

# 선박 디젤기관 원격제어 시스템을 위한 시뮬레이터 개발에 관한 연구

정 경 열\* · 류 길 수\*\*

## A Study on the Development of a Simulator for Ship Diesel Engine Remote Control Systems

*K.Y. Chung, K.S. Rhyu*

### Abstract

This paper introduces the development of a simulator for real-time systems, such as a ship engine remote control system. The real-time system, especially with a dead-time, needs to be tested whether it acts normally. The developed simulator is connected to a ship engine remote control system via PCL-812 boards, for acquiring data in real-time. In the simulator, digital and analog values are shown in different forms, such as ON-OFF, X-Y, or METER.

We made the time charts on control functions of a ship diesel engine, compared them with those by the simulator, and confirmed the same results.

## 1. 서 론

실시간 시스템의 필드측에는 센서, 액추에이터, 입출력 기기 및 다른 시스템들이 자리잡고 있으며, 이들 사이에서는 데이터들이 실시간으로 교환되고 있다. 최근에는 이러한 실시간 시스템들의 대부분은 마이크로프로세서 기반형으로 구성되는 경향이 있다.

선박 디젤기관 원격제어 시스템의 경우<sup>[1][2]</sup> 세 개의 서브시스템, 즉 BCS(Bridge Control

\* 한국기계연구원

\*\* 한국해양대학교

System), RCS(Remote Control System), SS(Safety System)로 구성되어 있으며, 실제로 NABCO사의 경우<sup>[3]</sup> Z80 MPU로 구성되어 있고, NORCON사의 경우에는<sup>[4]</sup> 8088 MPU로 구성되어 있다. 이 중에서 RCS는 선박 디젤기관을 기계측으로부터 멀리 떨어진 기관제어실이나 선교로부터 조종하기 위한 실시간 시스템으로서, 주기관의 기본적인 제어, 즉 주기관의 시동 및 정지, 전·후진, 증·감속, 이상운전시의 비상감속 및 비상정지, 위험진동대역의 자동회피 등의 기능들을 가지고 있다. 이러한 기능들은 정밀한 스캔스제어에 의해 해당 전자밸브를 작동시키게 된다. 따라서 이러한 실시간 시스템이 완성되면 정확한 동작을 수행하는 지를 확인하는 작업은 디젤기관을 보호하는 의미에서 대단히 중요한 단계중에 하나라고 할 수 있다.

본 논문에서는 선박 디젤기관 원격제어 시스템의 고기능화를 위해 하드웨어를 80186 MPU로 구성한 시스템의 동작확인을 위한 타임차트 시뮬레이터의 제작에 대하여 논한다. 시뮬레이터는 각종 전자밸브들의 온·오프 상태와 같은 디지털 값을 실시간으로 표시해 주기 위한 기능과 기관 회전수와 같은 아날로그 값을 실시간으로 표시해 주기 위한 기능으로 이루어져 있다. 또한, 이런 기능들을 PC상에서 윈도우 기반으로 구성하기 위한 방법과 시뮬레이션 결과에 대하여도 논한다.

## 2. 선박 디젤기관 원격제어 시스템의 구성

본 논문에서 이용하고 있는 선박 디젤기관 원격제어 시스템의 하드웨어<sup>[5],[6]</sup>는 80186MPU를

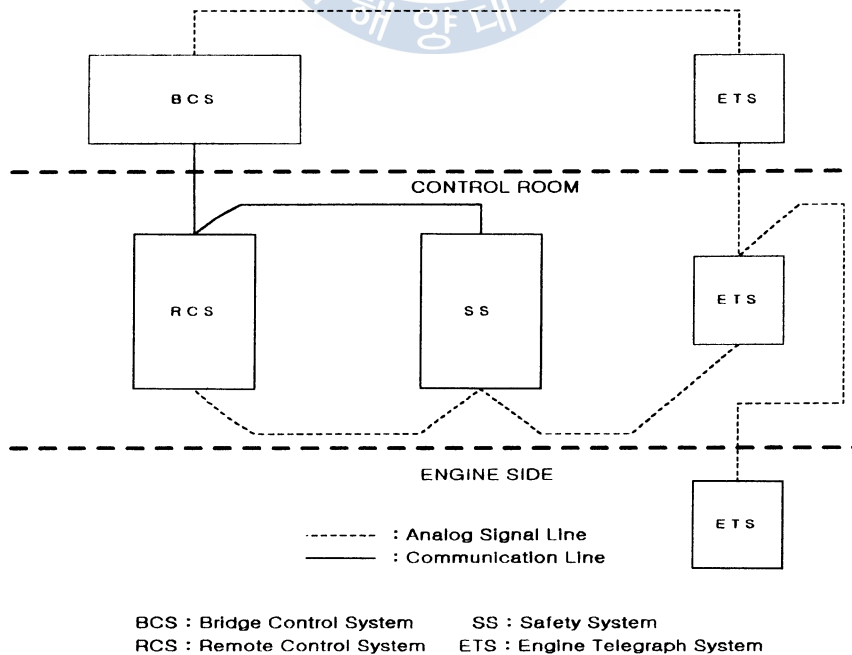


Fig. 1 Structure of a main engine remote control system

기반으로 하여 Fig. 1과 같이 3개의 서브시스템 (BCS, RCS, SS)과 ETS로 구성되어 있다. BCS에서는 RPM Command, 제어위치선택, 각종 스위치조작들이 가능하고, 이 값들을 RCS에 전송해 주며, 또한 디젤기관의 상태값 및 안전관련 값들을 전송받아 이것들을 패널에 디스플레이 해준다. RCS에서는 디젤기관의 원격제어기능과 함께, RPM 검출, 제어위치, 상태표시, 알람 표시 등을 해주며, 이 값들을 BCS와 SS에 전송해 준다. SS에서는 디젤기관을 감시보호하기 위해 RPM 및 Overspeed 검출, Shutdown, Slowdown, Emergency Stop 등의 시기를 결정해 주고, 이 값들을 RCS에 전송해 준다. ETS는 디젤기관을 기동 또는 정지하기 위한 약속신호를 발생함과 동시에 이 값들을 각각 BCS, RCS에 넘겨주게 된다.

이와 같이 3개의 서브시스템은 서로 유기적으로 데이터를 주고 받음으로써 자신의 기능을 원활히 수행한다. 각 서브시스템들은 서로 통신을 수행하기 위해 인터럽트용 8259소자와 타이머용 8253소자 각 1개, 시리얼 통신용 8251소자 두 개를 이용하여 RS-422A방식<sup>[7],[8]</sup>으로 연결되어 있다.

### 3. 시뮬레이터의 구성

시뮬레이션을 수행하기 위해 Fig. 2와 같이 I/O 신호박스를 통하여 RCS와 PC를 연결하였으며, RCS측과는 각 센서 또는 전자밸브 등으로 연결되어 있는 선과 직접 결선하고, PC측과는 PCL-812보드를 통하여 연결하였다. 엔진의 상태를 나타내기 위해 센서 55개, 동작시기를 나타내기 위해 전자밸브용으로 7개, 기관회전수를 위해 1개를 각각 연결하였다.

#### 3.1 데이터의 변환방법

다음은 PCL-812보드를 이용하여 기관회전수를 12비트 A/D 데이터로 읽어들이는 방법을 예

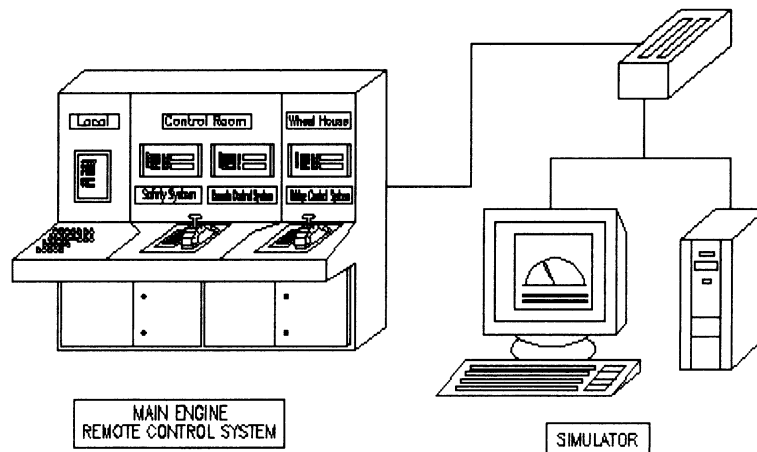


Fig. 2 Connection of a main engine remote control System and a simulator

시하고 있다. 처음 2줄은 PCL-812보드를 채널 1로 설정하여 놓고 소프트웨어 트리거를 걸어줌으로써 초기화 하는 것을 의미한다. 그 다음 do - while 문에서는 8비트의 데이터를 읽어들이 이것의 5비트째가 0이면 상위비트를 이미 읽어들었다는 의미이므로 이 문장을 빠져나와 다음 하위 8비트 데이터를 읽어들이 후 이것을 MAKEWORD함수에 의해 12비트 데이터로 변환시킨 다음, 최종적으로 0에서 300사이의 RPM값으로 변환하게 된다.

```

outportb(0x23A,1);
outportb(0x23C,0x00);
do {HiADvalue = inportb(0x235);} while ((HiADvalue & 0x10) != 0 );
LoADvalue = inportb(0x234);
ADvalue = MAKEWORD(LoADvalue, HiADvalue);
OrderRpm = (300L* (long)ADvalue) / 0xffff;

```

전자밸브의 경우에는 채널번호만 다르게 하고 나머지는 동일하다. 단 마지막으로 읽어들이 값을 디지털 값으로 변환시켜야 하므로 다음과 같은 비교문에 의해 읽어들이 값이 1024보다 크면 밸브가 온상태로 그렇지 않으면 오프상태로 인식하게 된다.

```

if (ADvalue > 1024) iSolCheck[i] = TRUE;
else iSolCheck[i] = FALSE;

```

### 3.2 기준값의 저장방법

실시간시스템이 시퀀스제어를 수행하기 위해서는 각 전자밸브를 온 또는 오프상태로 해야 할 기준 시간을 기억해 두어야 하며, 다른 센서들에 관해서도 설정값을 기억해 두어야 할 필요가 있다. 또한 시스템이 다운된 경우에도 빠른 복구를 위해서는 주요 데이터를 보존할 필요가 있다. 이를 위해 본 논문에서는 실시간 시스템측에 EEPROM을 장착하여 이용하고 있다. 이 EEPROM에 각 전자밸브의 개폐시기의 기준값을 저장해 두고 이것을 PC측에 넘겨주도록 하고 있다.

EEPROM을 하드웨어적으로 구성하는 경우 RAM과 동일한 세그먼트내에 두는 것이 불가능하며 RAM의 확장을 고려하여야 하므로, 메모리번지의 뒤쪽 즉 0x80000 - 0x87fff 번지에 연결하여 두는 것이 일반적이다. 따라서 소프트웨어적으로 이 번지를 조작하고자 하는 경우에는 직접 번지 액세스 기법이 요구된다.

Intel 80186 MPU에서의 메모리데이터 참조방법은 다음과 같이 구분된다.

```

┌ segment내 -> offset값을 이용한 near pointer (2 byte)
└ segment外 -> segment:offset값을 이용한 far pointer (4 byte)

```

C언어로 물리번지 0x80000를 직접 참조하기 위해서는 다음과 같은 기법이 요구되며, 반드시

far pointer로 정의되어야만 한다. 단, 직접 참조시에는 원하는 데이터가 물리적으로 어느 번지에 저장되어 있는지를 사전에 약속하여 놓을 필요가 있다.

```
char far *a;  
a = (char far *) 0x8000000L; /* 물리 address 0x80000 */
```



다음은 세그먼트외의 물리번지를 직접 Read / Write 하는 C 언어의 주요 부분만을 나타낸 프로그램이다.

**write 법 :**

```
static unsigned int buf[1000];  
movedata( FP_SEG(buf), FP_OFF(buf), 0x8000, 0x0000, 1000);
```

**read 법 :**

```
static unsigned int buf[1000] ;  
int i ;  
unsigned int far *inc ;  
inc = (unsigned int far *)0x8000000L ;  
for (i = 0 ; i < 1000 ; i++)  
    buf[i] = *(inc++) ;
```

#### 4. 시뮬레이터의 기능

전자밸브의 온·오프 시기를 시뮬레이션하기 위해 Fig. 3 과 같은 화면을 구성하였다. 전자

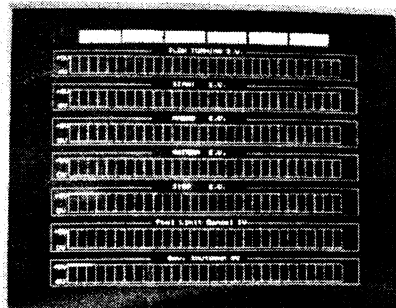


Fig. 3 Initial display menu of the proposed simulator

밸브의 신호가 5V이면 온 상태를 나타내고 0V이면 오프 상태를 나타내도록 하였다. 여기에 표시되는 전자밸브의 종류 및 해당 PCL-812보드의 채널번호는 표 1과 같다.

Table 1 Channel numbers of a PCL-812 board connecting to solenoid valves

	전 자 밸 브 종 류	채 널 번 호
①	SLOW TURNING S.V.	5
②	START S.V.	6
③	AHEAD S.V.	7
④	ASTERN S.V.	8
⑤	STOP S.V.	9
⑥	FUEL LIMIT CANCEL S.V.	3
⑦	GOV. SHUTDOWN S.V.	4

그림에서 각 메뉴의 기능은 다음과 같다.

- (1) START  
: 각 전자밸브의 상태를 실시간으로 나타내기 시작한다.
- (2) PAUSE  
: 현재 진행되고 있는 화면을 일시 정지한다.
- (3) CLEAR  
: 화면을 지우고 새로 그린다.
- (4) ANALOG  
: 시간축을 기반으로하여 기관회전수 요구량과 실 기관회전수를 나타낸다.
- (5) GAUGE  
: 메터 형태로 기관회전수 요구량과 실 기관회전수를 나타낸다.
- (6) EXIT  
: 프로그램을 종료한다.

Fig. 4는 ANALOG 버튼을 눌렀을 때 표시되는 화면이다. 그림에서 Governor는 기관회전

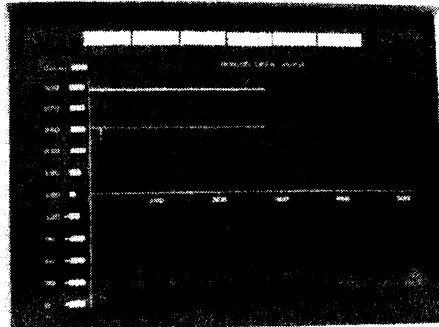


Fig. 4 Menu when ANALOG button is pressed

수 요구량 값을 의미하며, RPM은 실 기관회전수를 각각 의미한다. RPM은 기관회전 방향이 전진, 후진일 때 각각 다르게 표시하기 위해 전진방향 최대값을 250, 후진방향 최대값을 -250으로 표시하였다.

Fig. 5는 GAUGE 버튼을 눌렀을 경우에 나타나는 화면으로서 동일 게이지상에 기관회전수 요구량과 실 기관회전수를 표시할 수 있도록 하였다.

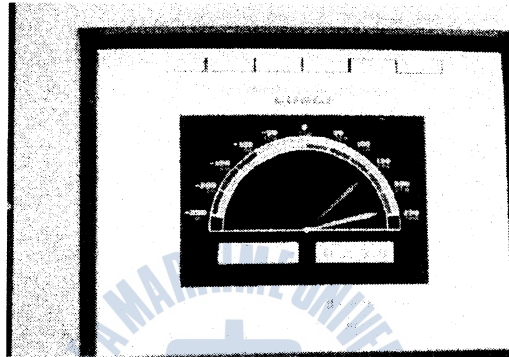


Fig. 5 Menu when GAUGE button is pressed

## 5. 시뮬레이션

Fig. 6은 본 논문에서 구성한 선박 디젤기관 원격제어 시스템과 시뮬레이터를 보여주고 있다. 좌측으로부터 시뮬레이터, 연결 박스, 직류전원장치, BCS, RCS, SS 순으로 배열되어 있다.

시뮬레이션을 위해 우선 원격제어 시스템의 전·후진 시동, 정지, 증·감속, 비상 감속, 비상 정지, 3회 시동실패, 비상 후진, 비상 전진 등에 관한 타임차트를 작성하였다. Fig. 7은 이 중에서 비상후진에 관한 타임차트를 보여주고 있다. 이 기능들에 대하여 시뮬레이터를 통해 테스트 하여 보았다. Fig. 8은 비상후진과 3회 시동실패에 관한 수행결과를 보여주고 있으며, Fuel

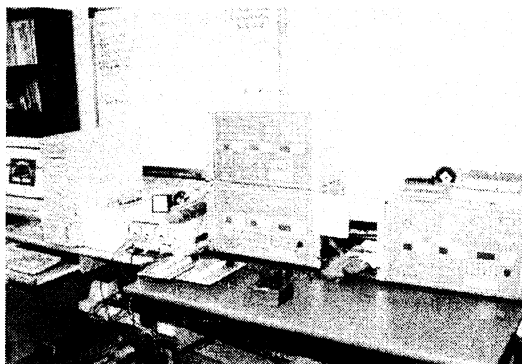


Fig. 6 Picture of the proposed system

limit cancel S.V. 가 온상태에서 오프상태로 바뀌는 첫 번째 위치가, 선박의 전진방향으로의 타력에 의해 비상후진이 실패한 것을 보여주는 시점이며, 이 때부터 3회 후진방향 시동에 들어가게 되는데 그림에서는 2회만에 정상적으로 시동완료하여 후진방향으로 운전하고 있다는 것을 의미하고 있다. Fig. 7과 Fig. 8의 그림에서 각 전자밸브의 작동시기를 비교하여 본 결과 정상적으로 동작되었다는 것을 알 수 있었다.

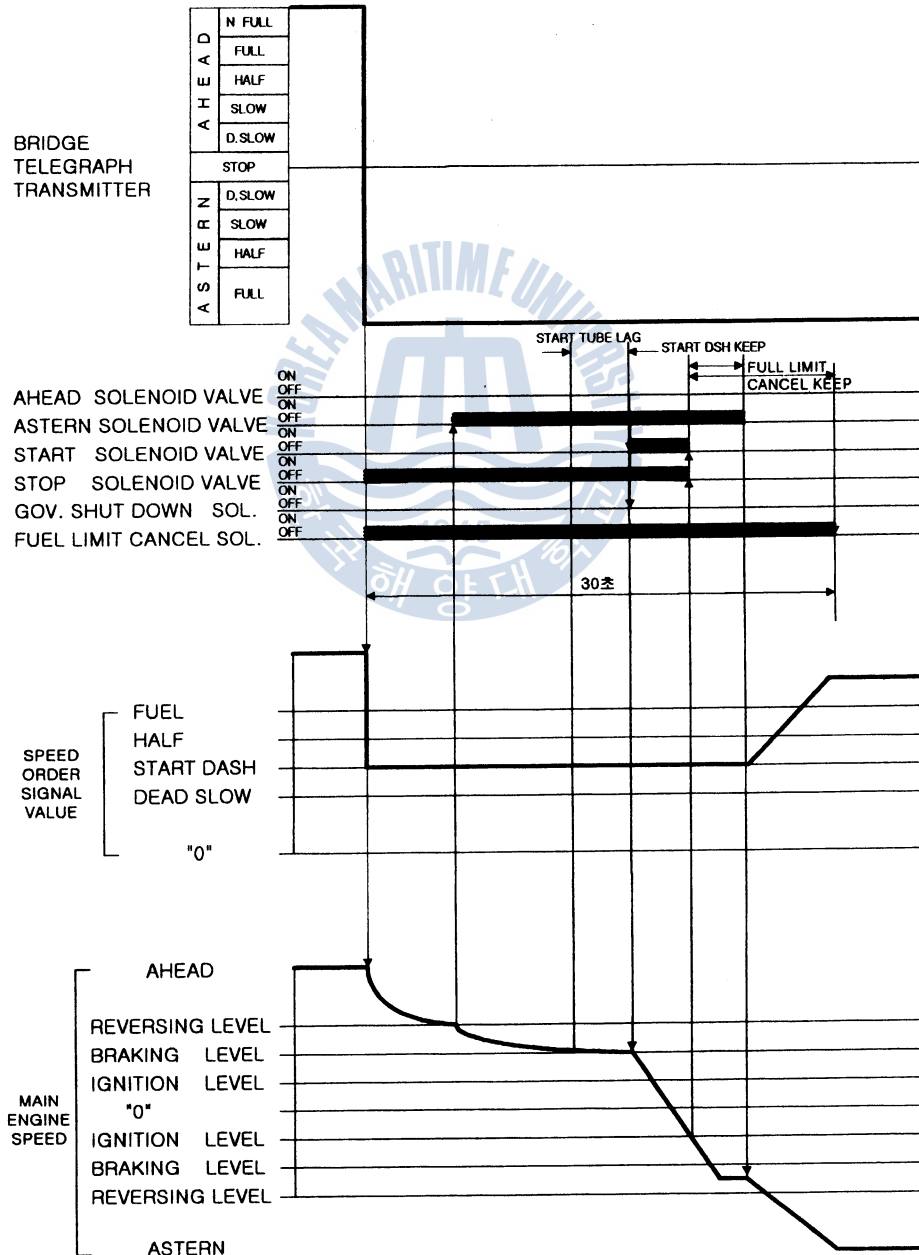


Fig. 7 Time chart of emergency astern



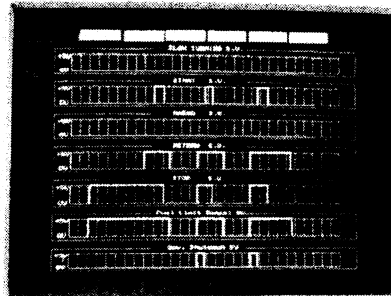


Fig. 8 Simulation result of emergency astern

## 6. 결 론

본 논문에서는 선박 디젤기관 원격제어 시스템과 같은 실시간시스템을 위한 시뮬레이터 개발에 관하여 논하였다. 지연시간을 가지는 실시간 시스템들은 개발이 완료된 후 정상적으로 동작되는지의 검증이 반드시 필요하다. 이를 위해서는 동작중인 시스템으로부터 실시간 데이터를 획득하여 시뮬레이터에서 표시하여 주어야만 하는데, 여기에서는 PCL-812 보드를 이용하였다.

시뮬레이터에서 표시해 주기 위한 기능으로서의 시간변화에 따른 디지털 값의 표시, 아날로그 값의 표시 방법을 제안하였다. 또한 선박 디젤기관의 제어기능에 관한 타임차트를 작성한 후 이것과 시뮬레이터에 의한 결과를 비교하여 보았으며, 그 결과 정확한 동작을 하고 있음을 확인할 수 있었다.

급후 시뮬레이터의 표시기능에 대하여 보다 더 다양하고 이해하기 쉬운 방법을 제공함으로써 사용자에게 편리성을 도모할 필요가 있을 것이다.

## 참 고 문 헌

- [1] 中川司, “船用コンピュータシステムと國際動向”, 日本舶用機關學會誌, 第32卷, 第2號, pp.79-85, 1997.
- [2] 細川成通, “主機關の自動化の現狀と將來動向”, 日韓學術交流講演會, 1993.
- [3] M-800B Bridge Maneuvering System Manual, NABCO.
- [4] Auto-chief 4 Main Engine Remote Control System Manual, NORCON.
- [5] 정경열외8명, “선박용 중소형 엔진 자동제어 시스템 개발”, 통산산업부연구보고서, 1996
- [6] 황동희, “실시간멀티태스크 처리기법을 이용한 선박엔진 안전시스템 개발에 관한 연구”, 한국해양대학교석사학위논문, 1995.
- [7] 宮崎誠一ら, “標準シリアルインターフェース”, トランジスタ技術 Special No. 8, CQ出版社, 1992.
- [8] 石川裕次, “シリアル轉送用LSIの使い方”, トランジスタ技術 Special No. 8, CQ出版社, 1992.

