

WWW을 이용한 이동로봇의 원격제어

박창욱* 박병훈* 홍진철** 박만식*** 이석규****
 * ****영남대학교 전기공학과 **안동정보대학 전기과 **대우중공업

Teleoperation of a Mobile Robot using World Wide Web(WWW)

C. W. Park, B. H. Park, C. C. Hong, M. S. Park, S. G. Lee

* ***Dept. of Electrical Engineering, Yeungnam Univ. ***Andong I.O.I.T **Deawoo Heavy Industries

Abstract - 지금까지의 시스템은 제한된 공간에서 구성되고 제어되어 왔으나, 공간적 제한을 극복하려는 원격 제어에 대한 연구가 한창 진행되어지고 있다. 원격 제어는 접속이 용이하여야 하며, 편리한 사용자 환경을 제공하여야 하며, 안정성이 보장되어야 한다. 최근 부각되고 있는 인터넷은 이러한 원격 제어의 요구를 잘 수용하고 있어 원격제어와의 접목이 시도되어지고 있다. 본 논문에서는 인터넷을 사용하여 공간적인 제한을 극복하는 열린 공간상에서 이동로봇을 제어함으로써 원격제어 시스템 설계와 구현에 관해 논한다. 구현된 원격 제어 시스템은 일반적인 웹서버를 사용하였고, 원격지와 이동로봇간의 정보를 양방향 통신으로 구현하였으며, 위와 같은 전체적인 시스템을 실제로 제작, 구현하여 실험함으로써 제안된 원격 제어 시스템의 실효성을 검증하였다.

1. 서론

지금까지의 원격제어란 한정된 범위내에 제어대상 시스템과 제어시스템을 함께 구축하여 직접 제어하고 있다. 인터넷에 있어서 WWW은 컴퓨터의 표준 GUI내에서 "Point and Pick"이라는 작업만으로 원격지의 Client에게 가장의 정보를 전달해 준다[1]. WWW은 원격지에서 컴퓨터상의 단말기를 통해 누구나 접근할 수 있는 용이성과 기존의 구축망을 그대로 이용할 수 있는 비용 절감의 효과가 있다[1][2]. 그리고, 데이터베이스 등 관련자원과의 접속도 용이하여 단순한 정보검색의 차원을 넘어서, 양방향 통신으로 우리가 제어하고자 하는 제어 대상의 정보를 획득하고 그 정보를 바탕으로 한 명령을 수행하게 함으로써 제어가 가능하고 이러한 과정을 반복적으로 수행하는 연속적인 작업이 가능하다. 그러나 많은 정보를 많은 사람이 실시간으로 접근할 수 있도록 하기 위해 아직 해결해야 할 점이 많다. 위에서 기술한 여러 장점들 때문에 WWW을 통한 원격제어에 관한 연구가 활발히 진행이 되고 있다. 과거에는 주로 전쟁터, 원자로, 공간탐사 등 현대산업사회에 많은 응용과 변화를 가지고 왔음은 누구나 부정할 수 없다[3]. 그리고, 현재의 산업체에선 분산환경으로 급격한 전환을 하고 있고, 시스템의 집중적인 통합 관리 및 제어의 필요성이 대두되고 있다. 1995년 이후, 자바로써 제작된 애플릿 프로그램은 작은 크기로서 이동이 쉬운 인터넷의 응용프로그램의 새로운 표준으로 부각되었다. 또한, 자바는 압축된 형태로서 강력하고 일목요연한 기능을 제공한다. 이러한 자바는 네트워크와 보안, 그리고 멀티스레드의 기능을 가지도록 고안되어 있어서 네트워크를 통한 분산 소프트웨어와 데이터의 접근이 용이하다. 위와 같은 동기를 가지고 원격지로부터 제한된 공간내의 탐사를 가능하게 하는 이동로봇을 제작하고, 원격지에서 GUI로 동작하는 자바애플릿의 제작 및 보편화된 PC상에 일반적인 웹서버의 설치와 또다른 제어서버의 제작등을 제시함으로써 원격제어시스템을 구축하고자 한다.

2. 전체 시스템의 개요

본 논문에서 구현된 전체 시스템을 그림1에 나타내었다. 또한, 앞서 서론에서 기술한 바와 같이 시스템의 전체적인 구성에 있어 네트워크 상에서 우수한 기능을 가진 자바로써 본 논문에서 제안한 시스템을 구축하여 향후 시스템의 확장에 중요한 이점을 가지도록 한다.

시스템의 구성은 크게 원격지의 Client영역과 두 개의 Server, 그리고 직접 제어서버에 연결된 제어대상으로 구성되어 있다. 본 논문에서 실험되는 제어대상은 이동로봇이다. 따라서 이 시스템은 크게 서버와 클라이언트간의 관계로 나타낼 수 있다.

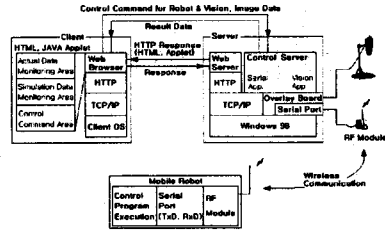


그림 1. 전체 시스템의 개요

또한 응용프로그램부분에서 처리된 제어명령과 데이터는 서버의 시리얼 포트를 통하여 이동로봇으로 넘겨지게 된다.

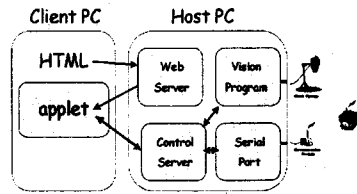


그림 2. 전체 시스템의 구성도

2.1 Client

자바애플릿은 원격지에서 호스트에 접속(Request)시 원격지로 이동(Downloading)하여 제어대상에 관한 제어응용프로그램으로 실행된다. 또한, 자바애플릿은 원격지의 시스템에서 동작하여 그 원격지의 시스템의 하부에는 전혀 영향을 미치지 않고, 제어서버측에 접속되어 제어명령이 수행되는 것을 보여준다.

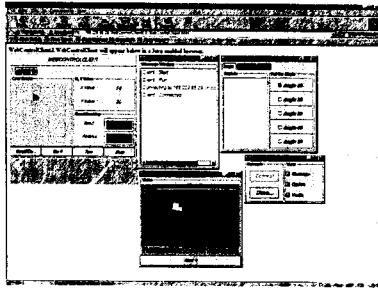


그림 3. 원격지에서 서버에 연결된 제어화면

위의 그림.3에서 나타나듯이 원격지에서 제어서버에 연결된 모습을 볼 수 있다. 원격지에선 서버에 접속하여, 제어대상의 위치정보를 파악하고 목적지를 설정할 수 있도록 되어 있으며, 각각의 창을 통해 접속상태 및 이동로봇의 이동각도와 거리, 속도의 정보를 지시할 수 있다.

2.2 Host

호스트영역은 전체적으로 크게 나누어 서버와 이동로봇으로 나눌 수 있다. 시스템에서 제안하고자 하는 것은 원격지에서 특정한 IP 주소의 제어서버에 연결하여 이동로봇을 실시간에 제어하고자 함이다. 그러므로, 원격지와 호스트의 연결을 담당하는 웹서버 그리고 원격지와 제어대상을 연결하는 제어서버로 나누어져 있다. 별도로 웹서버는 또 다른 PC에 설치하여도 전체시스템의 구성에는 상관이 없다.

2.2.1 Web Server

웹서버는 특별한 기능을 가지지 않고, 단순히 사용자가 호스트에 접속하여 클라이언트영역에서 기동될 애플릿만을 클라이언트영역으로 downloading 할 수 있으면 된다. 따라서 본 논문에서 이 부분은 기존에 나와있는 프리버전의 웹서버를 사용하여 구현하였다. - Sambar Server 4.2 Beta 6 Release

2.2.2 Control Server

제어서버는 5000port에 socket을 만들어 웹서버와 달리 PC에 부착된 제어대상을 제어할 수 있는 서버를 PC상에 구현하였다. 이러한 제어서버는 쓰레드를 사용하여 웹서버에서 원격지로 보내준 애플릿의 접속을 기다리고 있다. 원격지에서 한번 접속된 연결은 CGI와는 달리 연결상태를 지속적으로 유지하게 된다. 또한 자바로 만든 시리얼 통신 응용프로그램을 제어서버내에 함께 구현하였다.

본 논문의 시스템은 socket을 통해 받은 제어명령과 카메라를 통하여 들어오는 영상을 Server측에서 처리함으로써 이동로봇의 부담을 줄였고, 차후의 연구와 다른 제어대상에서도 적용할 수 있다. 또한 제어대상에서 일을 분담하여 처리함도 가능하다. 제어서버와 클라이언트간의 연결프로토콜은 Real-Time Transport Protocol(RTP)로써 실시간 데이터의 전송을 위한 엔드투엔드 네트워크 배달 서비스를 제공한다.

2.3 이동로봇

2.3.1 이동로봇의 개요

이동로봇은 Intel 80C196KC 프로세서를 기반으로 구성되어 있다. 모터구동을 위해 SLA7024 모터구동 전용 IC를 사용하고 있으며, 80C196KC에 내장된 UART를 이용하여 제어서버측과 RS-232C를 이용한 시리얼 통신을 하고 있다.

이동로봇은 제어서버의 시리얼 포트를 통해 넘어온 데이터를 받아 주변장치를 동작하도록 하고 있다. 즉, 제어서버로부터 이동거리, 회전각, 이동속도의 데이터를

받으면 이를 처리하여 모터의 펄스 및 펄스 간격을 조정하여 명령을 수행하도록 되어 있다. 내부적으로는 소프트웨어 타이머 인터럽트와 시리얼 포트 인터럽트를 이용하여 각 작업이 원활히 수행될 수 있도록 프로그래밍 되어 있다.

80C196KC는 20MHz 클럭으로 동작하고 있고, AD 컨버터, 고속 입·출력 등의 기능을 내장하고 있는 단일 칩 컨트롤러이다.

2.3.2 이동로봇의 알고리즘의 개요

그림.6은 이동로봇의 알고리즘이다. 이동로봇은 포트와 레지스터 및 변수에 대한 초기화를 끝낸 후 시리얼 포트를 통해 패킷이 넘어오는 동안 대기하고 있다. 패킷이 들어오면 이를 분석, check sum을 이용하여 오류 여부를 확인한 후 모터에 가해지는 펄스 및 펄스 간격을 조정하여 제어서버에서 주어진 명령대로 모터를 구동, 움직이게 된다. 여기서 모터 구동은 인터럽트를 사용하여 주기적으로 펄스를 주므로 일단 패킷이 들어오면 그 다음 상태와 상관없이 로봇은 움직이게 되어 있다.

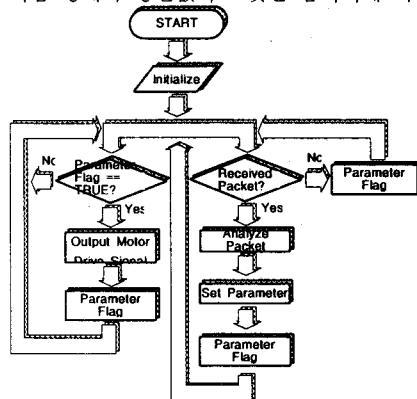
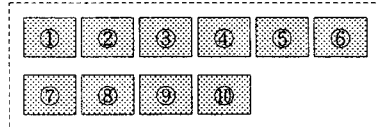


그림 6. 이동로봇의 알고리즘

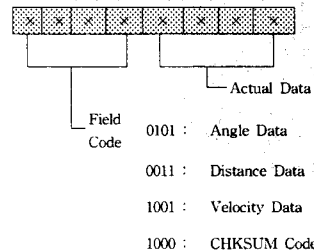
< 전송프로토콜 >

· 데이터 패킷의 구성

패킷은 다음과 같이 구성된다.



각 Data Field의 상위 4bit로 각 데이터를 구별하며, 하위 4bit는 각 Field의 실제 데이터이다. 이를 그림으로 나타내면 다음과 같다.



① Header Field (0x78)



② Packet ID (0101xxxx)



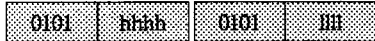
하위 4bit는 명령의 종류를 나타내며, code에 따라 다음 명령으로 구분한다.

가. Function의 종류

- 0101 : 이동명령
- 1111 : 리셋명령(next version)
- 0001 : 저장된 경로 송신 명령(next version)

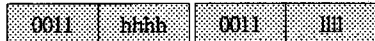
③, ④ Data Field - Angle (0101xxxx)

상위 4bit와 하위 4bit를 나누어 그림과 같이 구성한다. 상위 4bit는 회전각에 대한 데이터 ID를 나타내며, 하위 4bit는 실제 회전각을 나타낸다.



⑤, ⑥ Data Field - Distance (0011xxxx)

상위 4bit는 데이터 ID이고, 이동거리를 나타낼 경우 이미 얘기한 바와 같이 0x3의 값을 가진다. 하위 4bit는 실제 이동거리(0x00-0xFF)의 상위 4bit와 하위 4bit를 나타낸다.



⑦, ⑧ Data Field - Speed (1001xxxx)

상위 4bit는 데이터 ID, 하위 4bit는 이동속도를 나타낸다.



⑨, ⑩ CheckSum Field (1000xxxx)

통신시 패킷 전송에러 검출에 사용되는 Field로, 하위 4bit가 Check Sum을 나타내며 아래 그림처럼 2byte로 구성된다.



Check Sum의 상위 4bit Check Sum의 하위 4bit

Check Sum은 다음과 같이 구할 수 있다.
 CheckSum = Header Code + Function Code + Data Code + Control Code
 여기서 Check Sum의 변수형은 byte이며, overflow가 생길 경우 carry는 버린다.

3. 실험 및 검토

그림.1에 제시된 시스템을 이동로봇에 적용하여 아래의 그림.7과 같이 구성하였다.

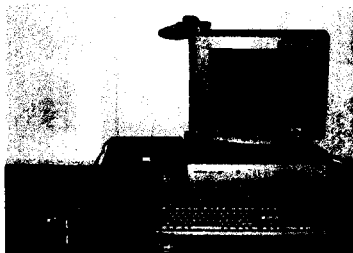


그림. 7 실험 장치

3.1 실험환경

- ▶ 서버측 PC의 사양.
- O/S : windows98, CPU: 셀러론 333MHz,
- RAM: 64MHz, OverlayBoard

3.2 실험

먼저 이 실험은 주로 연구실내의 인트라넷에서 행해졌다. 사용자가 서버측에 접속하여 피제어물의 동작환경을

보면서, 피제어물을 제어할 수 있도록 하였다. 연구실내의 환경이라는 점으로 인해 실시간으로 이동로봇을 제어할 수 있었지만, 어느정도의 지연은 피할 수 없었다.

3.3 검토

열악한 환경에서 사용자가 내린 명령이 빠르게 실행될 수 있도록 하기 위해 클라이언트와 서버간의 데이터량을 최소화하였다. 현재의 환경은 여러사람의 동시접속이 가능하도록 되어 있고, 이로 인해 일관된 명령을 내릴 수 없는 문제점이 있다. 이는 서버측의 접근을 제한하는 보안환경이 요구된다. 또한, 여러명의 접속으로 본 실험환경에서 사용된 PC의 사양으로는 만족스런 서비스를 제공하지 못하는 점도 있었다. 이는 서버의 성능을 향상시키므로써 어느정도 해결할 수 있을 것이다.

실시간환경에서 이동로봇의 움직임은 사용자의 명령에 의해 제어되는만큼 급작스런 환경변화에 대해 많은 문제점을 가진다. 이는 서버측에서 능동적으로 대처할 수 있는 시스템을 구현할 필요가 있다.

향후 연구에서는 카메라로부터 받은 영상정보로 이미 지처리와 회피알고리즘을 통한 서버의 직접적인 장애물 회피를 구현하고, 제어서버와 이동로봇간의 통신은 유선으로 이루어져 있으나, 무선으로 통신하여 이동로봇의 환경에 대한 자율성을 부여할 계획이다

4. 결론

본 논문에서는 제한된 공간내에서 구성된 원격시스템을 Web기술에 적용하여 인트라넷에서도 구현하여 제어하였다. 이러한 시스템 구현을 위해 자바기술을 사용하였고, 기존의 웹브라우저로 손쉽게 접속할 수 있음을 보였다. 또한 실험에서 나타난 약간의 시간지연은 네트워크망의 병목현상으로 피할수 없는 현상이었고, 실험에서의 이동로봇이 하계될 일에 대한 정의를 보다 더 자세하고, 간단히 할 때 본 실험에서 나타난 시간 지연은 큰 의미가 없을 것이다.

이외에도 준실시간영역에서 동작이 허용되는 시스템은 웹을 통해 충분히 제어가 가능함을 확인할 수 있었다.

참고 문헌

- [1] K.Goldberg, M.Mascha, S.Genter, C.Sutter, N.Rothenberg, J.Wielgley, "Desktop teleoperation via the World Wide Web", Proc.IEEE Int. Conf.Robotics and Automation, Nagoya, JAPAN, May, 1995.
- [2] 이명진, 문제철, 강순주, "인트라넷상에서 WWW을 이용한 무선 비행체 원격제어", 제어·자동화·시스템공학회 합동 학술 발표회 논문집, pp191-195, 1998.
- [3] Eric Paulos, John Canny, "Delivering Real Reality to the World Wide Web via Relerobotics", Proc.IEEE Int. Conf.Robotics and Automation, Minneapolis, Minnesota, April, 1996.
- [4] Sun Microsystems, Java Remote Method Invocation Specification bata draft, Dec 1996
- [5] Java WhitePaper - Moving from the Console to the Web in Real Time, March, 1997.
- [6] Engineering Web Technologies for Embedded application, IEEE Internet Computing, May·June, 1998.
- [7] Java Networking Programming, O'reilly, Elliotte Rusty Harold.
- [8] <http://java.sun.com>
- [9] <http://www.aglance.com>