위성통신을 위한 실시간 데이터 통신 컴포넌트의 구현

윤희철*

Implementation of Real-Time Data Communication Component for Satellite Communication

Hee-Chul Yun*

본 연구는 2004년도 (재)부산정보산업진흥원의 지원에 의하여 이루어진 연구로서, 관계부처에 감사 드립니다.

요 약

기존 이메일 프로그램은 비싼 통신요금과 저속의 위성통신환경에서 최대한 통신비를 줄이기 위한 목적으로 개발되었고, 그 결과 프로그램에 유연성이 떨어져, 새로운 기능의 추가나 업그레이드가 용이하지 않았다. 또한 기존 프로그램의 경우 이메일 전용으로서 다양한 육상과 해상간 업무용 프로그램과 연동시키는 데에 어려움이 많았으며, 개발자 입장에서도 무선위성통신의 특성상 개발이 쉽지 않았다. 따라서 본 논문에서는 기존 프로그램과는 달리 데이터를 패킷 분리하여 전송함으로써 프로토콜에 유연성을 주고, 실시간으로 여러 종류의 데이터를 전송할 수 있게 하여 동시에 E-mail뿐만 아니라, 모니터링 데이터, SMS, PMS를 위한 DB 데이터 등의 다양한 데이터를 전송할 수 있도록 구현하였다. 또한 프로그램 개발자들이 까다로운 위성통신환경을 몰라도 쉽게 개발할 수 있게 하고자, 컴포넌트를 윈도우즈 서비스 프로그램화 하였다.

ABSTRACT

The exist email program between Ship and Land has developed to reduce communication fee because of high satellite communication cost and low speed. Owing to this proper, the exist program has less flexibility and high speed. but to add or change functions is not easy. and it also have many difficulties for programmer to develop marine application because of delicate Satellite Radio commnucation. In this paper we design new protocol which consist of packet to give flexiblity and implement windows service program for programer to make marine applications easily like monitoring data, DB for Ship Management System, Planned Maintenance System.

키워드

Inmarsat, Satellite Communication, E-mail, Packet

Ⅰ. 서 론

현재 육상과 해상간의 통신은 비싼 통신요금과 2,400~9,600bps의 저속이 주류를 이루고 있는 낙

*드림포트(주)

접수일자 : 2004. 11. 15

후된 상황에서 통신이 이루어지고 있다, 이런 환경하에서 개발된 기존의 E-mail 전용 프로그램은 통신요금을 줄이기 위하여 최대한 통신 속도를 빠르게 하는데 목적을 두고 설계하다보니 프로토콜에 대한 유연성이 없이 개발되었다. 이런 이유로 인하여 통신 속도 면에서는 상당히 빠른 속도와 그에 따른 통신요금 절감 효과를 가져왔지만 유연성 없는 프로토콜 설계로 인하여 기능 업그레이드와 추가 시에 많은 애로사항이 발생한다. 또한 육상과 해상간의 업무용 프로그램이나 모니터링 프로그램 등을 개발하고자 할 때도 해상 통신의 특성을 잘 이해하지 못하여 응용 어플리케이션 개발이 힘든 상황이다.

그러나 지금은 각 국가간, 통신사업자간의 경쟁이 치열해져 위성 통신요금이 많이 내리고 있고, ISDN을 지원하는 새로운 장비의 등장으로 인해 과거와는 달리 통신요금 보다는 편리한 기능과 부가서비스에 대한 요구가 더 커지고 있다.

따라서 본 논문에서는 패킷방식을 도입하여 프로토콜에 유연성을 제공하는 통신 컴포넌트를 서비스 프로그램화 하는데 목적을 두고 구현하였다. 즉, 프로그램 개발자들이 이 통신 컴포넌트를 이용하여 E-mail, FTP(File Transfer Protocol), 파일/폴더 공유, 무선인터넷, 선박 업무자동화, 각종 부가서비스 등을 용이하게 개발 할 수 있는 위성통신용 다중 실시간 통신 컴포넌트를 구현하고 이를 이용한 응용 프로그램을 구현, 실험하였다

II. Inmarsat 서비스

초기의 서비스는 MARISAT(MARItime SA-Tellite) 위성을 사용하였으며, 1983년부터 임대 위성을 사용하게 되었다. 1990년까지는 전 세계 해역을 3개의 위성권으로 하여 운용하였으나, 대서양과 태평양 사이의 교신 불가능 해역을 극복하기 위해 1990년 11월 대서양 위성을 1개 더 추가하여 총 4개의 위성권으로 운용하고 있으며 Inmarsat 서비스 구성도는 그림 2-1과 같다.

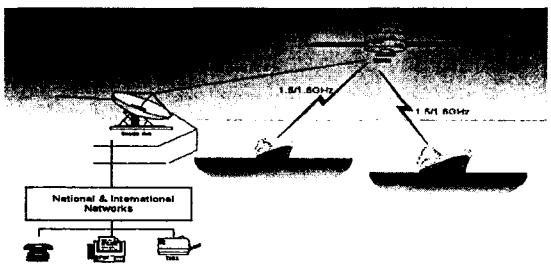

그림 2-1. Inmarsat 구성도
Fig. 2-1 Inmarsat Composition

1990년 Inmarsat는 자체 위성개발 계획을 수립하고, 영국의 위성회사에 발주하여 1990년 10월 인도양 상공에 처음 자체 위성을 발사하였다. 이어서 1991년 3월에는 동대서양 상공에, 1991년 12월에는 태평양 위성을 발사하였으며, 이들을 제2세대 위성이라 한다. 또한 1996년부터 1998년까지 제3세대 위성을 발사하여 현재에 이르게 되었고, 2004년부터는 14억불을 투입하는 제4세대 위성을 이용한 멀티미디어 통신 및 ISDN 서비스 접속이 가능하게 된다. 제4세대는 예비위성 1기를 포함한 총 3기로 구성되며, 운용위성의 궤도는 각각 W(west) 54° 와 E(east) 64°가 된다.

표 1과 2는 Inmarsat 서비스 현황을 나타내고 있으며, 현재 Inmarsat 서비스는 영국 Inmarsat 상업회사가 소유한 Inmarsat 정지궤도 위성을 이용하여 이동 중인 이동지구국(MES : Mobile Earth Station)과 육상의 가입자간의 전화, 팩스, 텔렉스, 데이터 통신을 제공하는 서비스이다. 우리나라는 KT가 LESO(Land Earth Station Operator)이며, 1991년 Inmarsat-A 서비스를 시작으로 1993년 Inmarsat-C 그리고 1997년에는 Inmarsat-B, M, Mini-M 서비스 지구국을 차례로 개통하고, 2001년에는 Inmarsat-GAN 지구국을 개통하여 서비스를 제공하고 있다[2 - 8].

표 1. Inmarsat 서비스 종류
Table. 1 Inmarsat service class

구 분	In - A	In - B	In - M	In - mM
서비스 이용	전화 텔렉스 데이터	전화 텔렉스 데이터	전화 텔렉스 데이터	전화 텔렉스 데이터
통신 속도	9600bps	9600bps HSD(64K)	2400bps	2400bps
전송 방식	아날로그	디지털	디지털	디지털
비고(개시)	1982년	1993년(A 대체)	1993년	1996년

표 2. Inmarsat 서비스 종류(계속)
Table. 2 Inmarsat service class(Cont.).

구 분	In - GAN	In - Aero	In - C
서비스 이용	전화 텔렉스 데이터	전화 텔렉스 데이터	전화 (텔렉스) 데이터
통신 속도	64Kbps	9.4Kbps	600bps
전송 방식	디지털	디지털	축적방식
비고(개시)	1999년	1993년(항공기)	1986년

III. 통신 컴포넌트 설계 및 구현

3.1 통신 컴포넌트의 설계

본 논문에서는 두 가지 관점에서 육상과 선박간의 위성통신 컴포넌트를 구현하였다.

첫째 육상과 해상간 업무용 프로그램을 개발하는 개발자들이 까다로운 위성통신 특성에 관계없이 쉽게 프로그램을 개발할 수 있도록 컴포넌트 프로그램을 구현하는 것이다. 즉, 구현한 통신 컴포넌트는 볼랜드 C++ 빌더, VC, VB와 같은 개발 툴에 포함시켜 사용하는 컴포넌트 라이브러리(component library) 형태가 아닌 윈도우즈 서비스 프로그램 형태로 구현하였다. 개발 툴에 포함시키는 라이브러리의 경우는 서비스 프로그램 형태에 비해 여러 개발 툴을 사용하는 응용 프로그래머 마다 다양한 여러 가지 복잡한 형태의 지원이 필요하나, 서비스 프로그램 형태의 경우는 서비스 프로그램과 응용프로그램 간의 적절한 통신프로토콜을 규정함으로써 컴포넌트 개발자와 이를 이용하는 응용프로그램 개발자간 독립적인 개발을 가능하게 해 줌으로써 생산성을 높이는 데에 도움이 되도록 하였다.

둘째, 패킷 다중화 개념을 도입하여 프로토콜에 유연성을 제공한다. 기존의 해상 E-mail 전용 프로그램의 경우는 통신비용을 줄이는데 중점을 두고 개발되어 통신비용을 줄이는 부분에는 소기의 목적을 달성하였지만, 이로 인하여 기능의 변경이나 추가 등에 상당한 애로사항이 많았다. 본 논문에서는 응용 프로그램에서 TCP/IP를 통하여 수신되는 데이터들을 패킷화하여 직렬포트로 전송하고, 직렬포트에서 수신된 내용을 다시 패킷으로 분할하여 각 해당 응용 프로그램에 전달함으로 해서 각각의 응용 프로그램이 통신 컴포넌트 프로그램을 직접 제어할 수 있도록 하였다. 이 통신 컴포넌트 프로그램은 Inmarsat 위성 단말기뿐만 아니라, 글로발스타(GlobalStar), 이리듐(Iridume)과 같은 위성통신장비에도 그대로 이용할 수 있다.

그림 3-1은 본 논문에서 구현한 시스템의 구성도이다. 육상과 해상의 선박간에 위성통신을 위한 다중 채널 실시간 데이터 통신 컴포넌트(서비스 프로그램)와 이 통신 컴포넌트를 이용한 E-mail 전송 프로그램, 메신저 프로그램, DB 액세스 프로그램 등의 응용 프로그램으로 구성된다. 본 논문에서는 통신 컴포넌트와 이를 이용한 메신저 응용 프로그램을 구현하여 육상과 해상의 선박간의 데이터 통신을 실험하였다.

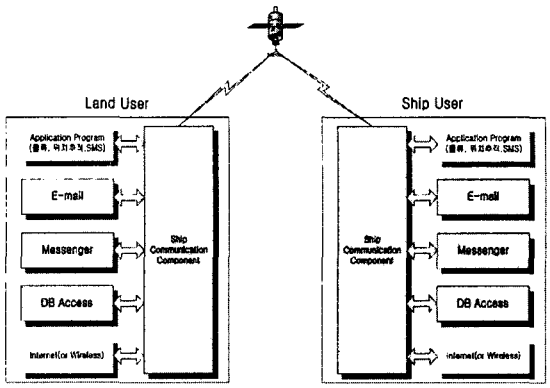

그림 3-1 시스템의 구성도
Fig. 3-1 Composition of System

3.2.1 통신 컴포넌트의 동작

그림 3-2는 통신 컴포넌트(이하, RB-SCCP : Rainbow Bridge-Satellite Communication Com-ponent Program)를 이용한 육상과 선박간의 데이터 전송 흐름을 나타내고 있다. 통신 컴포넌트는 육상용과 선박용의 구분 없이 동일한 프로그램을 사용하며, 프로그램의 내부 설정에 의해서 육상용과 선박용으로 구분된다.

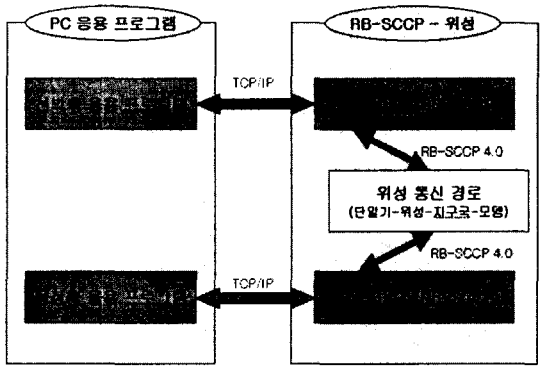

그림 3-2 통신 컴포넌트를 이용한 데이터 흐름
Fig. 3-2 Data Flow using Communication Component

RB-SCCP는 명령 포트, 메시지 포트, DB 포트, 제어포트, 기타포트의 5개의 포트를 지원하여 동시에 5개의 포트에서 어플리케이션이 동작할 수 있도록 설계를 하였다. 명령포트는 모뎀제어나 통신상태를 파악하는 데에 사용하고 메시지 포트는 메신저 구현, DB 포트는 SMS, PMS 등의 업무전산화 구현, 제어포트는 선박 모니터링이나 선박장비 제어를 위해, 기타 포트는 여유분으로 남겨두었다.

표 3-1. 메시지 정의(일부)
Table 3-1. Message Define

Command Type	STATE에 대한 Option1 값	STATE_DIALING_FAIL에 대한 Option2값
#define STATE 1	#define STATE_OFFLINE 1	#define DIALING_FAIL_NOCARRIER 1
#define DIALING 2	#define STATE_PORT_CLOSE 2	#define DIALING_FAIL_NODIALTONE 2
#define HANGUP 3	#define STATE_DIALING_FAIL 3	#define DIALING_FAIL_NOTREGISTER 3
#define MSG 4	#define STATE_PORT_ERROR 4	#define DIALING_FAIL_PASSERROR 4
#define FILE 5	#define STATE_PORT_OPEN 5	#define DIALING_FAIL_RSLDOFF 5
#define BUFFSIZE 6	#define STATE_MODEM_INI 6	
	#define STATE_MODEM_OK 7	
	#define STATE_DIALING 8	
	#define STATE_RING 9	
	#define STATE_CONNECT 10	
	#define STATE_STANDBY 11	
	#define STATE_ONLINE 12	

구조는 모두 동일하여, 명령포트를 제외하고, 어느 포트를 어떤 목적으로 사용이 가능하다.

RB-SCCP는 응용 프로그램들에서 데이터를 받아서 패킷화하여 전송하고, 수신된 데이터를 패킷으로 분류하여 해당하는 응용 어플리케이션으로 전송하는 역할과, 모뎀 제어의 역할을 하며, 어플리케이션에서 패킷 제어나 에러정정을 담당하도록 하였다.

3.2.2 통신 컴포넌트의 프로토콜

육상과 선박간의 RB-SCCP의 통신 프로토콜은 PPP(Point-to-Point)와 HDLC(High-level Data Link Control Procedure)를 변형, 단순화하여 위성통신에 맞게 재사용하였다. 프로토콜은 가변길이를 가지도록 설계하였고 프로토콜의 오버헤드는 8byte에 불과하고 나머지는 실제 데이터이다. 그림 3-3은 응용 프로그램과 RB-SCCP 사이에서 TCP/IP를 이용한 RB-SCCP의 프레임 구조를 나타내고 있다.

$	CmdID	Cmd Type	Option1	Option2	Data Size	Data	0

그림 3-3. RB-SCCP의 프레임
Fig. 3-3 Frame of RB-SCCP

"$"는 프레임의 시작, "0(zero)"는 프레임의 끝, ","(콤마)"는 구성요소 구분, "CmdID"는 Command ID로써 같은 날짜에는 유일한 숫자, "CmdType"은 STATE, DIALING, HANGUP, MSG, FILE, BUFFERSIZE 등 6가지를 의미, "DataSize"는 데이터의 크기, "Data"는 육상과 선박간에 실제 전송되는 데이터를 의미한다.

3.3 통신 컴포넌트의 구현

표 3-1은 RB-SCCP을 이용한 응용 프로그램간의 메시지 정의(일부)를 나타내고 있다. 응용 프로그램이 RB-SCCP에 연결하여 통신할 수 있는 TCP/IP 소켓 포트번호는 RB-SCCP에 설정된 Base Port부터 5개가 있다. 즉, 베이스포트가 2,000이면 명령포트는 2,000, 메시지 포트는 2001, DB포트는 2003, 제어포트는 2004, 기타포트는 2005가 된다. BasePort 번호는 Command 포트로서 주로 모뎀 및 중요하고 긴급한 통신제어에 이용되며, 현재의 위성통신 상태를 바로 알 수 있다.

각 포트 당 20개 패킷을 저장할 수 있는 버퍼(buffer)를 가지고 있으며, 이 RB-SCCP 안의 스케줄러에 의해서 각 포트를 순차적으로 액세스하여 버퍼에 쌓인 데이터를 읽어 들여 시리얼포트로 송신한다.

명령포트 이외의 포트는 모두 동일한 특성을 가지고 있으므로 사용상의 특정한 구분을 두지 않는다. 명령포트는 단지 모뎀제어나 상태 정보를 읽어오는데 사용하고 데이터 교환을 위해 사용하지 않

는다.

응용프로그램에서 상대방(육상이나 선박)에 데이터를 보내기 위해서는 먼저 각 소켓의 버퍼에 여유분을 확인하고 소켓에 데이터를 보내야 한다.

Inmarsat를 이용한 데이터 통신은 F77에서 지원하는 최고 56K bps가 있으나 아직 널리 사용되고 있지는 않고 일반적으로 사용하는 Inmarsat B는 9,600bps Mini-M은 2,400 bps에 불과하다. 따라서 응용프로그램에서 RB-SCCP의 통신 TCP/ IP의 고속통신이고 RB-SCCP 간의 통신은 2,400 bps의 저속통신이므로 속도를 조절해주기 위한 버퍼가 필요하다. 더구나 RB-SCCP에 연결되는 응용프로그램은 5개까지이므로 데이터를 보낼 때는 RB-SCCP의 통신 상태를 고려하여 전송할 필요가 있다. 따라서 응용프로그램이 RB-SCCP로 데이터를 보내기 전에 반드시 해당 연결포트의 버퍼에 남아 있는 여유 버퍼의 수를 확인하고 나서 전송해야 한다.

Command 포트인 BasePort는 각 응용 프로그램마다 RB-SCCP의 통신상태를 확인하거나, 여유 버퍼의 개수를 확인하기 위하여 여러 개의 소캣이 연결 가능하도록 설계를 하였으나 나머지 포트는 프로그램의 단순화를 위하여 오직 하나의 소켓만 연결 가능도록 하였다(그러므로 현재 최대 5개의 응용프로그램이 동시에 연결 가능 함).

그림 3-4는 RB-SCCP의 구현 초기화면으로 RB-SCCP는 통신상태, 송·수신 데이터 정보, 통신 환경 설정 그리고 디버깅을 위한 로그 창 등으로 구분된다.

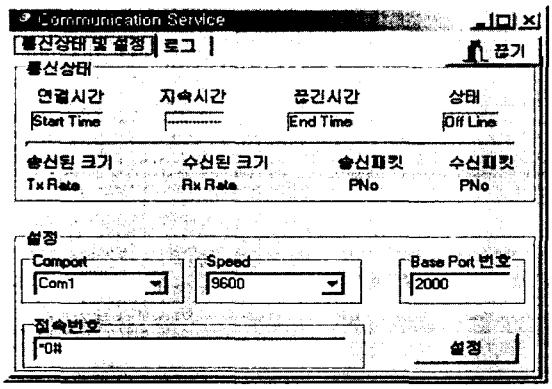

그림 3-4. RB-SCCP 구현 초기화면
Fig. 3-4 Initial Screen of RB-SCCP

상태표시 부분의 경우, 표3-1의 State에 대한 Option1에 나타나 있는 Off Line, On Line, Dialing, Connect 등의 값들로 나타난다. 송신패킷과 수신패킷은 응용프로그램에서 수신한 데이터를 RB-SCCP에서 패킷화 하면서 할당 된 패킷번호를 보여주는 것으로 모든 데이터에 대하여 일련의 연속된 패킷번호를 할당한다. Base Port는 RB-SCCP와 응용프로그램간의 TCP/IP 연결에서 사용할 베이스 번호를 말하며 베이스 번호부터 순차적으로 5번까지의 값을 갖는다[9].

IV. 실험 및 고찰

4.1 실험환경

본 논문에서 설계, 구현한 위성통신을 위한 통신 컴포넌트와 이를 이용한 육상과 선박간의 메신저 프로그램, 이메일 전송프로그램, DB액세스프로그램은 부산광역시 영도구 남항동에 위치한 (재)부산정보산업진흥원의 드림포트(주) 사무실의 육상 측과 옥상에서 선박의 위치를 대신하여, KT의 Inmarsat B 위성단말기를 사용하여 개발 및 테스트하였으며 그림 4-1과 같다.

실험 절차는 다음과 같다. 첫 번째, Inmarsat 단말기를 설치하여 안테나가 태평양 위성을 향하도록 하고, LES는 금산 위성지구국을 설정한다. 두 번째, 노트북과 단말기간에 RS-232C 케이블로 연결, 통신 서비스 프로그램을 시리얼 포트 COM1, 속도는 9,600bps로 설정 후 연결한다. 세 번째, 메신저 프로그램을 실행하여 통신 컴포넌트에 연결한다. 네 번째, 다이얼링 후, 육상 서버에 연결된 후 메시지를 송·수신한다. 다섯 번째, 양방향으로 파일을 송·수신한다.

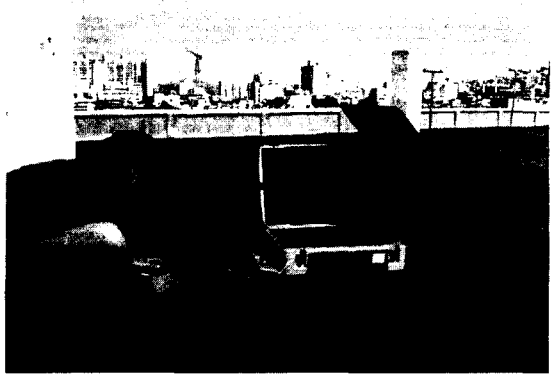

그림 4-1. 실험 환경 설정
Fig. 4-1 Experiment Environment Setting

그림 4-2는 RB-SCCP를 개발하는 과정 중에 사용된 테스트 환경으로 일차 구현 시는 비싼 위성통신요금 때문에 자체 내부 전화망을 사용하고 검증 시 위성단말기를 사용하였다. 그림 4-3은 RB-SCCP와 이를 사용하는 구현한 메신저를 이용하여 파일을 송·수하면서 메시지를 송·수신하는 모습을 보여주고 있다.

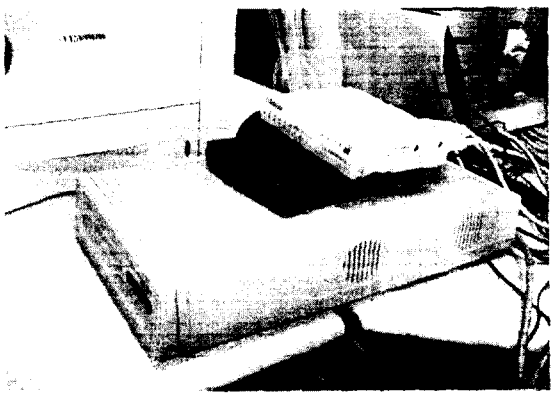

그림 4-2. 로컬 테스트용 모뎀
Fig. 4-2 Modem for Local Test

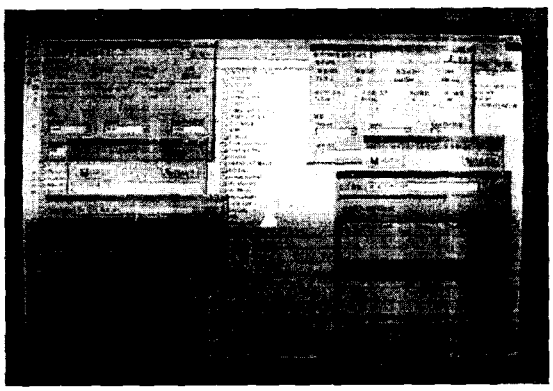

그림 4-3 로컬 테스트
Fig. 4-3 Local Test

4.2 실험결과

실험은 드림포트(주)의 선박이메일 서비스인 Rainbow Bridge에서의 파일 전송과 본 논문에서 구현한 RB-SCCP를 이용한 메신저 프로그램에서의 파일전송을 비교하였다.

그림 4-4는 RB-SCCP의 통신결과 화면이고 그림 4-5는 메신저를 이용한 파일 송신 과정을 나타내며, 그림 4-6은 메신저를 이용한 파일 수신 과정을 나타내고 있다.

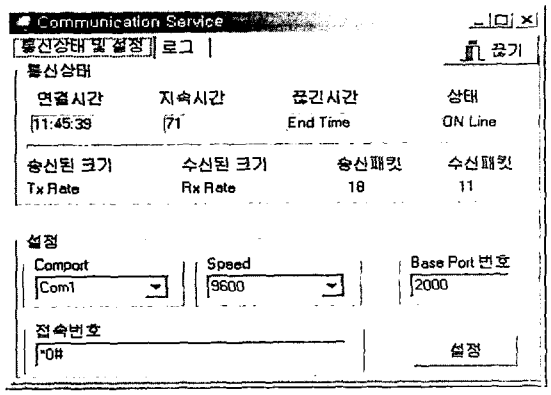

그림 4-4 연결 결과
Fig. 4-4 Connection Result

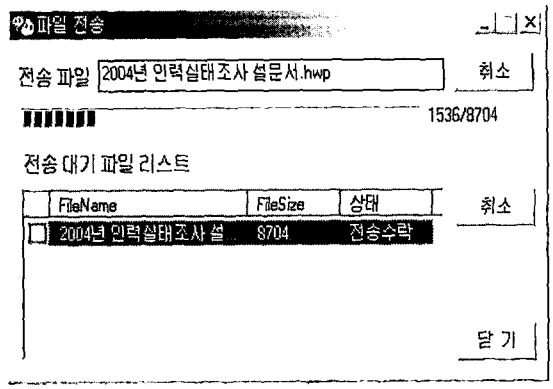

그림 4-5 메신저를 이용한 파일 송신
Fig. 4-5 File Transmission Using Messenger

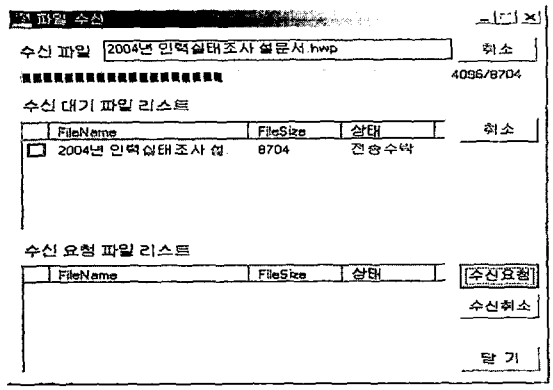

그림 4-6 메신저를 이용한 파일 수신
Fig. 4-6 Receipt of a File Using Messenger

표 4-1의 테스트 결과는 기존 E-mail 전용 프로

그램에 비해서 10% 정도의 시간이 더 소모됨을 알 수 있었다. 기존 E-mail 전용프로그램의 경우, 통신 비용과 서비스 비용이 모두 비쌌을 때 1초라도 통신 시간을 줄이기 위해 유연하지 못한 프로토콜을 사용하여 개발되었기 때문에 새로운 RB-SCCP를 이용한 메일 전송보다 성능이 앞설 것이라 예상은 하였지만 10% 정도의 차이를 보이고 있었다.

표 4-1. 실험 결과
Table 4-1. Test Result

구 분	기존 RainbowB	Email 어플리케이션
접속시간	평균 16초	평균 16초
1KByte	2초	3초
22300Byte 약 22KByte	25초	30초
413679Byte 약 400KByte	7분30초	8분10초
1MByte	18분57초	20분2초

그 원인을 고찰 해 보면 먼저 프로토콜 데이터 프레임상의 오버헤드로 인한 속도 저하가 있을 수 있다. 여기엔 윈도우즈 서비스 프로그램(RB-SCCP)간 또는 서비스 프로그램과 메신저 응용 프로그램간의 데이터 프레임상의 오버헤드를 생각할 수 있다. 그러나 앞에서 설명하였듯 RB-SCCP간의 오버헤드는 8byte에 불과하므로 RB-SCCP와 응용 프로그램간은 TCP/IP로 실제 통신 속도인 9,600bps보다 비교가 안될 정도로 고속이므로 무시해도 된다.

또 하나 고려해야 할 대상은 프로토콜 자체이다. RB-SCCP간 또는 RB-SCCP와 응용프로그램간의 프로토콜 교환의 효율성에 따라서 성능이 많이 좌우 될 수 있을 것으로 생각된다. 실제 테스트 결과 RB-SCCP간의 통신 시에는 원래의 성능을 발휘하는 걸로 봐서는 RB-SCCP와 응용 프로그램간의 프로토콜 교환에 문제가 있는 것으로 생각된다.

V. 결 론

본 논문에서는 기존 위성 이메일 서비스의 단점인 유연하지 못한 프로토콜과 육 해상간의 응용프로그램 개발이 어려웠던 문제들을 해결하기 위하여 프로토콜에 패킷 개념을 도입한 다중실시간 프로토콜을 설계하고, 프로그래머들이 업무용 프로그램을 쉽게 개발할 수 있도록 윈도우즈 서비스 프로그램 형태로 구현하였다. 기존의 선박이메일 서비스 프로그램의 경우 통신 속도를 최대한 빠르게 하기 위하여 부득이하게 유연성을 포기할 수밖에 없는 구조를 가지게 되었고 그로인해 새로운 기능의 추가와 업그레이드 등의 작업이 상당히 까다로웠고 어려웠다. 본 논문에서 구현한 위성통신을 위한 다중채널 실시간 데이터 통신용 컴포넌트는 통신 프로그램의 성능을 감소시키지 않고서도 다양한 기능추가나 작업을 가능하게 한다는 목표를 두고 구현되었으며, 실제 서비스 프로그램 구현결과 가능함을 충분히 발견하였다.

향후 연구로는 통신 컴포넌트와 응용프로그램간의 데이터 교환에 있어서의 프로토콜 개선을 통한 성능향상, 채널간의 대역폭 활용을 위한 방안이 연구되어야 할 것으로 보인다.

참고문헌

[1] 김기문 외 2인, "DSC용 MF/HF대 송·수신 장치 개발을 위한 제규정 연구", 연구보고서, 삼영전자부설 전자통신 연구소, 1995.

[2] 윤희철, "육상과 선박간 전자메일 시스템의 설계 및 구현에 관한 연구", 공학박사 학위논문, 한국해양대학교, 2003.

[3] 유형열, "DDS를 이용한 중단파대 국·영문용 DSC/NBDP 개발에 관한 연구", 공학박사 학위논문, 한국해양대학교, 1999.

[4] 윤희철, 임재홍, "해상용 전자메일 시스템의 설계 및 구현", 한국해양정보통신학회, 제6권, 제8호, 2002.

[5] 신은식, "인말새트 위성이동통신 세미나", 한국통신네트워크본부, 2000. 10.

[6] International Maritime Organization, "Introduction to the IMO", http://www.imo.org

[7] International Mobile Satellite Organization, "Introduction to the Inmarsat", http://www.inmarsat.org.

[8] KT/메스코전기전자(주), "인말세트-GAN(M4) 소개 및 장비 시연 세미나", 2002.

[9] 드림포트(주), "해양IT기술개발지원사업 과제수행 최종 결과보고서", (재)부산정보산업진흥원, 2004.

저자소개

윤희철(Hee-Chul Yun)

1992년 한국해양대학교 전자통신공학과(공학사)
1994년 한국해양대학교 전자통신공학과(공학석사)
2003년 한국해양대학교 전자통신공학과(공학박사)
1992~1993년 한국해양대학교 전자통신공학과 조교
1994~2000년 (주)사라콤 선박자동화연구소 팀장
2000~2003 한국해양대학교 시간강사
2001~현재 드림포트(주) 대표이사
2003~현재 한국해양대학교 공학교육인증준비위원회 자문위원
※관심분야 : 위성통신, 네트워크, 선박통신