

웹에 기반한 원격 감시 및 제어장치의 구현

장호성^{1*}

Implementation of Web Based Remote Monitoring and Control System

Ho-Sung Chang^{1*}

요 약 최근 컴퓨터 네트워킹 기술을 텔레미터/텔레컨트롤 시스템에 적용하여 고성능의 스마트 제어기능을 제공하려는 연구가 활발하게 진행되고 있다. 본 논문은 온실 모델에 다양한 적용을 위하여 특별히 설계된 리모트 및 로컬 제어 소프트웨어를 갖는 클라이언트/서버 구조를 제안한다. 서버 장치는 센서나 액츄에이터에 관련되는 각종 데이터를 수집, 집적하고 고품질의 원격 I/O 제어동작을 유지하기 위하여 데이터베이스에 저장한다. 이에 대해 클라이언트 장치는 인터넷 웹브라우저 상에서 제어 판넬과 같이 간단히 동작한다. 구현된 장치는 멀티미디어 정보가 포함된 온실의 감시 및 제어상태 모두를 대화형으로 원활하게 통신하였다. 본 연구에서는 패킷 크기에 의해 영향 받는 종합적인 스루풋과 전송시간에 관계되는 성능 중대의 한계도 분석하였다.

Abstract Recently, lots of research activities have been carried out to provide advanced smart control functions applying a computer networking technology to telemeter/telecontrol systems. In this paper, we suggest client/server architecture with specially designed remote and local control software for applicant those to greenhouse model. The server system collects and integrates various datas related sensor or actuator, and stores in database maintaining high quality of remote I/O control operation. Where as the client system simply operation such as control panel on the internet web browser. Implemented system well communicate interactively both monitoring and control status of greenhouse including multimedia informations. In this study we also analyze the extent of the performance enhancement in terms of overall throughput and data transfer time effected by packet size.

Key Words : Remote Control, Ubiquitous, Digital I/O, Image Information, Linux, Apache

1. 서론

텔레미터/텔레컨트롤(Telemeter/Telecontrol) 기술은 사물이나 사상을 더욱 정확하게 파악하고 정밀하게 조정하기 위하여 멀티미디어 통신 네트워크를 적용하는 동향을 보인다. 특히 internet infra의 확충과 broadband의 보급으로 인터넷 라디오, 인터넷 TV와 같이 streaming에 의한 영상, 음성은 물론 다양한 media 정보를 web browser상에서 간편하게 취급할 수 있어 이들 기술은 가전제품까지 확장 적용되고 있다^{[1],[2]}.

특히 시각정보는 빠르고 간편하며 확실하고 정보량도 풍부한 잇점에 의해 원격지의 감시, 조정 및 제어를 행하

는 영상정보 시스템에 널리 사용되고 있다. 고품질의 영상정보를 송출하면 정보전송량이 증대되어 넓은 대역의 통신로가 필요하게 되나 측정빈도나 정밀도를 높혀도 같은 효과를 나타낸다^[3]. 자연현상의 일반적인 경우는 항시 대량의 정보를 전송하는 것이 아니라 어떤 event가 발생하는 경우에 순시적으로 송출하는 것이 중요하게 된다.

결국 multimedia 정보를 취급하는 server system은 정보망 관리기능과 다중처리 제어성능이 강력해야 하며 정보통신 네트워크는 고속성과 광역성, 그리고 공중성에 의해 경제성이 결정된다^{[4],[5]}.

이와 같은 관점에서 본 연구는 높은 안정성의 감시 및 제어기능과 편이성을 제공하는 시스템 환경을 구축하여

이 논문은 2006년도 홍익대학교 학술연구조성비에 의하여 연구되었습니다.

¹홍익대학교 컴퓨터정보통신공학과(교수)

*교신저자: 장호성(hosung@hongik.ac.kr)

인터넷을 통해 양방향 원격 통신망을 구현하고 성능향상을 위한 방안을 연구한다.

2. 시스템 분석

2.1 제어 시스템 구조

네트워크를 이용하여 원격의 I/O 장치를 제어하기 위해서는 client-server 시스템 환경을 구축하고 인터넷이나 LAN에 접속되어야 한다. 클라이언트에는 웹 브라우저와 remote I/O 제어 프로그램이 설치되고 서버에는 웹 서버와 I/O 제어장치를 구동하는 local I/O 제어 프로그램이 설치되어 요구되는 동작을 각각 수행한다.

그러나 요구에 따라서는 웹 브라우저 대신 리모트 I/O 제어 프로그램만으로 서버를 제어하는 방법도 있으나 범용성이 떨어진다. 또한 양측 I/O 제어 프로그램이 직접 TCP/IP 통신을 행하는 방법도 있다. 이 방법은 통신이 직접적으로 개시되므로 파일 입출력이 없게 되나 통신 속도, 정확성 등 제어의 질이 크게 변화되므로 영상이나 음성 등 멀티미디어 정보의 전송 품질이 크게 떨어지게 된다⁶⁾.

따라서 본 연구에서는 감시 및 제어 시스템의 범용성과 안정성 확보를 위해 클라이언트-서버 시스템 구조로 설계하여 구현한다.

2.2 네트워킹 개발환경

인터넷을 통해 통신 네트워크를 구축하는 응용에 Linux는 강력한 OS로서 각광을 받고 있다. 리눅스는 특히 네트워킹과 하드웨어 제어가 간편하여 웹 서버는 물론 로컬 제어 I/O 프로그램 개발이나 시스템 구축에 널리 사용된다. 이와 같은 특징은 kernel을 비롯하여 풍부한 middleware와 다양한 application이 제공되고 있으며 실시간 clock 등 다양한 기능이 탑재되어 있거나 간단히 추가할 수 있는 장점이 있기 때문이다.

또한 리눅스는 각종 device driver가 풍부하게 개발되어 있고 ethernet controller의 PCI 디바이스가 미리 드라이버로 준비되어 있어 네트워크를 표준으로 지원한다⁷⁾.

그러나 리눅스는 마우스나 키보드로 대표되는 HID (Human Interface Device)와 스토리지 클래스의 일부가 지원되고 있으나 그 이외의 USB 인터페이스 등은 아직 이용이 곤란하다. 따라서 USB를 경유하여 HID나 스토리지 디바이스, 네트워크 인터페이스 등 디바이스를 사용하려면 이에 앞서 디바이스를 지원하는 디바이스 드라이버를 필요로 한다.

또한 리눅스는 화면용 드라이버가 그래픽 전체를 지원

하지는 않으므로 어떤 windows 시스템을 사용하는가에 따라 드라이버가 변하게 되므로 이에 대한 대책이 필요하다. 드라이버나 제어 프로그램 개발 tool은 리눅스 표준의 gcc를 기반으로 한 것을 사용하는 것이 간단하다. gcc는 윈도우즈의 C 컴파일러에 해당하며 대부분의 target을 지원한다.

2.3 Unix Emulator

윈도우즈 환경에서 리눅스를 기반으로 하는 시스템을 개발하기 위하여 윈도우즈용 유닉스 emulator인 Cygwin을 사용한다. 대규모 개발에서 유닉스 이몰레이터의 overhead 등이 문제로 되는 경우에는 리눅스 컴퓨터나 유닉스계의 워크스테이션을 이용하는 편이 좋으나 본 연구의 범위 정도의 용도에는 이몰레이터로 충분하다. 또한 Cygwin을 사용하면 gcc를 간편하게 이용할 수 있다⁸⁾.

Cygwin은 cygwin.dll이라는 dynamic link library를 사용하여 유닉스계의 시스템 call을 윈도우즈의 API로 변환하여 유닉스 환경을 emulating하고 있다. 따라서 cygwin.dll이 있으면 Cygwin의 프로그램을 윈도우즈의 consol application으로서 직접 구동할 수 있다.

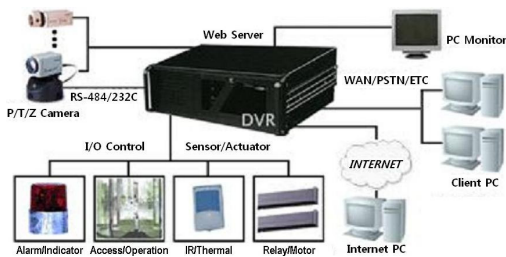
리눅스에 있어서 I/O 포트 제어가 간편한 것은 독특한 독자의 처리를 행하는 ioperm 함수이다. 이 함수를 호출하면 특정한 I/O 포트에 대해 지정한 범위의 포트만을 읽거나 쓸 수 있다. I/O 포트사용권의 취득후는 inb 함수와 outb 함수로 I/O 제어가 되도록 된다. 단, 프로그램 종료시에도 한번 ioperm을 호출하여 취득한 사용권을 개방할 필요가 있다.

3. 설계 및 제작

3.1 시스템 구성

하드웨어 제어를 수행하는 서버 시스템에서 취급하는 정보는 카메라로부터 전송되는 다량의 영상정보가 대부분이며 이벤트 발생시 음성 및 sensor와 relay 등의 디지털 I/O 정보가 포함된다. 한편 이들 데이터는 웹 브라우저 상에서 사용자가 필요한 정보의 형태로 제시되어 요구에 따라 사용자가 필요한 제어정보만을 취득하여 해당기기를 구동하는 구조로 설계하며 이들 기기는 USB나 IEEE394 등의 고속직렬 통신방식으로 전송된다.

그림 1에 외부 입출력 제어장치를 포함하는 웹에 기반한 원격 감시 및 제어 시스템 구성도를 보였다.

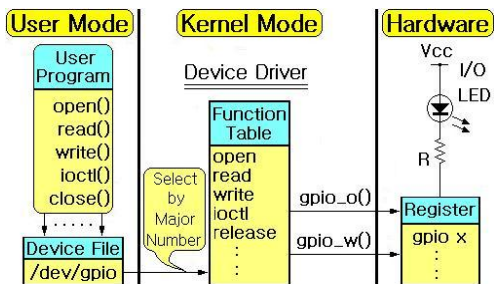


[그림 1] 시스템 구성도

3.2 디바이스 드라이버

유닉스계 OS에서 상위의 커널 모드는 시스템 콜이나 디바이스 드라이버 등 커널 프로그램이 동작하며 기능제한이 없고 특권명령도 사용된다. 이에 대해 하위의 유저 모드는 커널 이외의 프로그램이 제한적으로 동작되며 I/O memory access에 제한이 있고 특권명령은 사용할 수 없다. 결국 유닉스계 OS는 유저 프로그램이 직접 I/O 포트를 조작할 수 없게 된다. 그러나 유저 모드는 어떤 bug가 있어도 시스템이 충돌하는 일이 없기 때문에 디바이스 파일을 경유하여 커널내의 디바이스 드라이버를 사용하면 I/O 조작을 수행할 수 있다.

이와 같이 디바이스 드라이버를 경유하는 I/O 조작에는 기존의 디바이스 /dev/mem을 사용하여 불특정 어드레스의 메모리나 I/O에 액세스하는 방법과 특정기기 제어용 함수의 집합인 전용의 디바이스 드라이버를 제작하여 시스템에 이식시키는 방법이 고려된다. 그림 2에 I/O를 제어하는 디바이스 드라이버의 동작을 보였다. 함수 포인터를 미리 정한 구조체에 세트하고 대표 번호라고 부르는 수치와 함께 시스템에 등록한다. 유저 프로그램이 dev/gpio 등과 같은 디바이스 파일을 액세스하면 드라이버 파일에 등록된 대표 번호에 의해 디바이스 드라이버가 선택되고 등록된 함수가 호출되어 해당 기기의 조작을 행할 수 있다.



[그림 2] 디바이스 드라이버 동작

3.3 CGI 프로그램

통상의 응용 프로그램은 명령을 수행하면 당해 처리가 항상 즉각적으로 실행되어 화면 갱신 등의 동적동작을 행한다. 그러나 인터넷에서 간편하게 사용되는 Netscape나 Internet Explorer 등 대부분의 웹 브라우저에서는 단지 하나의 html(hyper text markup language) 문서를 하나의 화면만으로 표시하는 정적동작을 기본으로 하므로 매 화면마다 초기화한 후 신규 html 문서를 제시해야 한다.

이러한 정적동작은 웹서버 시스템을 매개로 하고 CGI(common gateway interface) 기능을 사용하면 갱신시마다 매번 구동되어 동적형태로 표현할 수 있다. 따라서 클라이언트에서는 홈페이지를 보는 것과 같이 간단한 방법에 의해 웹 브라우저를 감시 및 제어 pannel로서 응용이 가능하게 된다.

3.4 서버 시스템 구현

서버 시스템은 http 서버와 CGI 프로그램을 사용하여 웹 페이지상에서 하드웨어에 액세스하여 제어하는 방법을 사용한다. 이를 위해 웹 서버는 프리웨어로 보급되어 있는 Apache를 사용하며 http 서버는 특히 메모리 사용량이 적고 고속이며 효율적으로 설계되어야 하므로 Java로 구축한다.

http 서버 동작은 브라우저가 요구한 URL에 따라 통상은 html 문서파일을 브라우저에 보낸다. 그러나 http 서버는 CGI 기능에 의해 파일만이 아니라 외부 프로그램의 출력을 반송하는 기능도 수행한다. 따라서 CGI 프로그램의 실행과 종료에 따라 클라이언트의 브라우저에는 html 문서가 표시되므로 통상의 홈페이지와 같은 동작을 수행할 수 있다.

그러나 CGI 프로그램은 실행상태가 유지되어 있으면 웹 서버와 클라이언트의 브라우저가 busy 상태로 지속되므로 CGI 프로그램은 화면표시가 끝난 후 즉시 종료시킨다. 따라서 CGI 기능은 주 프로그램이 실행하고 있어도 호출할 때에만 단발적으로 동작하는 인터페이스를 실현한다.

4. 실험 및 고찰

4.1 원격 감시 및 제어

서버는 영상정보와 I/O 제어 데이터를 구분하여 처리한다. 영상정보는 DNS Technology 사의 전용 캡처 및 전송장비인 UDVR을 사용하고 디지털 베이스인 I/O 제어 데이터는 하드웨어와 별도 제작한 CGI 프로그램으로 링크하여 실험하였다.

그림 3에 UVDR 서버의 초기화면을 보였다. 화면에는

4대의 CCD 카메라에서 전송되는 영상과 기본적인 상태를 보이고 있으며 다양한 서버메뉴의 선택으로 환경설정 및 운영상태를 감시할 수 있다.

또한 그림 4는 서버 운영중에 KBS 인터넷 방송인 쿡을 실행시켜 오디오 신호를 부가하여 전송하는 화면을 보인 것이며 그림 5는 클라이언트에 표시되는 화면을 보인 것이다. 이들 화면은 사용자의 환경설정에서 요구하는 다양한 감시기능을 정확히 영상정보로 전송하였으며 릴레이, alarm 신호 등 제어동작과 상태도 실시간으로 정확하게 수행하였다.

한편, I/O 장치는 하드웨어 디지털 port에 센서 2 channel을 할당하고 2개의 출력 포트에 on/off 릴레이 구동 채널을 제공하여 SSR(solid-state relay) 소자로 모터 및 밸브 등 동력장치를 구동하게 하였으며 동작상태는 LED로 확인하게 하였다. 각 포트에 접속된 장치는 특성에 따라 부가적인 S/W나 API를 제작하여 구동시키며 시험에서는 제어 시뮬레이션에 의해 기능을 평가하고 성능을 보완하였다.



[그림 3] 서버 시스템 화면(4채널)



[그림 4] 오디오 부가 서버 시스템



[그림 5] 오디오 부가 클라이언트 시스템

4.2 통신 성능시험

4.2.1 네트워크 환경시험

서버측 인터넷 회선은 한국통신의 전화접속 네트워크인 megapass 라이트(50Mbps)급을 사용하였다. 감시 및 제어대상인 온실 model의 위치는 충북 괴산이며 DNS 서버로부터 약 20km 이상 원격지로 측정되어 네트워크 환경이 매우 열악하다고 판단된다.

따라서 실험에서는 가능한 높은 통신품질을 확보하기 위해 시스템 부가기능을 가급적 제한하고 네트워크와 통신환경을 최적화하는 가능한 조치를 선행하였다. 실측은 한국통신에서 제공하는 속도측정 프로그램을 이용하여 수행하였으며 데이터는 수회 반복 측정한 후 평균값으로 취득하였다.

표 1에 서버측 인터넷 회선의 일반적인 전송속도와 품질을 측정한 결과를 보였다. Down/Up의 속도품질이 평균 3Mbps/1Mbps를 유지하므로 열악한 상태이나 원격 감시 및 제어장치를 위한 네트워크 환경으로는 실용 가능하다고 판단된다. 한편, 감시장치 목적의 동영상 송출성능을 평가하기 위하여 500kb 동영상을 수신하는 스트리밍 테스트를 수행하였다. 결과는 기준대역폭 491.0kbps, frame 속도 29.89 frames/sec, fps로 각각 측정되었으며 수신 packet수가 2,387개에서 전체 복구 패킷수는 6, buffering 횟수는 1회, 손실 패킷수는 0으로 측정되어 비교적 양호한 전송상태를 보였다.

[표 1] KT-메가패스 속도측정

속도 품질	Max [Mbps]	Min [Mbps]	Avg [Mbps]	Delay & σ [ms]	
				Ping	UDP
Down	3.917	2.178	3.025	Avg 26 ($\sigma = 0.53$)	Avg 28 ($\sigma = 1.94$)
Up	1.344	0	0.993		

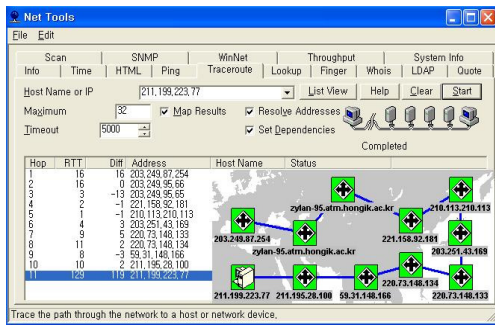
기타 : DNS 질의소요시간: 평균 23ms
CPU 부하율 : 7%, 메모리 점유율 : 73%

KT-Megapass 라이트(50Mbps)급

4.2.2 네트워크 전송시험

네트워크 시험은 Ipswitch사에서 개발한 네트워크 모니터링 유틸리티인 WhatsUp Gold Ver.8.0으로 수행하였다. 그림 6에 Net Tools의 Traceroute 기능을 사용하여 서버-클라이언트간에 설정된 네트워크 맵과 각각의 IP 및 측정된 전송시간을 보였다.

클라이언트는 충남 조치원에 위치하며 전용회선을 사용하는 LAN망에 접속되어 있다. 네트워크 map은 10hops를 구성하며 노드당 평균 20ms 내외의 전송지연시간을 보이고 있다.



[그림 6] 트레이서루트(WhatsUp Gold V8.0)

한편, 서버-클라이언트간 전송성능을 평가하고 최적 통신조건을 설정하기 위하여 샘플 데이터를 50개의 패킷으로 구성하고 패킷 크기를 256bytes에서 16kbytes까지 변화시키면서 전송량에 따른 전송시간과 throughput 및 ping 테스트를 수행하였다.

표 2는 측정결과를 보인 것으로 패킷 크기를 증가시키면 스루풋은 비례하여 증가하였으나 전송시간은 4kbytes까지는 소폭 증가하고 16kbytes 이상에서 비례하여 대폭 증가되었다. 이와 함께 핑 시험에서는 왕복지연시간은 4kbytes까지는 소폭 증가하고 16kbytes 이상에서는 대폭 증가함을 보인다.

[표 2] 성능 실험 (패킷=50개)

Size [Bytes]	Data Transfer test			Ping Test		
	Total [Bytes]	Time [ms]	Avg.Thr [kb/s]	Round trip time [ms]		
				max	min	avg
256	17K	6641	20.96	153	118	134
1k	54K	7019	62.20	168	124	147
4k	290K	8567	270.81	214	165	194
16k	901K	13195	546.58	389	375	378

이상의 시험결과를 고찰하면 네트워크 환경과 전송 패킷

의 크기를 고려하여 최적 전송조건을 설정해야 할 것으로 판단된다.

5. 결론

인터넷을 통해 원격지에 있는 온실의 상태를 실시간으로 감시하고 제어하는 원격제어 시스템을 구현하고 평가하였다. 원격지에 있는 서버 시스템은 웹 서버와 로컬 I/O 제어 프로그램을 탑재하여 4개의 CCD 카메라로부터 취득한 대용량의 화상정보를 영상처리하여 초당 3~30 프레임의 이미지 데이터로 가공하여 연속 전송하였다. 서버는 이와 함께 음향정보 및 릴레이, 센서 등 다양한 입출력 장치와 actuator 등 디지털 I/O data를 event에 동기하여 양방향으로 성공적으로 통신하였다.

이들 데이터는 인터넷에 공개하여 인터넷 연결이 가능한 임의의 원격지에서 실시간 감시 및 제어 동작을 수행하게 하였다.

실험에서는 서버 위치상 네트워크 환경이 대단히 열악하여 영상 데이터는 초당 10frame 이하의 성능에서 정상적인 감시기능을 구현할 수 있었다. 그러나 양방향 음향 및 디지털 신호는 약 200ms 이내의 응답 속도를 보여 실용상 제어기능은 실시간으로 구현되는 성과를 얻을 수 있었다.

본 연구의 결과는 다양한 multimedia data를 다중화하고 전송하여 웹 브라우저에서 종합적으로 관리할 수 있는 기술적 특성을 제시하였다. 따라서 향후 통신환경의 개선 및 초고속망의 확충으로 고성능 고기능 감시 및 제어 서비스를 제공하는 산업기기에 활용할 수 있을 것으로 기대된다.

참고문헌

- [1] H. Schulzrinne, X. Wu, and S. Sidiroglou, "Ubiquitous Computing in Home Networks," *IEEE Comm. Mag.*, Oct. 2003.
- [2] 1394 Forum, *Internet Protocol(IP) based Home Network over IEEE1394 Bus*, Ver 0.9, June, 2000.
- [3] S.T.Park, A.Khrabrov, D.M.Pennpck, S.Lawrence, C.L.Giles, L.H.Ungar, "Static and dynamic analysis of the Internet's susceptibility to faults and attacks", Proc. of INFOCOM, Apr. 2003.
- [4] J. Allard, V. Chinta, S. Gundata, G. Richard III, "Jini Meets UPnP; An Architecture for Jini/UPnP Interoperability," *Proceedings of IEEE Symposium on Applicati*

- ons and the Internet, pp.268-275, Jan. 2003.
- [5] K. Wacks, "Home Systems Standards; Achievements and Challenges," *IEEE Comm. Mag.*, Apr. 2002.
- [6] Paul Ferguson, and Geoff Huston, "*Quality of Service; Delivering QoS on the Internet and in Corporate Networks*", John Wiley & Sons. Inc. *998.
- [7] Alessandro Rubini/Jonathan Corbet,; *Linux Device Drivers*", 2nd ed., O'Reilly Associates, Inc., 2002.
- [8] 이희조, "인터넷 기반구조의 취약성 분석과 보안 강화 방안", *Telecommunications Review*, 제 13권 2호, pp.235-245, 2003년 4월.

장 호 성(Ho-Sung Chang)

[정회원]



- 1991 단국대학교 전자공학과 (공학박사)
- 1991 - 현재 홍익대학교 컴퓨터 정보통신공학과 교수.

<관심분야>

정보 멀티미디어 정보시스템, 텔레매틱스, ITS, 센서 네트워크, 무선통신