

# TM/TC를 탑재한 분산제어 시스템의 실시간 제어기 설계와 구현

## Design and Implementation of Real-Time Controllers of an Enhanced Distributed Control System with Embedded TM/TC

金 珍 泰\*, 朴 仁 甲\*\*  
( Jin-Tae Kim\* and In-Kap Park\*\* )

### 요 약

DCS는 분산된 컴퓨터 네트워크를 이용한 통합된 EIC의 다양한 응용기술을 적용할 수 있는 원격 감시 제어가 요구되고 있다. 현재까지는 DCS는 원격 사이트의 플랜트에 대해 실시간성 보장은 기대에 미치지 못한다.

본 논문에서는 효율적인 TM/TC 시스템을 위한 각 제어기를 기존의 PRS 하드웨어와 GWS, RPS, SRS로 광역화에 맞도록 설계하였으며, 또한 각 구현된 제어기의 실시간을 만족하는 통신처리 시간, 시스템 부하율을 분석하여 타당성을 검토하였다.

### Abstract

DCS require a remote supervisory control that applies a various applications with a period of integrated EIC(electric instrument control) using distributed computer network. Until now, DCS has leaved something to be desired in guaranteeing real-time required in remote site plant.

In this paper, each controller is designed by implementing the controller suiting with an wide area to enhance the hardware of existing PRS controller and GWS, RPS and SRS, for the efficiency of TM/TC system, Therefore, real-time emphasizes on the communication processing time, rate of system load analyzed and validity was evaluated for a each implemented controllers.

Keyword : DCS, TM/TC, GWS, RPS, SRS

### I. 서 론

현재 산업사회 발달로 인하여 빠른 처리속도와 입출력의 일괄처리가 요구되는 공정자동화와 제어시스템분야에서는 1990년대 근거리 통신망(LAN)의 보급, 응용과 자료의 투명성 제공을 통하여 어려운 여건에서도 플랜트 공정역할을 담당하는 자동화 분야의 핵심 산업으로 중요하게 자리

\* (주)네오게이트

(NeoGate Co., LTD)

\*\* 建國大學校 電子工學科

(Dept. of Electronic Eng., Konkuk Univ.)

接受日: 1999年3月10日, 修正完了日: 1999年6月3日

잡을 수 있는 시스템은 분산제어 시스템이다[1-5]. 현재의 자동제어 시스템의 형태는 크게 SCADA (supervisory control and data acquisition) 시스템[6] 과 분산제어 시스템으로 나눌 수 있다.

SCADA 시스템은 광역화에 따른 중앙집중 감시제어 형태로 단순 감시제어에 많이 적용되고, 분산제어 시스템의 경우에는 한정된 단위 지역내의 공정제어 분산처리 형태로서 고기능의 플랜트 공정 감시제어에 주로 적용되고 있다[7]. 그러나 현재 SCADA 시스템의 기능화된 현장제어장치를 통한 분산제어 요구와 분산제어 시스템에서의 광역화에 따른 실시간 적용 요구 등으로 두 시스템 간의 구분이 없어지고 있다. 이러한 면에서 분산제어 시스템은 광역화에 적용하고자 할 때, 공정처리에서 요구되는 실시간성을 보장하기에는 미흡한 점이 많아 시스템 구축 시 어려움을 가중시키고 있었다. 따라서, 고속 응답과 기능 수행을 위해서는 실시간 처리능력의 제어기 구현을 통하여 효율적인 통신과 제어이론을 적용할 수 있는 TM/TC용 분산제어 시스템의 개발이 필요하다.

본 논문에서는 이러한 광역화 시스템 구축을 위한 실시간 제어기 설계와 구현을 통하여 기존의 시스템에 신뢰를 더할 수 있는 시스템을 구현하였다[7][8].

II. TM/TC를 탑재한 실시간 제어기 구현

TM/TC는 최근에 데이터 통신의 급속한 기술 진보에 편승하여 설비의 신뢰성, 실시간 제어, 인력 절감 효과를 얻을 수 있기 때문에 각종 산업현장에서 활용 범위를 넓히고 있다[7][8].

본 논문의 TM/TC용 분산제어 시스템 구성은 상위 시스템과 제어기로 나눈다. 분산된 노드인 제어기의 결과값을 보기 위한 그래픽 환경의 중앙 감시시스템인 POS(process operating system)는 상위 시스템이고, 직접 제어를 담당하는 PRS(process remote system), RPS(remote PRS), GWS(gateway system) 또는 SRS(small remote system)등은 하위 시스템인 제어기이다. 플랜트 공정의 루프(loop)나 로직(logic) 시퀀스

를 처리하는 프로세스 장치는 PRS이고, 이 기종의 PLC(programmable logic control), SRS, RPS 또는 하위 GWS를 직렬통신을 통하여 데이터 수집용 중계장치는 GWS이다. 직렬과 LAN 통신을 이용하여 데이터를 처리하면서 하위에 I/O 포인트를 거느릴 수 있는 원격 프로세스 장치는 RPS이며, 원격지에서 상위의 GWS나 RPS와 직렬통신으로, 고속통신을 이용한 이 기종 시스템과는 HDLC(high-level data link control) 통신을 통해 감시 및 제어 데이터를 처리할 수 있는 소규모 원격 감시장치는 SRS이다. 그림 2-1은 구현한 TM/TC용 분산제어 시스템의 전체 구성도를 나타내고 있다.

본 연구대상의 TM/TC용 분산제어 시스템은 다양한 기능을 갖춘 시스템으로 분산구조, 계층구조, 모듈구조 및 실시간 처리의 특징을 갖도록 하여, 집중화된 제어를 처리하는 방식에 탈피하여 광역 시스템에 대한 원격 감시 및 제어를 관장할 수 있는 시스템으로, 기존의 집중된 제어에서 중간 제어를 통한 부하 분담 처리와 이에 결과를 중앙의 POS로 전달하여 전체공정을 하나의 제어 시스템에서 처리하는 것과 같은 결과를 낼 수 있도록 하는 통합 시스템을 구축할 수 있다.

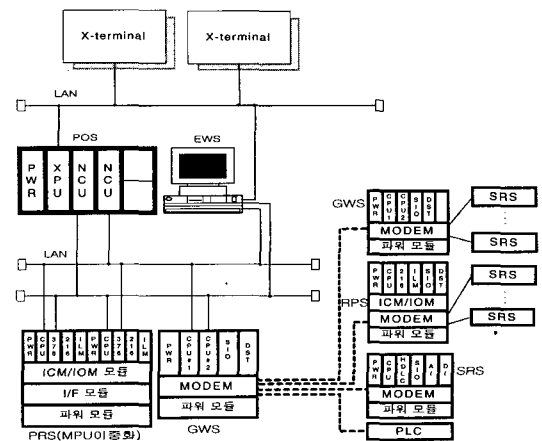


그림 2-1. TM/TC용 분산제어 시스템의 전체구성도

Fig. 2-1. Whole configuration block diagram of TM/TC DCS.

2.1 GWS 제어기 구현

본 논문에서의 GWS는 TM/TC용 분산제어 시스템에서 신뢰성을 보장하면서 하위 제어기와 다기능의 이 기종 시스템과의 통신을 관리하는 제어기이다. GWS는 SCADA의 통신제어장치(CCU) 또는 FEP의 원거리 데이터 수집기능을 보유하여 GWS, RPS, SRS, PLC등의 제어기와 직렬 통신을 이용하여 데이터를 전송하거나, 수집하는 시스템 및 통신관리 중계기용 제어기이다.

GWS의 하드웨어는 PRS와 RPS처럼 포인트 처리에 따른 감시와 제어 기능을 가지고 있지 않아서 특별한 포인트 저장의 목적인 메모리 카드를 사용하지 않는다. GWS는 I/O 포인트를 거느리고 있지 않기 때문에 브로드캐스트 영역내의 BIO, WIO영역을 설정할 필요가 없어 메모리 카드를 사용을 하지 않고, CPU 카드 내에 있는 자기 자신의 4Mbytes 메모리 일부를 할당하여 데이터 처리 및 다운로드된 데이터베이스를 저장할 수 있도록 하였으며, CPU는 이중화 구성과 통신 포트를 열어 주는 SIO(serial input output) 카드 그리고 포트 확장을 위한 DST(distributed terminal)을 두어 DST에서 나온 신호선은 전용회선 모뎀(MODEM)이나 직접 연결을 통하여 시스템을 구성할 수 있도록 하였다[9]. 그림 2-2는 GWS의 하드웨어 구성을 나타내고 있으며 GWS가 통신을 통한 데이터 취득이 가능한 제어기는 GWS, RPS, SRS, PLC이다.

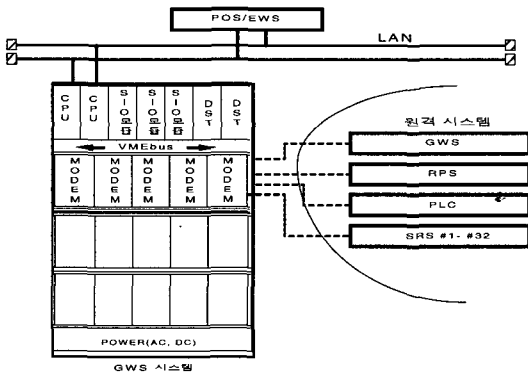


그림 2-2. GWS의 하드웨어 구성

Fig. 2-2. Hardware block diagram for GWS.

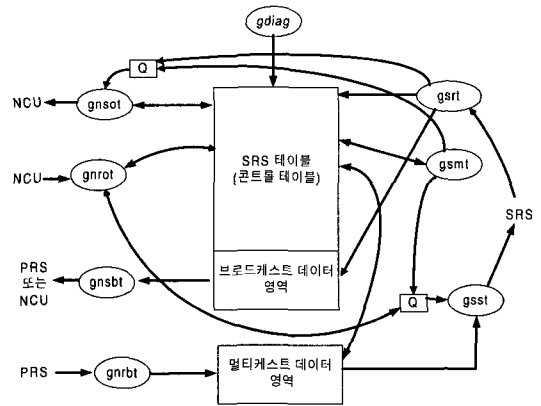


그림 2-3. GWS의 소프트웨어 TASK 구성

Fig. 2-3. Software task block diagram for GWS.

소프트웨어 구성은 PRS와 TASK 공유가 가능하면서 직렬통신을 관장할 수 있도록 새로운 TASK로 구성하였다. 먼저 PRS에서 기능 개선된 TASK는 gdiag, gnrbt, gnsbt, gnrot, gnsot들로 시스템진단 전용 TASK와 LAN과의 데이터 교환을 위한 TASK로 구성되며, 직렬통신을 위한 TASK는 gsst, gsrt, gsmt로 구성된다. TASK는 PRS, RPS와 공유하여 소프트웨어 갱신 시에는 전체가 같이 갱신할 수 있도록 하였다.

gsst TASK는 NVRAM영역을 참고로 자기 자신이 거느릴 SIO카드의 수를 파악하여 데이터 전송기능을 처리하며, gsrt TASK는 요구에 대한 응답 데이터 수신 처리를 담당하며, gsmt TASK는 데이터 송수신의 관련된 제전송과 SRS 테이블을 관리하는 기능을 갖는다. GWS의 TASK 구성은 그림 2-3에 나타내고 있다.

시스템 진단은 gdiag TASK에서 처리하면서, 직렬통신을 위한 SIO 카드의 상태를 점검과 여러 발생 시에 메모리 셋을 실시할 수 있도록 하였다. 또한 시스템 부하율을 계산하여 제어기의 안정된 운영상태를 주기적으로 확인할 수 있도록 하였다. MMI나 POSM의 제어를 RPS나 SRS로의 중계 역할을 하는 GWS는 먼저 상위 시스템으로부터 제어 명령을 받은 gnrot TASK는 명령을 분석하여

SRS 테이블에 등록여부 및 사용 가능여부를 결정하여, 미등록 시에는 SRS 테이블에 등록을 한다. 이때 GWS를 통해 SRS로 전송하는 데이터는 SBR 버퍼에 있는 *sbr\_list*에 등록하게 되며, GWS가 데이터 취득이 가능한 최대 시스템 수는 본 논문에서는 32대로 시스템을 구성하였다. 여기에 따른 멀티캐스트 영역의 *sbr\_list* 구조는 그림 2-4에 나타나고 있다.

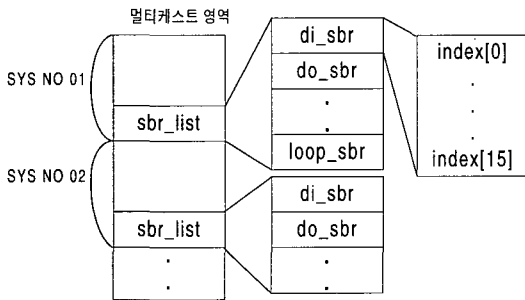


그림 2-4. 멀티캐스트 영역의 *sbr\_list* 구조  
Fig. 2-4. Structure of *sbr\_list* in multicast region.

*sbr\_list*의 구조는 각 시스템의 멀티캐스트 영역 내에 *sbr\_list*를 두고 있으며, *sbr\_list*안에는 각 포인트의 *sbr*이 존재하게 된다. 각 포인트 *sbr*은 16개의 인덱스를 두고 이 리스트에는 이미 데이터베이스에 정의된 수만큼을 이용하여 전송 데이터의 최대 크기인 1Kbytes를 넘지 않도록 하여 전송한다.

2.2 RPS 제어기 구현

본 논문의 RPS는 신뢰성을 보장하도록 현장 데이터 수집과 통신 처리 및 관리의 제어기로 상위의 GWS와 하위의 SRS를 동시에 직렬통신을 통한 데이터 전송 및 수집 기능을 수행하면서, LAN을 이용하여 LOS(local operation system)와 실시간으로 데이터를 처리한다. RPS 시스템의 성능평가는 루프 처리와 통신처리 시 RPS에 걸리는 시스템 부하율을 통한 평가와 POS-GWS-RPS간의 전체 데이터 교환의 실험을 통한 실시간 처리에 관하여 평가하였다.

하드웨어 구성은 입출력 포인트 수집을 위해 ILM(i/o link module) 카드와 VME버스를 통하여 수집된 데이터를 주기적으로 메모리 카드에 저장하며, ILM카드는 최대 12개의 ICM(i/o control module)카드와 고속 통신을 통하여 local PV(present value)값을 수집하고, ICM 카드는 각각의 12개의 IOM(input/output module) 카드에서 A/D 변환된 데이터를 수집하여 ILM 카드로 전송한다 [9].

CPU는 시스템의 부하 증가와 기존의 PRS보다 많은 타스크가 늘어남을 고려하여 시스템 메모리를 8Mbytes로 사용하였으며, 자기 자신의 제어 및 감시 시스템인 LOS와 LAN을 통하여 브로드캐스트 전송을 통한 데이터 처리 및 일치가 되도록 하였다. 따라서 다른 제어기와 비교할 때 시스템 부하는 높지만 기능의 다양화를 통한 제어를 처리할 수 있도록 설계하였다. 그림 2-5는 RPS의 하드웨어 구성으로 ILM, ICM, 각종 IOM 카드는 기존 분산제어 시스템의 PRS 제어기의 카드를 사용하였다. 수집된 데이터 처리를 위한 RPS의 소프트웨어 타스크 구성은 그림 2-6과 같으며, PRS에서 I/O 처리를 위해 기능이 개선된 타스크, LOS와 LAN 통신을 위한 타스크, 하위의 SRS와 통신 시 상위 시스템이 되어 데이터 요구를 보유할 수 있는 타스크와 GWS와 통신 시에는 하위 시스템이 되어 데이터 응답을 하는 통신 타스크로 구성하여 소프트웨어를 설계하였다.

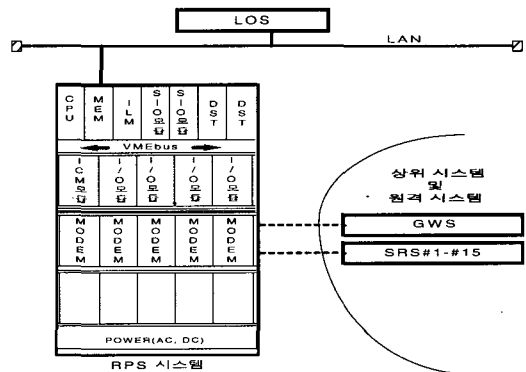


그림 2-5. RPS의 하드웨어 구성  
Fig. 2-5. Hardware block diagram for RPS.

타스크 구성을 보면 nsot, nrot, nsbt, nrbt 타스크들은 상위의 LOS와 통신을 유지하여 제어명령의 수신, 브로드캐스트 요구 및 이에 대한 응답을 처리하는 타스크이다. ssot, srot 타스크는 상위 GWS의 디스플레이 요구나 제어 요구에 대한 응답을 처리하는 타스크이며, sst, srt, smt 타스크는 하위의 SRS를 거느렸을 때 데이터 요구 및 제어 명령을 전송하는 타스크로 나뉘어져 있다. 따라서, 시스템은 POS-GWS-RPS, POS-GWS-RPS-SRS, LOS-RPS, LOS-RPS-SRS로 구성하여 제어를 전달하는 기능을 가진다.

그림 2-7은 데이터 요구 시 RPS의 예외처리 구조를 나타내고 있으며, diag 타스크에서는 new 버퍼와 old 버퍼를 각각 만들어 자신의 이전에 수집된 데이터를 old 버퍼에 저장하고, 새로운 데이터를 new 버퍼에 넣어 비교 후 갱신된 데이터 부분만을 예외버퍼에 저장하고, 정주기로 예외(exception) 데이터 요구 시에만 ssot 타스크를 통하여 주기적으로 상위의 GWS로 전송하며, 각 타스크간의 데이터 교환을 위한 메시지 큐를 생성하고, 예외 데이터 처리를 위한 EXCT 공유메모리를 할당과 SIO의 지연시간을 처리하는 DELA 공유메모리를 할당한다.

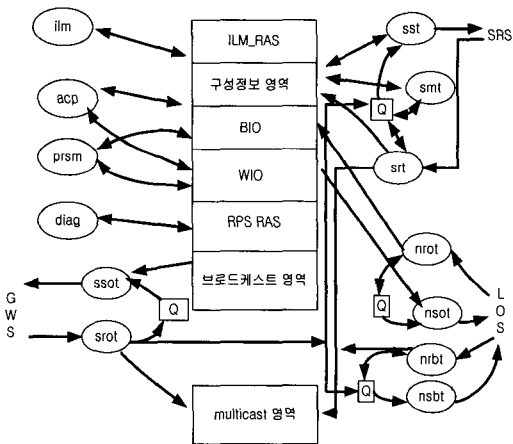


그림 2-6. RPS의 소프트웨어 타스크 구성

Fig. 2-6. Software task block diagram for RPS.

### 2.3 SRS 제어기 구현

SRS는 연속제어, 시퀀스 제어 등의 고급제어 기능을 내장하여 폭넓은 제어를 할 수 있는 제어기로, 기존의 LAN 통신을 이용한 한정된 지역에 대한 감시·제어를 광역화로 확장시킬 수 있는 제어기이다. 구성은 CPU 카드, GWS, RPS 등 상위 시스템과의 연결을 통한 원활한 감시제어를 위해 이중화된 통신카드, 고속의 HDLC(high-level data link control) 통신카드, 제어연산카드로 구성되어 있다. CPU 모듈은 상위 시스템에서 전송되어 온 데이터를 분석하여 관련 모듈을 제어 및 감시하고 복수의 통신 프로토콜을 갖추고 있어 다양한 이 기종 시스템에 연결하여 사용할 수 있도록 설계하였다. GWS와 RPS와의 통신 시스템 구성 시에는 이중화 지원이 가능하도록 하였고, CPU가 전체 모듈을 관장하여 DI, AI와 직렬 입출력 모듈로 들어오는 정보를 입력처리 후, 다시 직렬포트를 이용하여 GWS나 RPS로 전송하거나 HDLC 모듈과 E1급의 전용선을 이용하여 이 기종 시스템에서 모니터링하게 해 주며, GWS, RPS 또는 이 기종의 상위 시스템에서 보내는 제어 명령들은 직렬 출력포트와 DO, AO 출력모듈로 출력이 처리되도록 하였다.

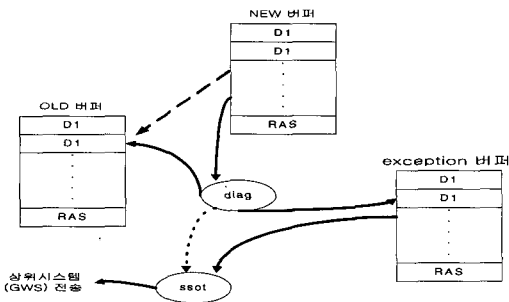


그림 2-7. RPS의 예외처리 구조

Fig. 2-7. Exception procedure structure of RPS.

2.3.1 SRS 하드웨어 구현

SRS 제어기의 설계 시 높은 신뢰성의 데이터 처리를 위해 CPU의 메인칩은 모토로라의 32비트인 MC68340을 채택하여 기존의 16비트 제어카드와 비교하여 32비트의 처리능력은 상당한 기능개선을 제공하였으며, CPU내 여러 단위 모듈기능들을 포함한 하드웨어 구성과 저렴한 가격으로 효율적인 제어기를 구성할 수 있다는 장점을 가지고 있다.

이에 따른 SRS의 CPU 내부 블록도는 그림 2-8에 나타내고 있으며, 시스템은 원격 다운로드를 통하여 기존의 원격 유지보수 시스템의 기술을 한층 향상시키기 위하여 플래시메모리 사용과 2Mbytes의 메모리를 사용하였다.

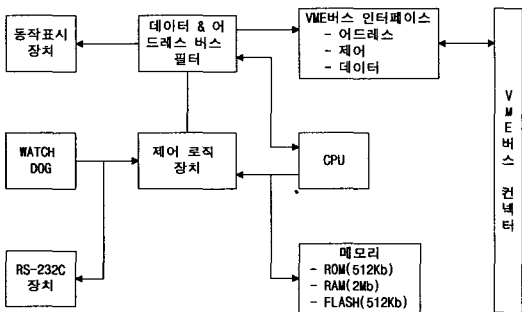


그림 2-8. SRS의 CPU 내부 블록도

Fig. 2-8. Internal block diagram of CPU for SRS.

그림 2-9는 SRS의 기능 블록도를 나타내고 있다. 이는 이 기종 시스템이나 GWS, RPS와 연계한 SRS의 데이터의 흐름을 보여주는 것으로 상위의 제어기와 HDLC 통신카드를 통한 E1급(2.048Mbps) 속도까지 통신이 가능하며, SIO의 2개 포트는 GWS나 RPS와 전용회선을 이용한 RS-232C 통신으로 이중화가 가능하며, 나머지 2개 포트로는 계측장비로부터 데이터를 직접 받아서 처리할 수 있도록 하였다. 메모리는 상위 제어기로부터 내려오는 프로그램을 플래시메모리로 다운로드하여 CPU의 시작 동작을 ROM에서 플래시메모리로 변경할 수 있도록 하였다.

2.3.2 SRS 소프트웨어 구현

SRS의 운영체제는 실시간 다중 프로세스 운영체제(real time operating system)를 사용함으로써 동시에 각 프로세스를 구동시킬 수 있으며, 운영체제를 이용한 제어 프로그램은 부팅과 동시에 필요한 프로세스를 가동시킨다. TASK 구성은 그림 2-10에 나타내고 있으며, 통신은 SRS 자체에서 일방적인 브로드캐스트 방식의 데이터 전송이 이루어지는 것이 아니라 상위의 GWS나 RPS 또는 이 기종의 FEP(front end processor)의 폴링 요구에 의해서만 데이터 전송을 한다. 하지만 통신 이상시는 로직 제어를 통한 단독으로 자동프로그램을 구동시킬 수 있도록 하였다. 각 TASK의 구성은 sio(SRS I/O)TASK는 GWS 또는 RPS를 통한 제어 명령을 받아 수행하고 그 결과를 되돌려주는 TASK이고, sdiag(SRS diagnostic) TASK는 SRS의 상태를 진단하는 기능을 수행하며, sacp(SRS ACP)는 SRS에서 동작하는 ACP(advanced control point) 구동용 TASK로서 PRS보다 포인트 규모가 작으며 태그 개념이 없고 사용 가능한 스텝수도 적으나, SRS의 자동제어 프로그램을 담당한다.

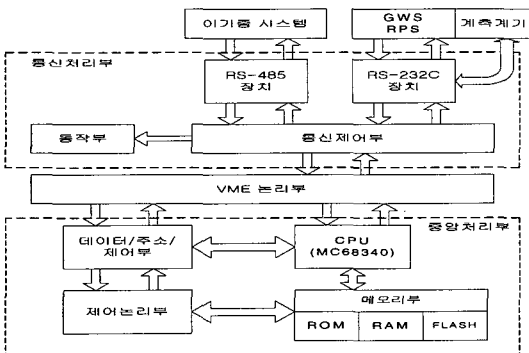


그림 2-9. SRS의 기능 블록도

Fig. 2-9. Function block diagram for SRS.

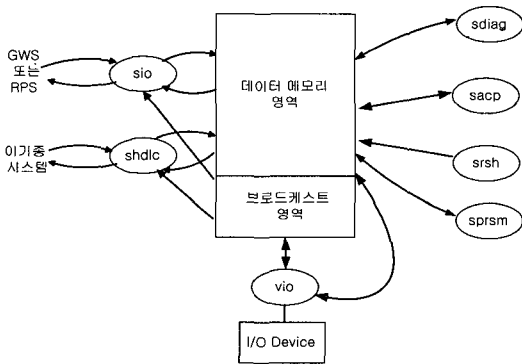


그림 2-10. SRS의 소프트웨어 구성

Fig. 2-10. Software block diagram for SRS.

GWS나 RPS로부터의 운영에 필요한 스크립트를 다운로드 받아 PRS와 RPS의 ACP 자동 프로그램과 동일한 과정을 수행하며, 각 프로그램 스텝은 6개의 스크립트로 구성되어지며 각 스텝은 서로 연결되어 1순회 주기 동안 수 개의 스텝이 수행된다. sprsm은 SRS의 기본동작 DI, DO, AI, AO, LOOP 등의 감시, 제어 기능을 수행하는 타스크이다. vio(virtual I/O)타스크는 SRS내의 I/O 모듈과의 통신을 담당하며 기존 PRS의 ILM 타스크 + ILM 카드 + ICM 카드의 기능을 수행하며, VME버스를 통하여 입출력 카드와의 입력과 출력을 관리한다.

shdlc(high-level data link control) 타스크는 SRS의 데이터를 고속통신의 56Kbps, 64Kbps, 1Mbps, 2Mbps로 전송하는 기능을 수행한다. 위에 나열된 각 타스크 운영프로그램들은 설치 후 재구성하고자 할 경우에는 ROM에 의한 프로그램 교체가 아니라 플래시메모리에 의한 프로그램 다운로드를 통하여, 재부팅 시 다운로드된 운영프로그램으로 작동되도록 하였다.

### III. 실험 및 고찰

본 논문에서는 광역화 적용에 맞추어 TM/TC를 탑재한 분산제어 시스템의 감시·제어기능, 통신의 신뢰성을 보장하는 실시간 제어기의 하드웨어와 소프트웨어에 대해서 설계와 구현을 하였다.

제어기 구현에 따라 POS 소프트웨어를 개선하여 시스템 성능을 향상시켰으며, 제어기 구동 소프트웨어는 GWS, RPS, SRS로 나누어 타스크 설계를 통하여 TM/TC 환경에 맞추어 분산제어 시스템과 SCADA 시스템의 통합화가 되도록 하였다.

#### 3.1 GWS의 성능평가

GWS는 광역화 요구와 실시간 통신처리를 만족할 수 있는 제어기로서, 기능은 POS와 RPS나 SRS의 통신 및 제어의 중계역할을 담당하며, PRS의 브로드캐스트 데이터를 하위의 제어기로 전송할 수 있도록 하였다. 이러한 광역화에 따른 데이터의 빠른 응답처리를 위하여 GWS의 예외처리 방식을 사용하여 성능 분석하였다. 성능 분석을 위해 GWS는 30대의 SRS와 2대의 RPS를 직렬통신을 이용해야 운영할 수 있도록 하여 데이터 요구는 모든 시스템에게 균등하게 요구되도록 하였다.

표 3-1의 예외처리에 의한 데이터 전송 에러율로 SRS와 RPS에 대한 GWS의 평균 통신에러율은 3.85%가 되고 이는 정확히 데이터를 전송 후 수신할 수 있는 확률을 96.15%로 추정할 수 있다. 이 데이터는 GWS가 얼마나 정확히 SRS나 RPS의 데이터를 수집하면서 LAN을 이용한 PRS나 POS의 전송여부와 각 PRS의 sbr 비트 세팅된 데이터를 SRS 멀티캐스트 영역에 저장하여 각 제어의 전송이나 디스플레이 요구를 처리한 경우의 실험 데이터다.

이와 같은 예외처리에 따른 에러 복구 기능의 타당성을 고려하면, GWS의 데이터 요구 응답 처리 분석과 에러복구 기능을 위해 데이터를 정확하게 시스템에서 인식할 확률을 P라 하면, 정확하게 전송될 확률의 합  $D_i$ 는  $D_i = i \cdot \dots \cdot N$ 으로써 표시될 수 있다. 이러한  $i$ 의 확률은 식 3-1과 같다.

$$P^i \times (1-P)^{(N-i)} \tag{3-1}$$

이때 GWS에서 정확하게 인식할 평균 시스템 활용성을  $A_r$ 이라 놓으면 식 3-2와 같이 된다.

표 3-1. 데이터 통신에러율

Table 3-1. Error rate of data transmission.

SIO를 통한 총 데이터 전송(횟수)	SRS 송신/수신 (횟수)	통신 에러율	RPS 송신/수신 (횟수)	통신 에러율
16,000	15,000/14,556	2.96 %	1,000/966	3.40 %
24,000	22,500/21,907	2.64 %	1,500/1,502	6.53 %
32,000	30,000/28,812	3.96 %	2,000/1,897	5.15 %
40,000	37,500/36,120	3.68 %	2,500/2,350	6.00 %
48,000	45,000/43,893	2.46 %	3,000/2,809	6.37 %
56,000	52,500/50,343	4.11 %	3,500/3,216	8.11 %
64,000	60,000/57,197	4.67 %	4,000/3,765	5.88 %
평균 에러율	252,828/262,500	3.49 %	17,500/16,405	5.92 %

$$A_{tr} = \sum_{i=0}^N C(N, i) \times (P^i \times (1-P)^{(N-i)}) \tag{3-2}$$

식 3-3에서 정확하게 인식될 확률 C(N, i)는

$$NC_i = \frac{N!}{i!} \text{ 와 같다.}$$

여기서 N은 총 데이터 전송(send) 횟수이고 i는 정확하게 인식된 데이터이다. 이때, 본 논문에서 데이터 인식을 위해 활용되는 알고리즘 이용시 처리되는 부가적인 시간을 t라 표시하면 시스템 활용성을 위한 확률 P' = P-t가 되고 정확한 데이터로 인식되지 않는 확률은 1-P' = 1-P+t가 된다.

따라서 본 논문에서 처리한 GWS의 예외처리 알고리즘 적용 후 활용성을 위한 확률 A<sub>tr</sub>'는 식 3-3과 같다.

$$A_{tr}' = \sum_{i=0}^N C(N, i) \times (P'^i \times (1-P')^{(N-i)}) \tag{3-3}$$

위의 식 3-3에서 총 데이터 전달횟수 N=5, 10, 20일 때 정확하게 데이터를 수신한 후 데이터 인식을 위해 알고리즘을 이용한 부가적인 에러확률 t를 P의 1%로 작용하고, 시스템의 신뢰성을 80%와 90%일 때를 고려하여 시스템의 활용성을 보면 식 3-2를 이용하여 계산한 경우는 신뢰성이 80%인

경우에는 N=5, 10, 20일 때 각각 73.7%, 67.8%, 63.0%로 처리된다. 신뢰성 90%에서 N=5, 10, 20일 때 각각 91.9%, 73.7%, 67.8%로 계산된다. 이 계산에서 예외처리 알고리즘을 적용 시에는 기존의 데이터 처리 알고리즘의 신뢰성과 비교를 위해서, 식 3-3을 이용하여 시스템의 신뢰성을 90%이고 t를 P의 1%로 계산된 활용성은 전송횟수 N=5, 10, 20인 경우는 90.6%, 70.1%, 62.6%이지만, 본 논문에서의 예외처리를 통한 활용성의 실험치는 t를 P의 1%일 때, 전송횟수 N=5, 10, 20을 10,000번 실행한 경우는 97.1%, 96.7%, 94.9%로 평균 96.2%의 실험치를 얻었다. 이 결과는 GWS에서 사용하고 있는 예외처리 알고리즘을 이용한 GWS와 SRS, RPS의 통신 시에는 에러 발생에 따른 효율적인 에러 복구기능을 가지면서 데이터 수집이 가능하다는 것을 알 수 있다.

### 3.2 RPS 중심의 TM/TC 성능평가

광역화에 따른 원거리의 LAN과 직렬통신을 이용하여 실시간 데이터 처리와 I/O 카드를 통한 데이터 수집기능을 수행하는RPS와 GWS간의 실시간 데이터 처리과정을 실험하고, 시스템 운영에 중요한 역할을 하는 RPS 시스템 부하율을 측정하여 제어기를 성능평가 하였다.

표 3-2는 통신에 따른 처리시간을 나타내고 있으며, 소프트웨어 TASK 구성에서 포인트 동작에 따른 POS-GWS-RPS간 소프트웨어 처리시간을 표시한 것으로 상위 POS의 MMI에서 디스플레이 요구를 받거나, 제어를 받는 경우의 처리 시간을 실험하였다. 그림 3-1은 시스템간의 데이터 처리과정을 도식하였다.

실험을 위한 통신 속도는 9600bps이며, 직렬통신을 이용한 데이터 갱신은 170ms이내에 요구와 응답을 처리함을 알 수 있다. 표 3-2의 결과를 보면 기존의 LAN 통신과 직렬통신을 이용 시에도 충분한 200ms이내에 데이터 처리를 통한 실시간 제어가 가능하다는 것을 알 수 있다. 이 결과는 기존의 분산제어 시스템에 TM/TC를 위한 활용을 광역화 적용에 시스템 효율성이 높다는 것을 알 수



있다. 하지만, RPS에 SRS와 연계되는 시스템 구성 시에는 현재 POS로 SRS로 데이터 전송되는 시간은 보다 더 많은 지연시간이 있어 실시간성이 떨어짐을 알 수 있으며, 이는 앞으로의 데이터 처리 시간 향상을 위한 새로운 알고리즘에 관하여 연구해야 할 것이다.

표 3-2. POS-GWS-RPS간 소프트웨어 처리시간  
Table 3-2. Software procedure time of POS-GWS-RPS.

단위:msec

DATA 흐름		기능	처리 시간
source 타스크	dest 타스크		
MMI(XPU)	ncusnd(XPU)	디스플레이요구 메시지 전송	10
ncusnd(XPU)	cmdsnd(NCU)	메시지 요구를 NCU에 전달	3
cmdsnd(NCU)	gnrot(GWS)	메시지 요구를 GWS로 요구 (실패조건은 고려안함)	3
gnrot(GWS)	SRS테이블(GWS)	수신된 요구를 SRS테이블에 저장	1
SRS테이블(GWS)	gsmt(GWS)	gsmt는 주기적으로 테이블 스캔	1
gsmt(GWS)	gsst(GWS)	gsmt는 gsst에 데이터 전송 통보	1
SRS테이블(GWS)	gsst(GWS)	gsst는 SRS테이블 데이터 읽기	1
gsst(GWS)	srot(RPS)	직렬통신을 이용한 데이터 전송	1,000
srot(RPS)	multicast 영역	수신 데이터 저장	1
prsm 정주기 실행(RPS)		RPS의 정주기 데이터 실행	20
prsm의 BR영역 읽기		RPS의 BR영역 읽기	10
BR 데이터	실영역	BR영역을 실제영역 쓰기	30
ilm정주기실행		ilm 타스크의 정주기 실행	20
ilm의 데이터 읽기		ilm 타스크가 데이터를 읽기	10
ilm(RPS)	ICM 메모리	ilm카드의 실장 데이터만 읽기	10
ilm(RPS)	RPS BR영역	ilm은 정주기 데이터를 RPS의 영역 복사	10
ssot(RPS)	gsst(GWS)	BR데이터를 GWS로 전송	1,000
gsst(GWS)	BR 데이터 영역	수신된 데이터를 GWS의 BR영역에 저장	10
BR 데이터 영역	gnsbt(GWS)	수신 데이터 검색	1
gnsbt(GWS)	bcsrv(NCU)	LAN을 통해 상위 NCU로 전송	3
bcsrv(NCU)	공유메모리	bcsrv에 의해 공유메모리로 저장	3
MMI 반영			10
prsm	ilm 영역	MMI가 실제 ilm의 영역을 read (입력포인트의 스캔주기)	10

위의 표 3-2의 데이터는 RPS 부하에 영향을 미치는 값이지만 신뢰성 평가에 필요한 수치이기도하다. 이러한 수치를 부하율로 분석하는 것은 어렵기 때문에 본 논문에서는 부하율 분석을 위하여 가장 우선 순위가 낮은 타스크를 RPS에서 처리하게 하여 전체 부하율을 계산하였다.

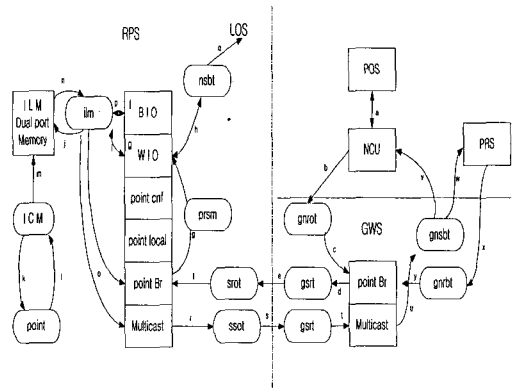


그림 3-1 POS(PRS)-GWS-RPS간 데이터 교환

Fig. 3-1. Exchange data of POS(PRS)-GWS-RPS.

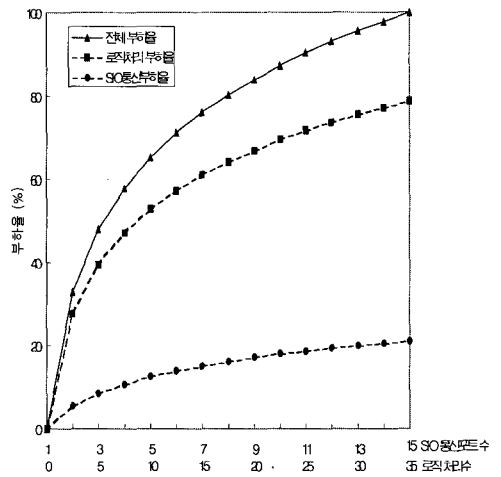


그림 3-2. RPS 시스템 부하율

Fig. 3-2. Rate of system overload for RPS.

이러한 부하율 분석은 그림 3-2에 나타내고 있

으며, 루프 32점, ACP 로직 처리 시 최대 1,000 스텝 처리와 직렬통신을 위해 16개의 SIO포트를 할당하여 15대의 SRS로부터 데이터 수집과 1대의 GWS로 데이터 전송과장의 부하율을 측정하였다. 처음 설계과정부터의 기존의 시스템의 부하율 증가를 고려한 8M의 메모리 할당의 타당성을 통하여 루프 32 점, ACP 로직의 1,000 스텝과 16대의 시스템 연결 시에 루프 32 포인트와 로직을 1,000 스텝의 실행을 수행 할 때는 80%에 가깝도록 부하율이 증가함을 알 수 있었으며, 또한, SIO 포트를 통한 채널을 한 채널 씩 증가한 경우에는 통신 요구를 진행 시에 최대 16대의 포트에 할당된 제어기와 통신에 따른 에러율은 20%에 가깝게 나타나 전체 시스템 부하율이 데이터 송·수신 처리 부하율을 포함하면 100% 가까운 부하율을 나타내고 있다는 것을 알 수 있었다.

### 3.3 SRS의 성능평가

시스템 사양의 다양한 요구로 점차 지역적인 공정처리에서 원격 감시 및 제어의 TM/TC 기능을 유지하는 SRS의 효율적인 감시·제어 진행과 통신의 효율성에 대한 성능평가는, 설계된 시스템에 대한 HDLC와 직렬 통신 통신에러율, 원격 다운로드 실행시간 실험을 통한 원활한 프로그램 전송으로 시스템 효율성을 분석하였다. SRS의 HDLC 통신성능 분석환경은 먼저, HDLC 카드를 이용한 56Kbps, 64Kbps, 1Mbps, 2Mbps의 속도로 국내 이동통신 업체에서 운영중인 HINA망을 이용하여 자체 제작한 HINA 시뮬레이터(simulator)와 데이터 송·수신을 통하여 통신에러율을 측정하였다. 썬크방식의 128 바이트의 데이터 요구에 대한 응답 처리과정을 측정하였으며, 시뮬레이터는 100ms 단위로 주기적인 요구와 응답을 처리하도록 하였다.

그 결과는 56Kbps, 64Kbps에서는 1%이하의 통신에러율로 통신을 판정할 수 있고, 고속인 1Mbps, 2Mbps 통신 시에는 통신에러율이 5%이하로 나타나 구현한 HDLC 카드는 고속통신의 효율

성 높다는 것을 알았다. 이에 대한 SRS의 HDLC 통신에러율은 표 3-3에 나타내고 있다.

표 3-3. SRS의 HDLC 통신에러율

Table 3-3. Communication error rate of HDLC for SRS.

전송횟수	56K바이트	64K바이트	1M바이트	2M바이트
1차 (500회)	3	4	28	24
2차 (450회)	0	3	26	27
3차 (600회)	3	2	31	33
4차 (750회)	2	4	28	35
5차 (750회)	1	6	23	29
평균에러율	0.30 %	0.63 %	4.53 %	4.93 %

SRS의 통신 시 주요한 특징은 소스 프로그램의 원격 다운로드를 통하여 프로그램 갱신을 이룰 수 있다는 장점을 가지고 있다. 다운로드 중에 에러가 발생하여 원격지에서 시스템 이상이 발생하더라도 5초 후에 ROM 소스 프로그램과 비교하여 다운로드된 데이터가 완전한 데이터가 아닐 경우에는 이전 버전의 프로그램으로 실행할 수 있도록 하였다. 원격 다운로드 실험에서는 400K 바이트의 응용 프로그램을 19,200bps 속도로 원격 다운로드를 실시할 때는 평균적으로 35분 정도의 시간이 소요됨을 알 수 있었다. 에러가 없을 경우의 이상적인 값은 초당 21 바이트 내외로, 전송되는 이상형의 처리시간은 20여분이지만 이는 일방적인 데이터 전송만을 고려한 시간이지만 응답시간을 주기적으로 부여하기 때문에 30여분의 데이터 전송 시간을 고려하여야 한다. 원격 다운로드 처리 시간은 원격지까지의 A/S 거리와 비용을 고려한 상태에서는 RMS에 사용 가능성이 높을 것으로 본다.

## IV. 결 론

본 논문에서는 기존 분산제어 시스템이 근거리 통신에 국한한 점에 비해 원격 감시제어를 이룰 수 있도록 한 운영체제 환경에서 TM/TC용 분산제어 시스템에서 실시간 제어기의 설계와 구현을 하였다.

GWS 제어기의 성능분석은 시스템 부하율이 50%이내인 GWS로 데이터 전송 시에 처리되는 통신에러율 분석으로 SRS는 평균 3.49%, RPS에는 5.92%의 통신에러율을 나타내었다. 이 데이터는 기존의 SCADA 시스템 운영시의 통신제어 장치의 평균에러율인 9%보다 4~6%가 낮음을 알 수 있었다. 실시간 데이터 전송을 위하여 예외처리를 통한 활용성의 실험치는 예외처리를 적용한 전송횟수 N=5, 10, 20을 10,000번 실행한 경우는 97.1%, 96.7%, 94.9%로 평균 96.2%의 실험치를 얻었다.

RPS 중심의 TM/TC 평가에서는 상위의 POS(PRS)-GWS-RPS간 시스템 연계에 따른 데이터 처리시간을 측정하였으며, 각 단계별 수행 시간을 거쳐 데이터 요구에 응답시간은 총 170ms가 됨을 알았다. 이 데이터는 제어기를 통한 실시간 통신을 통한 갱신된 데이터 수집이 이루어질 수 있음을 알 수 있었다.

SRS의 성능평가는 HDLC 통신 처리에 따른 통신에러율을 측정하여 128 바이트 데이터의 요구에 대한 응답처리과정에서 HDLC 시플레이터와 100ms 단위로 주기적인 요구와 응답을 처리한 결과, 56Kbps, 64Kbps에서는 1%이하의 에러율로 통신을 판장할 수 있고, 고속인 1Mbps, 2Mbps 통신 시에는 클럭 생성 때 동기화부에서 에러에 의한 에러율로 5%이하로 나타났다. 따라서, 구현한 HDLC 카드는 고속통신 시에 효율성과 또한 프로그램의 원격 다운로드 기능을 탑재하여 시스템 활용성이 높다는 것을 알 수 있었다.

앞으로 시스템 성능을 분석하기 위한 더 많은 것이 연구되어야 하겠지만, 본 논문의 주요쟁점인 원격 감시시스템 망에서의 구현된 제어기의 활용 능력은 타당하리라고 여겨지며 차후의 개방형 분

산제어 시스템의 설계를 위해서는 통신 시 적은 데이터를 유지할 수 있도록 할 수 있는 연구가 지속적으로 이루어져야할 것이다.

## 참 고 문 헌

- [1] Andrzej Goscinski, *Distributed Operating Systems The logical Design*, Addison-Wesley Publishing Company, Second Edition, 1991.
- [2] Georage Coulouris, *Distributed Systems Concepts and Design*, Addison-Wesley Publishing Company, Second Edition, 1994.
- [3] 성명기, "현장에서 신뢰성이 입증된 TM/TC 시스템," 월간 CONTROL, pp.63-67, 1996년 8월.
- [4] 김진태 김내진, "인터페이스 모듈화 설계 및 시스템 이중화 설계에 관한 연구," 건국대학교 대학원 학술논문 제41집, pp.398-407, 1995.
- [5] 김진태, 김내진, "분산제어 시스템에서 개방화 구조 설계," 건국대학교 대학원 학술논문 제 43집, pp.256-269, 1996.
- [6] 장태수, 김승진, 윤병주, 송변전 자동화를 위한 중앙제어장치 및 소프트웨어 개발(중간보고서), 한국전력공사, 1994년 7월
- [7] 이상철, "VSAT를 이용한 TM/TC 적용," 월간 CONTROL, pp.68-71, 1996년 10월
- [8] Motorola, *VMEexec-Volume 1*; Motorola Computer Group, 1992
- [9] 김내진, 박인갑 "신뢰성있는 실시간 분산제어 시스템의 성능분석에 관한 연구", Journal of IEEE Korea council, 제2권 2호, pp.105-112, Dec, 1998.
- [10] 김진태, 김내진, 박인갑, "실시간 분산제어 시스템에서 원격제어국의 신뢰성에 관한 연구," 대한전자공학회, 회로 및 시스템 연구회/전력 전자 연구회 합동학술 발표회 논문집, 제16권 1호, pp. 53-58, 1994
- [11] 김진태, 김내진, "실시간 분산제어 시스템의 인터페이스 모듈화 설계에 관한 연구," 대한 전자공학회 교육논문 제9권, 제1호, pp. 18-26, 1995

[12] WindRiver Systems, *VxWorks Programmer's Guide*, WindRiver Systems, Inc. Ver 5.3.1, 1997

[13] Jin Tae Kim, In Kap Park, "A study on the Fault Tolerance using Dual Processing in the Real-Time Distributed Control System," JTC-CSCC, pp.978-984, Jun, 1994

---

— 저 자 소 개 —

---



金 珍 泰 (學生會員)

1966년 3월 9일생. 1991년 2월 건국대학교 전자공학과 졸업(공학사).

1993년 8월 건국대학교 대학원 전자공학과 졸업(공학석사). 1999년 2월 건국대학교 대학원 전자공학과 졸업(공학박사). 1991년 1월 ~ 1992

년 3월 삼성전자 연구원. 1993년 8월 ~ 1997년 8월 (주)광명제어 연구소 과장. 1997년 9월 ~ 1998년 4월 (주)미래기술 연구소장. 1998년 4월 ~ 현재 (주)네오게이트 연구소장.

주관심분야는 DCS, TMS, ITS등임.



朴 仁 甲 (正會員)

1950년 2월 1일생. 1973년 고려대학교 공과대학 전자공학과 졸업(공학사).

1979년 3월 고려대학교 대학원 전자공학과 졸업(공학석사). 1986년 8월 고려대학교 대학원 전자공학과 졸업(공학박사). 1989년 7월 ~ 1990년 6월

Oregon State Univ. Post doc. 1980년 9월 ~ 현재 건국대학교 전자공학과 교수. 1995년 1월 ~ 현재 대한전자공학회 이사.

주관심분야는 Computer Network, Parallel Processing, Distributed Control System등임.