인터넷 정보 가전용 iRTOS™를 위한 TCP/IP 설계 및 구현

박희상[0], 차태성, 배장식, 이철훈
충남대학교 컴퓨터공학과
{hspark, tscha}@pplab.ce.cnu.ac.kr, {jsbae, chlee}@ce.cnu.ac.kr

Design and Implementation of TCP/IP based on iRTOS™ for Internet Appliance

Hi-Sang Park[0], Tae-Sung Cha , Jang-Sik Bae, Cheol-Hun Lee
Dept. of Computer Engineering, Chungnam National Univ.

요 약

인터넷 정보가전이란 유·무선 정보통신망에 연결되어 데이터 송·수신이 가능한 디지털 TV, 인터넷 냉장고, DVD, 디지털 비디오 등과 같은 차세대 가전제품을 말하며, 휴대 정보 단말기로 원격 제어하는 홈 시큐리티 기능뿐만 아니라 초고속 인터넷과 연결되어 영화, 음악, 부가정보 등을 전송 받을 수 있어 향후 정보화 사회의 핵심분야로 부각되고 있다. 본 논문에서는 iRTOS™가 인터넷 정보 가전에 응용될 수 있도록 TCP/IP 네트워크 부분을 설계하고 이를 구현한 내용에 대해 기술한다.

1. 서론

정보화 사회로 발전함에 따라 네트워크 기술, 정보기술, 그리고 가전기술로 복합된 정보가전 분야에서 다양한 신제품들이 출시되어 가정에서의 정보 서비스는 물론 언제 어디서나 자유롭게 정보를 교환할 수 있는 이동형 정보 서비스가 보편화 되고 있다. 디지털 TV, 인터넷 냉장고, 인터넷 전화기, PDA 등의 정보 가전은 떠오르는 차세대 정보 제공 제품들로 자리잡고 있고 이들의 기능을 제어하는데 필요한 핵심 소프트웨어인 실시간 운영체제는 날이 갈수록 그 수요가 급격히 증대되고 있다. 그러나 국내에서는 정보 가전용 실시간 운영체제가 개발되지 않아 외국제품의 수입에 의존하고 있고 따라서 정보가전 제품의 판매에 따른 막대한 로열티를 지불하고 있는 실정이다. 다행히도 최근 기술적 열위와 선진국의 전략적 제휴를 극복하기 위한 노력의 일환으로 몇몇 연구기관 및 업체에서 실시간 운영체제를 발표하였다. 본 연구팀은 그 중에서 안정성 및 성능에서 인정 받은 실시간 운영체제인 iRTOS™를 연구대상으로 하였고, 이를 위한 TCP/IP 네트워크를 설계하고 구현하였다.

실시간 운영체제에서 TCP/IP 네트워크를 지원하기 위해서는 같은 우선순위의 태스크(Task)를 허용하는 스케줄링 정책 및 세마포어(Semaphore), 메시지 큐(Message Queue), 메시지 메일박스(Message Mailbox) 등 ITC(InterTask Communications)를 지원해야 한다.

본 논문에서는 2 장에서 iRTOS™의 전체적인 구성을, 3 장에서는 TCP/IP 설계 및 구현 내용을, 4 장에서는 테스트 환경 및 결과에 대해 기술한다. 그리고 5 장에서는 결론 및 향후 연구 과제를 기술한다.

2. iRTOS™

iRTOS™는 실시간 운영체제의 핵심이라고 할 수 있는 멀티태스킹(MultiTasking) 및 ITC 환경을 제공한다. 태스크는 독립적으로 수행되는 실행 프로그램을 의미하며, 멀티태스킹 환경은 여러 개의 태스크가 동시에 수행될 수 있는 스케줄링 정책을 제공함을 의미한다. ITC 환경은 태스크와 태스크의 공동 작업(Cooperation)을 위한 동기화(Synchronization) 및 통신(Communication)을 위하여 세마포어, 메시지 메일박스, 메시지 큐등을 지원함을 의미한다[1].

2.1 태스크 스케줄링 정책

iRTOS™는 우선순위 테이블을 별도로 관리하여 정해진 시간내에 가장 높은 우선 순위의 태스크를 찾을 수 있다. 이를 기반으로 iRTOS™는 서로 다른 우선 순위의 경우, 우선 순위가 높은 태스크를 선점(Preemption)하여 실행하고, 동일한 우선 순위의 태스크들은 타임 슬라이스(Time Slice)동안 차례로 수행되는 라운드 로빈 방식을 따른다[1].

2.2 ITC(InterTask Communications)

iRTOS™는 세마포어, 메시지 메일박스, 메시지 큐를 지원한다. 이들 ITC는 자원(Resource) 및 메시지(Message)를 획득하지 못했을 경우 태스크가 Wait 상태로 대기하게 되는데, 보통 자원 및 메시지를 획득하지 못해도 Time Out 시간 후에는 자동으로 Ready 상태로 복원된다. 이를 통해 무한정 기다리지 않고 적절히 대처할 수 있도록 해준다. 그렇지만, 네트워크 관련해서 ITC를 사용하게 될 경우, 대부분 Time Out을 무한대로 설정하여 사용하게 된다[1].

- 세마포어(Semaphores)
  세마포어는 안전한 공유 자원 관리를 위해 사용될

뿐만 아니라, 동기화(Synchronization) 및 상호 배제(Mutual Exclusion)를 위해서도 사용된다.

- 메시지 큐
  메시지 큐는 특정 태스크나 ISR에서 다른 태스크로 여러 개의 메시지를 전송할 수 있다.

[그림 1] Message Queue

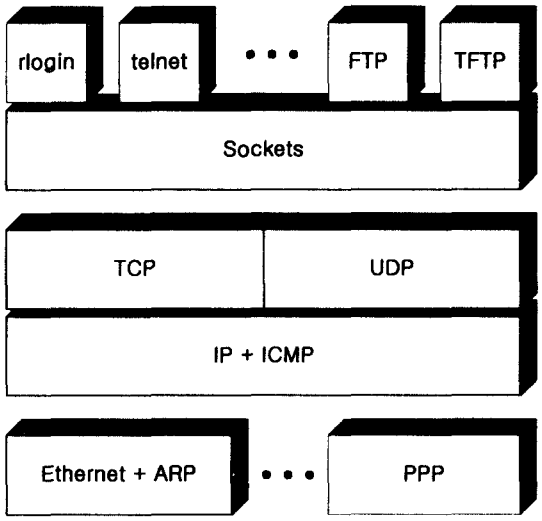
- 메시지 메일박스
  메시지 메일박스는 하나의 자원을 가지는 이진 세마포어(Binary Semaphore)와 유사하지만, 메시지를 수반한다는 점에서 서로 다르다. 메시지 메일박스는 주로 태스크와 태스크 간에 메시지를 포함하는 간단한 동기화를 위해 사용된다. 메시지 큐의 특수한 경우로 이해할 수도 있다.

3. TCP/IP 설계 및 구현

실시간 운영 체제인 iRTOS™를 포함한 멀티태스킹 기반의 운영체제에서 TCP/IP구현의 핵심은 적절한 ITC사용이라고 할 수 있다. 실제로 iRTOS™ 기반의 TCP/IP는 여러 개의 태스크로 구성되어 있는데, 각각의 태스크는 Wait 상태로 대기하고 있다가 처리할 패킷이 들어오게 되면 깨어나게 된다.

본 논문에서 구현한 TCP/IP는 Ethernet을 기반으로 한 것이지만 추후 PPP 등을 추가할 수 있도록 설계하였다. 또한 네트워크 응용프로그램(Application)개발이 용이하도록 소켓(Sockets)을 제공하고 있다.

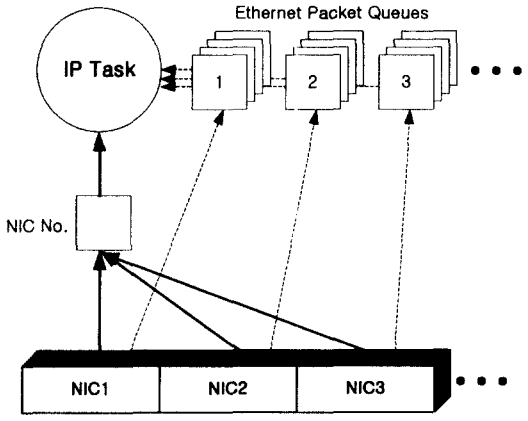
3.1 Protocol Layer



[그림 2] Protocol Layers

3.2 IP 패킷 수신

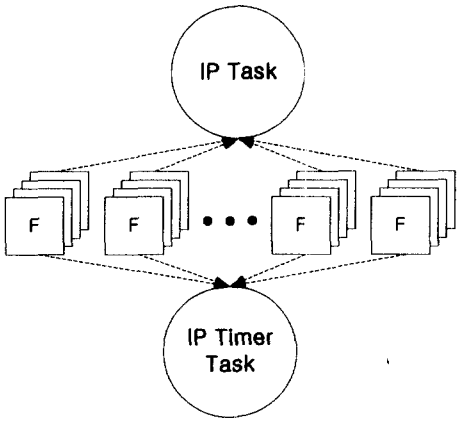
각각의 NIC(Network Interface Card) 카드는 IP와 IRQ(Interrupt Request) 번호를 별도로 가진다. 또한 송·수신에 필요한 각각의 큐를 가진다. IP 태스크는 메시지 메일박스를 생성한 후 Wait 상태로 대기하다가 ISR(Interrupt Service Routine)으로부터 메시지가 도착하면 깨어난다. 이때 받은 메시지는 NIC 카드의 번호가 되는데, 이 값을 인덱스(Index)로 해당 NIC 카드의 큐를 참조하여 수신된 패킷을 처리하게 된다. 반대로, ISR은 수신된 패킷을 해당 큐에 넣고 IP 태스크에 메시지(NIC 카드 번호)를 보낸다. 이로써 여러 개의 NIC 카드가 하나의 큐를 공유하게 될 때 메시지에 NIC 카드 번호를 포함시키기 위해 별도의 형식 (Format)을 갖추어야 할 필요가 없게 된다.



[그림 3] IP 패킷 수신

3.3 IP 패킷 처리

IP 패킷은 송·수신을 모두 IP 태스크에서 수행한다.



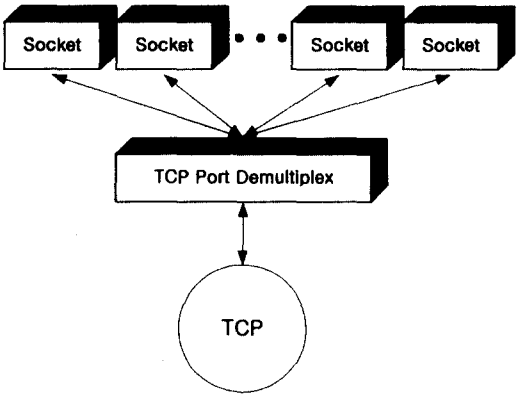
[그림 4] Fragment IP 패킷 처리

프레그먼트(Fragment) 패킷은 Flags 비트 중 MF 비트가 1(More Fragments)일 경우 MK 비트가 0(Last Fragment)이 설정된 패킷이 수신될 때까지 큐에서 대기한다. 만약 설정한

TTL(Time To Live)이 0 이 될 때까지 처리되지 않으면 IP Timer 태스크가 자동으로 리스트에서 제거하게 된다. 반대로 TCP 및 UDP 패킷을 여러 개의 IP 패킷으로 나누어 전송할 경우에는 IP 헤더(Header)를 복사한 후 TCP 패킷들을 복사하고 Flags 비트를 설정한 후 전송하게 된다[6].

3.3 TCP 패킷 처리

TCP 패킷은 수신 패킷과 송신 패킷을 각각의 태스크가 처리한다. [그림 5]는 TCP 수신 패킷 처리를 설명한 것이다.



[그림 5] TCP 수신 패킷 처리

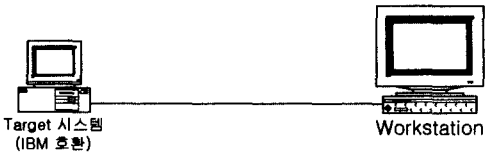
IP 태스크로부터 TCP 패킷을 수신하게 되면 TCP 태스크가 깨어나게 된다. TCP 태스크는 수신된 TCP 패킷의 소스 포트(Source Port)와 목적지 포트(Destination Port)를 확인하여 해당 소켓(Socket)을 찾아내게 된다. 각각의 소켓에는 버퍼(Buffer)를 포함하고 있는데, TCP 태스크는 수신된 데이터를 이 버퍼에 복사(Copy)하게 된다.

TCP 패킷 전송은 앞서 언급한 바와 같이 별도의 태스크에서 수행된다. 또 TCP의 경우 동기(SYNC)를 맞추기 위해 주기적으로 TCP 패킷을 전송해야 하는데 이를 위해 TCP Timer 태스크가 존재한다. TCP Timer 태스크는 주기적으로 깨어나 큐에 대기하고 있는 TCP 패킷이 딜레이(Delay)를 채운 경우 전송하게 된다.

4. 테스트 환경 및 결과

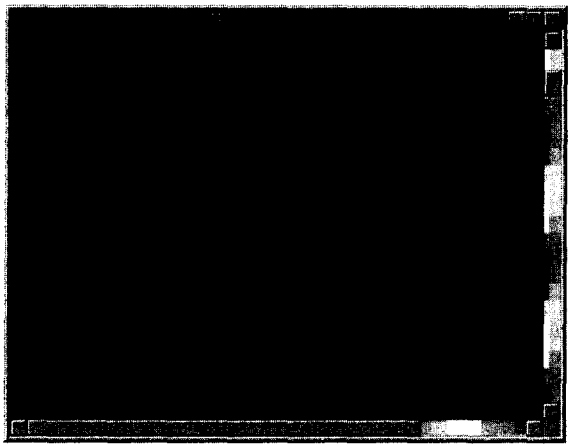
본 논문의 구현은 Microsoft Visual Studio™ 를 사용하여 컴파일(Compile) 하였다. 커널(36KByte)을 포함하지 않은 네트워크 이미지(Image) 크기는 ARP, IP, ICMP, TCP, UDP를 포함하여 약 44Kbyte이다. 참고로 Nuclueus NET의 경우, TCP, UDP, IP, ICMP, iGMP, ARP, RARP, BOOTP, DNS(resolver), DHCP(client), RIP/RIP2, TFTP(client)를 포함하여 약 75 Kbyte이다.

테스트 환경은 [그림 6]과 같이 3COM™의 3C509b NIC를 탑재한 IBM 호환 PC를 타겟(Target) 시스템으로 설정하였다. 그리고 Workstation에서 구현된 TCP/IP 소켓 프로그램을 실행하여 문자 메시지를 전송하면 타겟 시스템에서 에코(echo)하도록 하였다.



[그림 6] 테스트 환경

[그림 7]은 Workstation에서 타겟 시스템에 TCP 연결 후 메시지를 전송하면 이에 대한 에코(echo)된 메시지를 출력하는 화면이다. 그리고, 마지막에 ^ D를 눌러 연결을 종료한다.



[그림 7] Echo 테스트

5. 결론 및 향후 연구 과제

본 논문에서는 인터넷 정보 가전을 위한 선점형 실시간 운영 체제인 iRTOS™에서 TCP/IP 네트워크를 설계하고 이를 구현된 내용을 설명하였다. 이로써 iRTOS™가 정보 가전에 활용할 수 있는 토대를 마련했다고 할 수 있다. 정보 가전이 일반화 하면서 IPv6 에 대한 관심이 높아가고 있는데, 이에 대한 부분은 계속 연구되어야 할 것이다. 또한 인터넷 정보 가전과 관련한 JAVA 응용프로그램 개발에 관심이 커지고 있는데, 이를 위해 iRTOS™에 JVM(Java Virtual Machine)을 지원하는 부분도 계속 연구되어야 할 부분이다.

6. 참고문헌

[1] http://www.inestech.com
[2] Jean, J. Labrosse, " μ C/OS The Real-Time Kernel" , R&D Publications, 1992.
[3] D. Comer " Operation System Design VOL 1 : THE XINU APPROACH" , 1988.
[4] IEEE Std 1003.1b, " Portable Operating System" , 1993.
[5] W. Richard Stevens, " UNIX NETWORK PROGRAMMING" , 1997.
[6] RFC 791, 792, 793, 826 외