

인터넷을 이용한 선박기관의 SCADA 시스템 구축에 관한 연구

박인규¹⁾, 김윤식²⁾

A Study on the Development of Ship's Engine SCADA System using the Internet

In-kyu Park, Yoon-sik Kim

Department of Marine Engineering
The Graduate School of Korea Maritime University
Pusan, Republic of Korea

Abstract

This paper describes the development of ship's engine SCADA (Supervisory Control and Data Acquisition) system which can be controlled via the internet. In this paper, the SCADA system is composed of a number of microprocessor-based RTU(Remote Terminal Unit)s, a MMI(Man Machine Interface) host, a SCADA server, and SCADA clients. There are two protocols used in the system. Each RTU and the MMI host are connected by a RS-485 line and CSMA/CD(Carrier Sense Multiple Access/Collision Detection) protocol is used to communicate with each other. TCP/IP (Transmission Control Protocol/Internet Protocol) is used among the MMI host, the SCADA server, and SCADA clients. The equipments installed in the field are controlled by a number of RTUs. The function of the MMI host is to acquire real-time data

1) 한국해양대학교 기관공학과 전기제어 전공
2) 한국해양대학교 전기공학과 교수

from RTUs and control them. The SCADA server supports data transfer between the networked MMI host and the SCADA client on the web-server through TCP/IP. Data transfer is possible regardless of the type of network only if there are TCP/IP Winsock-compatible stack driver. The SCADA client is implemented as the shape of web-page by means of JAVA programming language. Therefore, it runs on a web-browser such as netscape and explorer, and allows a number of users to access this SCADA system.

1. 서 론

본 연구는 선박에 실용화되어 운용되고 있는 선박용 SCADA(Supervisory Control and Data Acquisition) 및 모니터링 시스템을 하드웨어나 네트워크의 확장 없이도 인터넷을 통하여 육상에서 감시 및 통제할 수 있는 시스템으로 확장하고자 하였다. 기존의 SCADA 및 모니터링 시스템의 대부분이 제한된 범위의 지역 네트워크에서 운용되고 있으며, 정해진 장소에서만 작업이 가능하다. 따라서 기존의 SCADA 시스템을 지역 네트워크는 물론 TCP/IP (Transmission Control Protocol/Internet Protocol)를 기반으로 하는 모든 형태의 네트워크에서 운용 가능하도록 함으로써, OA(Office Automation)와의 통합화가 가능하고 장소 및 거리의 제약이 최소화 될 수 있는 확장된 개념의 SCADA 시스템 구축을 제안하고, 그 실현 가능성의 확인을 목적으로, 실제 시스템을 구현한다. 그리고 현 연구단계에서는 인터넷의 사용에 따른 보안성의 문제는 고려하지 않았다.

본 연구는 기존의 SCADA 설계 방식에, 윈도우즈 소켓(socket)을 이용한 클라이언트(client)/서버(server) 구조에 의하여 SCADA 서버를 추가하고, MMI (Man Machine Interface) 호스트(host)의 사용자 인터페이스(user interface)를 웹 브라우저 상에서 동작하는 웹 페이지의 형태로 구현함으로써, 원격지의 작업자는 웹 브라우저를 통하여 원격감시를 위한 부가 설비 없이 현장의 데이터의 취득 및 산업설비의 감시제어가 가능하도록 한다. 즉, 시스템이 웹 브라우저 상에서도 운용 가능하게 되므로, 물리적 통신매체에 관계없이, 지역 네트워크는 물론 TCP/IP를 기반으로 하는 모든 형태의 네트워크에서 작업이 가능하게 된다.

2. 본 론

2.1 SCADA 시스템의 개요

SCADA 시스템은 집중 원격감시 제어장치로서 1개소의 중앙 제어소에 설치된 컴퓨터 장치와 다수의 피감시제어 대상에 원격소 장치(RTU : Remote Terminal Unit)를 연결하여 필요한 정보를 즉시 온라인(on-line)으로 취득함으로써 감시 제어 및 계측의 기능을 수행하는 시스템 즉, 운용자에게 필요한 정보를 제공하고 조작을 위한 높은 신뢰성을 보장하며 안전하고 경제적인 운용 수단을 제공하는 의미를 말한다.

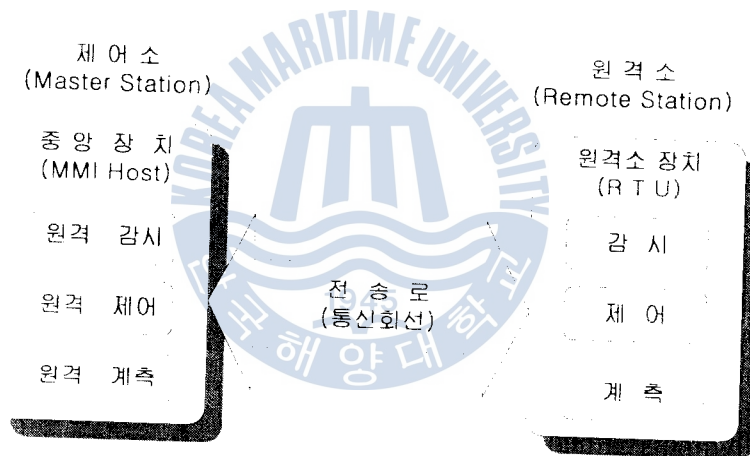


그림 1. SCADA 시스템의 개념도

[그림 1]에서 원격소 장치는 실제로 외부기기의 감시 및 제어계측의 기능을 수행하는 컴퓨터 장치로서 현장에 위치하여 제어소의 통제를 받으며, 감시 및 제어를 위한 각종 인터페이스(interface)와 디지털 및 아날로그 입출력 회로를 포함하고 있다. 제어소 장치는 원격지에서 전송로를 통하여 다수의 원격소 장치를 통제하는 컴퓨터 장치로서, 요구되는 명령을 원격소 장치에 전달하고, 원격소 장치에서 처리된 결과를 작업자에게 디스플레이 하는 역할을 수행하는 장치를 말한다.

2.2 인터넷과 감시제어

전세계에 흩어져 있는 개별 네트워크는 사용자의 상이한 요구사항을 만족시키도록 특정화되어 있다. 이러한 개별 네트워크들을 다양한 정보에 대한 필요성에 의해 하나의 가상네트워크(virtual network)로 연결시킨 것이 인터넷이다. 본 연구에서 말하는 인터넷이란 인터넷에 연결되어 있으며 인터넷 통신을 위한 표준과 도구를 사용하여 정보를 교환할 수 있는 TCP/IP를 지원하는 모든 형태의 네트워크를 의미한다. 인터넷에 의한 감시제어란 웹 브라우저에서 동작하는 SCADA 클라이언트의 작업자가 SCADA 서버에 의하여 인터넷 프로토콜(protocol)인 TCP/IP를 사용하는 상위계층의 네트워크를 경유해 MMI 호스트와 통신을 하여 원격소 장치를 동작시킴으로써 기기들을 감시 및 제어하는 것을 의미한다.

2.3 시스템의 설계 및 구현

2.3.1 시스템의 요구조건

본 연구의 SCADA 시스템이 인터넷을 통하여 안정적으로 감시 및 통제의 기능을 수행하기 위해서는 다음과 같은 요건을 만족시켜야 한다.

1. TCP/IP 프로토콜의 구조하에 동작될 수 있어야 한다.
2. 웹 브라우저에서 동작할 수 있어야 한다.
3. 웹 브라우저에서 해당 기기들의 상태를 감시할 수 있어야 한다.
4. 웹 브라우저에서 데이터의 송수신이 가능해야 한다.
5. 최신의 값으로 실시간 전송이 가능해야 한다.
6. 호스트에서 수신한 사용자의 명령이 실제 기기들을 구동시킬 수 있는 형태가 되어야 한다.

2.3.2 SCADA 시스템의 구성 모델

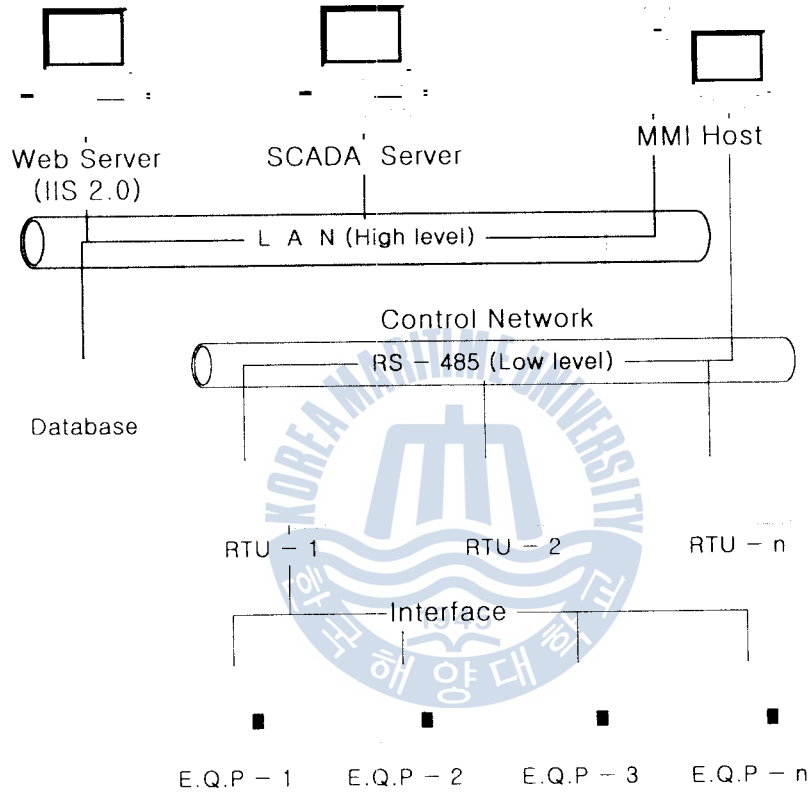


그림 2. SCADA 시스템의 구성 모델

2.3.2 SCADA 시스템의 구성 모델

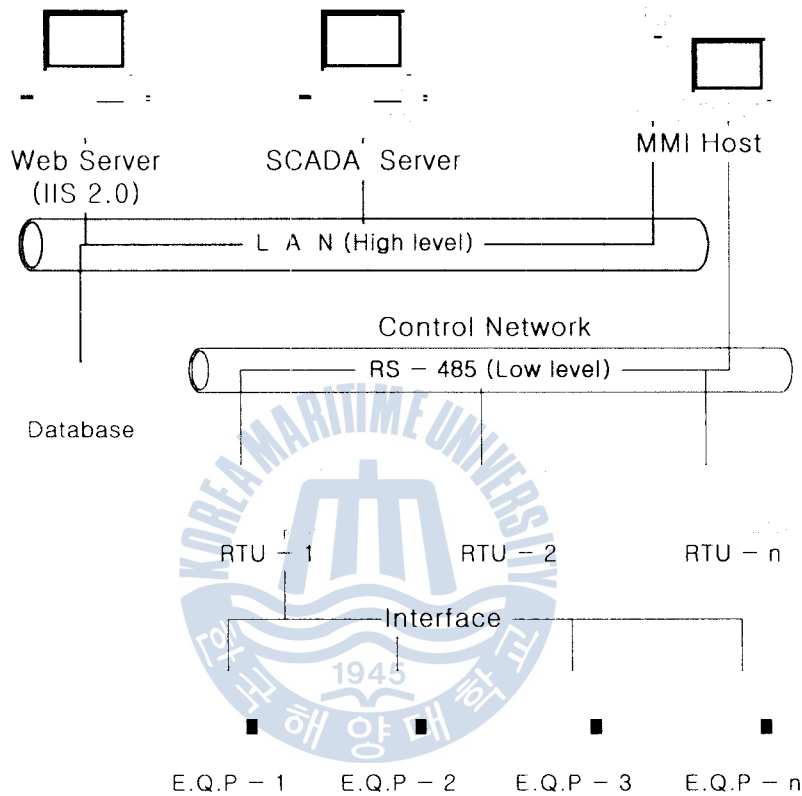


그림 2. SCADA 시스템의 구성 모델

이 그림 2는 본 연구에서 구현하고자하는 SCADA 시스템의 구현 모델을 나타낸다. 본 연구에서는 일반적으로 많이 사용되는 온도, 압력, 위치, 속도, 회전수 등을 제어대상으로 선정하여 기본적인 SCADA 시스템을 구성하였다. 전체적인 시스템의 설계는 분산제어 방식을 채택하였으며, 원격소 장치는 마이크로프로세서를 기반으로 설계하였다. 편의상 직접센서의 입력 및 제어를 담당하는 원격 마이크로프로세서로 설계된 원격소 장치를 RTU라고 하고, RTU를 통제하는 중앙 제어소 장치의 역할을 하는 컴퓨터 및 프로그램을 MMI 호스트, 웹 브라우저에서 실행되도록 자바 언어로 구현되어 웹 페이지의 형태로 웹 서버에 등록되어 있는 MMI 호

스트의 사용자 인터페이스를 제공하는 프로그램을 SCADA 클라이언트라고 하며 웹 서버 상의 SCADA 클라이언트를 통하여 사용자에게 시스템의 정보를 제공하는 서버 기능을 수행하는 프로그램을 SCADA 서버라고 정의한다. 그림에서 실선 부분은 기존의 SCADA 및 모니터링 시스템을 표현한 부분이며 SCADA 서버는 일반적인 인터넷 프로그래밍의 서버 역할을 하는 프로그램이 적재된 컴퓨터이다. 웹 서버는 윈도우즈 NT에서 제공하는 IIS(Internet Information Server) 2.0을 사용하였다. [그림 3]은 RTU의 하드웨어 구성을 나타낸 것이다.

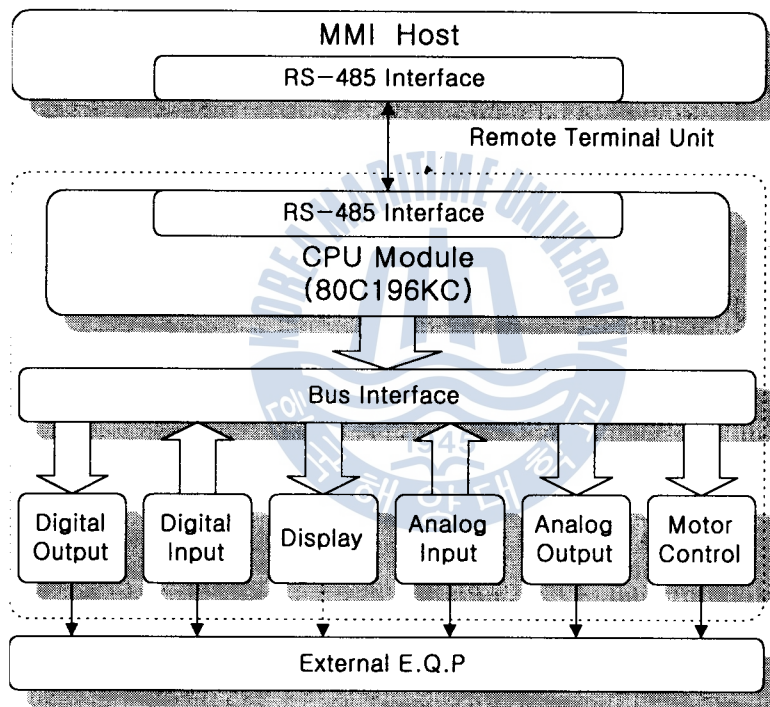


그림 3. RTU의 하드웨어 구성도

2.3.3 시스템의 소프트웨어 구성

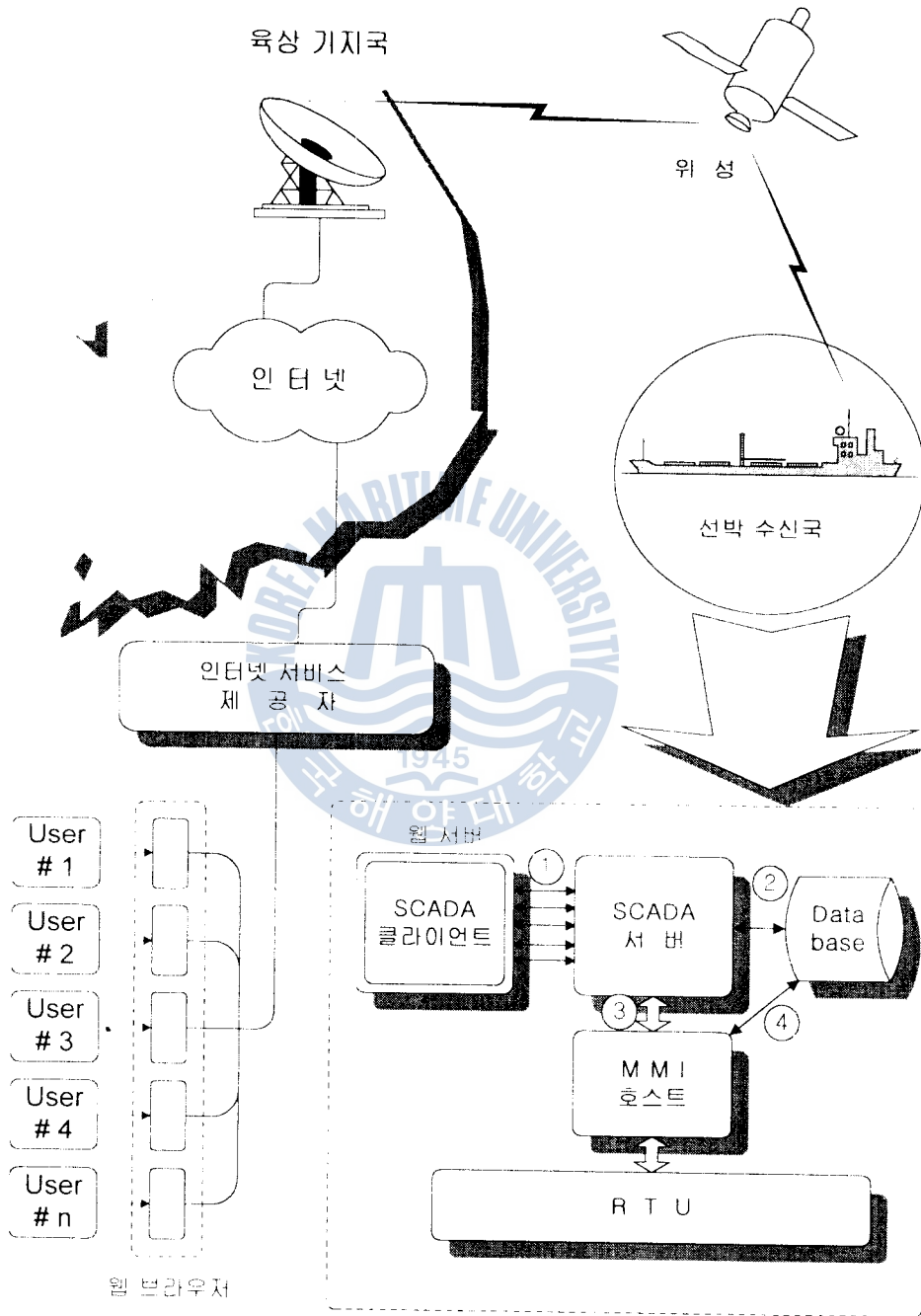


그림 4. 시스템의 소프트웨어 구현 모델

[그림 4]는 소프트웨어의 구현 모델을 나타낸다. 각각의 사용자는 Explorer나 Netscape와 같은 웹 브라우저를 사용하여 클라이언트에 접근함으로써 SCADA 서버에 접속하고, 접속된 사용자는 SCADA 서버를 통하여 시스템의 각종 정보를 액세스(access)하게 된다. 접속된 사용자가 시스템으로 제어명령을 전달하고자 할 때에는 해당명령을 SCADA 서버에 보내게 되며 명령을 수신한 서버가 다시 MMI 호스트에 전송함으로써 해당 기기를 동작시키게 된다. MMI 호스트는 RTU를 통제하며 RTU로부터 얻어진 데이터(④)를 데이터베이스에 저장하고 기기의 동작상태와 같은 실시간 정보는 SCADA 서버에 전송(③)하며 서버가 다수의 클라이언트 사용자의 화면을 갱신(①)함으로써 사용자에게 변경된 전보를 전달하도록 되어 있다. 그림에서 MMI 호스트와 RTU 프로그램은 기존의 SCADA 시스템을 구현한 것이며 ①은 각각의 사용자가 서버에 접속하는 소켓에 의한 접근 통로가 된다. ③은 MMI 호스트가 실시간 정보만을 SCADA 서버로 전송한다. ②는 사용자가 실시간 정보를 제외한 기타 요구되는 정보를 데이터베이스에서 얻기 위한 것이다.

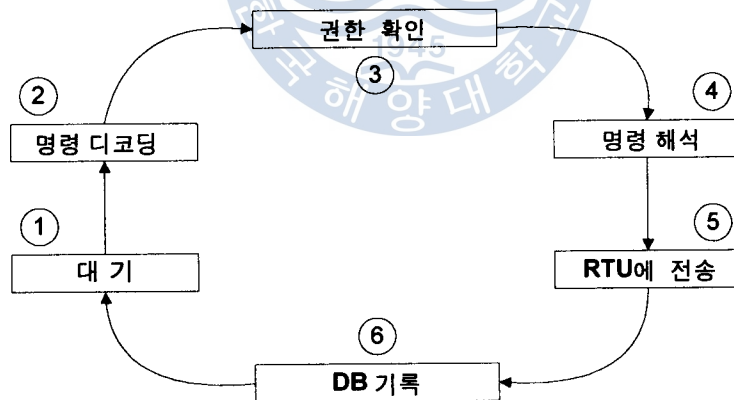


그림 5. MMI 호스트의 명령처리 흐름도

[그림 5]는 명령 수신에 따른 MMI 호스트의 활동을 요약하였다. MMI 호스트는 RTU로부터 얻어진 각종 정보를 데이터베이스에 저장하고, 기기의 동작상태와 같은 실시간(real-time) 정보는 SCADA 서버에 전송하며, 서버가 다수의 SCADA

클라이언트 사용자 터미널을 갱신함으로써 사용자에게 실시간 정보를 제공한다.

2.4 통신 시스템

MMI 호스트와 RTU간의 통신은 RS-485통신에 의해 버스형 LAN을 구성하고 자체 구현한 CSMA/CD(Carrier Sense Multiple Access/Collision Detection) 프로토콜을 적용하였다. SCADA 클라이언트와 SCADA 서버 및 MMI 호스트간의 통신은 기존의 LAN에, 윈도우즈 소켓에 의한 클라이언트/서버 모델을 구성하여, 인터넷 프로토콜인 TCP/IP을 적용하였다.

2.4.1 CSMA/CD 프로토콜

CSMA/CD 프로토콜은 자유 경쟁(contention)에 의한 방식으로써, 각 노드(node)는 송신에 앞서 버스에 반송파를 보내어 송신가능인가를 감지(carrier sense)한다. 송신 가능한 상태이면 곧바로 데이터의 송신을 시작하지만 공용 버스가 사용중이면 사용할 수 있을 때까지 기다린다. 복수 노드가 동시 또는 노드간의 선반지연시간 내에 송신을 시도하는 경우 상호가 송신 전에 반송파를 감출할 수 없어서 충돌이 발생하게되고, 충돌한 노드는 충돌 감지(collision detection)를 한 후, 각각 랜덤(random)시간 후 재 송신을 시도한다.

2.4.2 소켓에 의한 클라이언트/서버

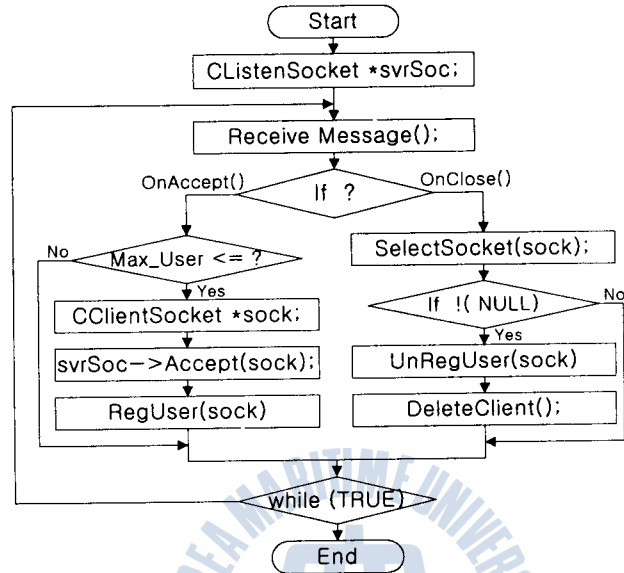


그림 6. 클라이언트/서버에 의한 사용자 접근

[그림 6]은 SCADA 서버와 SCADA 클라이언트 애플릿간의 클라이언트/서버 구조에 의한 사용자 접속/해지의 흐름을 표현하였다. 클라이언트/서버 구조는 네트워크 응용 프로그램을 설계하는데 가장 많이 사용되는 구조로써, 클라이언트는 서비스나 정보를 요구하는 측을 말하며, 서버는 클라이언트의 요구에 대한 서비스를 제공하는 측을 말한다. 구현된 시스템에서 웹 서버상의 SCADA 클라이언트와 SCADA 서버간에 윈도우즈 소켓에 의한 클라이언트/서버 구조를 적용하였다. 윈도우즈 소켓은 TCP/IP 네트워크 프로그램의 구현을 위하여 제공되는 표준 인터페이스로, 기존의 버클리 소켓에 윈도우즈의 메시지-드리븐(message-driven) 특성을 확장한 것이다.

2.5 실행 및 고찰

SCADA 시스템의 구축은 육상에서 선박자체를 통제하기 위한 목적으로 진행되 기보다는 선박의 상태를 감시하여 육상에서 이용한다는 측면에서 진행되었다. 즉,

선박 통제의 기능보다는 선박의 상태를 감시한다는 측면에서 인터넷을 이용하는 방법이 제안된 것이다. 구현된 시스템은 특정한 선박자체를 선정하여 설계한 것은 아니지만 인터넷을 통한 감시를 목적으로 선박에의 적용을 위한 확장성을 고려하였다. 연구의 목적이 기존의 선박에 탑재되어 있는 시스템을 그대로 이용하고 인터넷을 통한 선박 감시의 가능성을 확인하는 것에 있으므로 실제 실험단계에서의 실험장치의 구성에 SCADA 시스템의 성능이나 규모 등의 문제는 현 연구단계에서 중요한 요소로 작용하지 않는다고 생각되어 가능한 한 실험장치의 구성을 간단히 하였다. 실험은 선박에 LAN이 설치되어 있다고 가정하고 실험에 임하였다.

본 연구는 기존의 SCADA 기술과 웹 기술의 연동이라고 할 수 있으며 기존의 SCADA 시스템에서 얻어진 데이터를 인터넷을 경유하여 육상에 전달하고 육상의 제어 명령을 인터넷을 경유하여 전달하는 것이 핵심이며 이것을 목적으로 자바 언어 및 웹 페이지의 장점을 이용한 것이다.

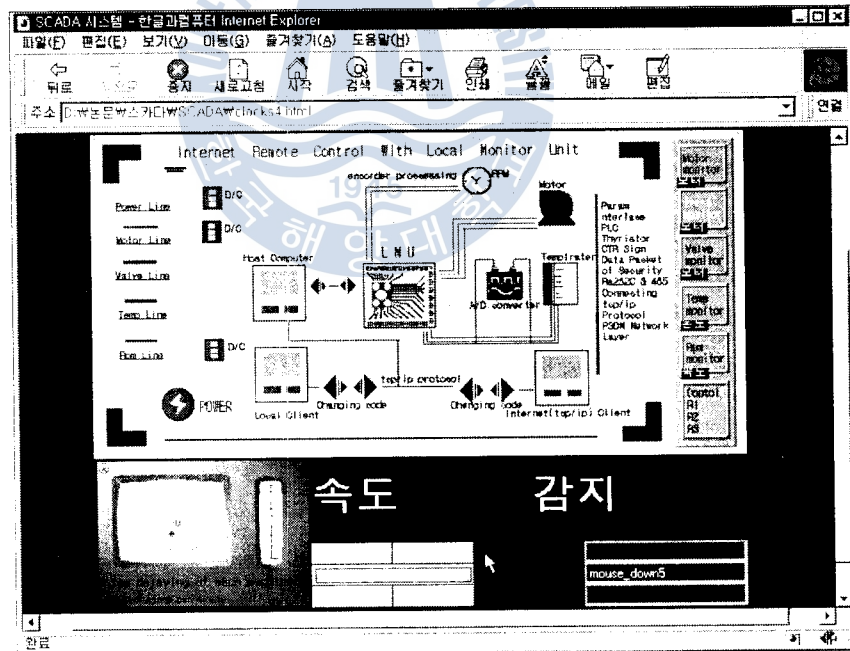


그림 7. RPM 측정화면

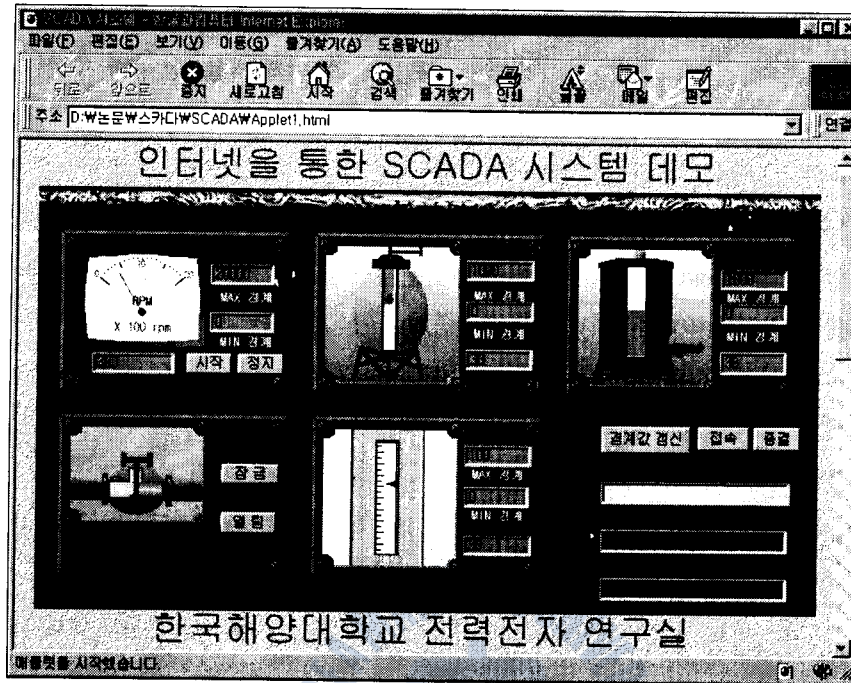


그림 8. 시스템의 실행을 위한 테스트 화면

3. 결 론

본 연구에서 구현한 SCADA 시스템은 선박에의 적용을 위하여 기존의 시스템을 인터넷 범위로 확장한 것으로써 선박에의 적용 가능성을 확인하기 위함이다. 구현된 시스템은 실험을 통하여 다음과 같이 정리하였다.

1. SCADA 시스템의 운용에 필요한 MMI 호스트 프로그램의 사용자 인터페이스를 웹 브라우저에서 실행되도록 구현함으로써 장소 및 거리의 제약을 최소화하였고, 실제 육상에서 선박의 감시 및 통제 가능성을 확인하였다.
2. 기존의 SCADA 및 모니터링 시스템은 단말장치 프로그램에 있어서 OS에 의존하므로 각각의 OS를 기반으로 하는 단말장치의 부가적인 프로그래밍이 요구되지만, 구현된 시스템은 웹 상에서 동작하기 때문에 터미널이 웹 브라우저의 장점을 흡수할 수 있으므로 OS에 따른 단말장치의 부가적인 프로그래밍이나 설치를 요구하지 않는다.
3. 선박의 특수성에서 기인하는 제반 문제에 대하여 그 신뢰성은 확인하지 못하였

고, 인터넷의 통신 속도와 해상외 물리적인 통신 수단에서 야기될 수 있는 문제점은 고려되지 않았다. 구현된 시스템에서 인터넷을 통한 통신속도는 자체 구성한 전용 네트워크의 속도와 비교하면 현저히 떨어지는 것이 사실이다. 그러나 SCADA 클라이언트의 사용자 터미널은 시스템에서 처리된 결과가 인터넷을 통하여 전달되면 화면으로 디스플레이 하는 역할과 사용자의 명령 전달의 기능만을 수행하므로, 인터넷의 통신속도가 시스템의 기본 기능인 감시제어기능에는 영향을 미치지 않을 것으로 사료되며, 앞으로 인터넷의 통신 속도는 향상될 것으로 기대된다.

참 고 문 헌

- [1] 강성억, "변전소 무인화 SCADA 시스템의 운영실적 및 신뢰도 분석조사 연구", 연세대학교 석사학위 논문, PP. 1-33, 1994.
- [2] 아한영, "CSMA/CD 방식을 이용한 마이크로 컴퓨터 통신망의 개발에 관한 연구", 연세대학교 석사학위논문, 1985.
- [3] 이상엽, "Internet Programming Bible", 영진 출판사, 1997.
- [4] Arthur Dumas, "Programming WinSock", SAMS, 1995.
- [5] 이민식, "윈도우즈 소켓 프로그래밍", 일원기술, 1996.
- [6] "電子技術基礎講座(51)", CQ出版社, 1995.

