

論文97-34S-7-2

대용량 통신처리시스템의 전화망 정합 장치의 통신 모듈 구현 및 성능 분석

(Implementation and Performance Evaluation of the Communications Module of TNAS in the Advanced CPS)

金 建 石 * , 趙 平 東 *

(Kun Suk Kim and Pyung Dong Cho)

요 약

본 논문에서는 대용량 통신처리시스템의 전화망 정합 장치 상에서 통신 모듈을 구현하였다. 전화망 정합 장치의 서비스 처리 보드에서 실행되는 태스크들, 채널 구분자 전달용 큐, 송신용 큐들, 수신용 큐들 등과 같은 자원들의 구성과 각 태스크들의 동작 절차를 제안하였다. 전화망 정합 장치에서 통신 처리 서비스를 담당하는 서비스 처리 보드의 시스템 이용률과 버퍼의 크기 등의 성능을 평가하기 위하여 트래픽 모델을 정의하고 시뮬레이션하였다. 대용량 통신처리시스템의 고속 스위칭 구조 서브시스템은 전화망, 데이터망, 프레임 릴레이망, ATM 망 등 다양한 망과의 접속을 지원해야 하는데, 본 논문에서 구현한 통신 모듈은 이러한 망 접속을 지원하는 망 정합 장치에서 서비스 처리 보드와 고속망 어댑터 사이의 인터페이스에 적용이 가능하다.

Abstract

In this paper, we implemented the communication module in the Telephone Network Access Subsystem(TNAS) of the Advanced Communications Processing System(ACPS). We defined some kinds of communication tasks and related resources like several queues which are executed in real-time operating system, and implemented the procedures for processing the user information. Through traffic modeling and simulation, the performance of the Service Processing board Assembly(SPA) is evaluated in the aspects of system utilization and buffer size. The ACPS should accommodate various public networks such as public switched telephone network, packet switched data network, frame relay network, and ATM network. The communications module proposed in this paper could be used in the interface between the SPA and the High Speed Network Adaptor of other network interface subsystems.

I. 서 론

고도 정보화 사회의 도래로 정보유통 서비스에 대한 수요가 급증하고 있으며 이에 따라 정보산업이 이미 중요한 국가산업이 되어 이들을 원활히 유통할 수 있

는 정보유통 서비스가 통신산업의 중요한 대상이 되고 있다. 정보유통을 위한 기존 부가서비스는 개인단말에 부착된PAD(Packet Assembly Disassembly)를 통해 패킷망과 접속하고 전화망 가입자에게 데이터 서비스를 제공하는 형태로 이루어져 왔으나 급증하는 이용자 수요를 충족하고 기존 2,400 bps 보다 향상된 9,600 bps 또는 14,400 bps급의 고속 서비스를 제공하기에는 기능 및 확장성에서 한계를 보여왔다.

통신처리시스템(Communications Processing

* 正會員, 韓國電子通信研究院

(Electronics and Telecommunications Research Institute)

接受日子: 1996年1月23日 수정완료일: 1997年7월3일

System: CPS)은 가입자망과 서비스 제공망을 연결하는 관문국 형태의 시스템으로서 교환기와 트렁크 레벨에서 접속되어 데이터 서비스를 요구하는 이용자들을 패킷망과 연결시키는 역할을 수행한다^{[1] [2] [3] [4] [5] [6]}. 통신처리시스템을 통해 이용자는 특별한 장치의 필요없이 모델변조를 통한 데이터 서비스를 이용할 수 있으며 이러한 형태의 공중 데이터 서비스망 구성을 통해 정보통신 서비스의 광역화 및 보편화를 앞당기는 계기를 마련하였다.

한편, 현재의 통신처리시스템은 전화망(Public Switched Telephone Network: PSTN) 교환기와 패킷망 교환기 사이의 직접적인 통신만을 제공하도록 되어 있는데 비하여 개방형 대용량 통신처리시스템(Advanced CPS: ACPS)은 가입자 측의 PSTN, 데이터망(Packet Switched Data Network: PSDN), ISDN 등의 교환기와 정보 제공자 측의 PSDN, 프레임 릴레이망(Frame Relay Network), ATM망, Internet 등의 연동을 제공하여야 한다(그림 1). 이와 같이 대용량 통신처리시스템은 다양한 공중망들을 수용해야 하므로 기존의 통신처리시스템과는 다른 구조를 가지게 되고, 보편적인 망 접속 인터페이스 기능을 필요로 한다. 대용량 통신처리시스템의 통신 방식에 있어서도 기존의 두 망 사이의 직접 통신과는 달리 모듈 간의 통신 형태로 바뀌었다.

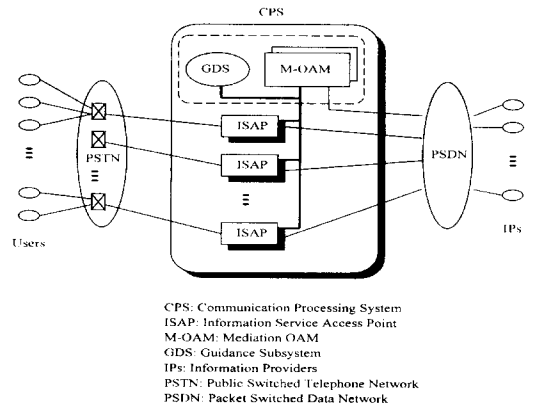
본 논문에서는 대용량 통신처리시스템의 새로운 구조에 적합하도록 설계된 전화망 정합 장치(Telephone Network Access Subsystem: TNAS) 상에서 데이터 통신을 담당하는 두 보드인 서비스 처리 보드(Service Processing board Assembly: SPA)와 고속망 어댑터(High Speed Network Adaptor: HSNA) 보드 간의 데이터 전송을 위한 통신 모듈을 설계하고 구현하였다. 그리고 SPA의 통신 모듈 상에서 사용되는 버퍼의 효율성과 서버 역할을 하는 프로세서에 대한 성능 평가를 하였다.

본 논문은 1장의 서론에 이어, 2장에서는 대용량 통신처리시스템의 전체적인 구조와 전화망 정합 장치에 대하여 설명하였다. 3장에서는 전화망 정합 장치의 고속망 어댑터 인터페이스 상에서 이용자의 정보 처리를 담당하는 통신 모듈을 제공하기 위하여 실시간 운영체제(Real Time Operating System: RTOS)에서 실행되는 통신 태스크(task)와 통신에 필요한 자원(resource) 등을 정의하고, 이용자 정보 처리 절차를

구현하였다. 4장에서는 TNAS에서 통신 처리 서비스를 담당하는 SPA의 성능을 평가하기 위하여 트래픽 모델을 정의하고 시뮬레이션하였다. 마지막으로 5장에서 결론을 다루었다.

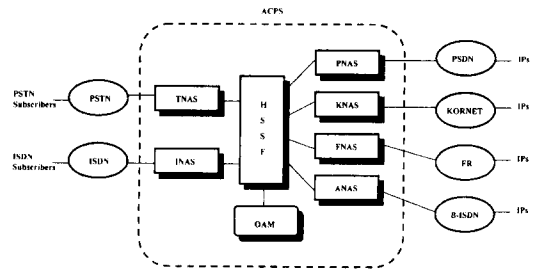
II. 대용량 통신처리시스템 및 전화망 정합 장치

개방형 정보 통신 기반 구조를 구축하려는 대용량 통신처리시스템은 정보 검색, 멀티미디어 정보 통신 서비스, 정보 이용료 회수 대행, 인터넷 접속 서비스 등과 같은 다양한 서비스들을 지원한다. 뿐만 아니라 전화망과 패킷망 접속에서 기존의 통신처리시스템에 비하여 고속 및 대용량화된 서비스 처리 기능을 제공한다.



CPS: Communication Processing System
 ISAP: Information Service Access Point
 M-OAM: Mediation OAM
 GDS: Guidance Subsystem
 IPs: Information Providers
 PSTN: Public Switched Telephone Network
 PSDN: Packet Switched Data Network

(a)



ACPS: Advanced CPS
 TNAS: Telephone Network Access Subsystem
 FNAS: Frame Relay Network Access Subsystem
 INAS: ISDN Access Subsystem
 HSNF: High Speed Switching Fabric
 ANAS: ATM Network Access Subsystem
 PSNA: Packet Network Access Subsystem
 KNAS: KORNET Access Subsystem

(b)

그림 1. 통신처리시스템 구조의 발전 모델
 (a) 기존 통신처리시스템 구조 (b) 대용량 통신처리시스템의 구조

Fig. 1. Evolutional model of CPS structure.
 (a) The structure of CPS (b) The structure of ACPS

그림 1은 대용량 통신처리시스템 구조의 발전 모델이다. 대용량 통신처리시스템은 PSTN, PSDN, 프레임 릴레이망, ATM 망 등과 같은 다양한 망 접속을 수용한다^[12]. 고속 스위칭 구조(High Speed Switching Fabric: HSSF) 서브시스템은 대용량 통신처리시스템을 구성하는 여러 종류의 망 정합 모듈들과 서비스 지원 모듈 및 통신처리 모듈을 서로 고속으로 연결하고 서비스 트래픽의 수송을 담당하는 내부 고속 상호 연동망의 기능을 제공한다. TNAS는 전화망에 접속되어 있는 이용자에게 대용량 통신처리시스템의 다양한 통신 서비스를 제공하기 위한 전화망 정합 장치이다.

TNAS는 전화망과의 인터페이스를 위한 전화망 정합(Telephone Network Interface: TNIF) 모듈, HSSF와의 인터페이스를 위한 HSNA 모듈, 그리고 이들을 제어하고 서비스를 처리하는 서비스 처리(Service Processing: SP) 모듈 등으로 구성되어 있다(그림 2). TNIF 모듈은 4T1(또는 3E1)을 기본으로 하여 서비스 접속 장치와 연결하는 트렁크 인터페이스 모듈과 가입자 호 요구 인지, 채널 상태 감시, 아날로그-디지털 변환, V.34 및 V.42/V.42bis 데이터 변복조, 채널별 TDLA 번호 발생 등의 기능들을 지원하는 가입자 모뎀 인터페이스 모듈(Telephony Dial-up Line Assembly: TDLA)로 구성되어 있다. 하나의 TDLA는 8 채널 가입자들을 수용하도록 설계되었다.

SP 모듈은 가입자 모뎀 속도 판별, 모뎀 오류 검출, 가입자 정보 송수신, 초기화 및 자체 진단 등의 기능들을 처리하는 DPA(Data Processing board Assembly)와 서비스 처리 및 제어, 전화망 정합 능력과 통신 기능(가입자 호처리 기능 및 장애 정보 수집 기능), DPA와 직렬 버스 인터페이스 기능, HSNA와 VME 버스 인터페이스 기능 등을 담당하는 SPA로 이루어져 있다. 하나의 DPA는 32 채널 가입자들을 수용 가능하고, SPA는 96 채널 가입자들을 수용하도록 하였다. 그리고 HSNA 모듈은 HSSF 인터페이스 기능, 프레임 송수신 기능, SPA와의 프로세스 간의 통신(Interprocess Communication: IPC)기능^[15] [16] 등을 수행한다.

TNAS의 SPA와 HSNA보드 사이의 인터페이스를 TNAS-HSNA 인터페이스라 한다. SPA는 HSNA보드를 초기화하고 상태를 확인하며 데이터를 송수신하

는데, 이 기능들을 수행하기 위해서는 TNAS-HSNA 인터페이스 프로토콜 및 통신 기능이 필요하다.

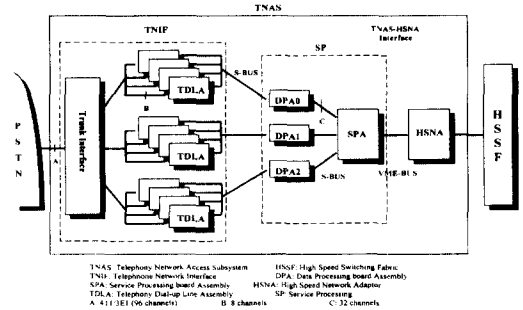


그림 2. TNAS 기능 구조도

Fig. 2. The functional diagram of TNAS.

III. 통신 모듈 구현

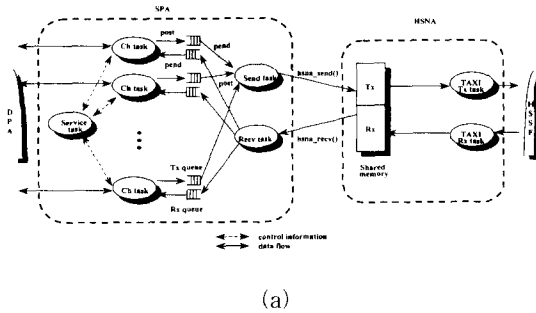
1. 통신 모듈의 구조

그림 3에서는 TNAS-HSNA 인터페이스 통신 모듈 구성도를 나타내었다. 서비스 태스크는 전화망 가입자의 접속이 있을 때마다 하나의 채널 태스크를 생성하여 데이터 처리 서비스를 할 수 있도록 하고 접속 해제시에는 채널 태스크를 제거하는 역할을 한다. 채널 태스크는 통신 모듈 내에 여러개 존재할 수 있으며 자신의 송신용 큐와 수신용 큐를 가진다.

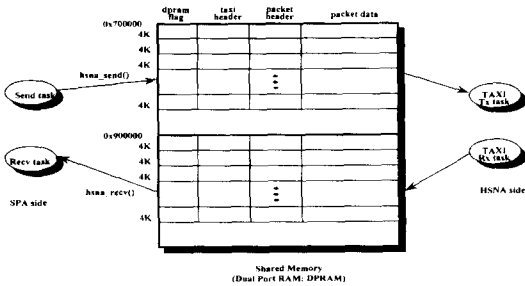
채널 태스크는 HSNA로 전송할 이용자의 정보가 있으면 그 정보를 패키징하여 이 채널의 송신용 큐에 포스트(post)한다. 채널 태스크는 자신의 수신용 큐를 펜드(pend)하여 수신된 패키지가 있으면 메모리에 저장한다. 송신용 태스크는 통신 모듈 내에 하나가 존재하며, 각 채널의 송신용 큐를 펜드하여 큐가 비어 있지 않으면 hsa_send 프리미티브를 사용하여 이용자 데이터를 HSNA로 송신한다. 수신용 태스크는 hsa_recv 프리미티브를 사용하여 HSNA로부터 패키지를 수신하여 해당 채널의 수신용 큐에 포스트한다.

SPA의 프로토콜 정합 능력은 SPA 보드 상에서 구현되며, HSNA 프로토콜 정합 능력은 HSNA 보드 상에 구현 탑재되어 있다. SPA와 HSNA 사이의 데이터 전송은 HSNA 상에 있는 공유 메모리인 DPRAM(Dual Port RAM)을 이용한다. HSNA로부터의 데이터 수신을 위해 SPA는 이 공유 메모리를 주기적으로 폴링하는 방식을 사용한다. SPA로부터 HSNA로의 통신을 위해 HSNA에는 공유 메모리의

0x700000 주소로부터 약 4 M 바이트 크기의 영역이 존재하는데, 이 중에서 0x700000 주소로부터 0x1000 바이트 크기의 블록이 32개 예약되어(reserved) 있다. 또한 HSNA로부터 SPA로의 통신을 위해 HSNA에는 공유 메모리의 0x900000 주소로부터 약 4 M 바이트 크기의 영역이 존재하는데, 이 중에서 0x900000 주소로부터 0x1000 바이트 크기의 블록이 32개 예약되어 있다.



(a)



(b)

그림 3. TNAS-HSNA 인터페이스 통신 모듈 구성도 (a) 통신 모듈 구조 (b) SPA와 HSNA 간의 IPC를 위한 공유 메모리 구조

Fig. 3. Structure of the TNAS-HSNA interface communications processing module.

(a) Communication module (b) Shared memory for providing the IPC between SPA and HSNA

그림 3에는 SPA와 HSNA 간의 통신을 위한 태스크들 간의 구조(그림 3a) 및 공유 메모리 구조(그림 3b)를 나타내었다. 그림 3b에 나타난 것과 같이 메시지 블록의 첫번째 필드(dpram flag)는 전송할 유효 메시지가 이 블록에 존재하는지의 여부를 표시하기 위한 플래그로 사용된다. 그리고 두번째 필드(taxi header)는 TAXI 프리미티브의 헤더 부분인데 TAXI 인터페이스 상의 제어 정보를 가진다. 세번째 필드(packet header)와 마지막 필드(packet data)는 각각

HSNA 프리미티브의 헤더와 데이터 부분을 나타낸다.

2. 통신 모듈 구현 환경: VRTX 시스템과 TNX 모듈

본 논문에서 구현한 TNAS의 통신 모듈은 고성능의 실시간 커널인 VRTX(Versatile Real-Time Executive) 상에서 수행되는데, 이 소프트웨어 요소(component)는 새김(embedded) 마이크로프로세서 응용들을 위한 실시간 및 멀티태스킹 운영 체계를 제공한다^{[9] [10]}. VRTX는 실시간 시스템을 지원하기 위하여 멀티태스킹 지원, CPU 스케줄링, 통신, 메모리 할당 등과 같은 메커니즘들을 제공한다. VRTX에서 기반한 시스템은 기능에 따라서 계층화되어 있는데, 각 계층은 하위 계층에서 제공되는 기능들을 사용한다(그림 4). 그림 4에서 검게 표시한 부분은 VRTX 운영 체제 메커니즘들이다. 인터럽트 서비스 루틴(Interrupt Service Routine: ISR)은 특정 주변 장치들에 대한 인터럽트를 제어하는 하드웨어 종속적인 코드 세그먼트이다. 사용자는 VRTX가 제공하지 않는 운영 체제 메커니즘들을 지원하기 위하여 시스템 호 제어거나 VRTX 기능 확장 등을 정의할 수 있다. 전체적인 시스템 구조는 수직적으로 세 부분으로 나누어지는데, 기본적인 시스템 호를 지원, 인터럽트 지원, 사용자 정의 기능 확장 메커니즘들이 있다.

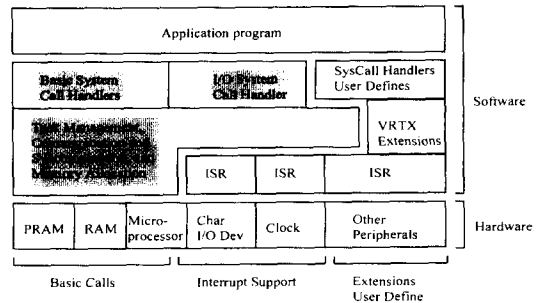


그림 4. VRTX 시스템 구조
Fig. 4. VRTX architecture.

TNX(TCP/IP Network Executive)는 VRTX에서 실행되는 응용 프로그램과 TCP/IP 통신 프로토콜을 사용하는 이더넷 연결상의 시스템 사이에 통신 기능을 제공한다^{[9] [11]}. TNX는 그림 5에 나타난 것과 같이 계층 구조로 이루어져 있다. TNX는 망 수준의 통신 수단을 제공하기 위한 프로토콜의 집합인 인터넷 프로

토콜 군(Internet protocol family: INET)을 지원한다.

TNX에서 지원하는 INET에는 데이터그램 프로토콜(User Datagram Protocol: UDP), 스트림 프로토콜(Transmission Control Protocol: TCP), 에러 메시지 프로토콜(Internet Control Message Protocol: ICMP), IP(Internet Protocol), ARP(Address Resolution Protocol), RARP(Reverse Address Resolution Protocol), 트레일러 프로토콜(Trailer Protocol) 등이 포함된다.

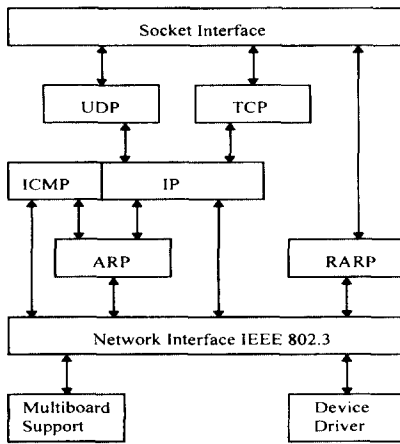


그림 5. TNX 구조
Fig. 5. TNX structure.

3. 통신 모듈 태스크들의 동작 절차

그림 6에서는 TNAS-HSNA 인터페이스 통신 모듈의 동작 절차를 나타내었다. 통신 모듈에서는 TNAS의 SPA와 HSNA 사이의 통신을 지원하기 위한 태스크들과 큐를 생성한다.

먼저, 채널 구분자(Channel Identifier) 전달용 큐를 생성한다. 이 큐는 채널 태스크가 생성될 때에 통신 모듈 내에 있는 여러 채널 태스크들이 각자의 고유한 구분자를 갖도록 하기 위한 것이다. TNAS-HSNA 인터페이스 상에서 송신을 담당하는 송신용 태스크와 수신을 담당하는 수신용 태스크를 각각 생성한다. 통신 모듈에서 지정한 수만큼의 채널 태스크가 생성되지 않았다면, 채널 구분자 전달용 큐에 생성할 채널의 구분자를 포스트하고 채널 태스크를 생성한다. 지정된 수 만큼의 채널 태스크의 생성이 완료되면 루프(loop)를 벗어나서 다음 동작을 수행하게 된다.

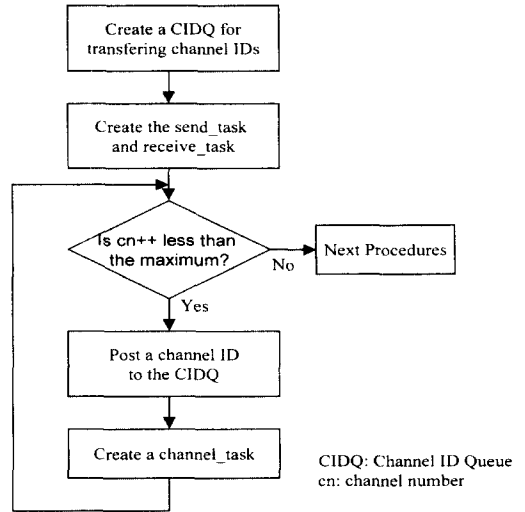


그림 6. TNAS-HSNA 인터페이스 통신 모듈 동작 절차
Fig. 6. Operating procedures of the TNAS-HSNA interface module.

그림 7에서는 채널 태스크의 동작 절차를 나타내었다. 채널 구분자를 얻기 위하여 채널 구분자 전달용 큐를 펠드한다. 각 채널 태스크는 송신용 큐와 수신용 큐를 생성하고 초기화한다.

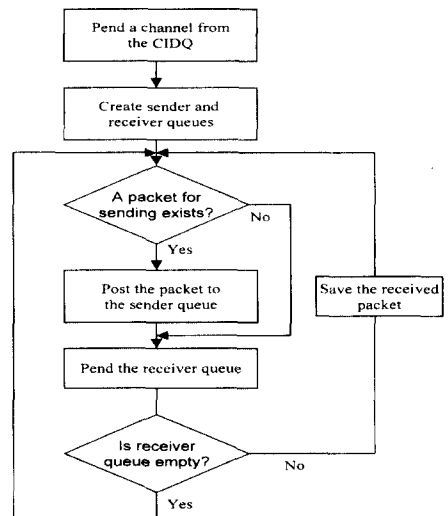


그림 7. 채널 태스크 동작 절차
Fig. 7. Channel task's procedure.

송신할 데이터가 있으면 송신 패킷을 만들고 채널의 송신용 큐에 포스트한다. 송신할 패킷이 없으면 바로

수신용 큐를 펜드하는 과정을 수행하게 된다. 채널의 수신용 큐를 펜드하여 비어 있지 않으면 수신한 패킷을 저장한다. 송신 패킷을 송신용 큐에 포스트하고 수신용 큐를 검사하여 패킷을 전송하는 위의 과정들을 반복 수행한다.

그림 8에서는 송신용 태스크의 동작 절차를 나타내었다. 송신용 포트(SEND_PORT)를 사용하여 SPA에서 HSNA로의 통신 세션을 설립한다. 통신 모듈 내에서 실행되는 채널 태스크들의 송신용 큐를 차례로 펜드하기 위하여 채널 번호 변수를 초기화한다.

채널의 송신용 큐를 펜드하여 큐가 비어 있는지를 체크한다. 큐가 비어 있지 않으면 송신 패킷을 hсна_send 프리미티브를 사용하여 HSNA로 전송한다. 다음 채널을 선택하기 위하여 채널 번호 변수를 변경한다. 송신용 큐를 펜드하여 패킷을 송신하는 위의 과정들을 반복 수행한다.

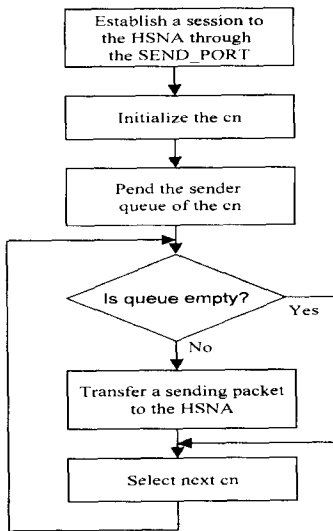


그림 8. 송신용 태스크 동작 절차
Fig. 8. Send task's procedure.

그림 9에서는 수신용 태스크의 동작 절차를 나타내었다. 수신용 포트(RECV_PORT)를 사용하여 HSNA로부터의 통신 세션을 설립한다.

hsна_recv 프리미티브를 사용하여 HSNA로부터 패킷을 수신한다. 수신한 패킷의 채널 번호 필드에서 해당 채널 번호를 구하여 그 채널의 수신용 큐에 포스트한다. 패킷을 수신하여 수신용 큐에 포스트하는 위의 과정들을 반복 수행한다.

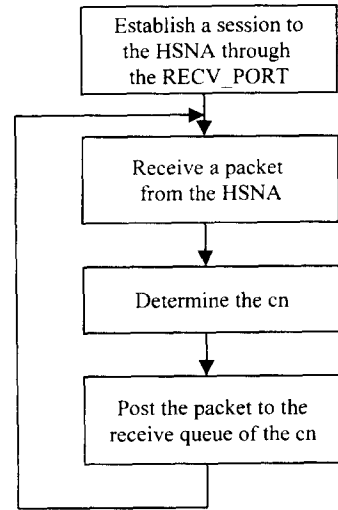


그림 9. 수신용 태스크 동작 절차
Fig. 9. Receive task's procedure.

IV. 시뮬레이션

1. 트래픽 모델링

본 논문의 시뮬레이션에서는 TNAS에서 통신 처리 서비스를 담당하는 SPA의 성능을 평가하기 위하여 트래픽 모델을 다음과 같이 정의하였다. 먼저, n 개의 상호 독립적인(mutually independent) 트래픽 소스들의 집합이 있다. 이들 중에서 i 번째 트래픽 소스의 패킷 도착 간격 시간인 X_i 는 도착률(arrival rate) $\lambda_i (i=1, 2, \dots, n)$ 를 파라미터로 가지는 지수 분포(exponential distribution)를 따른다. 여기서 포아송 프로세스와 지수적 도착 간격 시간(exponential interarrival time) 사이의 관계로 인하여, $i=1, 2, \dots, n$ 에 대하여 시간 구간 $(0, t]$ 에서 i 번째 트래픽 소스의 패킷 도착 수인 $N_i(t)$ 는 도착률이 인 포아송 프로세스가 된다. 이때 시간 구간 $(0, t]$ 에서 시스템에 도착한 전체 패킷의 수인 $N(t)$ 는 n 개의 독립적인 포아송 프로세스의 중첩(superposition of Poisson processes)이 된다^[13]. 즉, $N(t)$ 는 도착률이 λ 인 포아송 프로세스이다. 여기서, $\lambda = \sum \lambda_i$ 이다. 이것은 패킷 도착 간격 시간들은 도착률 λ 를 파라미터로 가지는 지수 분포라는 것을 의미한다.

본 논문에서 사용한 M/M/1 큐잉 시스템의 시뮬레이션 모델을 그림 10에 나타내었다. 시뮬레이션에서는

96개의 독립적인 트래픽 소스들을 사용하였다. 각 채널 태스크마다 FIFO(First Input First Output) 특성을 가지는 큐를 두었다. 시뮬레이션에서 SPA 서버는 128 바이트(bytes)로 구성된 패킷을 초당 100개 처리할 수 있다고 가정하였다.

시간 T 동안에 A 개의 패킷이 도착하였을 때에 도착률은 $\lambda = A / T$ 이다^[14]. 평균 서버 사용 시간(mean server busy time)을 B 라고 할 때에 서버의 이용률(server utilization)은 $U = B / T$ 이다. L_q 를 시스템의 버퍼 내에 있는 패킷의 평균 수이고, W_q 를 평균 큐잉 시간(mean queueing time)이라고 할 때에 리틀의 법칙(Little's Law)에 의하여 $L_q = \lambda W_q$ 이다.

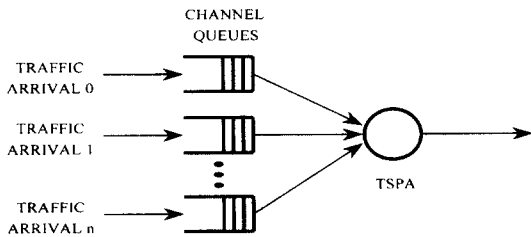


그림 10. 큐잉 모델
Fig. 10. Queueing model.

두 부류(class)의 트래픽 소스들이 있다고 가정한다. 클래스 1은 많은 대역폭을 요구하는 서비스이고 클래스 2는 적은 대역폭을 요구하는 서비스이다. 소수의 대량 데이터 전송 서비스는 클래스 1에 속하고, 대부분의 이용자 인터랙티브(interactive) 서비스는 클래스 2에 속한다. 동일한 부류의 트래픽 소스들을 가지는 경우와 두 부류의 트래픽 소스들을 가지는 경우로 나누어서 시뮬레이션하였다. 전자의 경우는 96개의 채널 태스크로부터 발생하는 트래픽 소스들이 동일한 도착률을 가진다고 가정하였다. 후자의 경우는 96개의 채널 태스크들 중에서 특정 수의 트래픽 소스들은 대량의 패킷을 발생하는 클래스 1 트래픽이고, 다수의 나머지 소스들은 적은 수의 패킷을 발생하는 클래스 2 트래픽이라고 가정하였다.

2. 동일한 부류의 트래픽 소스들에 대한 시뮬레이션

96개의 채널 태스크들은 동일한 부류의 트래픽을 발생시킨다고 가정한다. 도착률이 $\lambda_1 = \lambda_2 = \dots = \lambda_n (= \lambda)$ 인 n 개의 상호 독립적인 포아송 도착 스트림이 있고, 서비스률(service rate)이 μ 인 하나의 서버가 있다.

포아송 프로세스의 중첩 특성을 기반으로 하여 전체 도착률(λ_s)은 $\lambda_s = \sum \lambda_i = n\lambda$ 이다.

동일한 부류의 트래픽 소스들을 사용할 경우에, 다양한 도착률에 대하여 96개의 트래픽 소스들 중에서 임의의 한 트래픽 소스의 k 번째 패킷이 도착했을 때 그 소스의 버퍼에 대기하고 있는 패킷 수의 추이를 그림 11에 나타내었다. 여기서 서버의 서비스률은 이고, 한 트래픽 소스의 도착률은 $\lambda = 0.8, 0.9, \dots, 1.007$ 이다. 도착률이 증가함에 따라 버퍼 길이가 커지지만 그 증가폭은 작음을 알 수 있다.

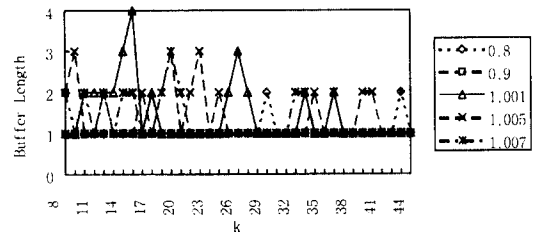


그림 11. 동일한 부류의 트래픽 소스를 사용하였을 경우에, 임의의 한 트래픽 소스의 k 번째 패킷이 도착했을 때 이 소스의 버퍼에 대기하고 있는 패킷의 수

Fig. 11. For any one of traffic sources of the same class, the number of packets in this source's own buffer at the time of the arrival of the k th packet.

표 1. 동일한 부류의 트래픽 소스를 사용했을 때, 도착률에 따른 서버 이용률 및 평균 버퍼 길이

Table 1. For traffic sources of the same class, server utilization and mean buffer length according to arrival rate.

Arrival Rate	Results	Server Utilization	Mean Buffer Length
0.6		0.5773	1.194
0.7		0.6772	2.121
0.8		0.7898	3.572
0.9		0.8678	6.043
1.001		0.9632	27.991
1.005		0.9772	31.047
1.007		0.9829	39.081

동일한 부류의 트래픽 소스들을 사용할 경우에, 다양한 도착률에 대한 SPA 서버의 이용률과 시스템 전체적인 버퍼의 평균 크기를 표 1에 나타내었다. 도착률

이 증가할수록 서버의 이용률은 증가하지만 시스템의 전체적인 평균 큐 길이도 증가한다.

패킷을 위한 버퍼를 각 채널 태스크마다 두는 경우가 전체 채널 태스크들이 공유하는 하나의 버퍼를 두는 것보다 훨씬 더 비효율적이라는 것을 알 수가 있다. 예를 들어, 그림 11과 같이 한 트래픽 소스의 도착률이 0.9인 경우에 각 소스에 할당해야 하는 최대 버퍼 크기가 4이므로 전체 96개의 채널 태스크들이 필요로 하는 버퍼 크기는 384가 된다. 한편, 공유 버퍼를 사용할 경우에는 표 1에 나타난 것과 같이 평균 버퍼 길이가 6.043이다.

3. 두 부류의 트래픽 소스들에 대한 시뮬레이션

96개의 채널 태스크들 중에서 일부는 클래스 1 트래픽을 발생시키고 나머지는 클래스 2 트래픽을 발생시킨다고 가정한다. 도착률이 λ_1 인 m 개와 λ_2 인 $(n-m)$ 개의 상호 독립적인 포아송 도착 스트림이 있고 서비스율이 μ 인 하나의 서버가 있다. 이 도착 스트림의 전체적인 도착률(λ_s)은 $\lambda_s = m\lambda_1 + (n-m)\lambda_2$ 이다.

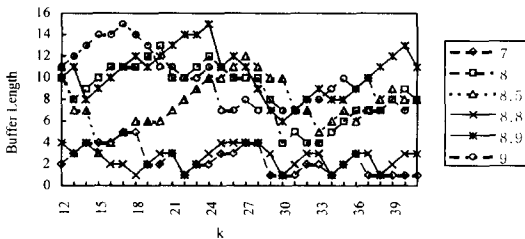


그림 12. 클래스 2 트래픽의 도착률을 고정하고 클래스 1 트래픽의 도착률을 변화시킬 경우에, 클래스 1의 임의의 한 트래픽 소스의 k번째 패킷이 도착했을 때 이 소스의 버퍼에 대기하고 있는 패킷의 수

Fig. 12. For any one of the class 1 traffic sources in case of fixing the arrival rate of class 2 ones and varying that of class 1 ones, the number of packets in this source's own buffer at the time of the arrival of the kth packet.

클래스 2 트래픽 소스들의 도착률을 $\lambda_2=0.5$ 로 고정시키고 6개의 클래스 1 트래픽 소스들의 다양한 도착률이 $\lambda_1=7, 8, 8.5, \dots, 9$ 인 경우에, 임의의 한 클래스 1 트래픽 소스의 k번째 패킷이 도착했을 때 그 소스의 버퍼에 대기하고 있는 패킷 수의 변화를 그림 12

에 나타내었다. 여기서 서버의 서비스율은 $\mu=100$ 이다. 도착률(λ_1)이 증가함에 따라 클래스 1 트래픽의 버퍼 길이가 커진다. 클래스 2 트래픽 소스의 버퍼 길이의 추이를 나타내는 그림 11과 비교해서 클래스 1 트래픽 소스가 훨씬 큰 버퍼를 필요로 함을 알 수 있다.

클래스 2의 도착률을 고정시키고 클래스 1의 도착률을 변화시킬 때, 다양한 클래스 1의 도착률에 대한 SPA 서버 이용률과 시스템 전체적인 버퍼의 평균 크기를 표 2에 나타내었다. 도착률이 증가할수록 서버의 이용률은 증가하지만 시스템의 전체적인 평균 큐 길이도 증가한다. 이 경우도 역시 공유 버퍼를 사용하는 것이 각 채널 태스크마다 각자의 버퍼를 두는 것보다 더 효율적임을 알 수 있다.

표 2. 클래스 2 트래픽의 도착률을 고정하고 클래스 1 트래픽의 도착률을 변화시킬 때 서버 이용률 및 평균 버퍼 길이

Table 2. In case of fixing the arrival rate of class 2 ones and varying that of class 1 ones, server utilization and mean buffer length according to arrival rate.

Results		Server Utilization	Mean Buffer Length
Class 1 Arrival Rate			
7.0		0.8932	8.339
8.0		0.9368	17.103
8.5		0.9524	22.043
8.8		0.9672	26.043
8.9		0.9836	36.730
9.0		0.9943	66.356

다양한 두 부류의 트래픽 소스들의 수와 도착률에 대하여, 임의의 한 클래스 1 트래픽 소스의 k번째 패킷이 도착했을 때 그 소스의 버퍼에 대기하고 있는 패킷 수의 추이를 그림 13에 나타내었다. 여기서 클래스 1과 클래스 2의 트래픽 소스들의 수는 각각 4:92, 5:91, 6:90, 7:89, 8:87, 9:85 등이고, 서버의 서비스율은 $\mu=100$ 이다. 트래픽의 전체적인 도착률이 비슷할 경우에 클래스 1 트래픽 소스의 버퍼 길이는 유사하게 변하고, 그림 11에 나타난 것과 같은 클래스 2 트래픽 소스의 버퍼 길이 보다는 훨씬 큰 변화폭을 가진다. 따라서, 각 채널 태스크마다 고유의 버퍼를 두는 것보다 전체 채널 태스크들이 공유하는 하나의 버퍼를 두는 것이 더욱 효율적임을 알 수 있다.

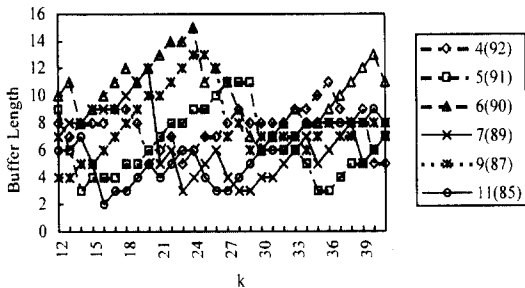


그림 13. 클래스별로 트래픽 소스의 수를 다르게 했을 경우에, 클래스 1인 임의의 한 트래픽 소스의 번째 패킷이 도착했을 때 그 소스의 버퍼에 대기하고 있는 패킷의 수

Fig. 13. For any one of the class 1 traffic sources in case of varying the ratio of the number of class 1 ones to that of class 2 ones, the number of packets in this source's own buffer at the time of the arrival of the th packet.

다양한 두 부류의 트래픽 소스들의 수와 도착률에 대한 SPA 서버 이용률과 시스템 전체적인 버퍼의 평균 크기를 표 3에 나타내었다. 전체적인 트래픽 도착률이 비슷할 경우에 서버의 이용률과 시스템의 전체적인 평균 큐 길이도 비슷함을 알 수 있다.

표 3. 클래스별로 트래픽 소스의 수를 다르게 했을 때 서버 이용률 및 평균 버퍼 길이

Table 3. Server utilization and mean buffer length according to varying the ratio of the number of class 1 ones to that of class 2 ones.

Class 1		Class 2		Result	
Rate	Source #	Rate	Source #	Server Utilization	Mean Buffer Length
10.5	4	0.6	92	0.9851	38.824
8.5	5	0.6	91	0.9801	40.266
8.9	6	0.5	90	0.9836	36.730
7.5	7	0.5	89	0.9747	33.045
7.0	9	0.4	87	0.9715	30.222
5.9	11	0.4	85	0.9714	34.390

V. 결론

본 논문에서는 대용량 통신처리시스템의 TNAS-HSNA 인터페이스 상에서 통신 모듈을 구현하였다. TNAS의 SPA에서 실행되는 태스크들, 채널 구분자

전달용 큐, 송신용 큐들, 수신용 큐들 등과 같은 자원들의 구성과 각 태스크들의 동작 절차를 제안하였다. 통신 모듈에서 최대 수용 가능한 채널 태스크들의 수를 변화시킴으로써 목적 시스템의 가변적인 환경에 잘 적응할 수 있다.

시뮬레이션을 통하여 TNAS에서 통신 처리 서비스를 담당하는 SPA의 시스템 이용률과 버퍼의 크기 측면에서 성능을 평가하였다. 패킷을 송수신하는데 필요한 버퍼를 각 채널 태스크마다 두는 것보다는 전체 채널 태스크가 공유하는 하나의 버퍼를 두는 것이 자원 이용 효율면에서 유리함을 알 수 있다.

대용량 통신처리시스템의 HSSF 서브시스템은 전화망, 데이터망, 프레임 릴레이망, ATM 망 등 다양한 망과의 접속을 지원해야 한다. 본 논문에서 구현한 TNAS-HSNA 인터페이스 상에서의 통신 모듈은 이러한 망 접속을 지원하는 TNAS, PNAS, FNAS, ANAS 등과 같은 망 정합 장치에서 SPA와 HSNA 사이의 인터페이스에 적용이 가능하다.

참고 문헌

- [1] K.S. Kim, J.D. Huh and D.U. Kim, "The Implementation of Billing Service Function in Communication Processing System," Proceedings of the 2nd APCC'95, Vol. 1 of 2, pp.42-46, Osaka, Japan, June 13-16, 1995.
- [2] K.S. Kim, J.D. Huh and D.U. Kim, "Implementation of IP Charging Function in Communication Processing System," Proceedings of KITE Summer Conference '94, Vol.17, No.1, pp.14-17, Jeonjoo, Korea, 1994. 7.
- [3] C.D. Shin, H.J. Kim, J.D. Huh and D.U. Kim, "The implementation of Guidance Service in Communication Processing System," Proceedings of KICS Fall Conference '93, Vol.12, No.2, pp.3-6, Soowon, Korea, 1993. 11.
- [4] C.D. Shin, H.J. Kim, J.D. Huh and D.U. Kim, "The implementation of IP Protocol in Communication Processing System," Proceedings of KITE Fall Conference '93, Vol.16, No.2, pp.80-83, Incheon, Korea, 1993. 11.

- [5] B.N. Yoon, T.J. Kim and C.H. Kang, "Application of the Communication Processing System to an Open Value Added Network," Proceeding of the 6th JC-CNSS, pp.34-38, 1993.
- [6] D.Y. Kim, K.J. Nam, H.S. Hong, C.B. Lee, S.J. Kim and B.N. Yoon, "Flexible Architecture for Communication Processing Service between PSTN and PSDN," JWCC-8, Taipei, 1993. 12.
- [7] B. Hebrawi, "OSI Upper Layer Standards and Practices," McGraw-Hill, Inc., pp.1-94, 1993.
- [8] D. E. McDysan, D. L. Spohn, "ATM: theory and application," McGraw-Hill, Inc., pp.65-167, 1995.
- [9] "Getting Started with VRTXvelocity," Ready Systems, California, USA, pp.2.1-2.43, July, 1992.
- [10] "VRTX32 User's Guide," Ready Systems, California, USA, pp.1.1-2.30, July, 1992.
- [11] "TNX User's Guide," Ready Systems, California, USA, pp.2.1-2.12, July, 1992.
- [12] K.S. Kim, S.J. Yoon, J.D. Huh and P.D. Cho, "Design of the TNAS-HSNA Interface in the Advanced CPS," Proceedings of KICS Fall Conference '95, Vol.14, No.2, pp.310-314, Seoul, Korea, 1995. 11.
- [13] Kishor S. Trivedi, "Probability & Statistics with Reliability, Queuing, and Computer Science Applications," Prentice-Hall, Inc., Englewood Cliffs, N.J., pp.109-468, 1982.
- [14] M.H. MacDougall, "Simulating Computer Systems Techniques and Tools," The MIT Press, USA, pp.1-128, 1987.
- [15] J. Bloomer, "Power Programming with RPC", O'Reilly & Associates, Inc., USA, pp.103-145, July, 1992.
- [16] "Berkeley IPC Programmer's Guide", Hewlett Packard Company, USA, pp.1.1-3.23, July, 1992.

 저 자 소 개



金 建 石(正會員)

1992년 경북대학교 컴퓨터공학과 졸업(공학사). 1994년 경북대학교 대학원 컴퓨터공학과 졸업(공학석사). 1994년 2월 ~ 현재 한국전자통신연구원 연구원. 관심분야는

Communication Processing

System, Multimedia communications, PSTN, PSDN, ISDN, and ATM network access systems, Internet protocols, High speed networks.

趙 平 東(正會員)

1980년 연세대학교 전자공학과 졸업(공학사). 1980년 한국전자통신연구원 입소. 1985년 충남대학교 대학원 컴퓨터공학과 졸업(이학석사). 1997년 ~ 현재 한국전자통신연구원 책임연구원, 뉴미디어시스템연구실장. 관심분야는 ISDN, B-ISDN, Internet, 통신프로토콜, 통신처리시스템, 지능망, Network architecture design.