

최적화된 큐에서의 NMEA 멀티플렉서의 설계 및 구현

김창수* · 정성훈* · 임재홍**

*한국해양대학교 전자통신공학과, **한국해양대학교 전파·정보통신공학부 부교수

Design and Implementation of NMEA Multiplexer in the Optimized Queue

Chang-Soo Kim* · Sung-Hun Jung* · Jae-Hong Yim**

*Dept. of Electronics & Communication Engineering, Graduate school of National Korea Maritime University

**Division of Radio and Information Communication Engineering, National Korea Maritime University, Pusan 606-791, Korea

요 약 : 현재 국내에서는 NMEA 멀티플렉서를 개발한 사례 및 제품이 몇몇 업체가 있으나, 외국 제품보다 인지도 및 제품판매의 실적이 미비하다. 그러므로 외국의 고비용 장비를 수입하여 사용하거나, 멀티 포트를 이용하여 NMEA 신호를 전송하는 프로그램을 소프트웨어적으로 구현하여 사용하고 있다. 그러므로 고비용이 지출되거나, 각 응용프로그램마다 별도의 처리 부분을 작성해야하는 문제점이 있다. 또한, NMEA 신호를 출력하는 각각의 장비들은 제조회사 및 플랫폼이 다르므로 이종의 자원낭비 및 손실 등을 초래할 수 있다. 본 논문에서는 경제적으로 NMEA 멀티플렉서 모듈을 구현하기 위하여 가상 시스템을 구성하여 실험을 실시하였다. NMEA 신호를 입력하기 위해 각 장비들은 가상의 데이터를 가진 노드로 대체하여 멀티플렉서 구현을 위한 가상의 시스템을 구성하고, 여러 문제들을 보완하기 위하여 신뢰성 있는 처리방법과 고성능의 단일 하드웨어 모듈로서 독립적으로 동작할 수 있다. 최적화된 큐의 설계를 이용하여 모듈의 메모리 효율을 높이며, 주요 항해 장비인 자이로컴퍼스, 에코 사운드, GPS 등과 실시간으로 RS232 멀티포트를 통하여 입력되는 신호를 실제 PC에서 출력하여 통신의 정확성을 유지할 수 있는 NMEA 멀티플렉서의 설계 및 구현에 대하여 제안하였다.

핵심용어 : NMEA 0183, 멀티플렉서, 큐, 멀티포트, GPS, RS232

Abstract : The National Marine Electronics Association(NMEA) is nonprofit-making cooperation composed with manufacturers, distributors, wholesalers and educational institutions. We use the basic port of equipment in order to process the signal from NMEA signal using equipment. When we don't have enough one, we use the multi-port for processing. However, we need to have module development simulation which could multiplex and provide NMEA related signal that we could solve the problems in multi-port application and exclusive equipment generation for a number of signal. For now, we don't have any case or product using NMEA multiplexer so that we import expensive foreign equipment or embody NMEA signal transmission program like software, using multi-port. These have problems since we have to pay lots of money and build separate processing part for every application programs. Besides, every equipment generating NMEA signal are from different manufactures and have different platform so that it could cause double waste and loss of recourse. For making up for it, I suggest the NMEA multiplexer embodiment, which could independently move by reliable process and high performance single hardware module, improve the memory efficiency of module by designing the optimized Queue, and keep having reliability for realtime communication among the equipment such as main input sensor equipment Gyrocompass, Echo-sound, and GPS.

Key words : NMEA 0183, MULTIPLEXER, QUEUE, MULTI PORT, GPS, RS232

1. 서 론

순항하는 선박과 항공기가 일정 항로에 따라 목표지점까지 가기 위해서는 여러 가지 제반 사항이 따른다. 그중에서 운송물을 빠르고 안전하게 수송하기 위해 현재의 위치 정보를 알려주는 GPS 수신기와 같은 장치가 필요하며, 또한 자이로컴퍼스(Gyrocompass), 풍향풍속계(Anemo meter), 스피드 로그(Speed Log), 타각지시기(Rudder indicator) 등 항해장비들도 필요하다(Hoffman-Wellenhof, 1997). 이런 항해장비들 간의

통신을 위한 인터페이스의 표준을 정하는 목적으로 미국해상 전자통신협회(NMEA : National Marine Electronics Association)라는 위원회가 발족되었다. NMEA는 제조업자들, 배급업자들, 도매업자들 그리고 교육 기관들로 구성된 비영리적인 제휴이다. 해상 장비 사이에 통신을 위하여 전기적 신호에 관한 인터페이스와 데이터 프로토콜로 정의할 수 있으며, 해상 전자 장치의 인터페이스에 널리 사용되고 있는 프로토콜이다(NMEA, 2002)(곽, 1999).

NMEA 프로토콜을 사용하는 장치로부터 출력되는 신호를

* 대표저자 : 김창수(정회원), cszzim@empal.com 051)410-4910 011)9539-6594

** 정회원, jhyim@mail.hhu.ac.kr 051)410-4318

처리하기 위해서 장비의 기본 포트를 이용하며, 부족할 경우 멀티포트를 사용하여 처리하고 있다. 그러나 수많은 신호에 대한 멀티포트의 적용과 전용장치로서의 출력의 제약 사항들을 해결하기 위하여 관련 NMEA 신호를 멀티플렉싱하여 제공할 수 있는 모듈 개발이 필요하다.

현재 국내에서는 NMEA 멀티플렉서를 개발한 사례 및 제품이 있으나 외국 제품에 비해 인지도 및 제품 판매의 실적이 미비하다. 그래서 고비용을 지불하더라도 안정적이고 인지도 있는 외국의 장비를 수입하여 사용하거나, 멀티 포트를 이용하여 NMEA 신호를 전송하는 프로그램을 소프트웨어적으로 구현하여 사용하고 있다. 이러한 방법은 고비용이 지출되거나 수입 의존적이며, 각 응용프로그램마다 별도의 처리 부분을 작성해야하는 문제점이 있다. 또한, NMEA 신호를 출력하는 각각의 장비들은 제조회사 및 플랫폼이 달라서 중복의 자원낭비 및 손실 등을 초래할 수 있다. 본 논문에서는 이를 보완하기 위하여 효율적인 NMEA 신호 처리방법과 고성능의 단일 하드웨어 멀티플렉싱 모듈로서 독립적으로 동작할 수 있고, 최적화된 큐의 설계를 이용하여 모듈의 메모리 효율을 높이며, 중요 입력 센서 장비인 자이로콤파스, 에코 사운드, GPS 등의 장비들과 실시간 업데이트 통신이 가능한 NMEA 멀티플렉서의 설계 및 구현에 대하여 제안하였다.

2. 관련 연구

2.1 NMEA 신호 종류

NMEA 신호는 NMEA 0180, 0182, 0183 등의 종류가 있으며, 현재 국제 표준으로 쓰이는 NMEA 0183을 중심으로 기술한다(SiRF, 2002).

NMEA 종류 중 단순 데이터 포맷은 7비트로 비트 0에서 비트 5까지는 크로스 트랙 에러를 나타내며, 비트 6에서의 값이 1이면 데이터가 유효함을 나타내는 것이고, 비트 7에서의 값 0은 단순 데이터 포맷을 나타내는 것이다. 복합 데이터 포맷은 아스키코드로 나타내어진 크로스 트랙 에러와 방위각, 현재의 경도/위도 및 2진 상태 바이트로 구성된 37바이트의 데이터 블록으로 이루어져 있다. 복합 데이터 포맷에서의 모든 바이트는 비트 7의 값이 1로서 단순 데이터 포맷형식과 구별되어진다. 그리고 단순 데이터와 복합 데이터를 동시에 전송이 가능하며, 복합 데이터 블록 중간에 단순 데이터의 삽입이 가능하다(Brookhouse, 2004). 그리고 NMEA 0183은 데이터 전송을 ANSI 표준에 따라서 직렬 비동기 방식으로 전송을 한다. Fig. 1은 NMEA 0183의 비트구성을 나타내며, 첫 번째 비트는 시작 비트이며, 다음에 따라 오는 비트는 데이터 비트, 마지막 비트는 정지 비트로 이루어져 있다(곽, 1999).

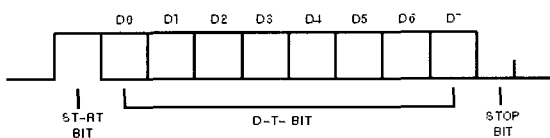


Fig. 1 NMEA 0183 Data Transmission

2.2 NMEA 신호 형식

NMEA 0183은 ASCII 코드의 모든 문자를 사용하며, 데이터는 문자형태로 전송이 되어진다. 각 문장은 '\$'로 시작을 하며, 송신자 ID를 나타내는 두 개의 문자와 문자 3개로 표현이 되는 데이터 ID, 콤마로 구분된 데이터 필드가 오며, 추가적으로 체크섬과 캐리지 리턴 및 라인피드로 종결 된다. 그리고 한 문장은 \$와 <CR> <LF>를 포함해서 82개 이내의 문자로 구성되어 있다(NMEA, 2002).

\$	GP	APB	189.0,03.6,M,04,W	CLRF
헤더	송신자 ID	데이터 ID	데이터 필드	종단

Fig. 2 NMEA 0183 Data Format

```
$GPGLL,3957.3089,N,12651.1101,E,074405.359,A*33
$GPGGA,074405.36,3957.3089,N,12651.1101,E,1,04,2,0,-0017,M,,,*31
$GPRMB,A,0.01,L,SIM001,SIM002,3959.1290,N,12651.0980,E,001.8,000.,021.7,V*05
$GPRMC,074405.36,A,3957.3089,N,12651.1101,E,21.7,002.7,050704,08.,W*78
$GPAPB,A,A,0.0,L,N,V,V,0.0,T,SIM002,359.7,T,357.0,T*7E
$GPGSA,A,3,01,02,03,04,,,,,,,,,2.0,2.0,2.0*34
$GPGSV,3,1,08,22,67,215,,14,53,315,,30,50,161,,05,49,071,*7E
$GPGSV,3,2,08,18,41,170,,09,33,055,,25,16,251,,15,13,230,*72
$GPGSV,3,3,08,,,,,,,,,,,,,*71
```

Fig. 3 NMEA 0183 Output of Parametric Sentences

만약에 필드의 데이터가 유효하지 않으면 해당 필드를 생략하거나 콤마로 범위를 정하여 전송을 하며, 이때 문자 사이에 빈 공간은 없어야 한다.

체크섬은 '*'와 두 문자의 16진수로 구성되며, 이것은 '\$'와 '*'를 포함하지 않는 모든 글자의 Exclusive-OR의 결과이다. Fig. 2는 일반적인 NMEA 0183 데이터 형식을 나타내며, Fig. 3은 NMEA 신호의 sentences의 출력 형식을 나타낸다(Brookhouse, 2004) (Kozlov, 1997).

3. NMEA 멀티플렉서의 설계 및 구현

3.1 NMEA 멀티플렉서의 시스템 구성

Fig. 4의 NMEA 멀티플렉서 시스템은 실제 GPS 및 각종 장치들로부터 RS232 포트 및 다른 포트로부터 NMEA 데이터를 받아들여 컴퓨터에서 신호를 확인하는 구성도를 나타내고 있다(가남사, 1990).

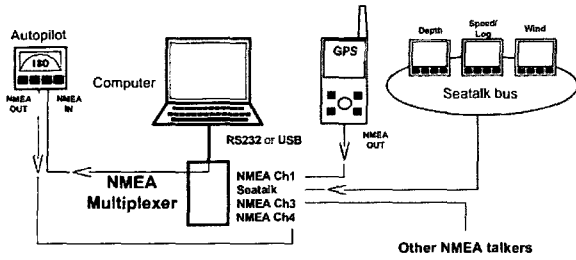


Fig. 4 NMEA Multiplexer System Structure

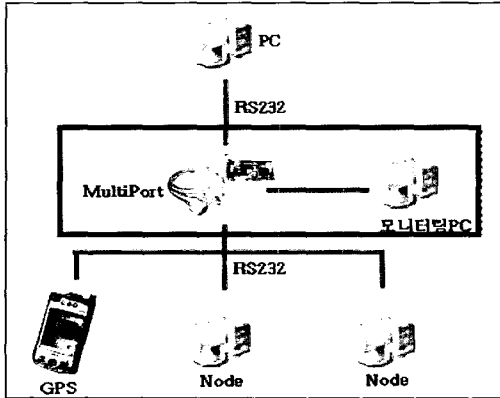


Fig. 5 Simulation System Structure

구성도에서 노드를 구성하는 장비들은 수심 측정기, 알파 레이더, 전자해도표시시스템(ECDIS) 등의 장치들이다. 본 논문에서 NMEA 멀티플렉서를 구현하기 위해서 그림에서와 같이 각 항해장비들의 데이터를 가진 가상의 노드들로 대체한다. 그리고 Fig. 5의 굵은 실선에 해당하는 부분은 본 논문에서 구현하는 NMEA 멀티플렉서를 멀티포트와 컴퓨터로 구성한 것이다. 이때 모니터링 컴퓨터에서는 각 노드들로부터 들어오는 데이터가 정상적으로 상위의 컴퓨터에 전송되는지 확인할 필요가 있다(Catsoulis, 2003).

3.2 NMEA 프레임의 처리 기법

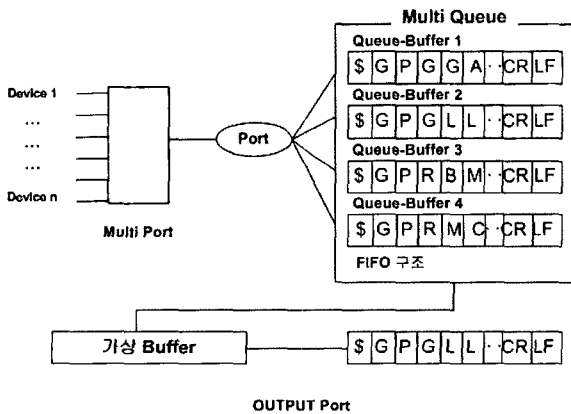


Fig. 6 Structure of Multi Queue

멀티 큐 구조는 선입선출(FIFO)방식으로 운영된다. 데이터 신호가 버퍼로 입력 될 때, 효율적인 메모리 관리를 위하여 버

퍼의 길이는 가변적인 크기를 가지게 된다. NMEA 멀티플렉서의 특성상 장치로부터 오는 데이터 중 최신의 데이터가 가장 중요하므로, 정보가 실시간으로 업데이트되도록 구성되어 있다. 그리고 '\$' 구분자 수를 판별하여 정상적인 데이터인지 잘못된 데이터 인지를 판별하도록 구성되어 있다.

3.3 대기 큐 알고리즘의 설계

각각의 장치들로부터 들어온 데이터를 입력포트로 받아들여 대기 큐를 활성화시키고, 그 대기 큐에서 프레임을 얻어온 후 각 포트에 해당하는 버퍼에 버퍼링 시켜주는 과정을 Fig. 7에서 큐 블록 다이어그램으로 보여주고 있다. 먼저 입력포트에서 대기 큐로 전송을 활성화하는 과정을 보면 NMEA 멀티포트에서 입력된 데이터 중 리스트 박스에서 한 줄 얻어온다. 그리고 GPS, 에코사운더(Echo Sounder), 자이로콤파스(Gyro Compass), 레이더(Radar) 등과 같은 데이터 프레임들을 해당 장비의 종류별로 버퍼링한다.

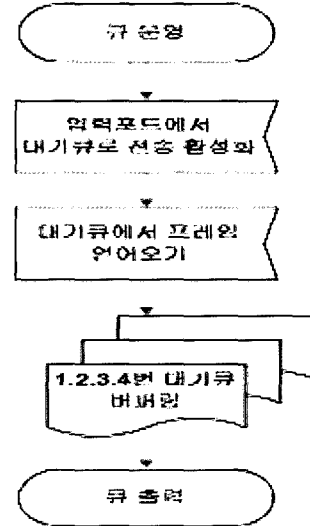


Fig. 7 Queue Block Diagram

Fig. 8은 대기 큐에서 프레임을 얻어오는 과정을 흐름도로 나타내고 있다. 먼저 프레임을 얻기 위하여 'EA'라는 배열을 선언하고 노드의 위치를 식별하기 위한 'Npos'를 지정한다. 각 항해장비별로 버퍼링을 하기 위해 변수(shpBuffer)를 선언하며, 각 장비별로 대기 큐 내에서 완전한 프레임, 즉 '\$'로 시작해서 '\$'로 끝나는 프레임이 있는지를 Case문과 'EA', 'Npos'의 값으로 For 반복문을 돌린다.

그리고 반복 루프 내에 해당 버퍼의 개수만큼 처리 루틴을 구현하여야 하며, 서브루틴으로 프로시저를 작성하였다. 프레임이 인터럽트가 발생할 때 자동적으로 수신된 바이트만큼 확보된 대기 큐에 입력이 완료되면 검색 루틴에서 '\$'신호를 검사한다. 그 이유는 NMEA 신호의 시작문자로 유효 프레임을 식별하기 용이하기 때문이다.

다음에 마지막에 들어오는 <CR><LF>가 포함되어 있다면 유효 프레임이 대기 큐 내에 존재한다는 것을 의미한다. 다음

과정에서는 정상 프레임과 미완료된 상태의 프레임과 절단 (Truncation)프레임을 구분해주는 부분으로 색상을 통하여 화면에 나타내도록 하였다. 이렇게 얻어진 유효 프레임은 단일 출력 포트를 통하여 출력될 수 있도록 이전에 얻어진 유효 프레임 저장장소에 계속 추가 시킨다. 그리고 입력포트에서 대기 큐로 전송 활성화하는 과정은 NMEA 멀티포트에서 입력된 데이터 중 리스트 박스에서 한 줄 얻어온다. 그런 다음 GPS, 에코사운드, 자이로콤파스 등과 같은 데이터 프레임들을 장비별로 버퍼링한다.

각각의 Case문 0번부터 3번까지는 각 포트들로부터 입력될 항해장비의 큐 4개를 나타낸다. 이 Case문에서는 데이터 프레임들이 한 번에 전송될 때, 전송한 프레임의 양(버퍼 크기)을 대기 큐에 뿌려주는 양을 체크하는 부분이다. Case문을 빠져 나오면 마지막으로 활성화 큐의 버퍼 량을 체크하고 시뮬레이션 한다. 대기 큐에서 프레임을 얻어오는 과정을 살펴보면, 우선 각각의 프레임 구분자인 '\$'와 '\$'의 개수를 정의하는 'EA' 배열과 각 위치를 위한 변수를 정의한다.

다음으로 들어온 데이터 프레임 중 완전한 프레임이 있는지 조사하기 위하여 For문을 사용하고 각 대기 큐 별 '\$'의 개수를 모두 조사한다. 유효한 프레임이 있는지 조사하여 각 버퍼의 색상을 결정하여준 뒤 유효한 프레임을 파싱하여 출력하고 출력 리스트박스 하단에 보여주는 구조로 이루어져 있다.

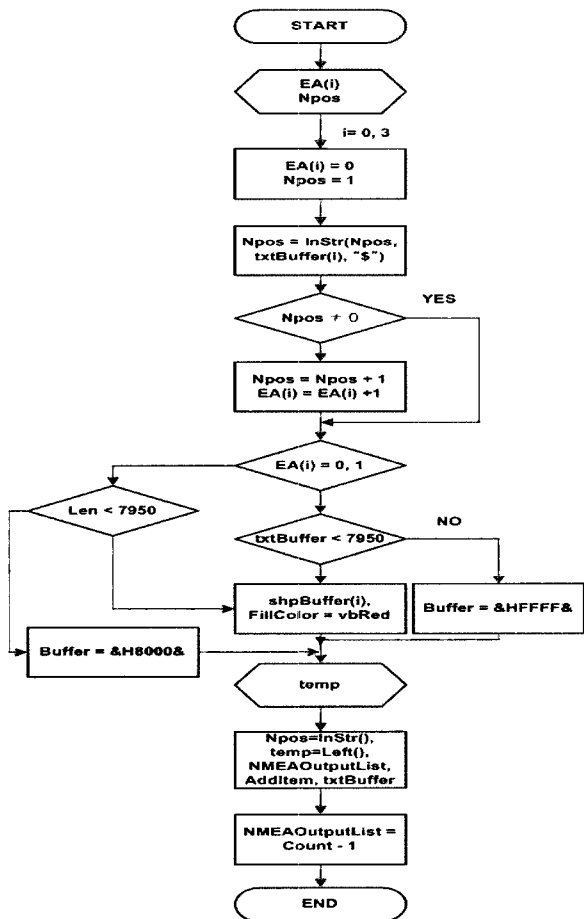


Fig. 8 Flow Chart from Wait Queue Frame

3.4 NMEA 멀티플렉서 구현을 위한 시뮬레이터

시뮬레이터의 모델링은 자이로콤파스나 풍향풍속계와 같은 항해장비들을 연결하여 시스템을 구현하기 위한 것으로 각 장비들이 NMEA 신호 모니터링 컴퓨터에 보내는 NMEA 데이터들을 각 해상장비에 해당하는 가상의 노드 컴퓨터가 장비들을 대신하여 데이터를 보내도록 Fig. 5와 같이 시스템을 구성하였다. 이 노드 컴퓨터들이 RS232를 통한 멀티포트로 NMEA 신호들을 보내면 여러 포트별로 들어온 NMEA 신호들을 통합하여 NMEA 신호 모니터링 컴퓨터에 데이터들을 보내게 되며, 이 멀티포트에 연결되어 있는 모니터링 컴퓨터에서는 들어온 데이터들을 버퍼 상에 저장한 뒤, 다시 이 데이터들을 파싱 과정을 거치게 된다. 그리고 이런 과정에서 각 항해장비에 해당하는 노드 컴퓨터들이 보내는 신호 전송속도 보다 처리하는 속도가 느리다면 이때는 데이터들이 버려지는 경우가 생기게 된다. 이 때 최근 정보를 계속 업데이트하고, 또는 버려지는 데이터들의 양을 확인 할 필요가 있다. 이를 위해 모니터링 컴퓨터상에 NMEA 멀티플렉서라는 프로그램을 구성하였다.

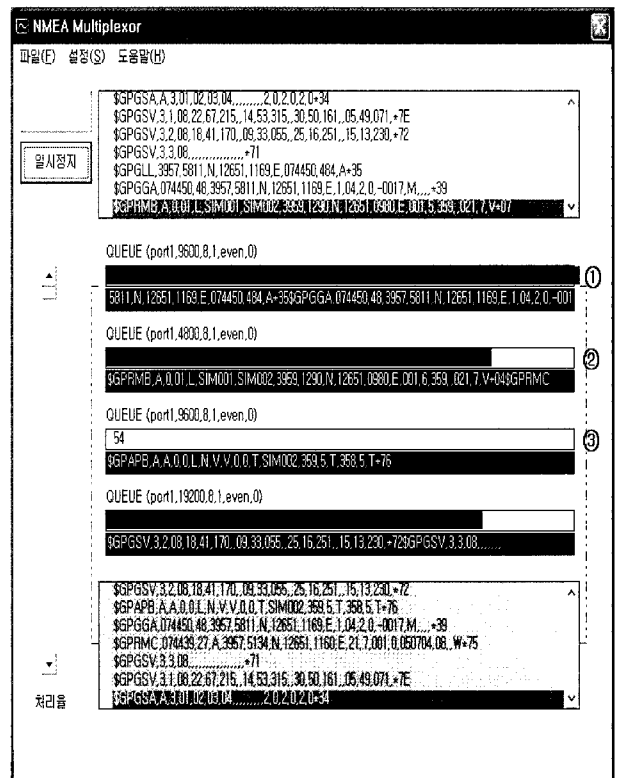


Fig. 9 Main Form of Monitoring PC

전체적인 화면 구성은 Fig. 9와 같다. 각각의 가상의 노드 컴퓨터들이 보내온 데이터들을 여러 없이 정상적으로 처리하고 있는지, 들어온 데이터들이 어떤 내용인지를 확인하고 가장 효율적인 처리율을 알 수 있게 설계되어 있는 폼 구조이다. 그림의 메인 폼 상단의 신호는 GPS 수신기를 통하여 수신되고 있는 NMEA 신호이며, 멀티포트와 연결되어 있는 기타 장

비로부터 입력되는 신호를 포함한다. 폼의 하단부 구성은 멀티 큐의 운영에 대한 시뮬레이션이다. 본 실험에서는 4개의 대기 큐를 사용하여 배치하였다. 대기 큐의 처리량에서 대기 큐 버퍼로부터 정상 프레임이 발견되었을 때, 파서에 의해서 파싱된 문자열을 버퍼에 연속적으로 추가하여 나타낸 모습이다. 밀의 텍스트 박스는 실제로 채워지는 텍스트 스트림이다. 이 실험에서는 처리량에 대한 최적화 처리율을 각 버퍼의 색깔로 버퍼 사이즈 변화량을 확인할 수 있다.

Fig. 9는 모니터링 컴퓨터의 메인 폼을 나타낸다. 정상 처리 못할 때는 버퍼가 노란 색(Fig. 9의 ①)으로, 정상처리 경우는 버퍼가 녹색(Fig 9의 ②)으로, 잘림 현상이 발생했을 때는 버퍼가 빨강색(Fig. 9의 ③)으로 표현되어 있다.

잘림 현상이 발생하였을 경우 해결책으로 유효 프레임을 다시 받기 위하여 손실된 프레임을 버리고 버퍼를 비워준다. 만약 유효 프레임과 손실된 프레임이 존재할 경우 유효 프레임을 인증하기 위하여 NMEA 신호의 마지막 부분의 순환 중복 검사를 위한 바이트를 파싱하여 앞부분의 데이터와 연산을 수행한다. 이때 오류 검사에서 정상적인 프레임이 존재할 경우 대기 큐 알고리즘에 따라 처리하게 된다.

4. 실험 및 고찰

프레임의 실시간 처리를 위하여 구축한 시뮬레이션 노드들과 GPS 수신기, 특정 NMEA 신호들을 보내주는 가상의 노드 컴퓨터로 구성하여 실험을 하였다. 다중 장치들로부터 동시에 입력되는 NMEA 신호들은 멀티포트를 통하여 구현된 시뮬레이션에서 해상 장비의 신호별 파싱과 유효 작업 분류 등을 통하여 다중화되어 정상적인 신호를 처리하는 과정을 보였다.

Table 1 Change of Frame Loss and Throughput

구 분	Port 1	Port 2	Port 3	Port 4
전송속도	9600	4800	9600	19200
버퍼크기	1000	1000	1000	1000
프레임 손실율	5.40%	2.45%	4.08%	0.52%
치 리 율	94.60%	97.55%	95.92%	99.48%

Table 1은 NMEA 가상 신호와 GPS 수신기로부터 수신되는 입력데이터를 60분 동안 수집하고 대기 큐 알고리즘을 통하여 처리된 결과로서 포트별 전송속도와 버퍼크기를 고정하고 입력된 프레임의 유효 데이터를 검출한 것이다.

프레임의 손실율은 버퍼의 크기를 초과하는 데이터 스트림이 발생할 경우 생기는 잘림 현상으로 버려지는 프레임이 발생한 빈도이다. 처리율은 수신되는 입력 데이터를 파싱하여 유효 프레임을 얻었을 경우 대기 큐에서 단일 포트로 복사된 결과로서 손실되지 않는 정상 프레임의 추출에 대한 비율을 나타낸다. 처리율과 프레임 손실율의 관계를 나타내는 자료를 바탕으로 그래프를 그려보면 처리율이 높을수록 프레임의 손실율은 낮아지는 것을 확인할 수 있다. 버퍼의 크기를 크게 한

다면 프레임 손실율은 없으며 데이터 신뢰성은 높지만, 전송 시간이 길어진다는 단점이 있을 것으로 예상된다. 그러므로 최적의 전송시간과 적은 프레임의 손실율을 얻기 위해서는 동적 큐의 할당 기법을 통해 넓은 대역폭이 필요한 장비에 대하여 충분히 수용할 수 있는 크기의 버퍼 량을 부여하면 적당한 큐 크기 설정이 가능하며 최소의 프레임 손실과 많은 공간의 메모리를 확보할 수 있다는 것을 알 수 있었다.

또한 항해장비들로부터 들어오는 데이터프레임의 크기보다 큐의 버퍼를 크게 하면 낭비되는 메모리 공간이 발생한다는 것을 의미한다. 그러나 무작정 처리율을 줄인다면 버퍼의 크기가 줄어들기 때문에 버퍼의 크기보다 큰 데이터 프레임들은 모두 버려지는 문제가 발생한다.

본 논문에서는 버퍼의 크기를 일정하게 하고 큐의 크기를 가변적으로 처리함으로써 처리율을 증가시키는 실험을 실시하였다. 그러나 향후 각 항해장비의 버퍼 크기를 변화 시킨다면 프레임 손실율이 문제가 될 것으로 예상되며, 실제적인 NMEA 모듈 설계 시에는 고려해야 할 것으로 생각된다.

5. 결 론

해상에서 작업하는 장비들이 다양해지고 더욱 많은 센서 장비들로부터 입력되는 신호들을 처리하기 위하여 수용하는 장비들의 경우 많은 수의 포트를 지원해야 한다. 그러나 여건상 제한적인 포트를 지원할 경우 수많은 항해 장비들을 연결하기 위하여 별도의 멀티 포트의 기능을 수행하는 장비가 필요하다.

본 연구에서는 4개의 멀티 큐와 가상 멀티 대기 큐를 설계하였다. 그리고 큐의 운영 알고리즘의 기법 적용과 메모리 크기를 동적으로 할당하여 적은 자원 환경에서도 다수개의 포트로부터 입력되는 신호들을 최소의 손실을 유지하면서 단일 포트로 신호를 통합하여 출력하는 NMEA 멀티플렉서 시스템의 시뮬레이션을 연구하였다. 입력되는 신호의 대역폭에 비례하여 수용하는 장비의 병목현상을 방지하고 최적의 데이터를 처리하기 위한 대기 큐 기법을 연구했으며, 입력되는 다수 개의 포트로부터 단일 또는 다수개의 포트로 신호를 출력시켜줄 수 있는 시스템에 대한 시뮬레이션을 통해 멀티플렉싱이 가능함을 연구하였다.

향후 NMEA 멀티플렉서가 임베디드 환경에서 개발될 경우 제한적인 시스템의 메모리량에 따라 대기 큐의 운영이 최적화될 수 있도록 하기 위해서는 가변적인 큐의 운영이 필수적이며, 데이터 처리의 분산 기법을 통하여 프레임 손실율이 없도록 모듈을 개발해야 할 것으로 생각된다.

본 논문에서의 통계를 분석해보면 실제 큐의 크기를 재조정할 때 트래픽의 부하를 줄이며, 신뢰성을 높일 수 있는 방법으로 큐를 동적으로 할당하는 과정이 고정 크기의 버퍼 방법보다 높은 효율과 프레임 처리 향상이 있다. 그러나 각 포트의 버퍼 크기를 고정해서 동적 큐 할당 알고리즘을 통하여 처리율 및 프레임 손실율의 결과를 도출 하였다. 그러므로 향후 각 항해 장비마다 버퍼의 크기를 가변적으로 했을 때의 결과를

고려해야 할 것으로 생각된다.

그리고 본 논문에서 구현한 NMEA 멀티플렉서는 외국의 고비용 장비 및 국내 제품 판매 저하 등을 고려하여 더욱더 경제적으로 개발하려는 목적으로 적절한 동적 큐 알고리즘을 이용한 멀티플렉서의 구현을 목적으로 시뮬레이션 하였다. 그러나 항해장비의 경우는 안전항해에 지대한 영향을 미치므로 논문내의 프레임 손실율을 없애는 방안을 향후 연구 과제로 다시 연구해야 할 것으로 사려 된다.

후 기

본 연구는 산업 자원부의 지역혁신 인력양성사업의 연구결과로 수행되었음.

참 고 문 헌

- [1] 가남사 편집부(1990), RS232C 인터페이스 사용법
- [2] 박재하, 김천곤(1999), GPS 측량의 이해, 부산정보대학 출판부, pp.105-115
- [3] 정보 조사 분석팀(2000), GPS 기술/시장 보고서, 한국전자통신연구원
- [4] Brookhouse NMEA multiplexer 제조사(2004), URL: <http://www.brookhouseonline.com>
- [5] D. Kozlov, M. Tkachenko(1997), Instant RTK cm with Low Cost GPS+GLONASS Receivers, Proc. of ION GPS-97, pp.1559-1569
- [6] Hoffman-Wellenhof, B., Lichtenegger, H., and Collins, J. (1997), Global Positioning System Theory and Practice, Springer Wien New York
- [7] John Catsoulis(2003), 임베디드 하드웨어 이해와 설계, 한빛미디어
- [8] NMEA 0183 Version 3.01(2002), STANDARD FOR INTERFACING MARINE ELECTRONIC DEVICES
- [9] SiRF Technology Inc.(2002), NMEA Reference Manual pp.1-5

원고접수일 : 2004년 12월 29일

원고채택일 : 2005년 2월 18일