, Yeong-gu Oh, and Dal-jung Kim, *A study on real-time sharing platform of PLC data*, Korea Information Processing Society, Vol. 21, No. 2, pp. 1177-1180, 11. 2014.
- [5] <http://www.opcfoundation.org>
- [6] Suk Keun Cha, Jae Young Yoon, Jeong Ki Hong, Hyun Gu Kang, and Hyeon Cha Cho, *The system architecture and standardzation of production IT convergence for smart factory*, Jonrnal of KSPE Vol. 32, No. 1, pp. 17-24, 2015.
- [7] Lee Sungwoo, Lee Jeakyung, Nam sojung, and Park Jongkyung, *The application of the field data collection system for the implementation of MES sample*, The Journal of Korean Society of Mechanical Engineers Vol. A, No. 35-9, pp. 1063-1070, 2011.
- [8] Jin Sukmyung, Seo Sangki, Yang Galam, and Wang Jinam, *Process fault detection and diagnosis system for PLC manufacturing*, Korea Institute of Industrial Engineers in the Spring, pp. 909-915, 2012.
- [9] Ko Minsuk, Park Sangchul, and Wang Jinam, *PLC program validation framework for Automated Manufacturing Line using validated devices data*, Korea Institute of Industrial Engineers, pp. 67-72, 2007

- [10] Kim, Sung Hak, *A study on effective test methods for semiconductor MES system project*, Hanyang Univ., 2014.

생산공정의 서로 다른 PLC 데이터의 ODSP 설계 및 구현에 관한 연구

김달중¹, 김평중², 하수철¹

¹대전대학교 컴퓨터공학과

²충북도립대학교 컴퓨터융합공학과

요 약

본 논문은 생산설비 중 서로 다른 PLC(Programmable Logic Controller) 데이터의 OPC 공유 플랫폼(ODSP : OPC Data Sharing Platform)을 설계 및 구현에 대한 논의이다. 이 플랫폼은 자동화 설비를 보유하고 운영 중인 수출집약 기업에서 원산지증명을 위해 사용되는 품질데이터, 시험성적성 등을 FTA(Free Trade Agreement) 체결 수입 상대국에 제출할 수 있도록 시스템으로 제공할 수 있다. 또 생산설비의 OPC 서버가 갖는 실시간 데이터를 수집하는 ODSP를 활용하여 MES(Manufacturing Execution System)에서 생성되는 다양한 실시간 데이터로 실시간 공정 그래프, 생산관리 모니터링 및 설비 관리와 생산공정에 대한 자가진단이 가능하게 한다. 따라서 ODSP는 중소기업이 생산관리, 공정관리, PLC 데이터공유 등에 필요한 수준별·분야별로 실시간 제조운영관리 시스템의 확대 보급이 가능하고, 생산설비의 실시간 제어가 가능함을 보여 주었다.



Dal-Joong Kim received the bachelor's degree in the Department of Computer Engineering from the Daejeon University in 1997. He received the M.S. degree in the Department of Computer Engineering from the Daejeon University in 1999. He is working in Sayinfo Company. His current research interests include MES, Card Print, S/W Engineering. He is a member of the KKITS.

E-mail address: djkim@sayinfo.co.kr



Phyoung-Jung Kim received the bachelor's degree in the Department of Computer Science and Statistic from the Chungnam National University in 1985. He received the M.S. degree in the Department of Computer Science from the Korea Advanced Institute of Science and Technology (KAIST) in 1995. He received the Ph.D. degree in the Department of Computer Engineering from the Chungnam National University in 2000. From 1988 to 1998, he was a Senior researcher at Electronics and Telecommunications Technology Institute (ETRI). He has been a professor in the Department of Computer Integrated Engineering at Chungbuk Provincial College since 1998. His current research interests include computer network, multimedia middleware platform, mobile agents.

E-mail address: pjkim@cpcu.ac.kr



Soo-Cheol Ha is the dean of College of Engineering and the head of Multimedia Content Technology Center. He has been a professor in the Department of Computer

Engineering at the Daejeon University since 1987. He received the bachelor's degree, the M.S. and Ph.D. degree in the Department of Computer Engineering from the HongIk University. He was a visiting professor of the Florida State University and the University of Texas Arlington, and a visiting scholar of the Washington State University. He was a vice president of Korea Game Society. His current research areas include software engineering, game engineering, multimedia application, and object-oriented engineering.

E-mail address: soocha@dju.kr