



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

工學博士 學位論文

ECS 신뢰도 확보를 위한  
함정 추진체계용 시뮬레이터 개발

The development of warship propulsion system simulator  
for ECS reliability



2016年 2月

韓國海洋大學交 大學院

메카트로닉스 공학과

鄭 星 泳

# 목 차

List of Tables .....	vi
List of Figures .....	viii
Abstract .....	xii

## 1. 서 론

1.1 연구 배경 .....	1
1.2 연구 동향 .....	5
1.2.1 ECS 개발 과정과 시뮬레이터의 활용 .....	6
1.2.2 함정 추진체계용 시뮬레이터 모델링 .....	8
1.3 연구 내용 및 구성 .....	11

## 2. 가상 함정에 대한 정의

2.1 개요 .....	12
2.2 국외 함정 추진체계 분석 .....	13
2.3 가상 함정의 추진체계 .....	16
2.3.1 가스터빈 .....	17
2.3.2 감속기어 .....	19
2.3.3 추진전동기 .....	22
2.3.4 축계 및 프로펠러 .....	23

<b>3. 시뮬레이터를 이용한 ECS 검증 방안</b>	
3.1 개요 .....	25
3.2 ECS의 정의 .....	25
3.3 ECS 검증 방안에 대한 검토 .....	27
3.3.1 ECS 주요 검증항목 .....	27
3.3.2 주요 검증항목에 따른 알고리즘 .....	28
3.4 ECS 주요 검증항목에 따른 시뮬레이터 개념 설계 .....	38
<b>4. 함정 추진체계용 시뮬레이터 개발</b>	
4.1 시뮬레이터 개발 방향 .....	39
4.2 하드웨어 설계 .....	44
4.3 소프트웨어 설계 .....	47
4.3.1 동적 시뮬레이션을 위한 OS .....	47
4.3.2 RT OS의 우선순위 .....	49
4.3.3 소프트웨어 설계 구체화 .....	51
4.4 연동장비 모델 개발 .....	55
4.4.1 기본 모델 .....	55
4.4.2 연동장비 제어기 모델 .....	69
4.4.3 기타 장비특성 모델 .....	79
4.4.4 신호 입출력 모델 .....	79
4.4.5 시스템 통합 .....	84
<b>5. 함정 추진체계용 시뮬레이터 검증</b>	
5.1 가상의 ECS 구축 .....	87
5.2 가상의 ECS를 이용한 시뮬레이터 검증 .....	91
<b>6. 결 론</b>	
<b>참고문헌</b> .....	113

## Nomenclatures

$\rho_w$	: 해수 밀도 [ $\text{kg/m}^3$ ]
$c$	: 블레이드의 현의 길이 [m]
$e_r$	: 감속기어 에너지 전달 효율
$k$	: 기어비
$l$	: 블레이드의 가장 두꺼운 부분의 두께 [m]
$n_g$	: 가스터빈의 회전속도 [rpm]
$n_p$	: 프로펠러 회전속도 [rps]
$A_E/A_O$	: 블레이드 비
$D$	: 프로펠러 직경 [m]
$J_a$	: 전진계수
$K_T$	: 프로펠러 추력 계수
$K_Q$	: 프로펠러 토크 계수
$P/D$	: 프로펠러의 피치와 직경의 비
$P_p$	: 프로펠러 출력 [kW]
$Q_a$	: 축에 설치된 발전기 토크 [Nm]
$Q_e$	: 엔진의 토크 [Nm]
$Q_p$	: 프로펠러 토크 [Nm]
$R_n$	: 레이놀드 수
$T_p$	: 추력 [N]
$V$	: 함속 [m/s]
$Z$	: 프로펠러 블레이드 개수

AI	: Analog Input
AO	: Analog Output
API	: Application Program Interface
CCG	: Cross Connect Gear
CODAD	: COmbined Diesel And Diesel
CODAG	: COmbined Diesel And Gas turbine
CODLAG	: COmbined Diesel-eLectric And Gas turbine
CODOG	: COmbined Diesel Or Gas turbine
COGAG	: COmbined Gas turbine And Gas turbine
COGOG	: COmbined Gas turbine Or Gas turbine
DAU	: Data Acquisition Unit
DI	: Digital Input
DO	: Digital Output
ECS	: Engineering Control System
EMS	: Engine Management System
EPCAMS	: Electric Power Control and Management System
FIFO	: First In First Out
FREMM	: European multi-purpose frigate
FPP	: Fixed Pitch Propeller
GT	: Gas Turbine
HILS	: Hardware In the Loop Simulation
IA	: Interface Agreement
IFEP	: Integrated Full Electric Propulsion
MCAS	: Machinery Control and Alarm System
MILS	: Model In the Loop Simulation
MIT	: Model Interface Toolkit
MRG	: Main Reduction Gear
OS	: Operating System
PM	: Propulsion Motor
PSI	: Propulsion System Integration
PXI	: PCI eXtensions for Instrumentation
RG	: Reduction Gear
SCM	: Simulation Control and Monitoring
SILS	: Software In the Loop Simulation
SIT	: Simulation Interface Toolkit
SSS Clutch	: Synchro Self Shifting clutch
TDMS	: Technical Data Management Streaming

## List of Tables

Table 1 Specification of Type 26 ship .....	15
Table 2 Specification of MT30 .....	17
Table 3 I/O list of MT30 .....	18
Table 4 I/O list of MT30 auxiliary machinery .....	18
Table 5 Specification of reduction gear .....	19
Table 6 I/O list of reduction gear .....	20
Table 7 Specification of propulsion motor .....	22
Table 8 I/O list of propulsion motor .....	23
Table 9 Specification of shaftline & FPP .....	24
Table 10 I/O list of shaft & FPP .....	24
Table 11 Sequential algorithm of ECS .....	27
Table 12 Individual components of system .....	40
Table 13 Hardwired signal for simulator .....	43
Table 14 Component of central simulator .....	45
Table 15 Inter-process communication .....	50
Table 16 Input for GT basic model .....	59
Table 17 Output for GT basic model .....	59
Table 18 Input for PM basic model .....	60
Table 19 Output for PM basic model .....	60
Table 20 Input for RG basic model .....	63
Table 21 Output for RG basic model .....	63
Table 22 Value of coefficients $K_T$ and $K_Q$ .....	65
Table 23 Input for propeller basic model .....	67
Table 24 Output for propeller basic model .....	67
Table 25 I/O signal for GT internal check .....	70
Table 26 I/O signal for GT normal start .....	70
Table 27 I/O signal for GT normal stop .....	71

Table 28	I/O signal for RG internal check .....	73
Table 29	I/O signal for RG brake system .....	74
Table 30	I/O signal for RG SSS clutch control algorithm .....	75
Table 31	I/O signal for PM control algorithm .....	77
Table 32	Concept of MODBUS memory block .....	80
Table 33	Allocation of MODBUS memory block (1) .....	80
Table 34	Address allocation of MODBUS memory block (2) .....	82
Table 35	I/O signal for basic model .....	84
Table 36	Configuration of cRIO hardware .....	87



## List of Figures

Fig. 1	Change of royal navy combatant propulsion system .....	2
Fig. 2	Global navy combatant propulsion types .....	2
Fig. 3	Technology of propulsion system .....	3
Fig. 4	SEASTEMA FREMM simulators .....	5
Fig. 5	Design procedure for propulsion controller .....	6
Fig. 6	Components for hardware in the loop simulation .....	8
Fig. 7	Ship propulsion dynamics .....	9
Fig. 8	Shaft line dynamics .....	9
Fig. 9	Percentage of propulsion system for global warship .....	13
Fig. 10	Propulsion system for Type 26 .....	14
Fig. 11	Propulsion system for virtual warship .....	16
Fig. 12	MT30 component .....	17
Fig. 13	Operation of SSS clutch .....	20
Fig. 14	SSS clutch servo-actuated lockout .....	22
Fig. 15	Configuration of ECS .....	25
Fig. 16	ECS control & monitoring signal .....	28
Fig. 17	Sequential control algorithm for GT start .....	31
Fig. 18	Sequential control algorithm for GT stop .....	32
Fig. 19	Sequential control algorithm for PM start .....	33
Fig. 20	Sequential control algorithm for PM stop .....	34
Fig. 21	Sequential control algorithm for GT to PM changeover .....	35
Fig. 22	PM to GT changeover sequential control algorithm .....	36
Fig. 23	Sequential control algorithm for GT emergency start .....	37
Fig. 24	Simulation method .....	39
Fig. 25	Procedure of controller verification .....	41
Fig. 26	Configuration of ECS for warship .....	41
Fig. 27	Concept of HILS simulator .....	42

Fig. 28	PXI(PCI eXtensions for Instrumentation) .....	44
Fig. 29	Configuration of simulator hardware .....	46
Fig. 30	Photo of simulator .....	46
Fig. 31	Jitter in general purpose OS .....	48
Fig. 32	Comparison of general purpose OS and real-time OS .....	48
Fig. 33	Separation of deterministic and non-deterministic task .....	49
Fig. 34	Sharing data on RT target .....	50
Fig. 35	Conceptual diagram of simulator (1) .....	51
Fig. 36	Conceptual diagram of simulator (2) .....	52
Fig. 37	Development process of simulator program .....	53
Fig. 38	Configuration of equipment model .....	53
Fig. 39	Conceptual diagram of ship dynamic simulation .....	55
Fig. 40	Conceptual diagram of basic model interaction .....	57
Fig. 41	Step response of GT, PM .....	58
Fig. 42	Program for GT basic model .....	59
Fig. 43	Program for PM basic model .....	60
Fig. 44	Program for RG basic model .....	62
Fig. 45	Fitting of coefficient ( $K_T$ and $K_Q$ ) .....	66
Fig. 46	Coefficient $K_T$ and $K_Q$ for propeller RPM .....	67
Fig. 47	Program for shaft and FPP basic model (1) .....	68
Fig. 48	Program for shaft and FPP basic model (2) .....	68
Fig. 49	Program for GT internal check .....	70
Fig. 50	Program for GT normal start .....	71
Fig. 51	Program for GT normal stop .....	71
Fig. 52	Overall program for GT control logic .....	72
Fig. 53	Program for RG internal check .....	73
Fig. 54	Program for RG brake system .....	74
Fig. 55	Program for SSS clutch control algorithm .....	75
Fig. 56	Overall program for RG control logic (1) .....	76
Fig. 57	Overall program for RG control logic (2) .....	76
Fig. 58	Program for PM control algorithm .....	77
Fig. 59	Overall block program for control logic .....	78
Fig. 60	MODBUS master-slave structure .....	79

Fig. 61	Signal input/output model program .....	83
Fig. 62	Overall program for PXI .....	85
Fig. 63	cRIO & NI 9215, 9264, 9401 .....	87
Fig. 64	Hardwired signal diagram between PXI and cRIO .....	88
Fig. 65	Photo of PXI and cRIO .....	88
Fig. 66	Front panel for ECS simulation program .....	89
Fig. 67	Block diagram for ECS simulation program .....	90
Fig. 68	Data storage & DIAdem .....	91
Fig. 69	GT start simulation output - GT .....	92
Fig. 70	GT start simulation output - shaft and clutch .....	93
Fig. 71	GT start simulation output - PM and FPP .....	94
Fig. 72	GT stop simulation output - GT .....	95
Fig. 73	GT stop simulation output - shaft and clutch .....	96
Fig. 74	GT stop simulation output - PM and FPP .....	97
Fig. 75	PM start simulation output - GT .....	98
Fig. 76	PM start simulation output - shaft and clutch .....	99
Fig. 77	PM start simulation output - PM and FPP .....	99
Fig. 78	PM stop simulation output - GT .....	100
Fig. 79	PM stop simulation output - shaft and clutch .....	101
Fig. 80	PM stop simulation output - PM and FPP .....	101
Fig. 81	GT to PM changeover simulation output - GT .....	102
Fig. 82	GT to PM changeover simulation output - shaft and clutch .....	103
Fig. 83	GT to PM changeover simulation output - PM and FPP .....	104
Fig. 84	PM to GT changeover simulation output - GT .....	105
Fig. 85	PM to GT changeover simulation output - shaft and clutch .....	106
Fig. 86	PM to GT changeover simulation output - PM and FPP .....	107
Fig. 87	GT emergency start simulation output - GT .....	108
Fig. 88	GT emergency start simulation output - shaft and clutch .....	109
Fig. 89	GT emergency start simulation output - PM and FPP .....	109

# The study of warship propulsion system simulator design

Jung, Sung Young

Department of Mechatronics  
Graduate School of Korea Maritime University

## Abstract

In the past, diesel engines, steam turbines or gas turbines were the typical mechanical propulsion system that was used on the warship. Therefore, the power train was simple and it was easy to repair and integrate into a system.

However, due to recent awareness of survival performance and automation, propulsion system of warships is evolving to hybrid mechanical-electrical propulsion system. As a result, the requirements of the propulsion control system have become more complicated, and warships ECS(Engineering Control System) have gained more attention.

During the operation of the ship, controlling the propulsion system is getting more complicated. And this is one of the most important factor to consider to while maintaining or rapidly changing the speed of the warship.

Since this ECS is manufactured only from foreign companies (L-3 Mapps,

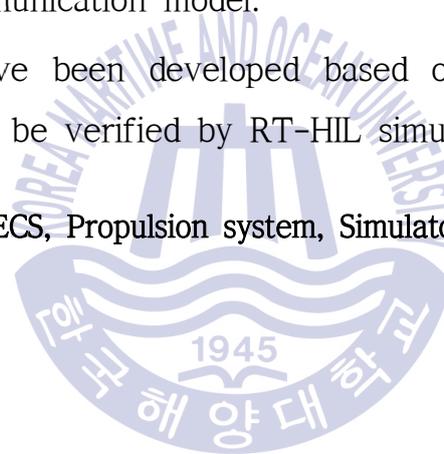
SEASTEMA), localizing its technology has become important. And ensuring the reliability by using the simulator is the critical factor of localization.

This paper is a study of warship propulsion system simulator design. In order to develop this simulator, the procedure is based on four main steps.

First, in order to collect core technologies and informations, a case study is carried out by internet and conference papers focusing on SEASTEMA and University of Genoa. Second, to develop the simulator, virtual warship propulsion system such as defining equipment specifications has been defined based on real operating data from the warship propulsion system. Then, RT-HIL(Real Time Hardware In the Loops) simulator was developed by LabVIEW and PXI. And it contains 4 sub-models; Basic model, logic model, etc model, communication model.

Finally, scenarios have been developed based on the warship's critical conditions and ECS will be verified by RT-HIL simulator.

**KEY WORDS : Warship, ECS, Propulsion system, Simulator, RT-HILS**



# 제 1 장 서 론

## 1.1 연구 배경

전함(battle ship)시대 이후 화약을 이용한 무기체계는 미사일 기반의 전자기 무기체계로 진화하였고, 이러한 무기체계의 변화는 함정에서의 전력 수요를 급격하게 증가시키게 되었다[1].

또한, 1980년 중반에 속도제어가 가능한 반도체 적용 교류 가변속 드라이브의 출현으로 고출력, 속도 제어 추진전동기와 고정주파수 발전기 시스템의 조합이 가능하게 됨에 따라 해군 함정의 추진체계는 디젤 엔진 등 기계식 추진체계뿐만 아니라 추진전동기 기반의 전기식 추진체계를 활용한 함정이 개발되기 시작하였다[2].

이러한 두 개의 기술적 발달이 복합적으로 이뤄진 결과, 함정 추진체계는 디젤엔진, 가스터빈 등 기존의 기계식(mechanical) 추진체계에서 디젤엔진과 추진전동기, 가스터빈과 추진전동기 등 복합식(hybrid) 추진체계로 변해가고 있는 추세이다.

이러한 복합식 추진체계는 함정 운용 목적에 따라 적절한 추진체계를 사용함으로써 기존의 기계식 추진체계에 비해 최적의 성능을 낼 수 있으며, 연료 효율 측면에서도 상대적으로 높다는 장점을 가진다.

Fig. 1은 영국 해군의 추진체계의 변화를 나타내고 있다. Fig. 1에 따르면, 과거 기계식 추진체계에서 1988년 이후 복합식 추진체계의 비율이 계속해서 높아지고 있으며, 2002년부터 전기식 추진체계의 비율이 점점 증가하는 것을 확인할 수 있다[3].

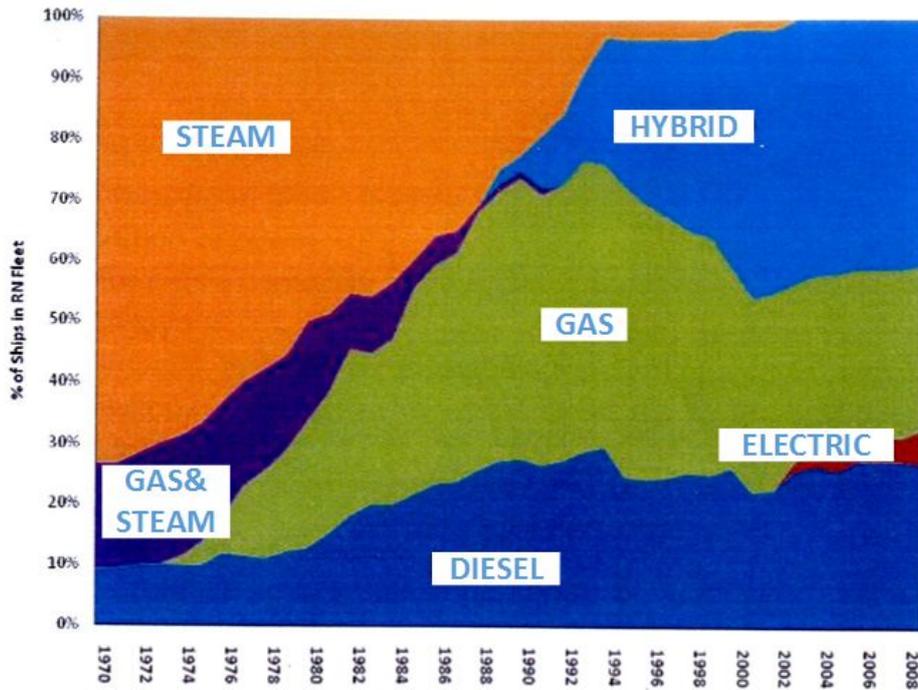


Fig. 1 Change of royal navy combatant propulsion system

Fig. 2는 전 세계 주요국가에서 건조되거나 건조중인 함정을 나타내고 있다. 대부분 복합식 또는 전기식 추진체계를 선택하고 있으며, 우리나라도 이러한 세계적인 추세에 맞추어 복합식 추진체계의 함정을 건조하고 있다.

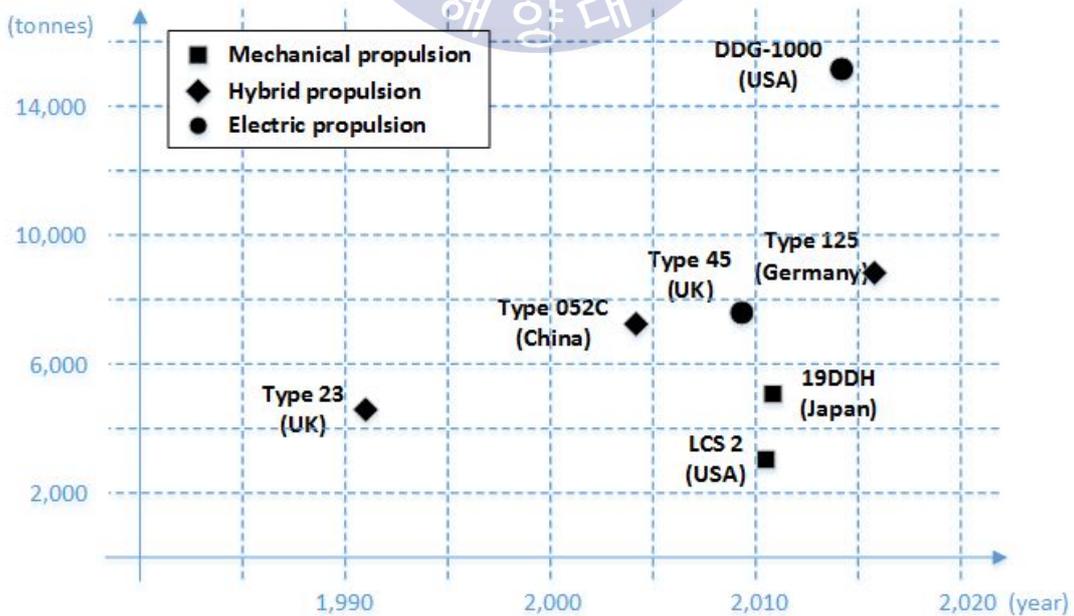


Fig. 2 Global navy combatant propulsion types

이러한 복합식, 전기식 추진체계는 함에 탑재된 모든 발전기의 출력을 추진계통과 함내 부하에 공통으로 사용하기 때문에 전력공급의 신뢰도가 높다. 또한, 부하에 따른 출력조정이 용이하기 때문에 탑재에너지의 효율적인 사용이 가능하므로 저소음의 장점과 함께 미래의 추진방식은 대부분 복합식, 전기식 추진체계가 될 것으로 전망하고 있다.

함정 추진기술은 추진동력원기술, 추진동력전달기술, 추진기기기술, 추진제어이며, Fig. 3은 함정의 추진기술을 나타내고 있다.

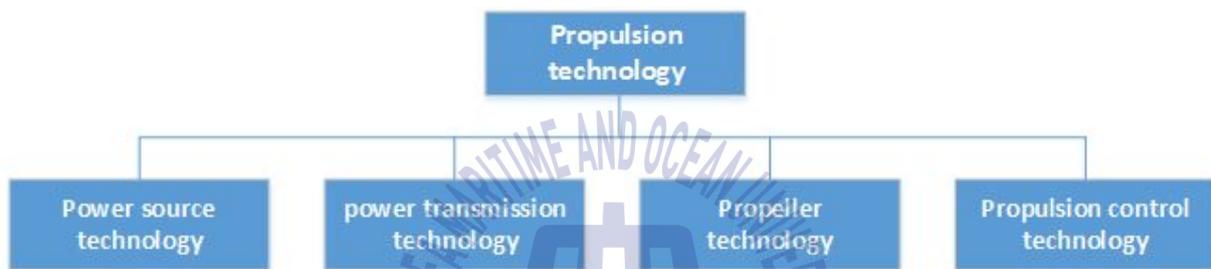


Fig. 3 Technology of propulsion system

각 기술별 우리나라의 기술 달성도를 보면 추진기 기술은 80%로 선진권 수준이며, 추진동력원, 추진동력전달 기술은 각각 77%, 71%로 중진권의 보통수준이다. 그러나, 추진제어 기술은 61%로 국내 개발능력이 부족하다[4].

이러한 추진제어 기술은 함정의 추진체계를 전반적으로 제어/감시하기 위해 정보 네트워크 기술을 활용하여 기존 함정에 비해 월등히 향상된 체계 운용성, 신뢰성, 효율성을 확보함으로써 함정 운용 비용의 절감과 함정 자동화 확대를 통한 승조원 감소 등 함정 운용에 있어서 매우 중요한 역할을 담당하고 있다. 이러한 시스템을 ECS(Engineering Control System)이라 하며, 캐나다의 L-3mapps나, 이탈리아의 SEASTEMA 등이 함정용 ECS 관련 주요 업체이다.

최근 우리나라 함정의 추진체계가 기존의 기계식 추진체계에서 복합식 추진체계로 발전함에 따라 추진체계 시스템 통합 관리 및 운용의 관점에서 추진제

어기술이 차지하는 비율은 점점 넓어지고 있다. 그러나, 추진제어를 수행하는 주 장비인 ECS는 모두 국외 업체에서 개발한 장비를 적용하고 있고, 이 결과 함정 건조 예산 과대화, 함정 추진체계 관련 기밀 정보 유출 등 다양한 문제가 발생하고 있다. 최근, 우리나라에서는 위와 같은 문제를 해결하기 위해 ECS 개발을 위한 기본연구를 수행하고 있다.

ECS 기술 국산화에서 가장 중요한 점은 신뢰성을 확보하는 것이다. ECS는 함정의 전반적인 시스템을 관리 및 제어하기 때문에 ECS의 고장 및 알고리즘 오류가 발생하는 경우 추진체계에 심각한 손상을 가져올 수 있다.

국외 ECS 개발 업체의 경우 여러 시뮬레이션 검증 방법을 이용하여 ECS의 제어 알고리즘과 하드웨어에 대한 신뢰성을 확보하고 있다.



## 1.2 연구 동향

현재까지 함정 추진체계용 ECS는 국외 전문업체에서 개발하였기 때문에 국내에는 ECS 검증용 시뮬레이터에 대한 연구 개발사례가 없다. 그러나 국외의 경우, 이탈리아의 SEASTEMA에서는 ECS 검증용 시뮬레이터를 2000년대 초반부터 개발하여 현재 HILS 기반의 시뮬레이터를 이용하여 ECS 개발에 활용하고 있다.

이 업체는 Fincantieri 조선소에서 만든 항공모함 Cavour와 FREMM(Frigate European Multi-Mission)급 함정의 ECS와 PSI 업무를 수행하였으며, ECS 개발 과정에서 실시간 HILS 시뮬레이터를 개발하여 검증을 수행함으로써 ECS의 신뢰성에 기여했다[5].

Fig. 4는 SEASTEMA가 개발한 FREMM 함정용 시뮬레이터 개발 과정에 대해 나타내고 있다.

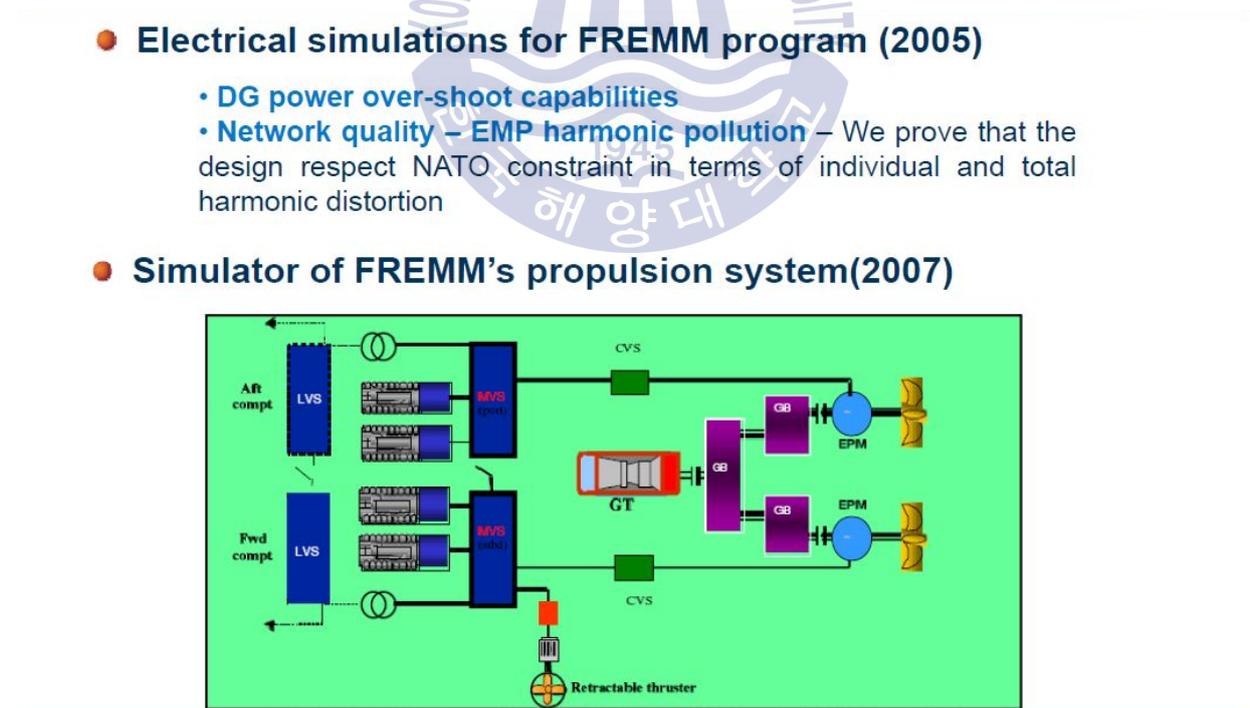


Fig. 4 SEASTEMA FREMM simulators

또한, SEASTEMA는 Genova 대학의 M Figari, M Altosole, G Benvenuto 등과 함께 2000년부터 2013년까지 시뮬레이터 개발에 관한 논문 11편을 학회에 제출하였으며, 2009년 항공모함 Cavour 함정을 대상으로 한 논문[6]과 2011년 FREMM 함정을 대상으로 한 논문[7]에서 함정 추진체계용 시뮬레이터 개발에 대한 과정과 함정 모델링에 대하여 설명하고 있다.

### 1.2.1 ECS 개발 과정과 시뮬레이터의 활용

참고문헌 [6]은 ECS의 개발 과정을 Fig. 5와 같이 크게 4단계로 나누고 있다.

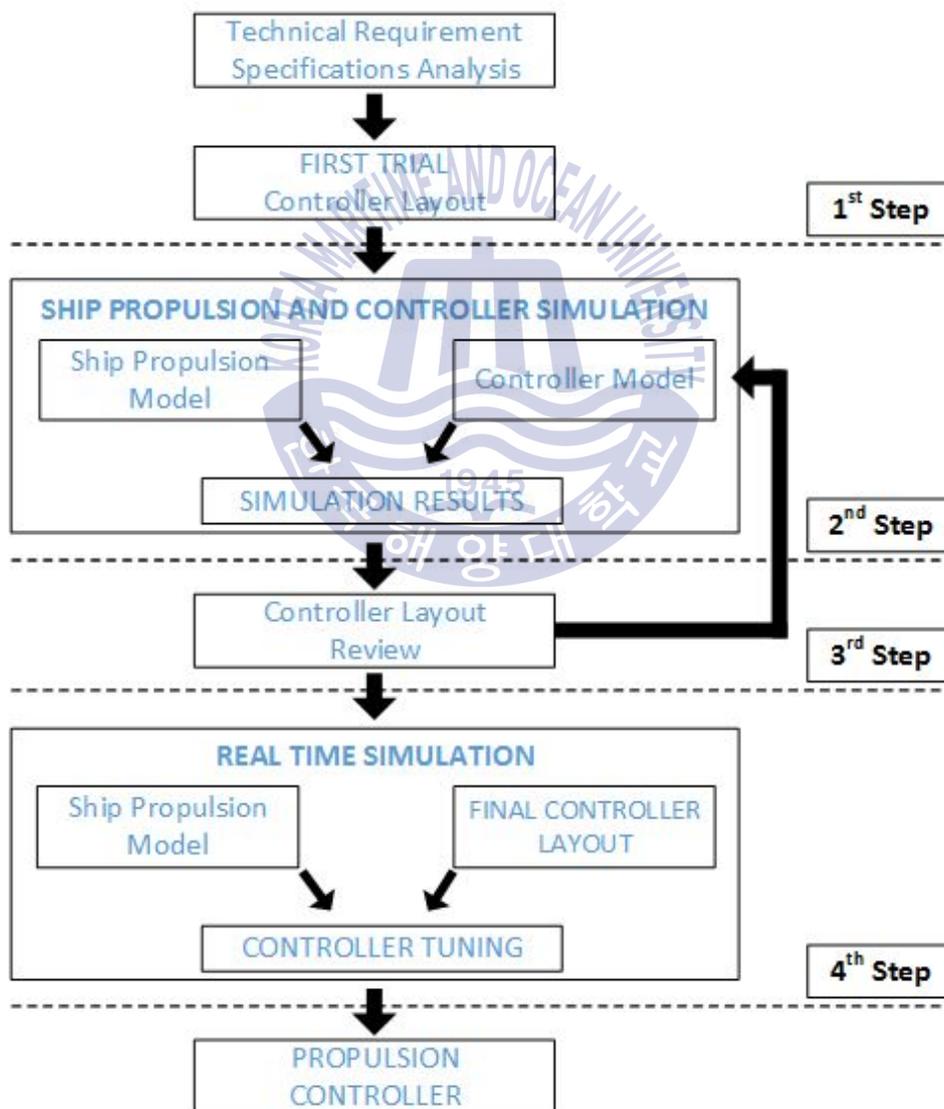


Fig. 5 Design procedure for propulsion controller

단계 1 : 조선소나 PSI 전문기관이 가지고 있는 경험과 제어 시스템에 대한 관련 지식을 기반으로 소비자 요구사항 및 기술사양 분석하여 1차 시제품용 제어기 개발

단계 2 : 컴퓨터 시뮬레이션을 통한 ECS 제어 알고리즘 검증

단계 3 : 검증 결과를 1차 시제품용 ECS 제어기에 반영하여 향상된 2차 ECS 제어기 개발

단계 4 : 2차 ECS 제어기를 실시간 시뮬레이션하여 제어기 값 동조

위 ECS 개발 절차를 보면, 1차 시제품용 ECS 제어기를 설계하여 소프트웨어 시뮬레이션을 통해 검증 및 보완한다. 이후, 2차 ECS 제어기를 설계하여 실시간 시뮬레이션을 통해 검증 및 보완하여 최종 제품을 만든다.

이 과정에서 가장 중요한 역할을 하는 부분은 단계 4인 실시간 시뮬레이션으로 볼 수 있다. 실제 함정의 추진체계는 다양한 종류의 장비로 구성되며, 이와 관련된 신호는 약 3,000개 이상이 된다. ECS는 이러한 신호를 처리하면서 통합 제어 및 모니터링을 수행해야 함으로 신호를 송수신하는 하드웨어를 포함한 시스템 전반의 신뢰성이 낮을 경우 함정에 적용하기 매우 어렵다. 특히, 국내에서는 ECS 실용화 경험이 없기 때문에 하드웨어를 포함한 시스템의 신뢰도 확보를 위해서는 실시간 시뮬레이션이 매우 중요하다.

SEASTEMA와 Genoa 대학은 ECS의 신뢰성 확보를 위해 실시간 시뮬레이션용 시뮬레이터를 NI(National Instrument)사의 LabVIEW와 Mathworks사의 Matlab, Simulink를 이용하여 구축하였다.

Fig. 6은 시뮬레이터 구성 요소를 나타낸 것이다.

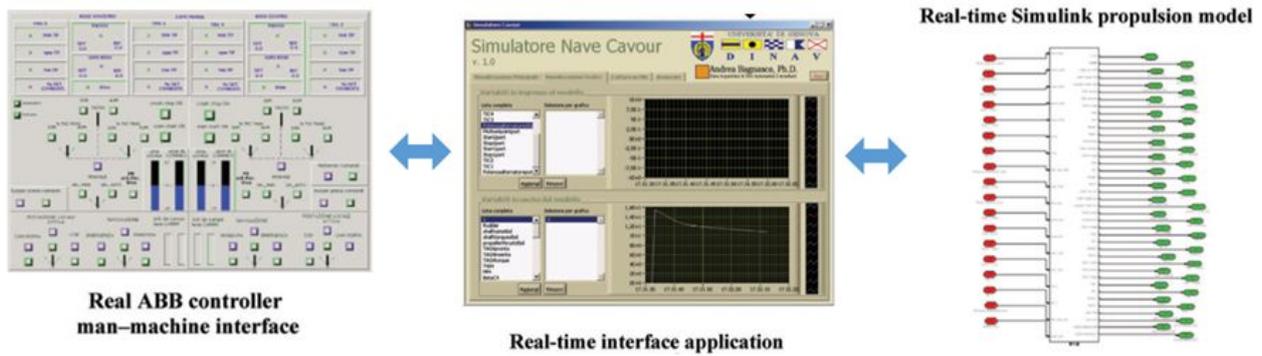


Fig. 6 Components for hardware in the loop simulation

Fig. 6을 보면, ABB에서 제작한 ECS 제어기와 Real-time simulink propulsion model (simulation model)를 실시간 인터페이스 응용프로그램(LabVIEW)을 이용하여 연결한 것을 skxksosek. 이 내용을 기반으로 시뮬레이터의 구조를 재구성해보면, ECS 검증을 위한 추진체계를 Simulink를 이용하여 구성하고, LabVIEW를 이용하여 시뮬레이션의 환경 제어, 데이터 저장, 상태 모니터링 등을 수행한 것으로 보인다.

참고논문 [6]에서는 LabVIEW의 SIT(Simulation Interface Toolkit) 모듈을 이용하여 추진체계 모델과 제어기간의 신호를 처리했는데(2009년 당시), 현재 이 모듈은 MIT(Model Interface Toolkit)으로 변경되어 NI사의 VeriStand 기능에 통합되어 있다.

### 1.2.2 함정 추진체계용 시뮬레이터 모델링

ECS의 제어 알고리즘에 대한 검증은 함정 추진체계에 대한 자료를 기반으로 개발된 모델을 이용하여 검증되어야 한다. 그러나, 함정의 추진체계에 관련된 자료는 군 기밀사항으로 분류되어 상세한 자료를 확보하기 어려우며, 일반적으로 시뮬레이션에 사용되는 열역학, 동역학적 모델은 많은 시간이 소모되기 때문에 실시간 시뮬레이션에 적용하기 어렵다는 문제점이 있다.

참고논문 [6]은 이와 같은 문제를 해결하기 위해 함정의 추진체계 모델을 단순화하는 방안을 선택하였다.

Fig. 7과 Fig. 8은 함정의 추진체계 동적 모델을 나타낸 것이다.

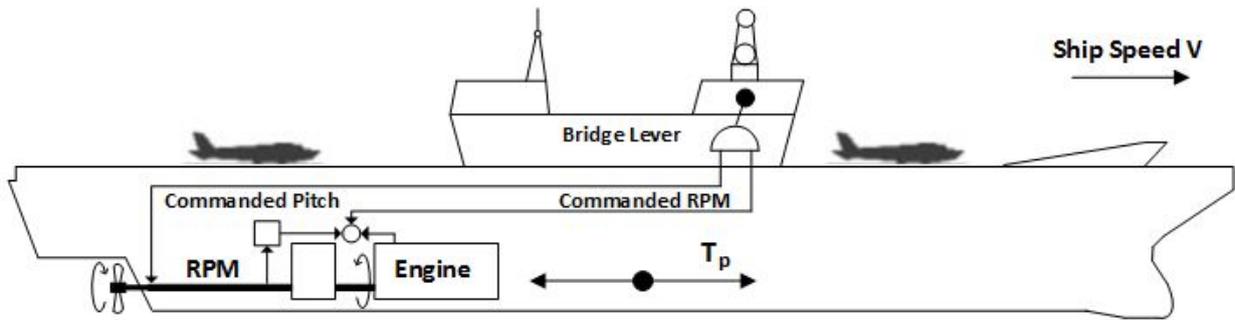


Fig. 7 Ship propulsion dynamics

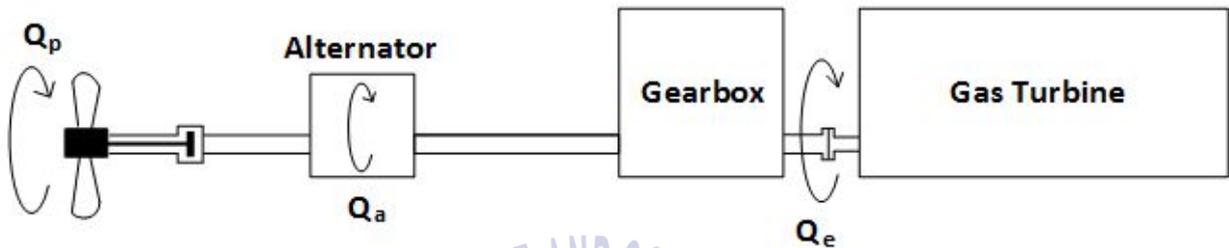


Fig. 8 Shaft line dynamics

$T_p$ 는 추력 [N],  $Q_e$ 는 엔진의 토크 [Nm],  $Q_p$ 는 프로펠러 토크 [Nm],  $Q_a$ 는 축에 설치된 발전기 토크 [Nm]를 나타낸다.

함정 추진체계용 시뮬레이터 관련 연구결과를 분석한 결과로부터 얻어진 핵심적인 요소를 정리하면 다음과 같다.

- 함정용 통합 추진체계인 ECS의 개발의 핵심은 ECS에 적용된 각 장비 운용 알고리즘이며, 이에 대한 신뢰성 확보가 매우 중요하다.
- ECS 기술을 국산화하기 위해서는 국내 업체에서 개발한 ECS를 함정에 실제로 적용하여 신뢰성에 대한 확보가 필요하다.
- 해외 ECS 개발 업체의 경우, ECS의 제어 알고리즘에 대한 신뢰성 확보를 위해 시뮬레이터를 개발하였으며, 이 시뮬레이터를 이용해서 검증을 수행한다.

- 이때 활용된 시뮬레이터는 동적 시뮬레이션을 포함한 실시간 시뮬레이션으로 구성된다.
- 실시간 시뮬레이터에 적용되는 모델은 실제 시간과 동일하게 시뮬레이션이 가능해야 함으로 연산시간이 적게 소요되는 모델을 적용해야 한다.

본 논문은 위에서 제시한 핵심 요소를 토대로 HILS(Hardware-In-the Loops) 기반의 함정 추진체계용 시뮬레이터를 개발하고, 시뮬레이터를 이용한 ECS 검증 방안을 제시한다.



### 1.3 연구 내용 및 구성

본 논문은 총 6개의 장으로 구성된다.

1장은 연구 배경과 국외 연구 개발 사례를 기술한다. 이를 위해 국외 함정 추진체계의 발전 경향과 국내 함정 기술 현황에 대해 조사하며, 국외 ECS 개발 전문 업체의 ECS 개발 사례 분석을 통해 ECS 개발에 필요한 핵심 요소를 정리한다.

2장은 시뮬레이터 개발을 위해 가상 함정의 추진체계의 개략적인 성능과 입출력 신호, 주요 장비에 대한 제어 알고리즘을 정의한다. 2장에서 정의한 가상 함정의 추진체계는 3장, 4장에서 개발되는 시뮬레이터의 기반 자료로 활용된다.

3장은 ECS의 핵심적인 추진체계 제어 알고리즘을 분석하여 7개의 검증 항목을 정의하고, 이에 따른 ECS 제어 알고리즘을 구체화한다. 구체화된 ECS 제어 알고리즘은 5장의 시뮬레이터를 이용한 ECS 검증의 평가 항목으로 활용된다.

4장은 함정 추진체계용 시뮬레이터의 하드웨어와 소프트웨어 설계에 대한 수행 과정을 기술한다. 시뮬레이터의 하드웨어는 중앙 시뮬레이터, 연동장비 시뮬레이터, SCM으로 구분하며, 소프트웨어는 기본 모델, 제어기 모델, 기타 모델, 통신 모델로 구분하여 개발한다.

5장은 가상의 ECS를 개발하고, 개발된 ECS와 시뮬레이터를 연동하여 3장에서 정의한 7개의 검증 항목에 대해 시뮬레이션을 수행한다. 또한, 가상의 ECS와 시뮬레이터간 시뮬레이션 결과를 분석하여 기술한다.

6장은 1장부터 5장까지 수행한 연구 결과에 대해 요약하고, 추후 연구 진행 방향에 대하여 기술한다.

## 제 2 장 가상 함정에 대한 정의

### 2.1 개요

ECS를 검증하기 위한 함정 추진체계용 시뮬레이터를 만들기 위해서는 ECS의 대상인 함정에 대한 다양한 자료를 필요로 한다. 대표적으로 연동장비와 ECS간의 신호를 정의한 IA(Interface Agreement), 각 연동장비별 운용 매뉴얼, 제어 알고리즘, 기타 함정 추진체계 운용을 위한 펌프, 전동기 등 다양한 장비들에 대한 식별과 운용 알고리즘에 대한 자료 등이다.

이러한 자료들은 일반적으로 함정 상세설계에서 ECS 제작사 및 각 연동장비 제작사, PSI 전문기관, 조선소 등이 모이는 연동회의(interface meeting)를 통해 결정되며, ECS 제작사는 이러한 자료를 바탕으로 함정 추진체계 운용을 위한 제어 알고리즘과 신호처리 체계 등을 개발하게 된다.

이때 만들어진 다양한 자료들은 함정의 작전성과 직접 또는 간접적으로 연결되어 있기 때문에 대부분의 자료는 군사보안으로 설정되어 있으며 확보가 매우 어려운 실정이다.

이러한 문제를 해결하기 위해 본 연구에서는 실제로 존재하는 함정들의 추진체계를 분석하여 본 연구에 적합한 함정 추진체계를 선정하고, 각 연동장비의 성능 및 연결, 운용 알고리즘을 정의함으로써 ECS 검증을 위한 함정 추진체계용 시뮬레이터 개발의 기본 자료로 활용하고자 한다.

## 2.2 국외 함정 추진체계 분석

전동기, 배터리 등의 전력 전자 관련기술의 발전에 따라 90년대 이후 군함의 추진체계는 과거 기계식 추진체계에서 기계·전기식이 함께 사용되는 복합형 추진체계로 진화하고 있다. 특히 영국의 Type 23, 프랑스와 이탈리아의 FREMM의 경우 가스터빈과 추진전동기를 복합한 추진체계를 사용하고 있다.

이와 같은 복합식 추진체계는 저속에서 추진전동기를 이용하여 운용함으로써 연료 효율을 높일 수 있으며, 기계식 추진체계를 사용하는 경우에 비해 수중 방사소음이 적기 때문에 대잠전 등 다양한 임무를 수행할 수 있다.

Fig. 9는 전 세계 함정 추진체계의 비율을 나타낸 것이다.

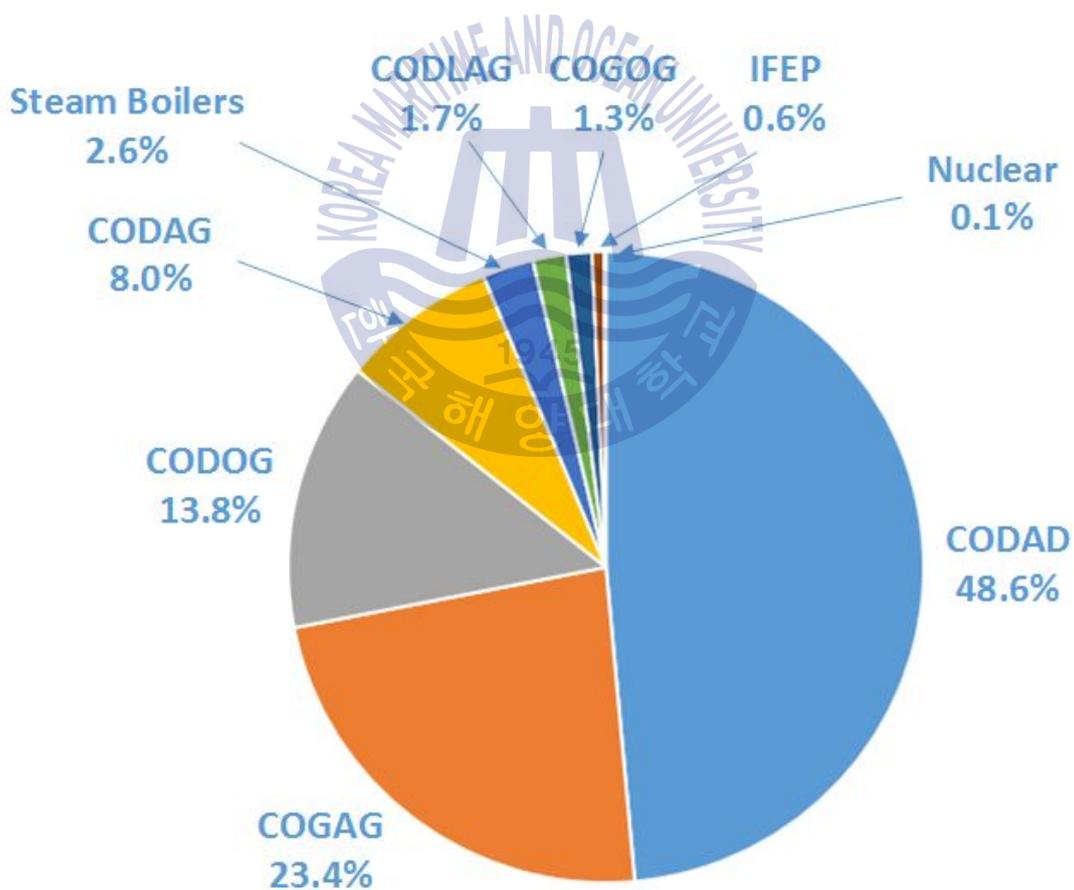


Fig. 9 Percentage of propulsion system for global warship

최근 전세계적으로 함정용 추진체계에 대한 연구 개발은 전동기를 이용한 추진시스템에 집중되고 있는 경향을 보이며, 가스터빈과 추진전동기를 복합한 추진체계에는 두 추진장비를 동시에 사용하는 CODLAG(COMBined Diesel-eLectric And Gas turbine)와 동시에 사용하지 않고 각각 따로 사용하는 CODLOG(COMBined Diesel-eLectric Or Gas turbine)가 있다.

CODLAG의 경우, CODLOG에 비해 최대 출력이 크다는 장점이 있지만, 출력 부하 분담이 어렵고, 실질적으로 얻는 최고 속도의 상승이 적다는 문제점이 있기 때문에 최근 건조되고 있는 영국의 Type 26의 경우 CODLOG 추진체계로 개발되고 있다.

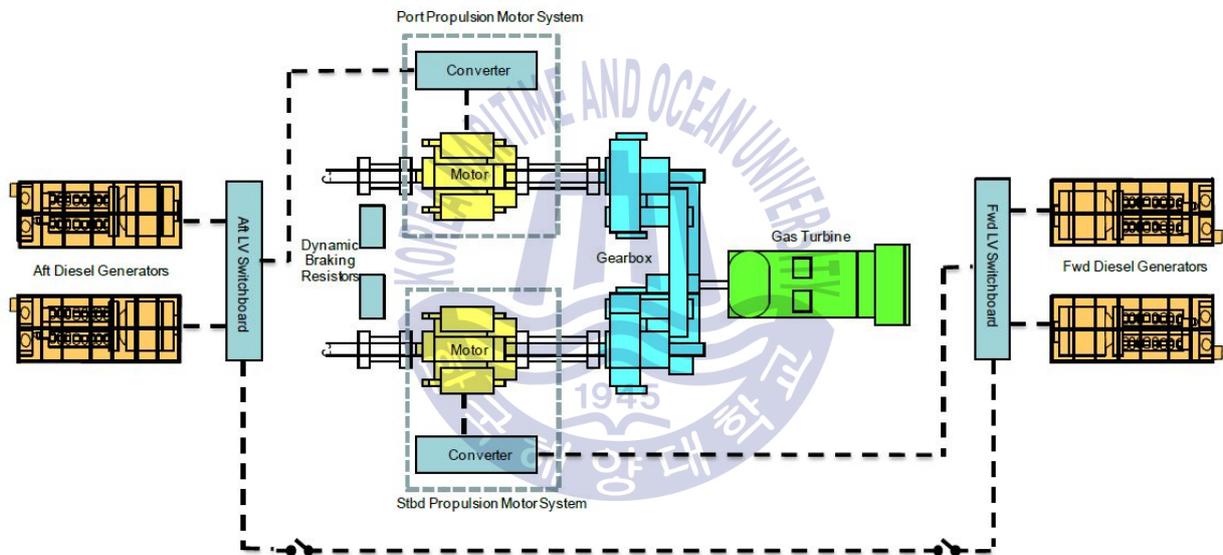


Fig. 10 Propulsion system for Type 26

Type 26의 추진체계는 가스터빈 1대와 추진전동기 2대, 축 2개로 구성된 3기 2축 추진체계를 채택하고 있으며, 추진전동기는 축에 바로 연결하여 구동할 수 있도록 구성되어 있다.

Table 1은 Type 26의 상세사양을 나타낸다.

Table 1 Specification of Type 26 ship

Parameter	Value
Length overall	148 [m]
Displacement	5,400 [tonnes]
Top speed	28 [knots]
Maximum electric speed	15 [knots]
Complement / crew	130
Operation range	7,000 [nm] at 15 [knots]
Configuration	CODLOG
Propeller	FPP

본 연구에서 정의하는 가상 함정의 추진체계는 Type 26의 추진체계를 기반으로 세부적인 내용을 정의하고자 한다.



### 2.3 가상 함정의 추진체계

본 연구에 적용되는 가상 함정은 세계적인 함정 기술개발 추세에 따라 가스 터빈과 추진전동기를 복합하여 운용하는 CODLOG 추진체계를 기반으로 구성한다. 본 함정의 추진체계는 가스터빈 1개와 감속기어 1개, 추진전동기 2개와 2개의 축, 고정 피치 프로펠러로 구성된 3기 2축의 구조를 가지고 있다.

Fig. 11은 가상 함정의 추진체계를 나타내고 있다.

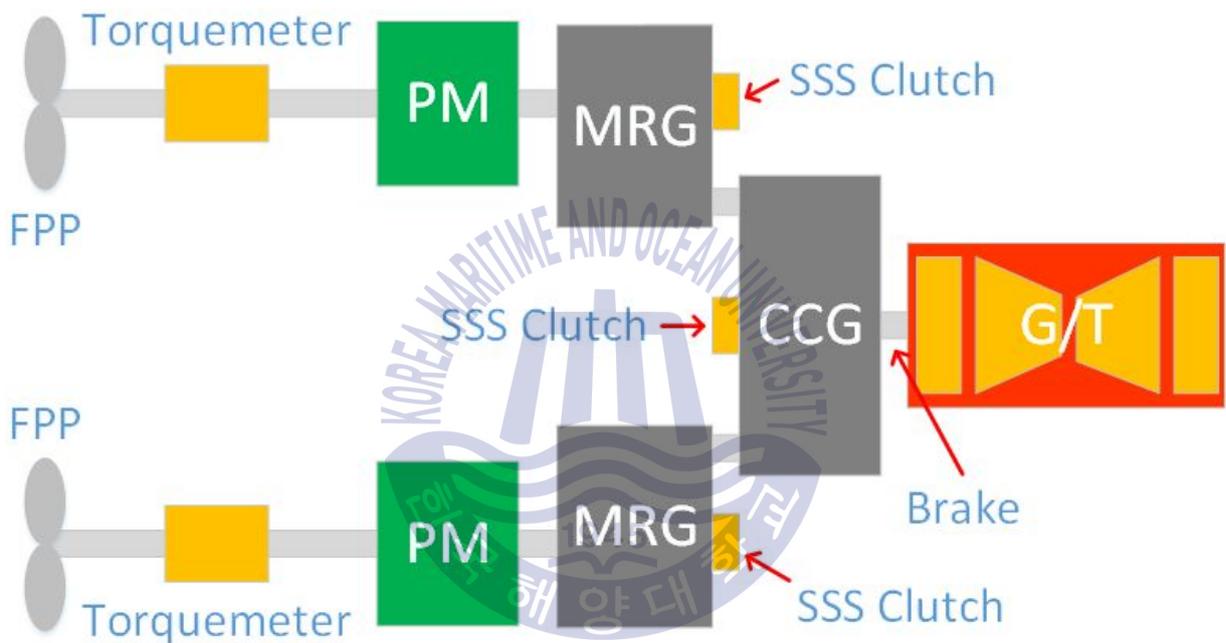


Fig. 11 Propulsion system for virtual warship

가스터빈의 추력을 전달하기 위해 감속기어에는 3개의 SSS 클러치와 브레이크가 설치되어 있으며, 축계에 토크미터가 설치되어 축계의 토크를 확인할 수 있도록 구성되어 있다.

### 2.3.1 가스터빈

가스터빈은 함정이 대출력을 필요로 할 때 사용하는 엔진으로서 Type 26 함정의 경우, Rolls Royce사의 MT30을 사용한다. 이 가스터빈은 최대 3600[rpm]에서 40 [MW]의 출력을 내며, 항공기 엔진을 기반으로 개발하였기 때문에 높은 신뢰성과 안정성을 가지고 있다[13].

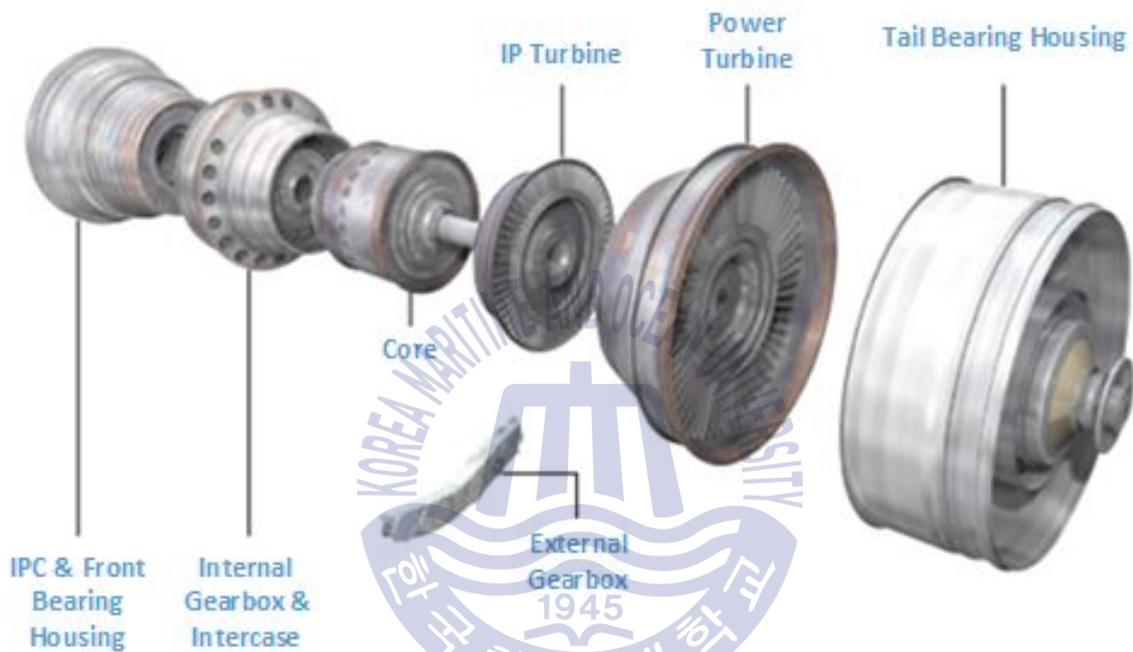


Fig. 12 MT30 component

Table 2 Specification of MT30

	Specification	Description
<b>Name</b>	MT3040	
<b>Structure</b>	2 Gas generator, 1 Power turbine	3 Shaft
<b>MCR(Maximum Continuous Rating)</b>	40 [MW] at 3,600 [rpm]	Idle : 900 [rpm]
<b>SFC(Specific Fuel Consumption)</b>	0.207 [kg/kWh] at MCR	
<b>Starting</b>	Electric motor	

가스터빈에는 가스터빈을 제어하기 위한 제어기(EMS, Engine Management System)가 설치되어 있으며, ECS는 가스터빈을 제어 및 모니터링하기 위해 EMS와 다양한 신호를 주고 받는다. 실제 가스터빈의 경우, 약 500개 이상의 신호를 실시간으로 처리하지만, 본 연구에서는 시뮬레이터 제작을 위해 가스터빈의 주요 제어기 신호에 대해서만 정의한다.

가스터빈과 가스터빈 구동용 기타 보기의 I/O는 Table 3, Table 4로 정의한다.

Table 3 I/O list of MT30

Name	Type	Range
Power demand [%]	AI	0~100
Normal start	DI	0=stop, 1=start
Normal stop	DI	0=start, 1=stop
Emergency start	DI	0=normal, 1=emergency
Turbine speed [rpm]	AO	0~3,600
GT torque [kNm]	AO	0~10,000
GT power [kW]	AO	0~20,000
Ready for start	DO	0=not ready, 1=ready
Idle warm-up	DO	0=normal, 1=warm-up
Start warm-up timer	AO	
Stop warm-up timer	AO	

\* AI : Analog Input, DI : Digital Input, AO : Analog Output, DO : Digital Output

Table 4 I/O list of MT30 auxiliary machinery

Name	Type	Range
Fan start	DI	0=off, 1=on
Damper open	DI	0=close, 1=open
LO pump on	DI	0=off, 1=on

### 2.3.2 감속기어

감속기어는 가스터빈의 출력을 축계에 전달하는 역할을 수행한다. 가상함정은 3기 2축의 CODLOG 추진체계를 적용하고 있으므로 본 추진체계에 사용하는 감속기어는 입력축 1개, 출력축 2개의 구조를 가진다. 또한, 가스터빈에 연결되는 CCG(Cross Connect Gear)와 각각의 축계와 연결되는 MRG(Main Reduction Gear)로 구분되며, CCG와 가스터빈을 연결하기 위한 No.1 SSS 클러치, MRG와 축계를 연결하기 위한 No.2, No.3 SSS 클러치가 각각 연결되어 있다.

Table 5 Specification of reduction gear

	Specification	Description
Type	CODLOG, Cross Connect Gear	1 GT input 2 Shaft output
Efficiency	GT mode : 98% at MCR PM mode : 99%	
Gear ratio	18 : 1	GT : 3,600 [rpm] Shaft : 200 [rpm]
Rotation	Input : clockwise Output : counter-clockwise	
Clutch	SSS Clutch, 3sets	1 GT input 2 Shaft output

감속기어에는 세이프티, 인터락, SSS 클러치, 브레이크 등의 장비가 설치되어 있으며, 이와 관련된 신호를 ECS, 가스터빈 및 추진전동기와 주고받는다. 특히, 감속기어의 윤활유 저압은 감속기어를 파손시키는 주 원인이 될 수 있기 때문에 윤활유 저압이 발생하는 경우, 가스터빈을 강제로 정지시키는 신호를 보낸다.

Table 6은 감속기어의 I/O를 정리한 것이다.

Table 6 I/O list of reduction gear

Name		Type	Range
Cross Connect Gear	Hydraulic brake system	DI	0=stop, 1=start
	Brake active	DI	0=stop, 1=start
	SSS clutch speed [rpm]	AO	0~3600
	SSS clutch engaged	DO	0=dis, 1=en
	Brake engaged	DO	0=dis, 1=en
	Brake ready	DO	0=not, 1=ready
	Gear ready	DO	RG ready
STBD Gear	SSS clutch lockout	DI	0=release, 1=lockout
	SSS clutch speed [rpm]	AO	0~200
	SSS clutch engaged	DO	0=dis, 1=en
	SSS clutch ratcheting	DO	Ratcheting
	SSS clutch locked out	DO	Lockout
	Gear ready	DO	RG ready
PORT Gear	SSS clutch lockout	DI	0=release, 1=lockout
	SSS clutch speed [rpm]	AO	0~200
	SSS clutch engaged	DO	0=dis, 1=en
	SSS clutch ratcheting	DO	Ratcheting
	SSS clutch locked out	DO	Lock out
	Gear ready	DO	RG ready

SSS 클러치는 입력축과 출력축의 속도 차에 의해서 스스로 연결되는 특징을 가지고 있는 장비이다. Fig. 13은 SSS 클러치의 연결 과정을 나타낸 것이다 [14][15].

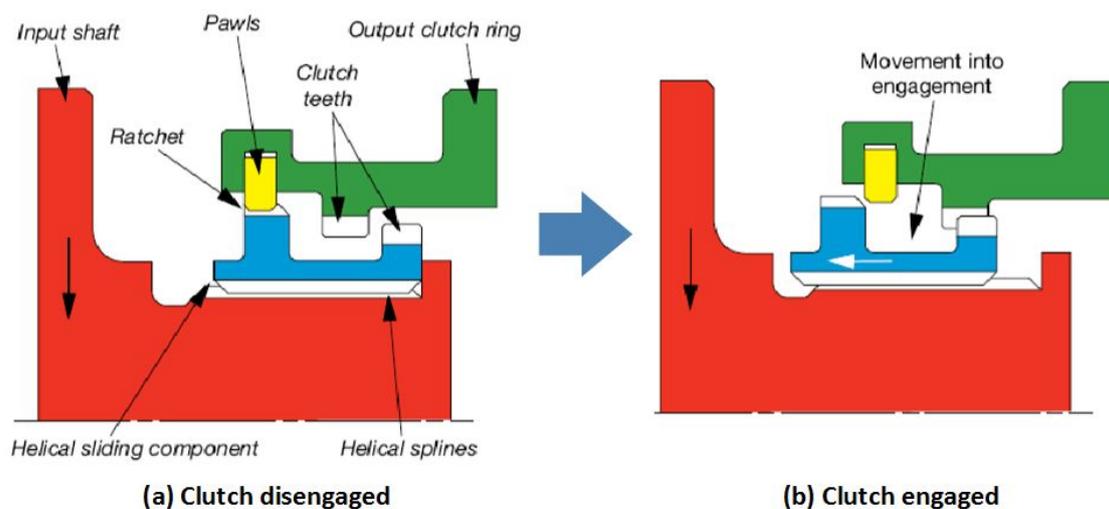


Fig. 13 Operation of SSS clutch

MRG에 설치된 SSS 클러치는 ECS의 명령에 따라 3가지 상태로 나뉘게 된다.

- lockout : 유압 시스템을 이용하여 입력측 라쳇트(input shaft ratchet)와 출력측 클러치 링 (output clutch ring)이 연결되지 못하게 하는 상태로 서보-제어 유닛(servo-control unit)이 슬라이딩 컴포넌트(sliding component)를 당겨서 입력측 라쳇트와 출력측 클러치 링에 있는 폴(pawl)이 서로 만나지 못하게 함으로써 속도차가 발생하더라도 입력측과 출력측이 연결되지 않도록 하는 상태
- ratcheting : 서보-제어 유닛이 슬라이딩 컴포넌트를 밀어서 입력측 라쳇트와 출력측 클러치 링의 폴이 서로 만나 있는 상태로, 입력측 라쳇트와 출력측 클러치 링의 속도차가 발생하면 슬라이딩 컴포넌트가 이동하여 입력측과 출력측이 연결될 수 있는 상태
- engaged : 슬라이딩 컴포넌트가 이동하여 입력측과 출력측이 연결된 상태

Fig. 14는 SSS 클러치의 lockout 장치를 나타내고 있다.



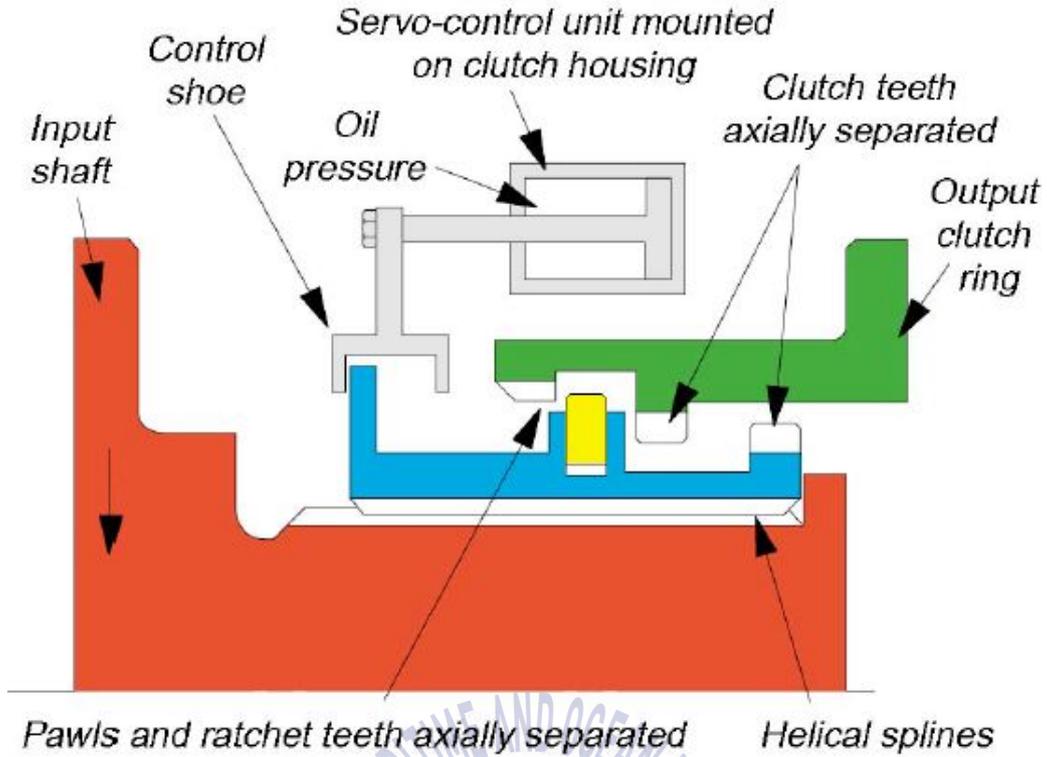


Fig. 14 SSS clutch servo-actuated lockout

### 2.3.3 추진전동기

추진전동기는 순항속력 이하인 저속(0~15 knots)에서 추진에 필요한 동력을 발생시키기 위하여 운용된다. 추진전동기는 감속기어 구동 없이 프로펠러에 추진 동력을 전달하며, 함정이 가스터빈 모드로 운용될 때 전동기와 컨버터의 전기적 연결을 끊어 축이 자유 회전하도록 한다[16].

Table 7, Table 8은 추진전동기의 사양과 I/O List를 나타낸다.

Table 7 Specification of propulsion motor

	Specification	Description
Type	Shaft mounted type	
Power	1.5 [MW]	
RPM	PM mode : 0 ~ 100 [rpm] GT mode : 0 ~ 200 [rpm]	

\* PM : Propulsion Motor, GT : Gas Turbine

**Table 8** I/O list of propulsion motor

Name		Type	Range
STBD	Motor speed [rpm]	AI	0~100
	Standby command	DI	
	Running command	DI	
	Idle command	DI	
	Shaft speed [rpm]	AO	0~200
	Shaft torque [kNm]	AO	0~1,000
	Shaft power [kW]	AO	0~1,500
	State standby	DO	
	State running	DO	
	State idle	DO	
PORT	Motor speed [rpm]	AI	0~100
	Standby command	DI	
	Running command	DI	
	Idle command	DI	
	Shaft speed [rpm]	AO	0~200
	Shaft torque [kNm]	AO	0~1,000
	Shaft power [kW]	AO	0~1,500
	State standby	DO	
	State running	DO	
	State idle	DO	

### 2.3.4 축계 및 프로펠러

축계 및 프로펠러는 감속기어와 추진전동기를 통해 전달되는 동력을 전달받아 함정의 추진력을 발생시킨다. 가상 함정의 프로펠러는 직경 4,500 [mm]로 5개의 날개를 가지고 있으며, 추진전동기 모드에서는 0~100 [rpm], 가스터빈 모드에서는 0~200 [rpm]으로 축당 최대 20 [MW]의 출력을 낼 수 있다.

Table 9와 10은 축계 및 프로펠러의 사양과 I/O List를 나타낸다.

**Table 9** Specification of shaftline & FPP

		Specification	Description
Propeller	Diameter	4.5 [m]	
	Number of blade	5	
Rotation		STBD : clockwise PORT : counter-clockwise	
RPM		PM mode : 0 ~ 100 [rpm] GT mode : 0 ~ 200 [rpm]	20 [MW]

**Table 10** I/O list of shaft & FPP

Name		Type	Range
STBD	Shaft torque [Nm]	AO	0~1,000
	Shaft RPM [rpm]	AO	0~200
	Shaft power [kW]	AO	0~20,000
PORT	Shaft torque [Nm]	AO	0~1,000
	Shaft RPM [rpm]	AO	0~200
	Shaft power [kW]	AO	0~20,000

## 제 3 장 시뮬레이터를 이용한 ECS 검증 방안

### 3.1 개요

함정 추진체계용 시뮬레이터는 개발하고자 하는 대상 함정의 ECS 검증을 위해 개발된다. 본 장에서는 ECS의 기능에 대해 간단히 설명하고, 검증하려는 ECS 기능과 검증 방법에 대해 제안한다.

### 3.2 ECS의 정의

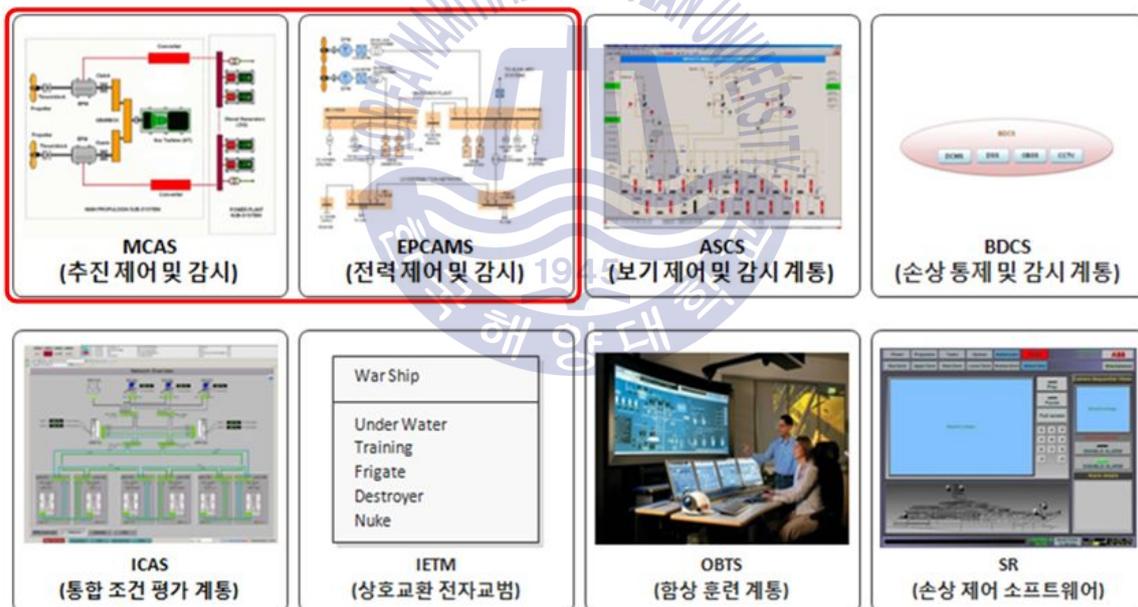


Fig. 15 Configuration of ECS

Fig. 15는 ECS의 구성을 나타낸다. 일반적으로 ECS는 함정 내부의 다양한 시스템에 대한 제어, 모니터링, 알람, 데이터 저장 등의 기능을 수행한다. 조향 (steering) 시스템을 제외한 대부분의 추진시스템은 ECS에 프로그래밍되어 있는 내부 제어 알고리즘에 따라 제어되며 그 결과가 모니터링된다.

ECS의 내부 시스템은 총 8개의 세부 시스템으로 구분되며, 추진제어 및 감시 시스템(MCAS : Machinery Control and Alarm System)과 전력제어 및 감시 시스템(EPCAMS : Electric Power Control and Management System)은 함정의 핵심 제어 시스템으로 다음과 같은 기능을 수행한다[17].

① MCAS : 추진계통의 기기와 보기 추진을 다루는 추진 제어 및 감시계통이다. MCAS는 기동, 정속, 예인, 해상 보급, 항해 등과 같은 추진 모드를 관리한다. 또한 엔진 시동/정지 절차관리, 추진 속도 관리, 축 오버토크와 기계의 오버로드로부터 보호, 추진부 체계 모니터링 및 통제, 연료 소비 통제 등을 수행한다.

추진 제어 및 감시의 범위는 가스터빈·보조 시스템, 가스터빈 연결을 위한 기어, 크로스 커넥터 감속기어, 2개의 추진전동기, 2개의 축계, 프로펠러, 연료펌프, 해수펌프, 오일펌프, 윤활유 펌프이다.

② EPCAMS : 전력제어 및 감시계통으로 전력 생산과 주요 분배 체계를 담당한다. 주로 전력 관리 기능과 발전기의 제어 및 감시기능을 수행한다.

전력 제어기능은 디젤발전기의 기동·동기화·병렬연결, 발전기 간 부하 공유 활성화·비활성화, 육상전원 연결 관리, 주요 설비·구성품·감시기기, 구동 중인 발전기의 정지 시 즉각적인 부하분산을 수행한다. 감시계통의 대상은 디젤발전기와 연계하여 가용한 물리적·전기적 신호의 매개변수와 경보 감시, 전압·전류·주파수·유효 전력·반응 전력 감시, 드라이브(주파수 변환기) 감시, 추진체계와 함내를 포함한 부하 등이다.

본 연구에서는 ECS의 기능 중 추진체계와 연관되어 있는 MCAS에 대한 검증이 가능한 시뮬레이터 개발을 목표로 한다.

### 3.3 ECS 검증 방안에 대한 검토

추진체계용 시뮬레이터는 ECS(MCAS)에 대한 검증을 목표로 한다. 따라서, 시뮬레이터 기능을 정의하기 위해서는 ECS의 검증 항목을 구체화하고, 검증 항목에 따라 필요한 시뮬레이터 기능을 정의해야 한다.

#### 3.3.1 ECS 주요 검증항목

ECS는 함내 설치된 추진장비를 함정 운용 목적에 맞도록 적절하게 제어 및 모니터링하는 역할을 수행한다. 과거 기계식 추진체계에서는 각 추진체계의 세부적인 제어를 중앙 제어기에서 모두 수행하였으나, 현대 함정은 대부분 자동 제어 시스템이 도입되어 세부적인 제어는 각 장비별 제어기에서 처리하고, on/off 등 전체적인 제어만 ECS에서 수행하고 있다. 따라서, ECS를 검증하기 위해서는 함정이 가지고 있는 특성에 따른 주요 운용 알고리즘에 대해 항목을 만들고 구체화하는 과정이 필요하다.

본 연구에서는 가상 함정의 가스터빈, 감속기어, 추진전동기, 축계 및 프로펠러에 적합한 ECS의 주요 운용 알고리즘을 임의로 정의하였다. 실제 ECS는 다양한 제어 알고리즘을 포함하고 있으나, 일반적으로 이에 대한 데이터는 확보하기가 매우 어렵기 때문에 본 연구에서는 다양한 논문을 바탕으로 핵심적인 알고리즘을 분석하여 7개의 항목을 도출하였다. Table 11은 ECS의 주요 시퀀스 알고리즘의 명칭을 나타낸 것이다.

Table 11 Sequential algorithm of ECS

No.	Name of sequential algorithm
1	GT start
2	GT stop
3	PM start
4	PM stop
5	GT to PM changeover
6	PM to GT changeover
7	GT emergency start

위 기능을 MCAS에서 제공하기 위해서는 MCAS와 추진체계간의 신호에 대해 정의되어야 한다. Fig. 16은 MCAS의 기능을 제공하기 위한 입력 신호와 출력 신호를 개략적으로 정리한 것이다.

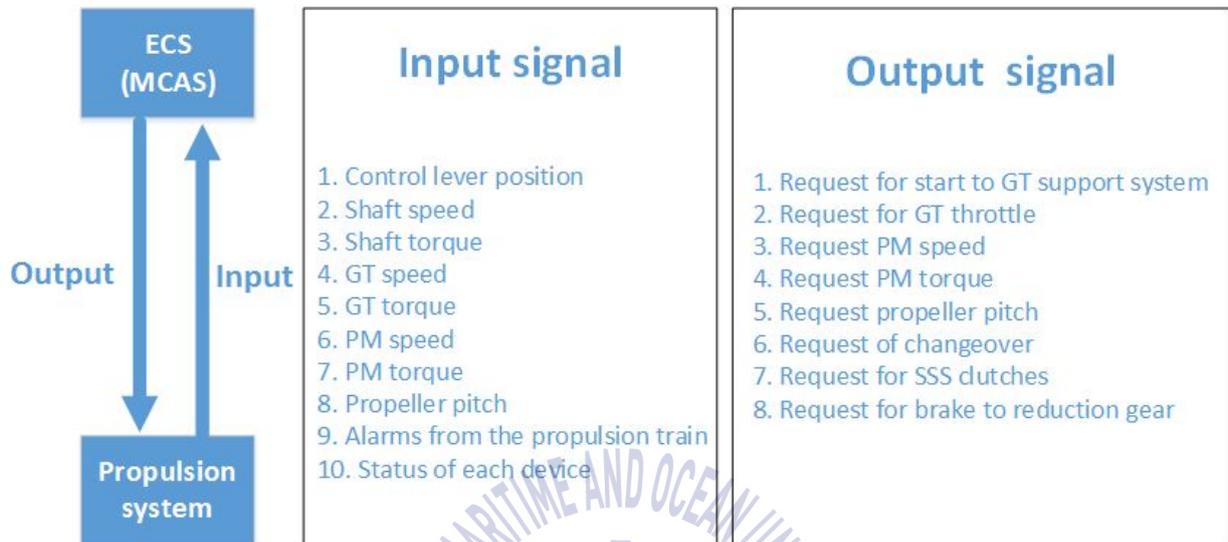


Fig. 16 ECS control & monitoring signal

### 3.3.2 주요 검증항목에 따른 알고리즘

3.3.1에서 정의한 검증항목은 총 7개로, ECS와 가스터빈, 감속기어, 추진전동기가 각 제어기의 제어 알고리즘에 따라 연동장비를 운용하고, 그 결과를 상호간 신호로 주고 받는다. 축계 및 프로펠러는 실제로 운용을 위한 유압펌프 등이 있다.

ECS를 개발하기 위해서는 함정 운용 목적에 따라 각각의 연동장비를 적절하게 운용하는 제어 알고리즘을 정확하게 개발해야 한다. 이러한 과정은 일반적으로 PSI 연동회의에서 각 장비 제작사들과 PSI 전문기관, 조선소, 그리고 ECS 제작사가 모여서 서로의 제어 알고리즘을 확인하고, 장비 운용 알고리즘과 상호간 인터페이스를 정의한다.

그러나, 본 연구의 목적은 ECS 개발이 아니라 ECS 개발을 위한 알고리즘 검증용 시뮬레이터 개발이다. 물론, 개별 장비의 제어 알고리즘을 모두 파악하여

시뮬레이터를 구성하는 것이 가장 효과적이지만, 실제로 PSI에 참석하는 업체나 ECS 개발 업체가 아닌 이상 각 장비 제작사로부터 제어 알고리즘을 확보하는 것은 어려운 실정이다.

따라서, 본 연구에서는 일반적으로 알려진 장비 운용 알고리즘과 다른 자료 [18][19]로부터 유추할 수 있는 장비 운용 알고리즘을 이용하여 3.3.1에서 정의한 7개의 검증항목에 필요한 가상의 장비 운용 절차와 그에 따른 신호를 정의한다.



## (1) GT start

GT start는 가스터빈 시동을 위한 장비 운용 절차를 나타낸 것이다.

가스터빈을 시동하는 경우 가스터빈 운용을 위한 여러 가지 보조 시스템이 정상동작을 하고 있어야 한다. 실제로 가스터빈 운용을 위한 보조 시스템은 상당히 많은 것으로 파악되나, 본 연구에서는 대표적으로 팬, 댐퍼, 윤활유 펌프에 대해서만 정상 동작하면 기동이 가능한 것으로 가정한다.

일반적인 가스터빈의 경우, 공회전(idle) 상태에서 각 기기의 온도 등의 상태를 정상상태까지 올리는데 일정 시간동안의 워업(warm-up)을 필요로 한다. 따라서, 시동관련 인터락이 해제되면 ECS는 5분간 가스터빈 워업을 수행하도록 한다. 이러한 가스터빈 워업은 비상 시동의 경우 생략할 수 있으나 가스터빈의 수명에 좋지 않은 영향을 미치기 때문에 정상적인 시동의 경우에는 일정 시간 동안 워업을 수행한다.

가스터빈 워업을 수행하는 동안 감속기어에서는 가스터빈의 출력이 감속기어로 전달되지 않도록 브레이크를 동작시킨다.

가스터빈 워업이 끝나고 브레이크가 해제되어 있으면 ECS는 가스터빈의 출력을 제어할 수 있다. ECS에서 출력을 높여서 감속기어가 회전하기 시작하면 MRG SSS 클러치가 연결되어 축계를 통해 프로펠러를 회전시키게 된다.

위 절차가 모두 끝나면 브레이크 시스템을 종료한다.

Fig. 17은 가스터빈을 이용한 함정 추진체계 운용 절차를 나타내고 있다.

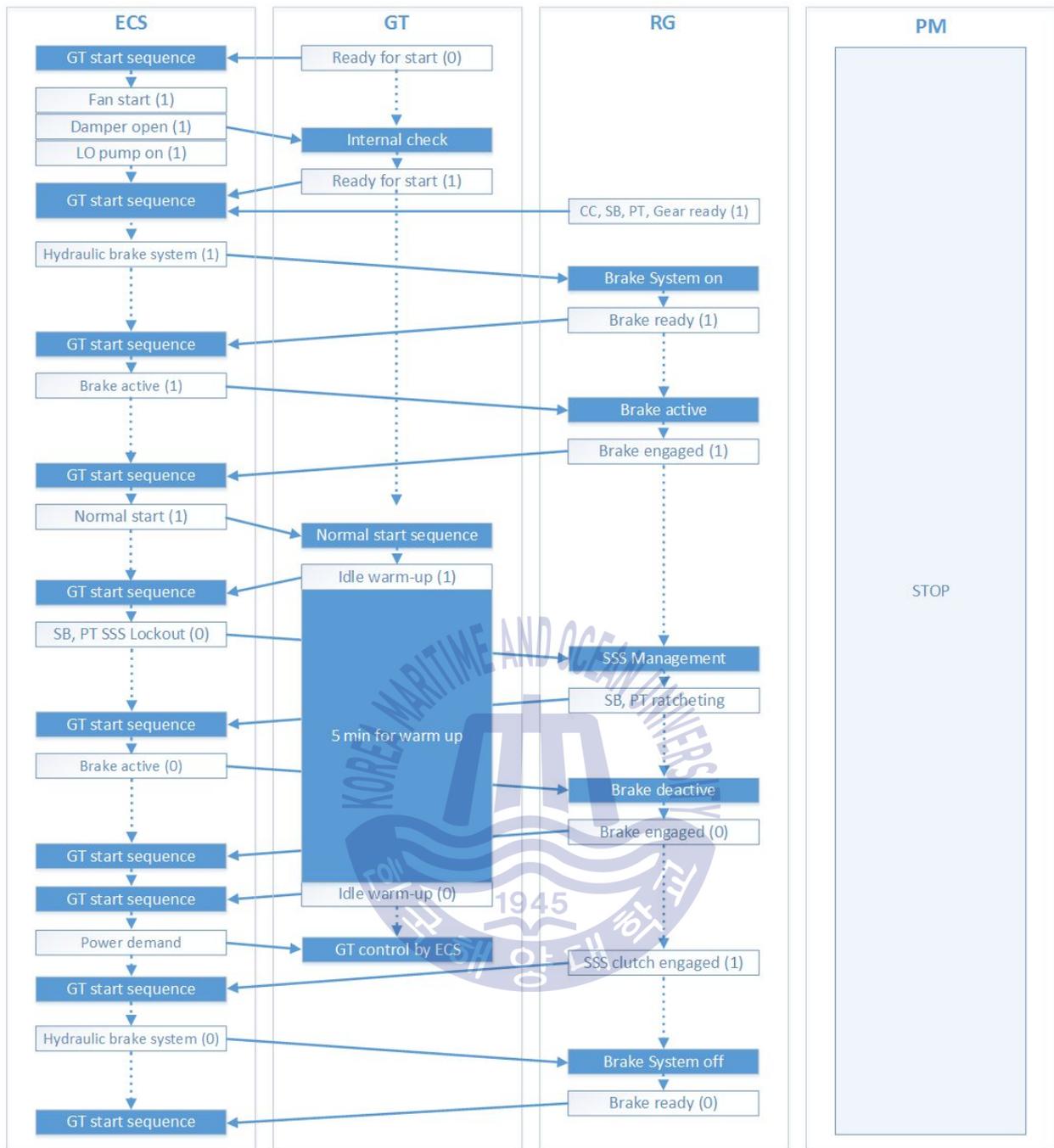


Fig. 17 Sequential control algorithm for GT start

GT start에서 추진전동기는 축계에 연결되어 강제 회전되며 이때 전력이 생산되면 가스터빈 입장에서 부하로 동작하기 때문에 주 배전라인과 절연이 되도록 구성되어야 한다.

(2) GT stop

GT stop은 GT start와 대부분 유사하다. ECS에서 가스터빈 출력 명령(GT power demand)을 0으로 주고 normal stop 신호를 가스터빈으로 보내면 가스터빈은 공회전 상태에서 5분간 냉각을 실시한다.

Fig. 18은 GT stop 절차를 나타내고 있다.

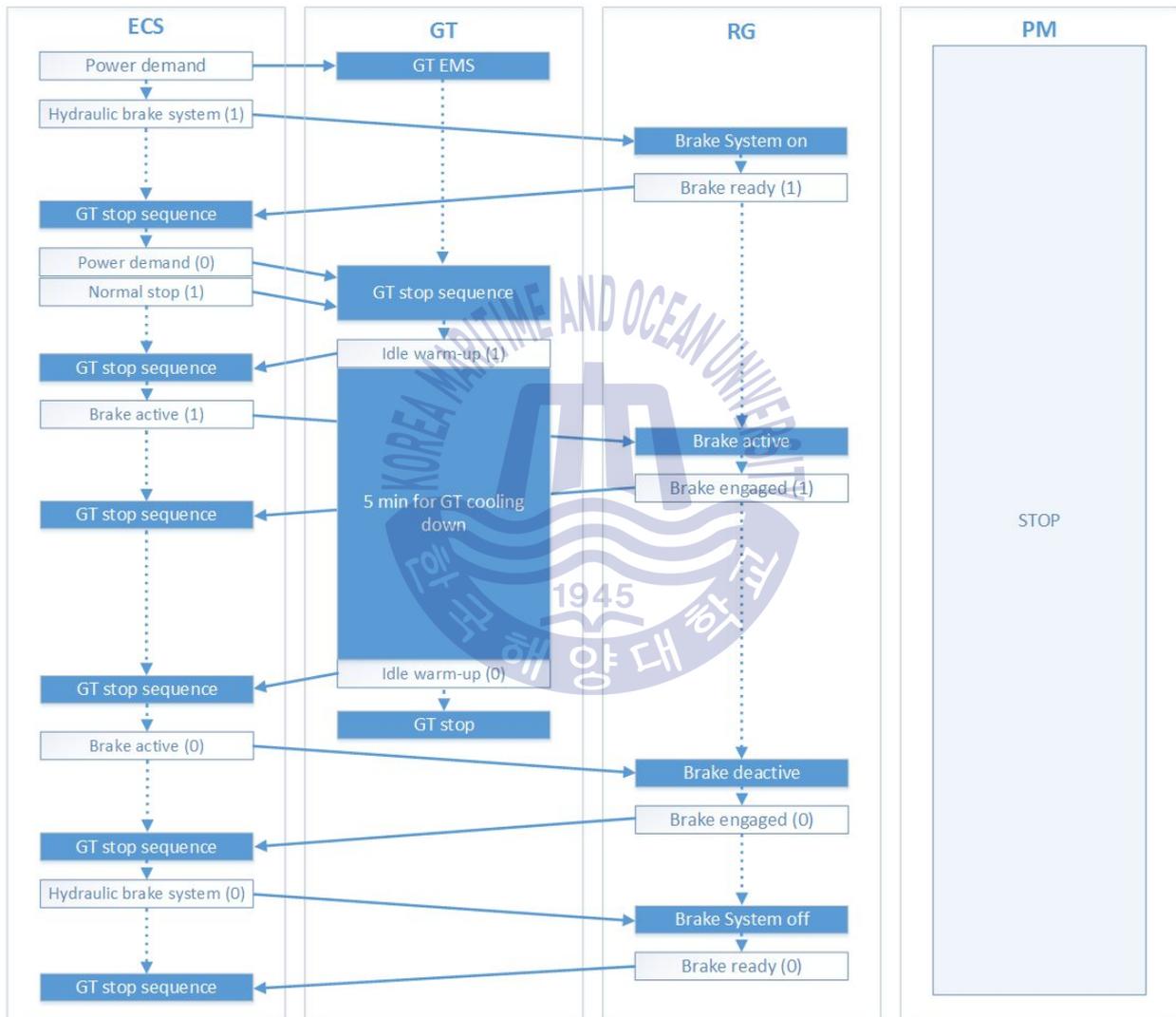


Fig. 18 Sequential control algorithm for GT stop

(3) PM start

추진전동기는 정지(idle), 준비(standby), 동작(running) 상태로 구분되며, ECS에서 운용하기 위해서는 MRG SSS 클러치를 축과 분리하여야 한다. MRG SSS 클러치가 연결되어 있는 경우, 추진전동기는 감속기어를 같이 회전시키기 때문에 에너지 낭비가 발생한다.

Fig. 19는 PM start의 절차를 나타내고 있다.

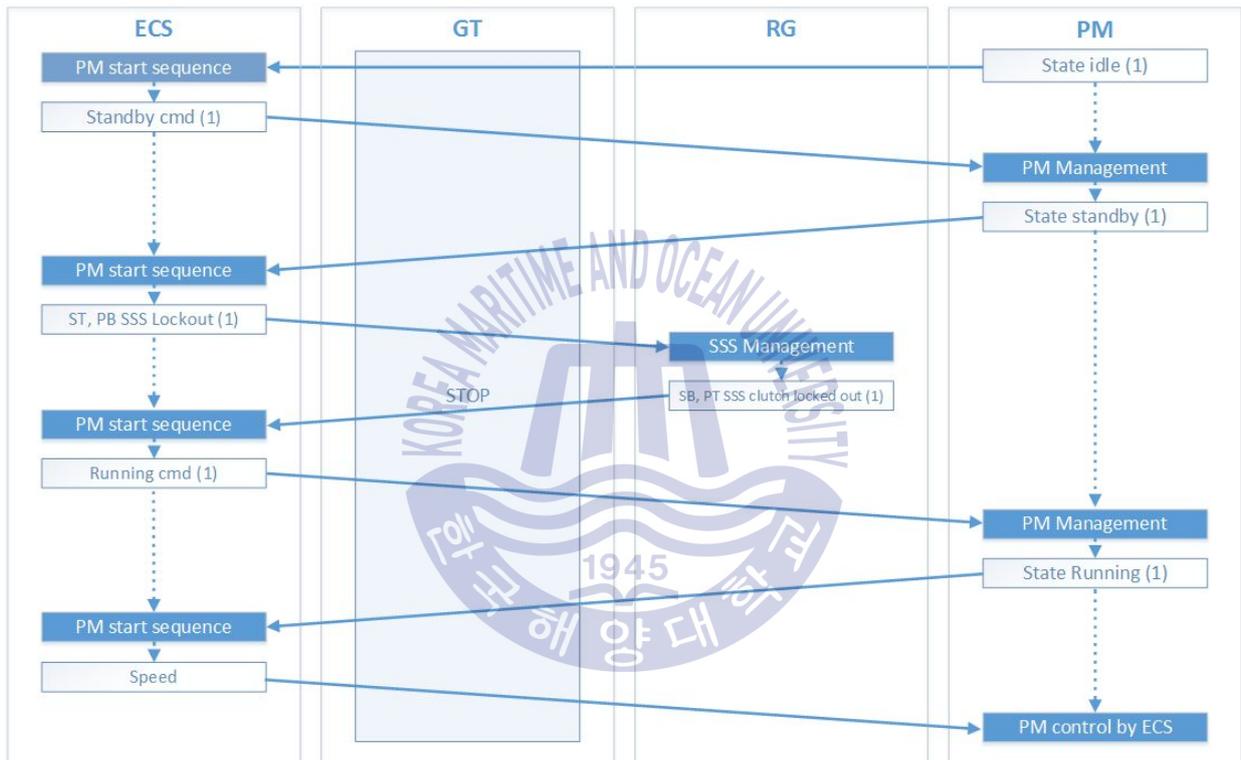


Fig. 19 Sequential control algorithm for PM start

(4) PM stop

PM stop은 속도값을 0으로 하고 준비 상태에서 정지 상태로 넘어가면서 추진 전동기가 정지하게 된다.

Fig. 20은 PM stop 절차를 나타내고 있다.

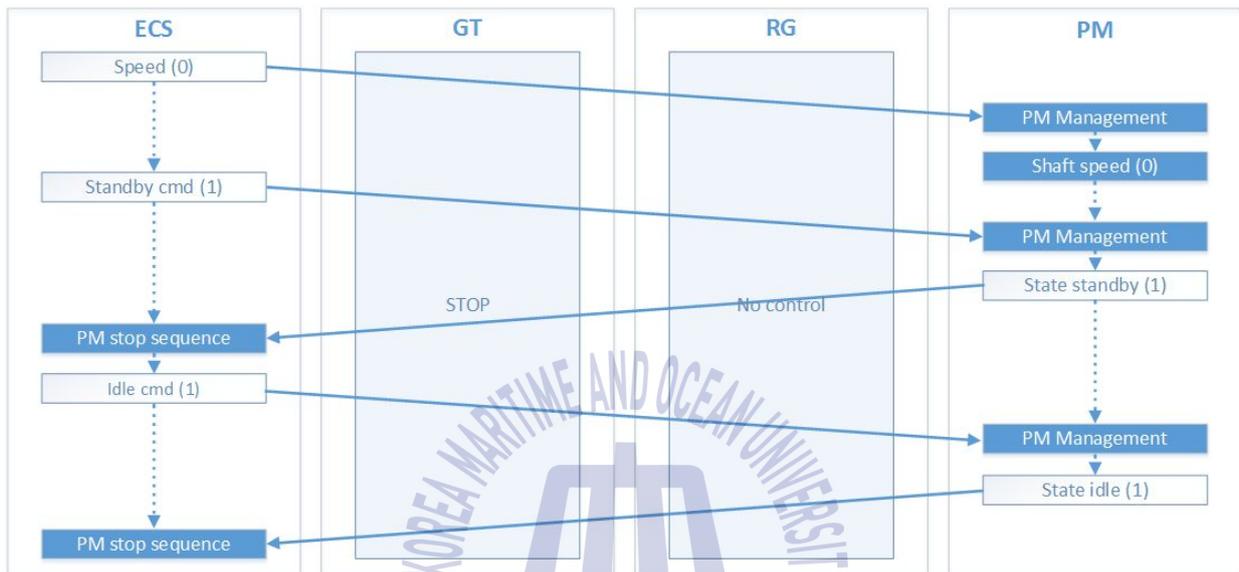


Fig. 20 Sequential control algorithm for PM stop

(5) GT와 PM간의 전환

복합식 추진체계인 CODLOG의 경우, 상황에 따라 가스터빈 모드에서 추진전동기 모드로, 추진전동기 모드에서 가스터빈 모드로 전환하는 경우가 발생한다.

GT to PM changeover는 가스터빈 모드에서 추진전동기 모드로, PM to GT changeover는 추진전동기 모드에서 가스터빈 모드로 전환하는 절차이다. 각 전환 절차 과정에서 ECS는 가스터빈과 감속기어, 추진전동기를 통합하여 제어 및 모니터링을 수행한다.

Fig. 21은 GT to PM changeover 절차를, Fig. 22는 PM to GT changeover 절차를 나타내고 있다.

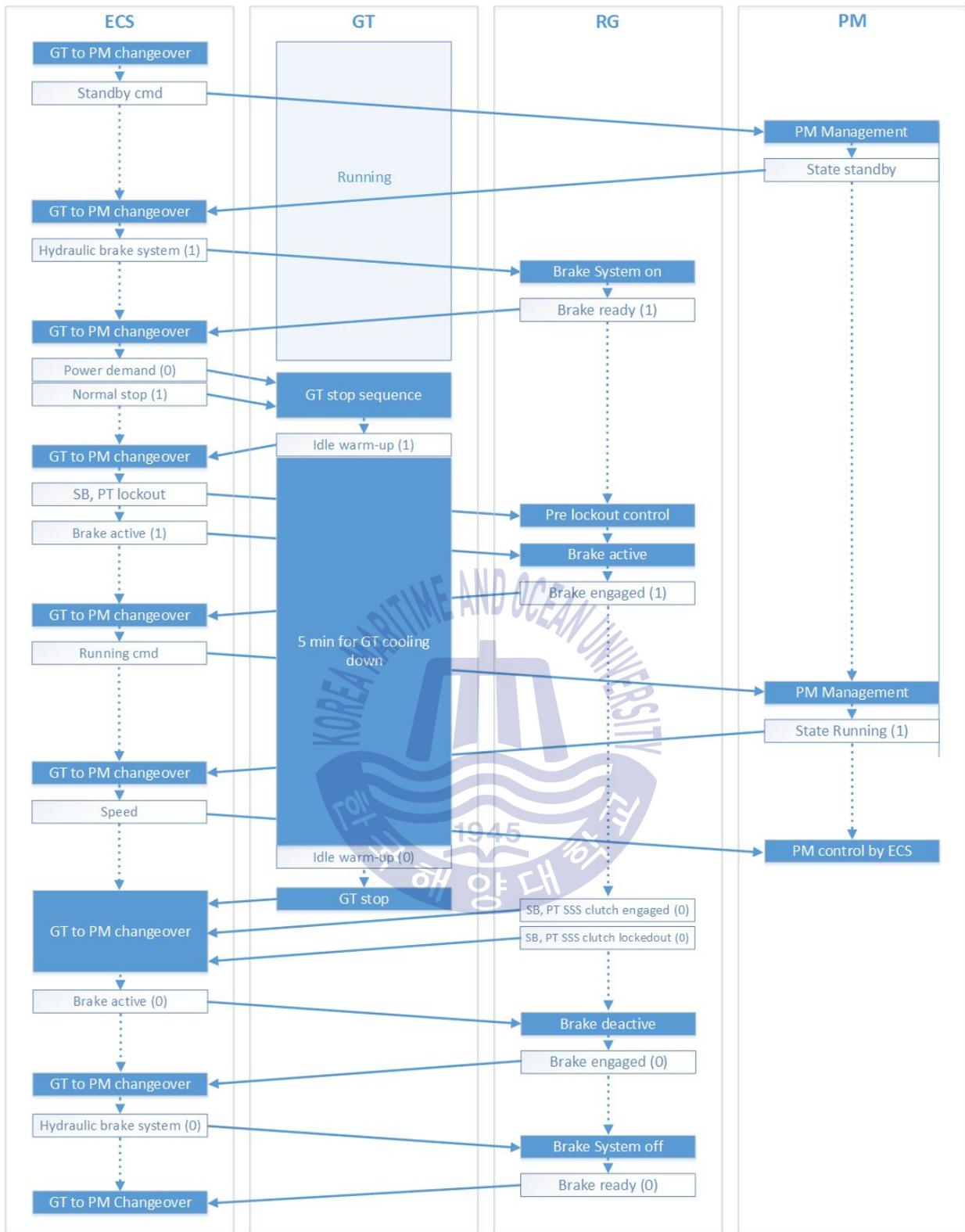


Fig. 21 Sequential control algorithm for GT to PM changeover

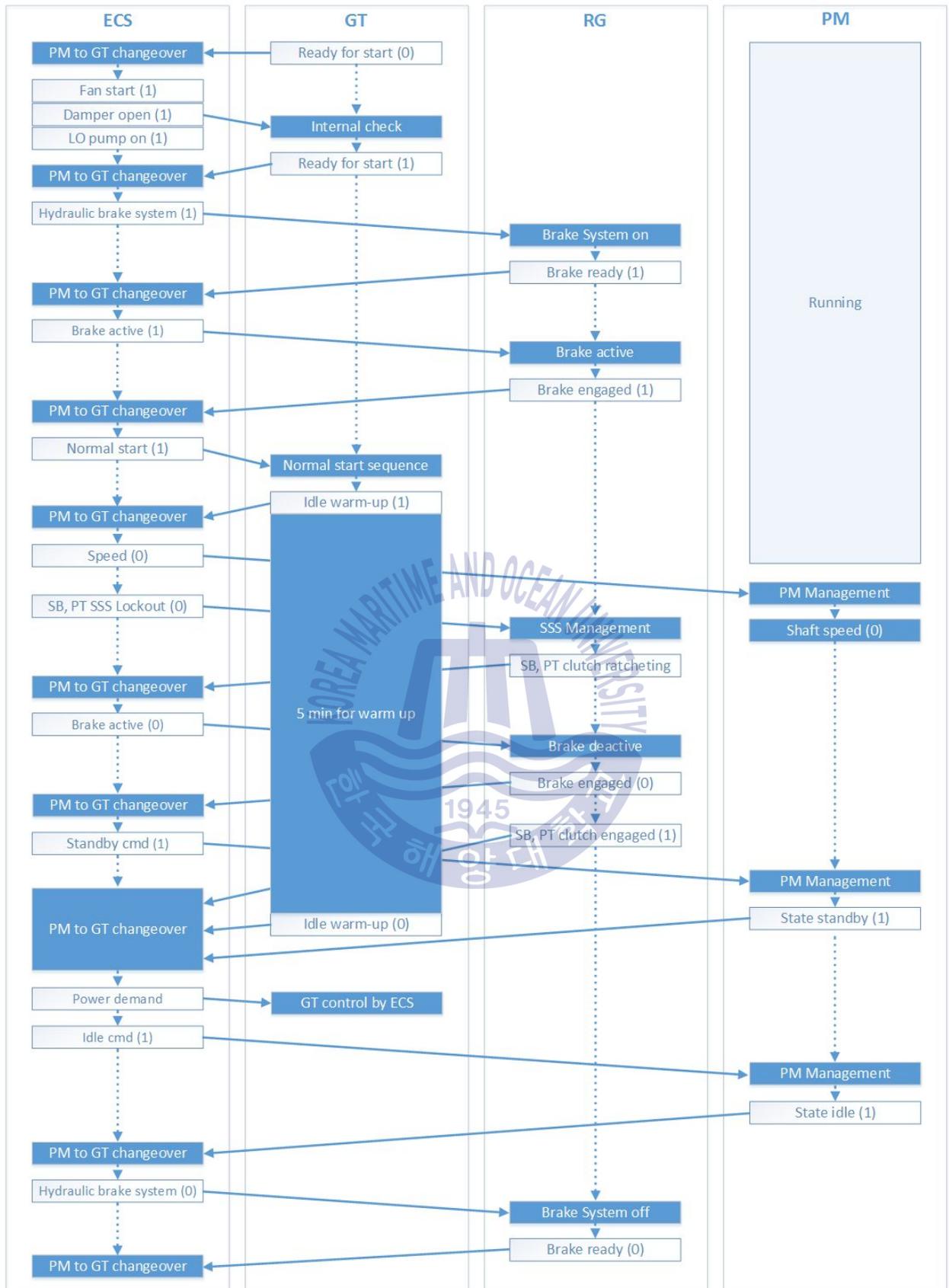


Fig. 22 PM to GT changeover sequential control algorithm

(6) GT emergency start

합정은 상황에 따라 정지상태에서 빠르게 기동해야 하는 경우가 발생할 수 있다. 합정에 설치된 가스터빈의 경우, 워업을 하지 않고 가스터빈의 출력을 바로 올림으로써 이와 같은 기능을 확보한다. 본 연구에서도 긴급 시동 기능을 추가하여 워업을 하지 않고 바로 출력을 올릴 수 있도록 하였다.

Fig. 23은 GT emergency start 절차를 나타내고 있다.

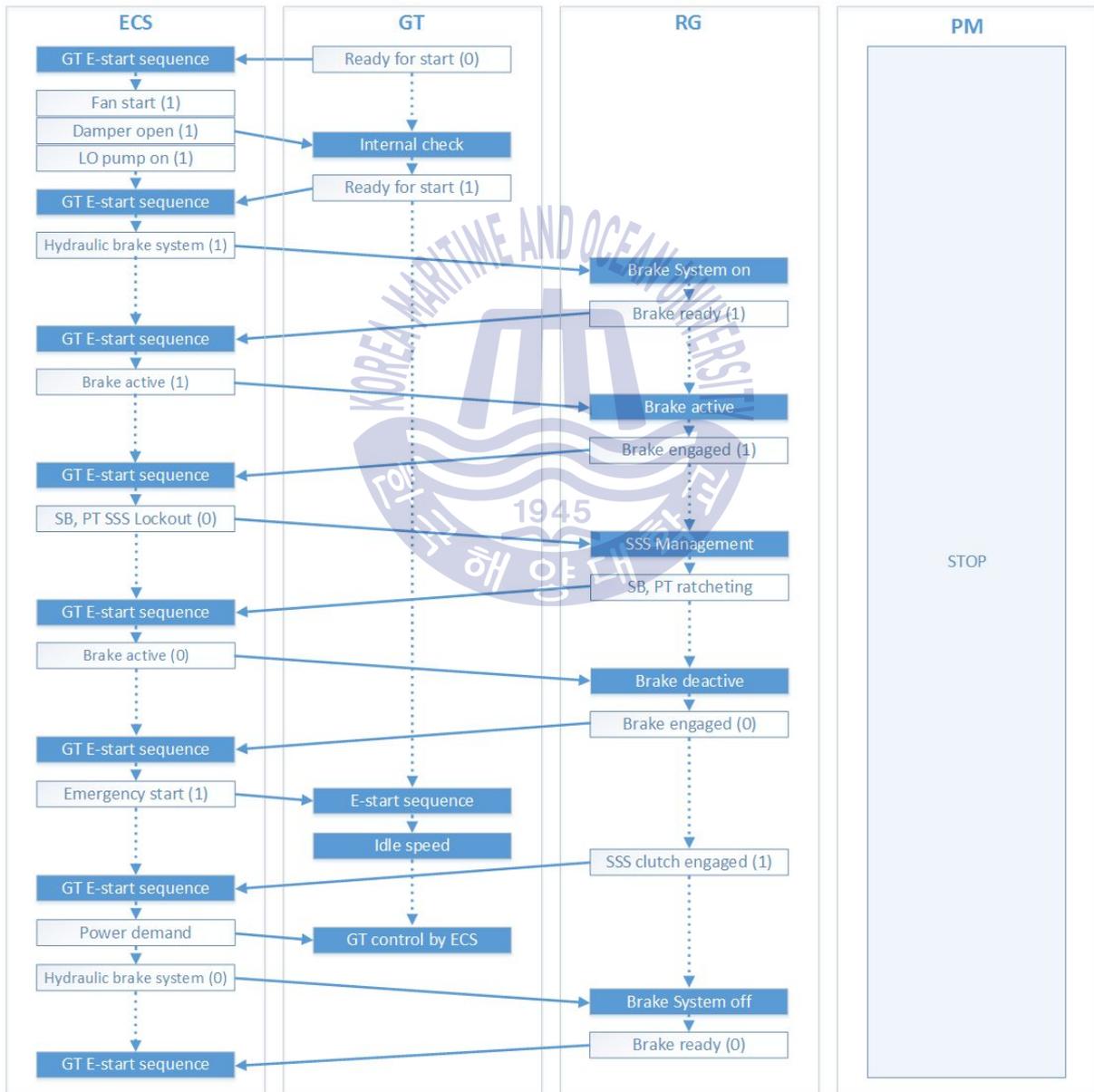


Fig. 23 Sequential control algorithm for GT emergency start

### 3.4 ECS 주요 검증항목에 따른 시뮬레이터 개념 설계

추진체계용 시뮬레이터는 3.3절의 7가지 제어 알고리즘에 대해 검증하는 것을 목표로 한다.

이를 위해 시뮬레이터 소프트웨어를 4개의 개체로 구분하고, 각각의 연동장비 모델의 기능을 간략히 정리하면 아래와 같다.

#### ① 가스터빈

- 주변 장비(팬, 댐퍼, 윤활유 펌프)가 모두 동작하면 ready for start 신호를 보냄
- 시동에는 정상 시동과 긴급 시동이 있으며, 선택에 따라 워업 여부가 결정됨
- 정상 정지 신호가 입력되면 5분간의 쿨링 다운 시간이 있음
- 필요에 따라 감속기어의 브레이크 시스템을 활용

#### ② 감속기어

- CCG SSS 클러치는 ECS에서 제어하지 않으며, MRG SSS 클러치는 필요에 따라 제어
- 가스터빈 시동과 정지시 가스터빈 운용을 위해 브레이크 시스템을 제어

#### ③ 추진전동기

- 추진전동기는 3가지 상태(정지, 준비, 동작)로 구분됨
- 추진전동기는 MRG SSS 클러치를 lockout 상태로 두고 ECS의 명령값에 따라 제어

## 제 4 장 함정 추진체계용 시뮬레이터 개발

### 4.1 시뮬레이터 개발 방향

본 장에서는 국외 함정 추진체계용 시뮬레이터 연구사례와 ECS 검증 방안에 따른 시뮬레이터 기능 정의를 기반으로 함정 추진체계용 시뮬레이터를 개발한다.

함정 추진체계용 시뮬레이터는 크게 두 가지 기능을 포함하는데, 하나는 함정 추진체계의 동특성에 대한 시뮬레이션이고, 다른 하나는 ECS의 운용 알고리즘 검증에 대한 시뮬레이션이다. 함정 추진체계용 시뮬레이터는 이 두 가지 기능을 실시간으로 구현하여 ECS를 검증한다.

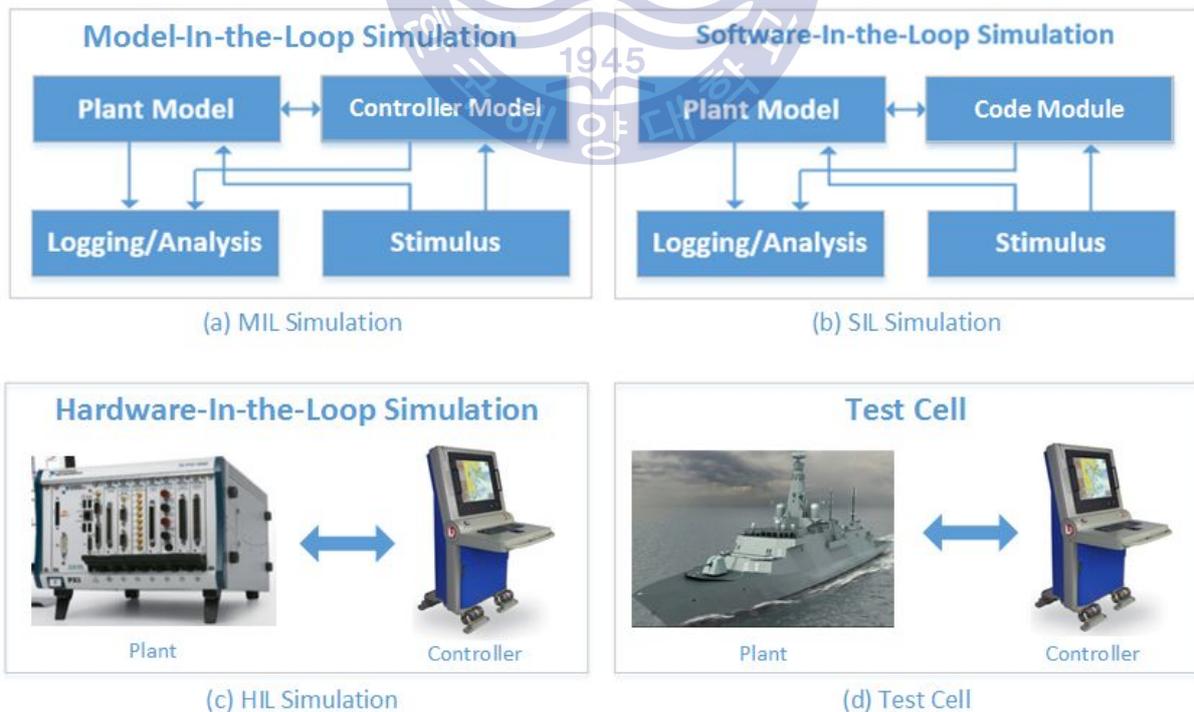


Fig. 24 Simulation method

일반적으로 제어기 개발에 있어서 시뮬레이션을 이용한 검증 방법은 Model In the Loop Simulation(MILS), Software In the Loop Simulation(SILS), Hardware In the Loop Simulation(HILS), Test cell로 구분되며, 각각의 시험 방법은 아래와 같다.

① MILS : 플랜트 모델과 제어기 모델을 개발하고 확인하거나, 구현 가능 여부를 테스트하는 방법으로 실제 프로그램 코드가 아닌 알고리즘 단계에서 검증한다.

② SILS : MILS를 통해 만들어진 제어기 모델은 프로그램 코드로 변환되고 각각의 기능 검증을 위해 단위별 모듈로 구분되어 테스트를 수행한다.

③ HILS : 하드웨어를 포함한 제어기 검증 방법으로 컴퓨터와 DAU 장비를 이용하여 실제 장비와 유사한 환경을 구축한 후, 제어기와 실시간으로 신호를 주고받으면서 제어기의 하드웨어와 소프트웨어에 대한 테스트를 수행한다.

④ Test cell : 최종 제품 장착전에 실제 플랜트의 하드웨어와 제어기를 연계하여 검증하는 단계로 실제와 동일한 환경에서 제어기 테스트를 수행한다.

Table 12는 검증 방법에 따른 제어기와 플랜트의 개별 구성 요소를 나타내고 Fig. 25는 제어기 검증 절차를 나타내고 있다.

**Table 12** Individual components of system

Name	MILS	SILS	HILS	Test cell
Controller	Simulation	Simulation	Real	Real
Plant	Simulation	Simulation	Simulation	Real

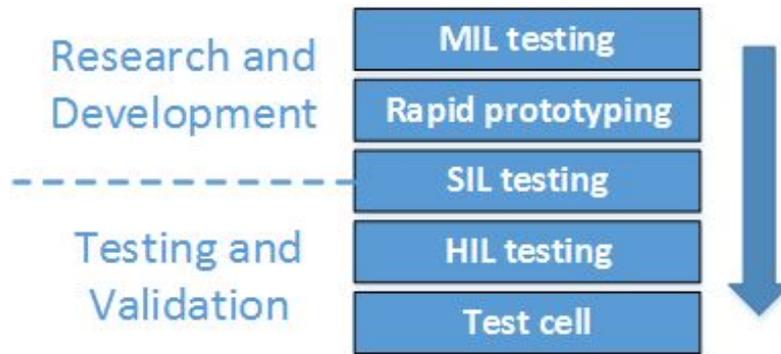


Fig. 25 Procedure of controller verification

MILS와 SILS는 컴퓨터 시뮬레이션을 이용하여 소프트웨어에 대해 검증하며, HILS는 하드웨어를 포함하여 검증함으로써 시스템 전체를 검증한다. Fig. 25의 검증 절차에 따라 제 2장에서 조사한 ECS 개발과정을 분류하면 SILS는 단계 2에 해당되고, HILS는 단계 4에 해당한다.

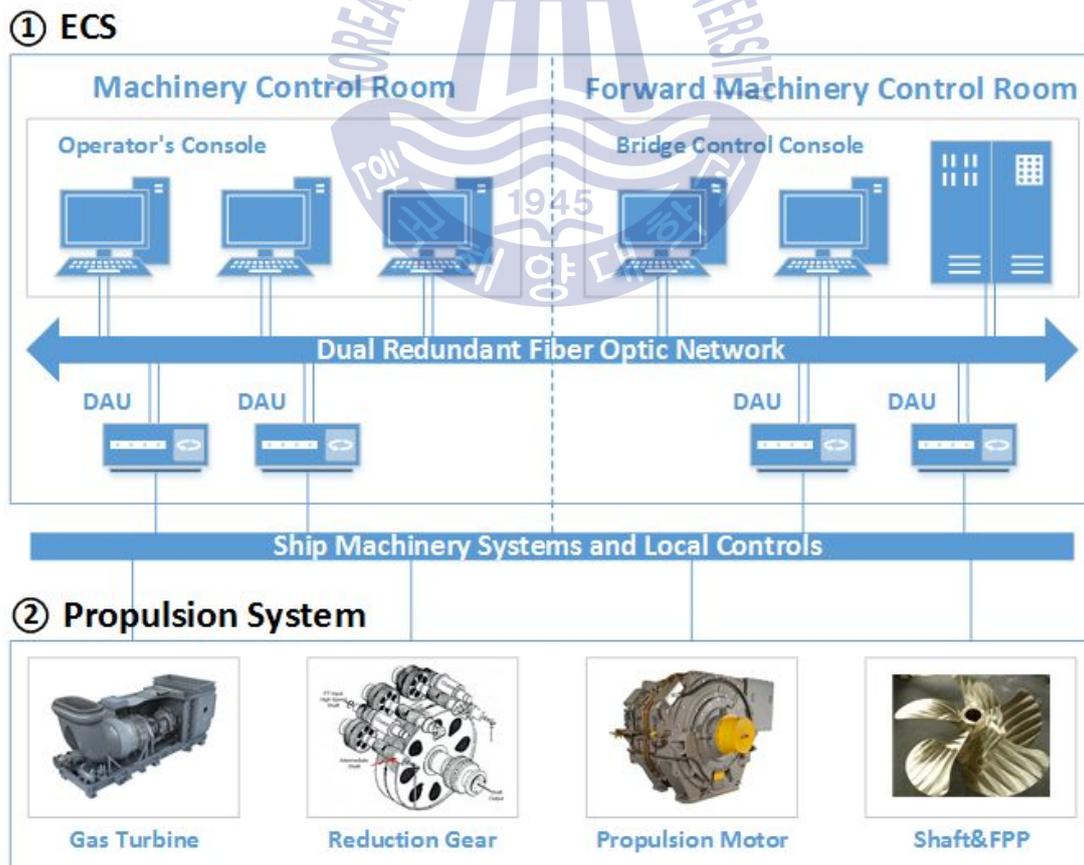


Fig. 26 Configuration of ECS for warship

ECS는 콘솔과 데이터 수집 장치(DAU, Data Acquisition Unit)로 구분되며, 콘솔과 데이터 수집 장치간의 내부 네트워크는 고장 등을 고려gkdu 이중 광통신 네트워크로 구성되어 있다. 통신방식은 일반적으로 TCP/IP를 이용하여 구축한다.

추진체계는 각각 제작사의 제어기에서 통신 또는 물리적 신호(hardwired signal)를 이용하여 ECS의 데이터 수집 장치와 데이터를 송수신하도록 구성된다.

HILS 시뮬레이터는 Fig. 26의 ‘② Propulsion System’에 따라 각 추진장비별 제어 알고리즘에 따라 만들어지는 신호와 동일한 신호를 실시간으로 구현해야 한다. 이러한 시뮬레이터를 구축하기 위해서는 RT 기반의 OS(Operating System)와 통신, 물리적 신호를 처리할 수 있는 데이터 수집 장치가 필요하다. 또한, 시뮬레이션의 초기값 설정 및 이벤트 발생, 시뮬레이션 동작 상태를 확인할 수 있는 시스템도 구축되어야 한다.

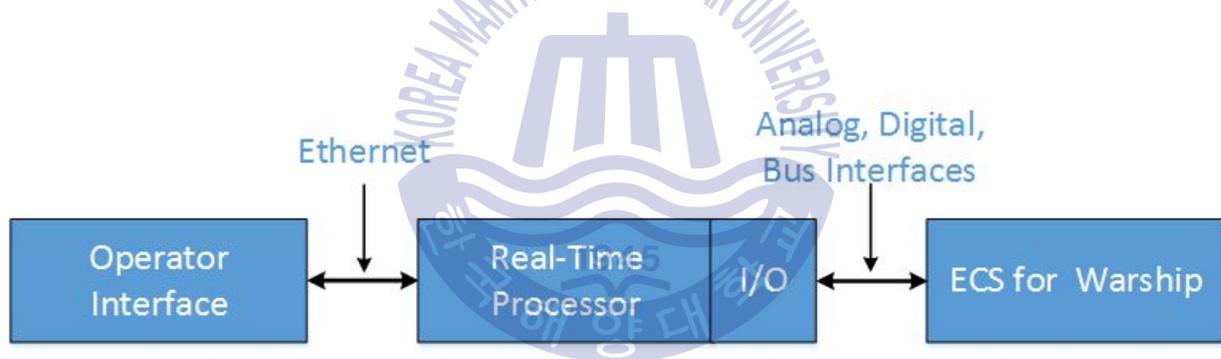


Fig. 27 Concept of HILS simulator

특히, 함정 추진체계는 각 추진장비 제작사별 통신 방식, 물리적 신호 등의 종류가 다양하고 개수가 매우 많기 때문에(2,000개 이상) 시뮬레이터를 구축할 때 이와 같은 점을 고려하여 하드웨어를 선정해야 한다.

본 연구에서는 연동장비의 모든 물리적 신호를 구현할 수 없기 때문에 아래와 같이 주요 신호를 선별하여 물리적 신호로 구축한다.

Table 13은 시뮬레이터의 물리적 신호를 정리한 것이다.

Table 13 Hardwired signal for simulator

Name	Type	Equipment	Range
Emergency start	DI	GT	0, 5V
Power demand	AI	GT	0~10V
PM PORT speed	AI	PM	0~10V
PM STBD speed	AI	PM	0~10V
PORT Torque	AO	Shaft & FPP	0~10V
STBD Torque	AO	Shaft & FPP	0~10V

실제 함정에서는 주로 전류신호를 많이 사용하나, 본 시뮬레이터에서는 시스템 구성을 단순화하기 위하여 전압신호를 사용하여 구성하며, 필요에 따라 물리적 신호를 확장하여 사용할 수 있도록 구성한다.



## 4.2 하드웨어 설계

4.1절의 설계 방향과 제 2장의 국외 시뮬레이터 개발 과정을 고려하여 하드웨어 설계를 진행한다. 함정 추진체계용 시뮬레이터의 하드웨어는 중앙 시뮬레이터, 연동장비 시뮬레이터, SCM(Simulation Control and Monitoring)으로 구분된다.

### (1) 중앙 시뮬레이터

물리적 신호 및 직렬 통신 기능을 통하여 주고받은 데이터 입출력을 처리하고, 내부 개별 장비 모델에 따라 시뮬레이션을 수행하는 장비로 본 연구에서는 NI사의 PXI(PCI eXtensions for Instrumentation)를 이용하여 구성한다.

Fig. 28의 PXI는 다양한 통신 규격과 물리적 신호처리가 가능한 모듈을 제공하고, 추후 확장이 용이하다는 장점이 있다.



Fig. 28 PXI(PCI eXtensions for Instrumentation)

중앙 시뮬레이터는 TCP/IP, RS422/485, 물리적 신호(Digital, Analog)를 처리할 수 있도록 다양한 모듈을 선정하여 설치한다. Table 14는 중앙 시뮬레이터의 구성을 정리한 것이다.

Table 14 Component of central simulator

Name	EA	Description
NI PXIe-1078	1	9-slot chassis
NI PXIe-8115	1	Core i5-2510E 2.5GHz controller
NI PXI-2722	1	5-channel 16-bit programmable resistor
NI PXI-6221	1	16AI, 24DIO, 2AO
PXI-8431/8	1	RS422/485, 8 port serial interface
SCB-64A	1	Shielded I/O connector block
CB-37F-HVD	1	Terminal block
SH37F-37M-2	1	Female to male shielded I/O cable

(2) 연동장비 시뮬레이터

각 연동장비의 동작을 모니터링하는 장비로, 본 연구에서는 2대의 PC를 연동하여 모니터링 한다.

(3) SCM(Simulation Control and Monitoring)

함정 추진체계용 시뮬레이터의 전반적인 시스템 관리 역할을 수행하며 구체적인 기능은 다음과 같다.

- 추진장비 시뮬레이션의 동작 상태 모니터링 및 데이터 로깅 기능
- 추진장비 시뮬레이션의 동작 제어(on/off, RPM 변경 등) 기능
- 추진장비 시뮬레이션의 모델변수 변경 및 모델 변경 기능
- 환경조건(사고 발생 등 여러 가지 상황 모사)을 시간에 따라 자동으로 제어하는 기능

중앙 시뮬레이터와 연동장비 시뮬레이터, 중앙 시뮬레이터와 SCM은 TCP/IP를 이용한 내부 네트워크로 데이터를 송수신하도록 구성하며, 중앙 시뮬레이터는 각각 추진장비의 통신 방식에 따라 ECS의 DAU를 통해 데이터를 송수신하도록 구성한다. Fig. 29는 함정 추진체계용 시뮬레이터 하드웨어 구성도이며, Fig. 30은 실제 설치 사진을 나타낸다.

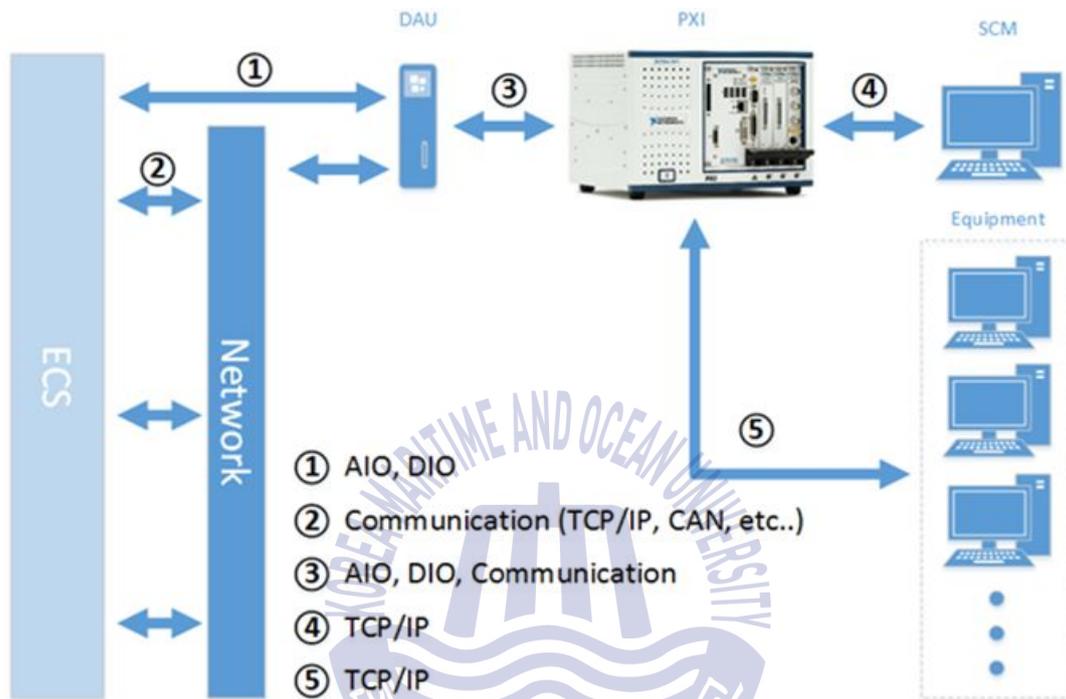


Fig. 29 Configuration of simulator hardware

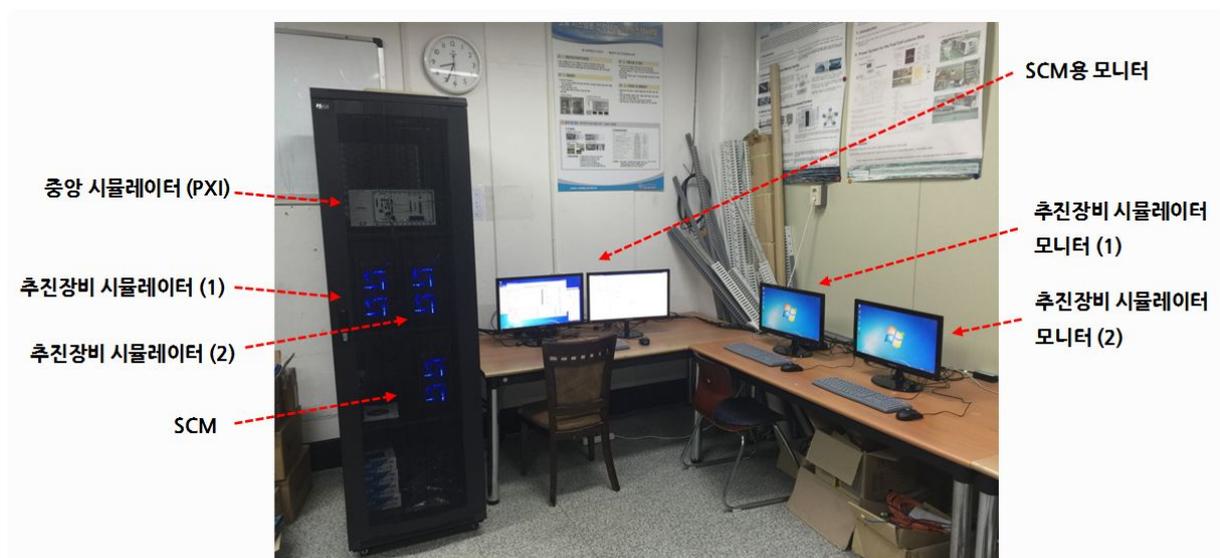


Fig. 30 Photo of simulator

### 4.3 소프트웨어 설계

함정 추진체계용 시뮬레이터의 소프트웨어는 크게 추진장비 동적 시뮬레이션을 위한 모델과 추진장비 제어기의 제어 알고리즘 시뮬레이션을 위한 모델로 구분할 수 있다. 추진장비 제어기의 제어 알고리즘은 시간의 오차가 시뮬레이터의 신뢰성에 큰 영향을 주지 않지만, 동적 시뮬레이션의 경우, 시간의 오차는 매우 중요하다.

따라서 동적 시뮬레이션이 운용되는 컴퓨터의 운영체제(Operation System, OS)에 대한 검토가 반드시 필요하다.

#### 4.3.1 동적 시뮬레이션을 위한 OS

동적 시뮬레이션의 경우, 모델의 시뮬레이션 결과값에 대한 신뢰성 확보를 위해 정시성(determinism)이 확보되어야 한다. 정시성이란 루프에 정해진 동작 시간을 정확하게 유지하는 기능으로 동적 시뮬레이션을 위해서는 반드시 확보되어야 한다.

그러나 일반 OS에서는 시뮬레이션 프로그램 이외에도 다양한 백그라운드 프로그램이 동작중이며, 일반 OS에서는 이러한 프로그램을 운용하기 위해 시분할 방식을 이용하여 CPU 자원을 할당하게 된다. 따라서 시뮬레이션 프로그램에서 지터(jitter)가 자주 발생하게 되고, 결과적으로 시뮬레이션의 정시성이 확보될 수 없다는 문제점을 가지고 있다.

이러한 문제를 해결하기 위해 동적 시뮬레이션의 경우 실시간(Real Time, RT) OS에서 수행한다. 실시간 OS는 일반 OS에 비해 지터가 적게 발생하며, 중요한 루프의 경우 우선순위를 할당할 수 있기 때문에 일반 OS에 비해 정시성 확보가 용이하다.

Fig. 31은 일반 OS에서 발생하는 지터에 대해 나타낸 것이다.

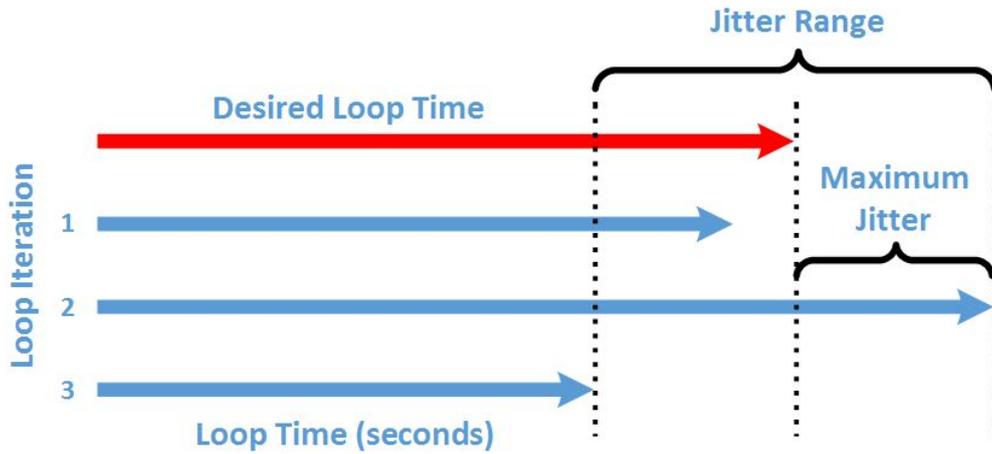


Fig. 31 Jitter in general purpose OS

Fig. 32는 일반 OS와 실시간 OS의 차이점을 비교하여 나열한 것이다.

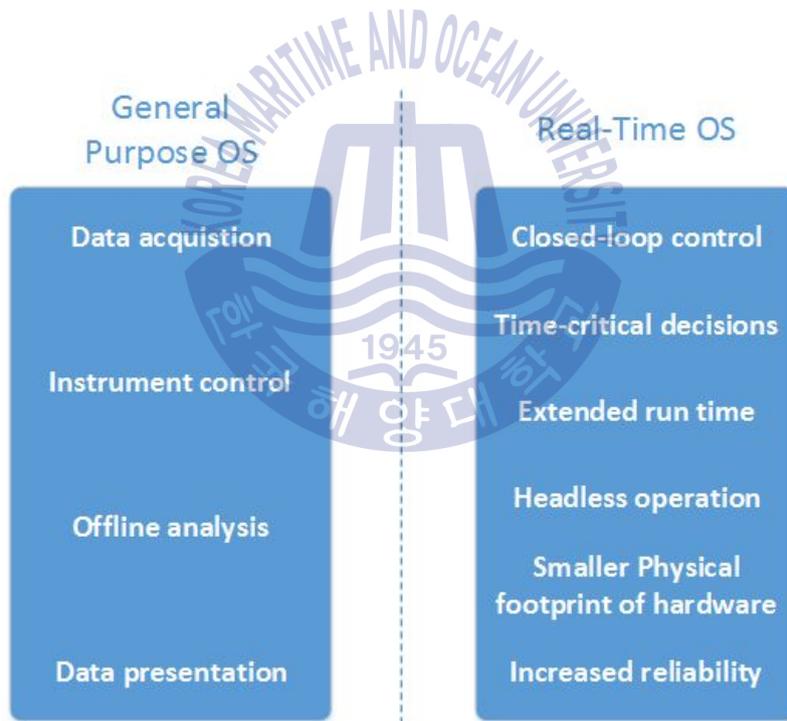


Fig. 32 Comparison of general purpose OS and real-time OS

이러한 점을 고려하여 본 연구에서는 RT OS의 하나인 Phar Lap ETS를 이용하여 시뮬레이션을 수행한다.

### 4.3.2 RT OS의 우선순위

일반 OS에 비해 RT OS는 지터가 비교적 적게 발생하지만, 발생할 가능성을 여전히 가지고 있다. 따라서, 시뮬레이션 소프트웨어 중 정시성이 필요한 모듈과 정시성이 중요하지 않은 모듈을 나눠서 우선순위를 정의할 필요가 있다.

시뮬레이터의 기능에 따라 모듈을 나누면 통신, 동적 시뮬레이션(dynamic simulation), 제어기 알고리즘 모델, 데이터 수집, 데이터 저장, 유저 인터페이스로 구분할 수 있다. 여기서 정시성이 필요한 모듈은 데이터 수집 모듈과 동적 시뮬레이션 모듈이 여기에 해당된다.

제어기 알고리즘 모델, 통신 모델, 데이터 저장 모듈 및 유저 인터페이스 모듈은 비정시성 모듈에 해당하며 유저 인터페이스 모듈은 SCM에 설치된다.

Fig. 33은 시뮬레이터의 모듈에 따른 분류를 나타낸 것이다.

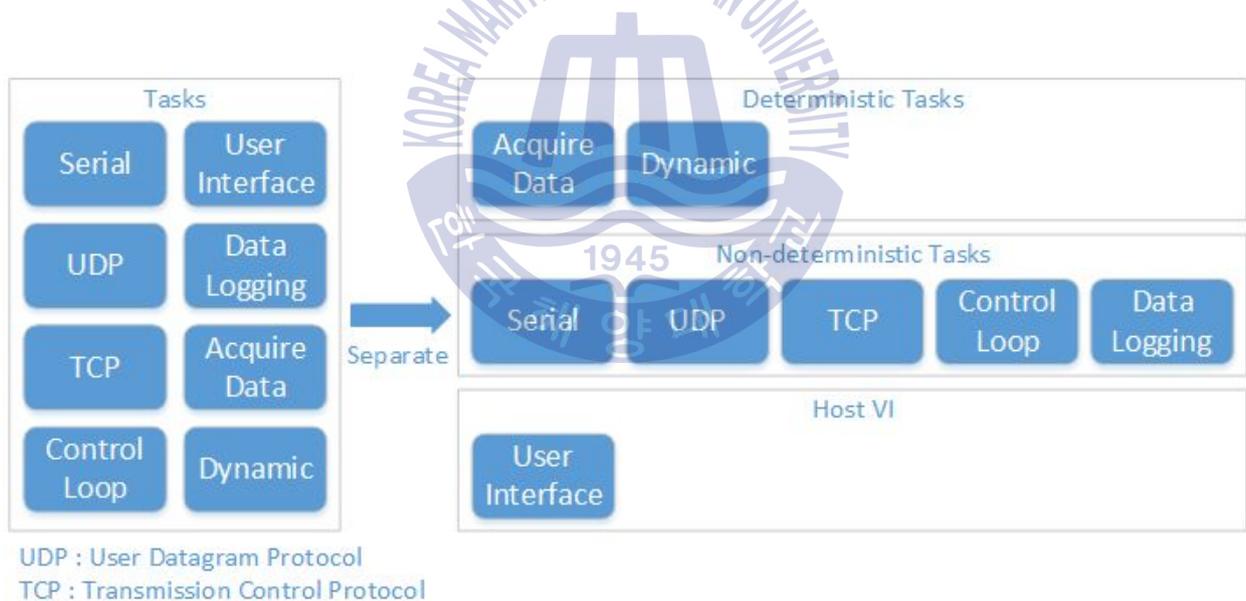


Fig. 33 Separation of deterministic and non-deterministic task

실시간 OS 기반의 시뮬레이션(RT HILS)을 구축하는 경우 실시간 OS와 데이터 수집 장치, 통신기능 등이 포함된 시뮬레이션용 PC와, 시뮬레이션 동작 상태, 제어 등을 관리하는 관리용 PC로 구분하여 각각의 PC간 데이터 공유, PC 내부의 프로그램간 데이터 공유에 대한 검토가 필요하다.

특히 실시간 OS에서 운용되는 프로그램은 정시성을 유지한 상태에서 데이터 공유가 수행되어야 함으로 이에 대한 기술적 검토가 필요하다.

Fig. 34는 실시간 OS에서 내부 프로그램 루프간의 데이터 공유에 대한 것을 나타낸 것이다.

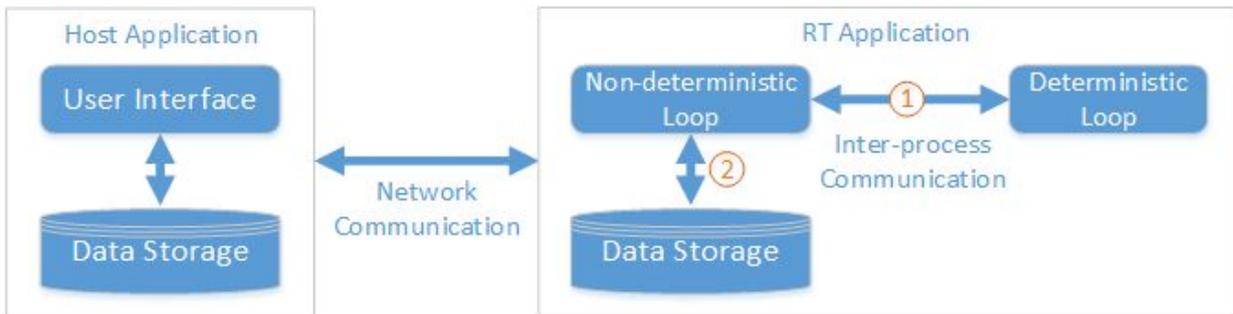


Fig. 34 Sharing data on RT target

데이터를 공유하는 방법은 데이터 공유의 목적과 데이터를 주고받는 루프의 종류에 따라 달라진다.

정시성 루프와 비정시성 루프간의 데이터 공유는 일반적으로 RT FIFO(First Input First Output)를 이용한다. RT FIFO는 일반 OS에서 큐와 유사한 것으로 일종의 버퍼를 가지고 있어서 데이터의 손실 없이 각 루프간 데이터를 주고받을 수 있다.

Table 15는 루프간 데이터를 주고받는 방법에 대해 정리한 것이다.

Table 15 Inter-process communication

	Deterministic loop and Non-deterministic loop	Non-deterministic loop and Non-deterministic loop
Latest value	- RT FIFO (single element)	- Single process shared variable - Local variable - Function global variable
Every value	- RT FIFO (multi element)	- Queues

정시성 루프와 비정시성 루프간 데이터 공유시 루프 정지 신호나 I/O 데이터 값의 경우 RT FIFO(single element)를 사용하고, 시뮬레이션 결과, 데이터 로깅 등 중요한 데이터 다수를 보내는 경우 RT FIFO(multi element)를 사용하여 구성한다.

### 4.3.3 소프트웨어 설계 구체화

본 연구에서 개발하려는 함정 추진체계용 시뮬레이터는 실시간 기반의 HILS 로 구현한다. 하드웨어는 4.2절에서 정리한 바와 같이 중앙 시뮬레이터, 연동장비 시뮬레이터, SCM으로 구성되며 중앙 시뮬레이터는 NI사의 PXI 연동장비 시뮬레이터와 SCM은 일반 PC로 구성한다.

Fig. 35와 Fig. 36은 함정 추진체계용 시뮬레이터의 소프트웨어 개념도이다.

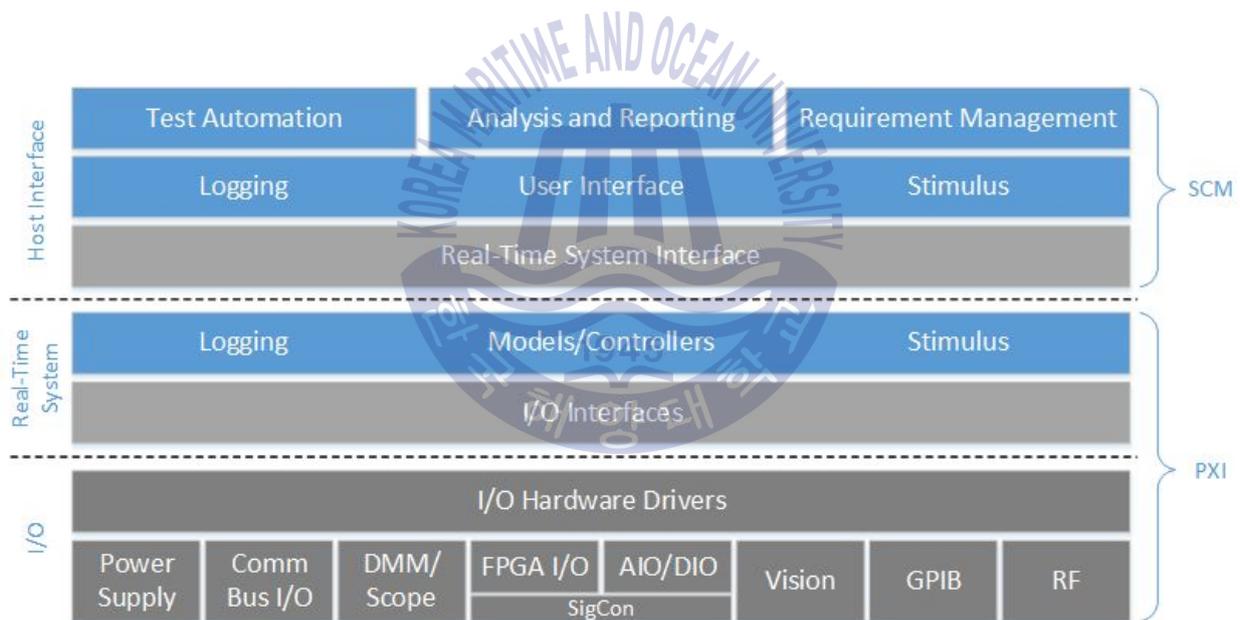


Fig. 35 Conceptual diagram of simulator (1)

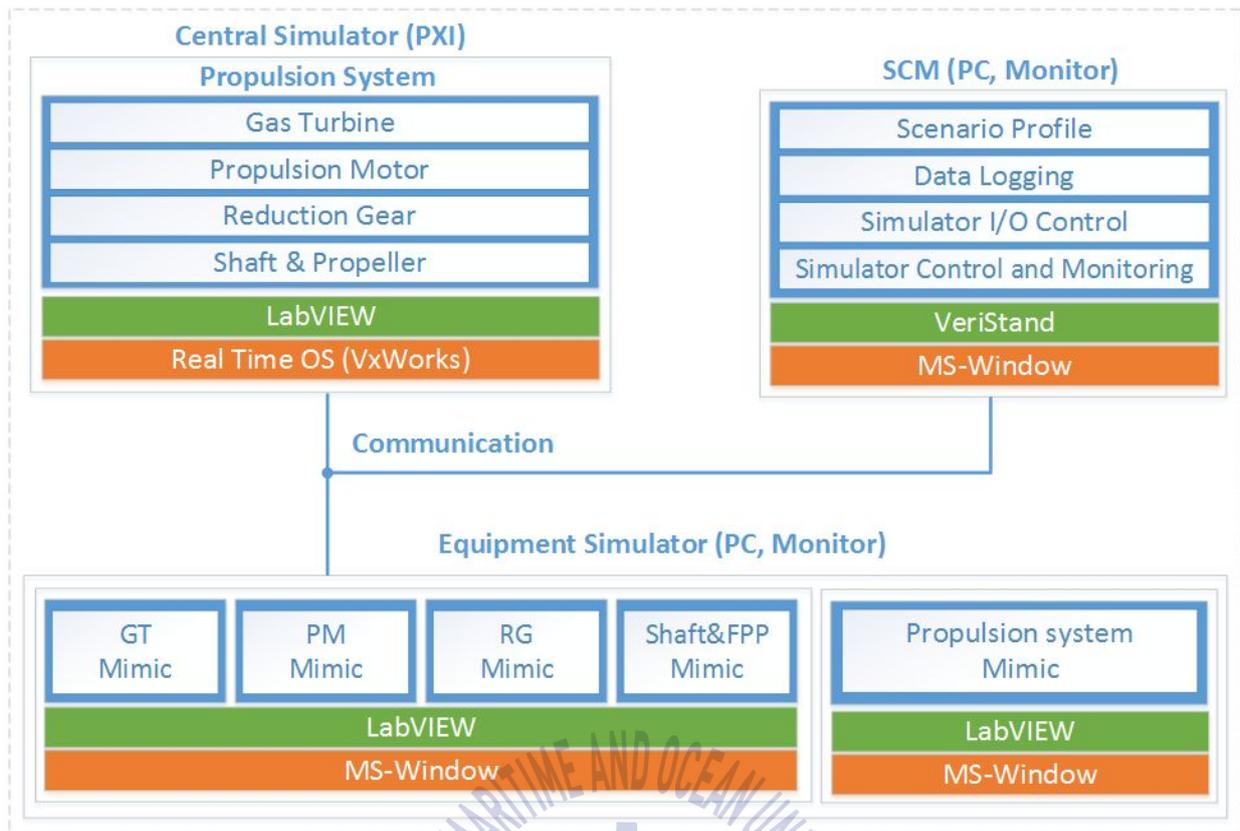


Fig. 36 Conceptual diagram of simulator (2)

시뮬레이터에서 관리 역할을 하는 SCM은 데이터 저장, 데이터 분석, 시뮬레이션 자동화, 유저 인터페이스 등의 역할을 수행하고 PXI과 실시간 인터페이스를 통해 데이터를 주고 받는다.

시뮬레이션을 수행하는 주체인 PXI는 데이터 저장, 시뮬레이션, I/O를 통한 데이터 입출력 처리 등의 기능을 수행한다.

연동장비 시뮬레이터는 2대의 PC로 구성하며 1대의 PC에서는 각각의 연동장비에 대한 세부적인 동작들을 나타내고, 다른 1대의 PC는 전체 시스템을 통합해서 나타내는 역할을 수행한다.

Fig. 37은 함정 추진체계용 시뮬레이터 프로그램의 개발 과정을 나타낸 것이다.

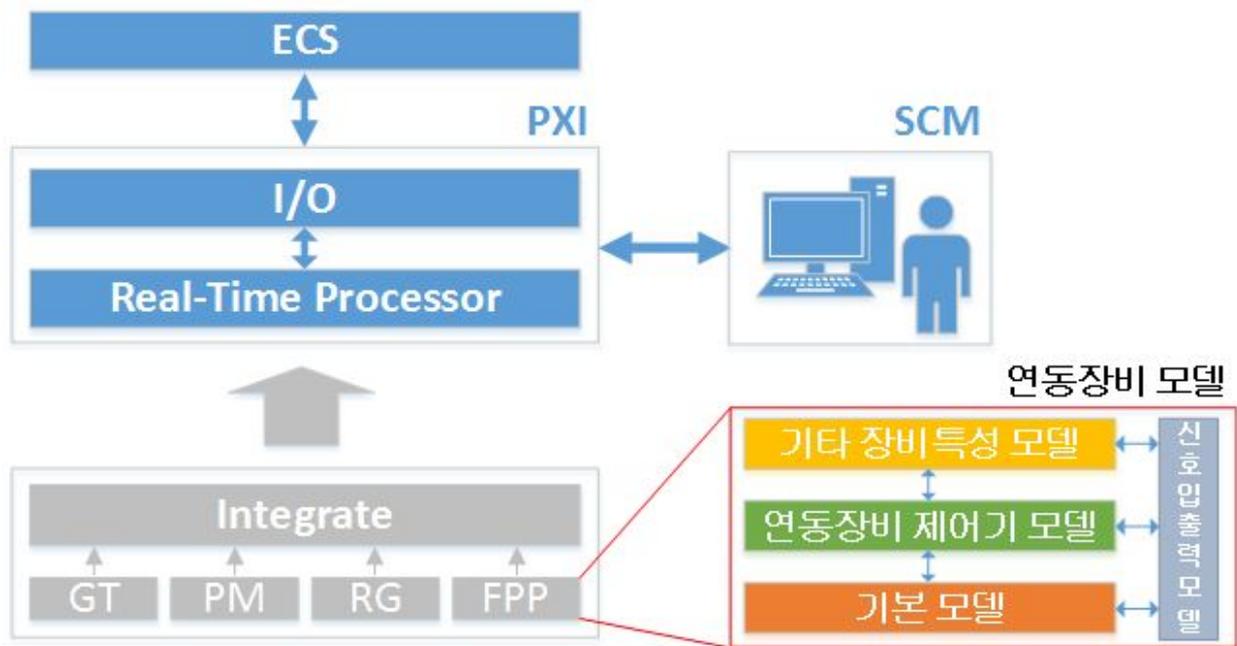


Fig. 37 Development process of simulator program

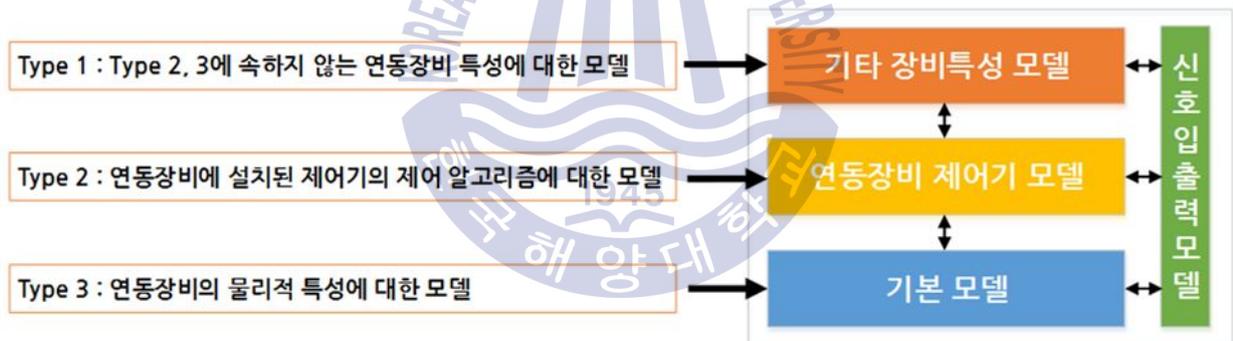


Fig. 38 Configuration of equipment model

CODLOG 추진체계의 연동장비인 가스터빈, 추진전동기, 감속기어, 축계 및 고정피치 프로펠러는 각각 하나의 연동장비 모델로 구성된다. 연동장비 모델은 Fig. 38과 같이 기능에 따라 기본 모델, 연동장비 제어기 모델, 기타 장비특성 모델, 신호 입출력 모델로 구분된다.

(1) 기본 모델 : 각 연동장비의 물리적 특성을 모사하는 수학적 모델로, 가스터빈의 회전속도, 토크 등이 여기에 속한다.

(2) 연동장비 제어기 모델 : 각 연동장비에 설치된 제어기의 제어 알고리즘을 프로그램으로 구현한 것으로 각 장비별 인터락, 보조기기 제어 등의 기능을 포함한다.

(3) 기타 장비특성 모델 : 기본 모델과 연동장비 제어기 모델에 속하지 않으나, 연동장비 운용을 위해 반드시 필요한 기능을 포함한다. 감속기어의 브레이크, SSS 클러치 등에 대한 모델이 여기에 포함된다.

(4) 신호 입출력 모델 : 연동장비 모델로 송수신되는 모든 데이터를 변환, 관리하는 역할을 수행한다. TCP, RS485 등 통신에 대한 기능을 지원한다.



## 4.4 연동장비 모델 개발

Fig. 38에서 정의한 바와 같이 연동장비 모델은 기본 모델, 연동장비 제어기 모델, 기타 장비특성 모델, 신호 입출력 모델로 구분된다. 이 절에서는 연동장비 모델의 신호 입출력, 제어 알고리즘을 구분하고 프로그램으로 개발한다.

위 4가지 연동장비 세부 모델중 기타 장비특성 모델의 경우 기본 모델, 연동장비 제어기 모델만으로 각 연동장비의 특성을 모델링할 수 없을 때 나머지 기능을 구현하기 위해 만들어 놓은 것으로 개발자의 필요에 따라 사용한다.

### 4.4.1 기본 모델

연동장비 모델중 기본 모델은 각 장비의 물리적 특성을 모사한 것으로 일반적으로 가스터빈, 감속기어 등 다양한 장비들에 대해서 다양한 모델링이 연구되어 있다[27][31][32]. 그러나, 제 2장에서 검토한 바와 같이 RT- HILS 시뮬레이터는 실시간으로 입출력 처리와 시뮬레이션이 가능해야 하기 때문에 기존의 열역학 기반의 모델 대신 단순화된 모델을 사용해야 한다.

추진체계에 대한 동적 시뮬레이션은 출력을 발생시키는 엔진 모델, 엔진에서 발생한 회전 에너지를 밀어내는 힘(추력, thrust)으로 전환하는 프로펠러 모델, 그리고 프로펠러의 추력을 이용하여 함정의 속도를 계산하는 선체 모델로 구분할 수 있다.

Fig. 39는 함정 추진체계의 동적 시뮬레이션을 위한 개념도를 나타내고 있다.



Fig. 39 Conceptual diagram of ship dynamic simulation

엔진 모델은 ECS로부터 제어 신호를 받아서 토크와 회전속도를 발생시킨다. 그러나 모델링에서는 시뮬레이션에서는 둘중에 하나를 발생하는 주체로 보고

구성하여야 한다. 본 시뮬레이션에서는 회전속도를 발생시키는 주체로 구성하였다.

함정 추진체계의 동적 시뮬레이션의 목적은 각 장비별 물리적 특성을 고려하여 함속, 각 축 및 장비의 토크, 회전속도, 추력을 시뮬레이션을 통해 계산하고 ECS에서 이러한 값을 받아 적절하게 제어 하는 제어 알고리즘을 개발하기 위해 활용된다. 따라서, 각 연동장비의 온도, 전압 및 전류의 맥동 등에 대한 세부적인 정보는 각 연동장비의 성능, 안정성을 평가하는데 있어서 매우 중요한 요소지만, 이러한 요소는 각 장비 제작사에서 장비 성능 예측 또는 개선을 위해 검토되어야 할 부분으로 본 시뮬레이터에서는 구현하지 않는다. 또한, 시스템을 단순화하기 위해 함정의 전, 후진 움직임에 대해서만 고려한다.

Fig. 40은 Fig. 39와 연계하여 본 연구에서 개발하고자 하는 시뮬레이터의 기본 모델간 관계도를 나타낸 것이다.



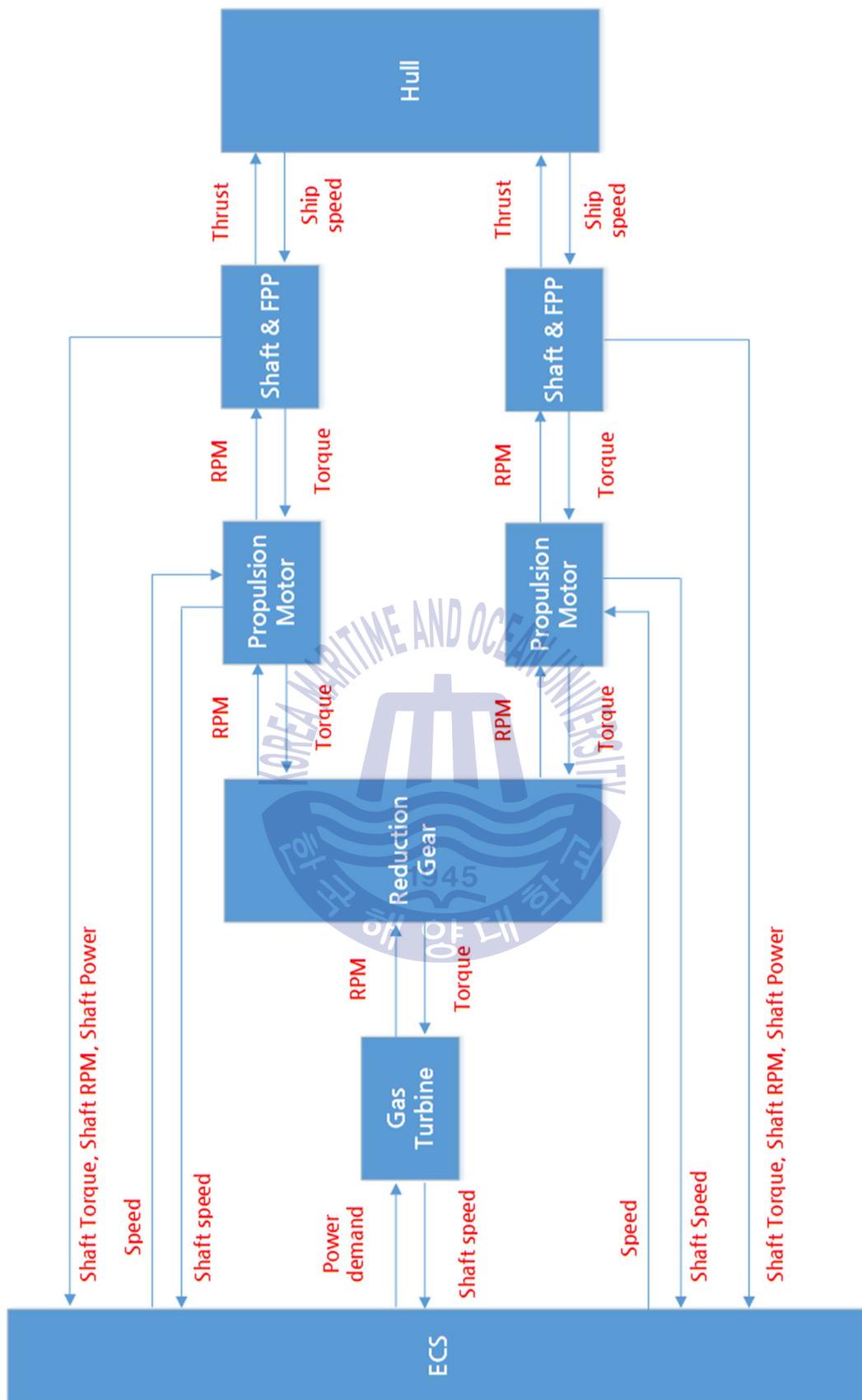


Fig. 40 Conceptual diagram of basic model interaction

### (1) 가스터빈과 추진전동기

Fig. 39의 동적 시뮬레이션 개념에서 바라볼 때 가스터빈과 추진전동기는 동력을 생성하는 엔진에 해당한다고 볼 수 있다. 일반적으로 가스터빈, 추진전동기 등에 대한 다양한 동역학적 모델링이 개발되어 있으나, 본 시뮬레이터는 각 장비의 운동 특성을 예측하는 것이 아니라 함정 전체의 운동 특성을 예측하는 것이므로 가스터빈 각 부의 온도, 손실, 압력 등에 대한 결과를 예측하는 모델링은 필요하지 않다.

따라서 본 연구에서는 이러한 모델링을 단순화하기 위해 ECS에서 속도 명령값에 따른 가스터빈과 추진전동기의 회전 속도 변화를 시간 진행에 따라 변화도록 구성한다.

Fig. 41은 일정한 속도입력을 주었을 때 가스터빈과 추진전동기의 출력 회전 속도의 변화를 나타낸 것이다.

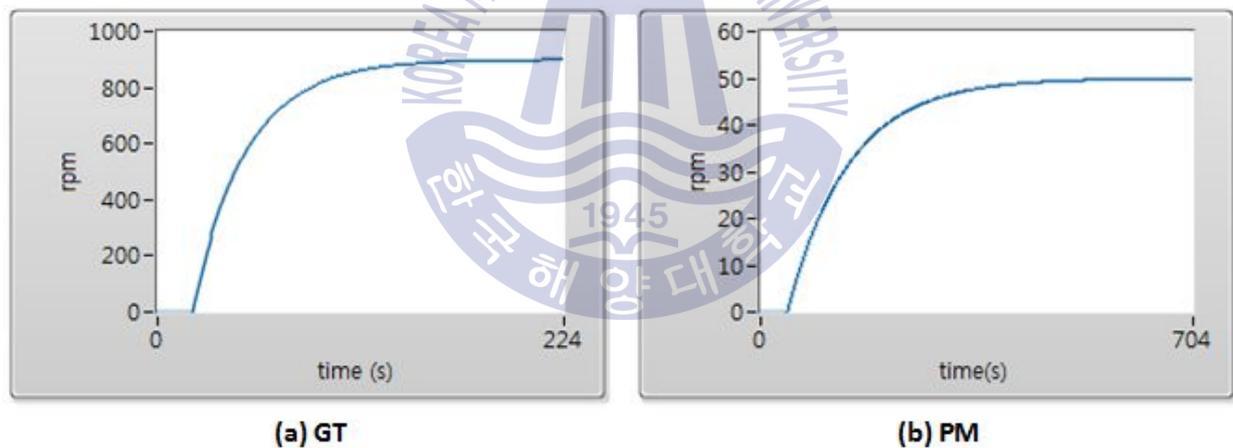


Fig. 41 Step response of GT, PM



Table 18, Table 19는 PM에 대한 입출력을 나타낸 것이고, Fig. 43는 PM를 위한 프로그램을 나타낸다.

Table 18 Input for PM basic model

Signal	Type	From
PM PORT speed [rpm]	AI	Communication
PM STBD speed [rpm]	AI	Communication
PM PORT torque [kNm]	DI	Fixed Pitch Propeller
PM STBD torque [kNm]	DI	Fixed Pitch Propeller

Table 19 Output for PM basic model

Signal	Type	To
PM PORT shaft speed [rpm]	AO	Communication
PM PORT shaft torque [kNm]	AO	Communication
PM PORT shaft power [kW]	AO	Communication
PM STBD shaft speed [rpm]	AO	Communication
PM STBD shaft torque [kNm]	AO	Communication
PM STBD shaft power [kW]	AO	Communication

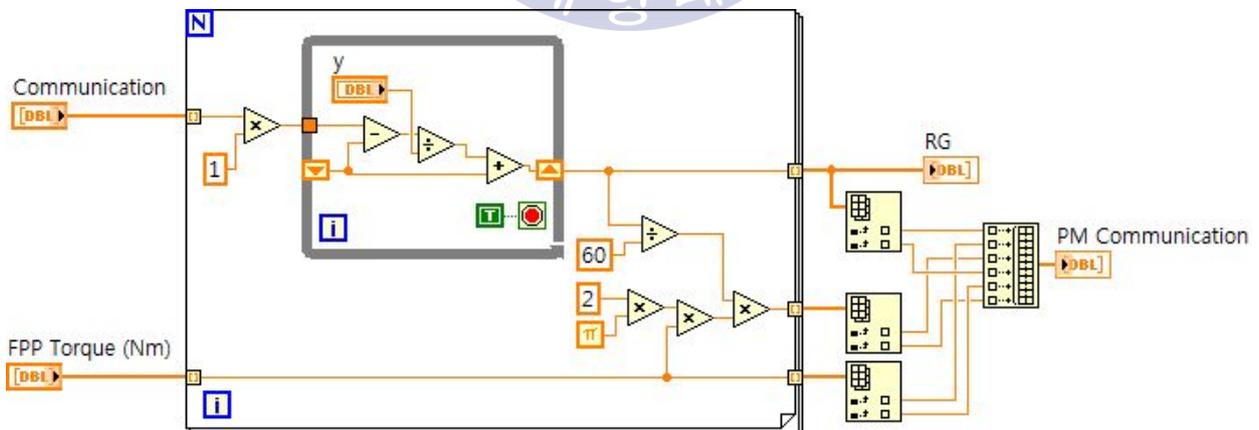


Fig. 43 Program for PM basic model

## ② 감속기어

감속기어의 경우 내부에 설치된 베어링, 기어형상, 연결 방법에 따라 다양한 시뮬레이션 모델을 개발한다. 그러나, 감속기어를 동력 전달과 기어비에 따른 회전속도와 토크의 변화 관점에서 보면 감속기어 모델은 회전속도에 관한 식과 토크에 관한 식으로 간략화 할 수 있다[18].

$$k = \frac{n_g}{n_p} \quad (1)$$

$$Q_p = kQ_g e_r \quad (2)$$

여기서  $k$ 는 기어비,  $Q_p$  및  $Q_g$ 는 각각 프로펠러 및 가스터빈의 토크 [Nm]를 나타내며,  $e_r$ 는 에너지 전달 효율이다.  $n_p$  및  $n_g$ 는 각각 프로펠러 및 가스터빈의 회전속도 [rpm]를 나타낸다.

또한, 실제 감속기어에서는 전달하는 에너지와 회전속도에 따른 효율곡선이 존재한다. 최근 개발되는 감속기어의 경우 대부분의 구간에서 98% 이상의 에너지 전달 효율을 보이고 있음으로 본 연구에서도 감속기어 효율을 98%로 가정하고 모델을 개발한다.

또한, 감속기어에는 CC SSS 클러치, PORT SSS 클러치, STBD SSS 클러치 총 3개의 SSS 클러치가 설치되어 있으며, Fig. 14와 같이 입력축 축 회전속도가 출력축 축 회전속도보다 높아지는 경우 서로 연결되고, 낮아지는 경우 떨어지는 구조를 가지고 있다. 감속기어 기본 모델에서는 이와 같은 동작을 구현할 수 있도록 해야 한다.

또한 브레이크 신호에 따른 동작 특성도 구현되어야 한다.

Fig. 44는 감속기어의 입출력 신호와 프로그램을 나타내고 있으며, Table 20, 21은 감속기어에 대한 입출력을 나타낸 것이다.

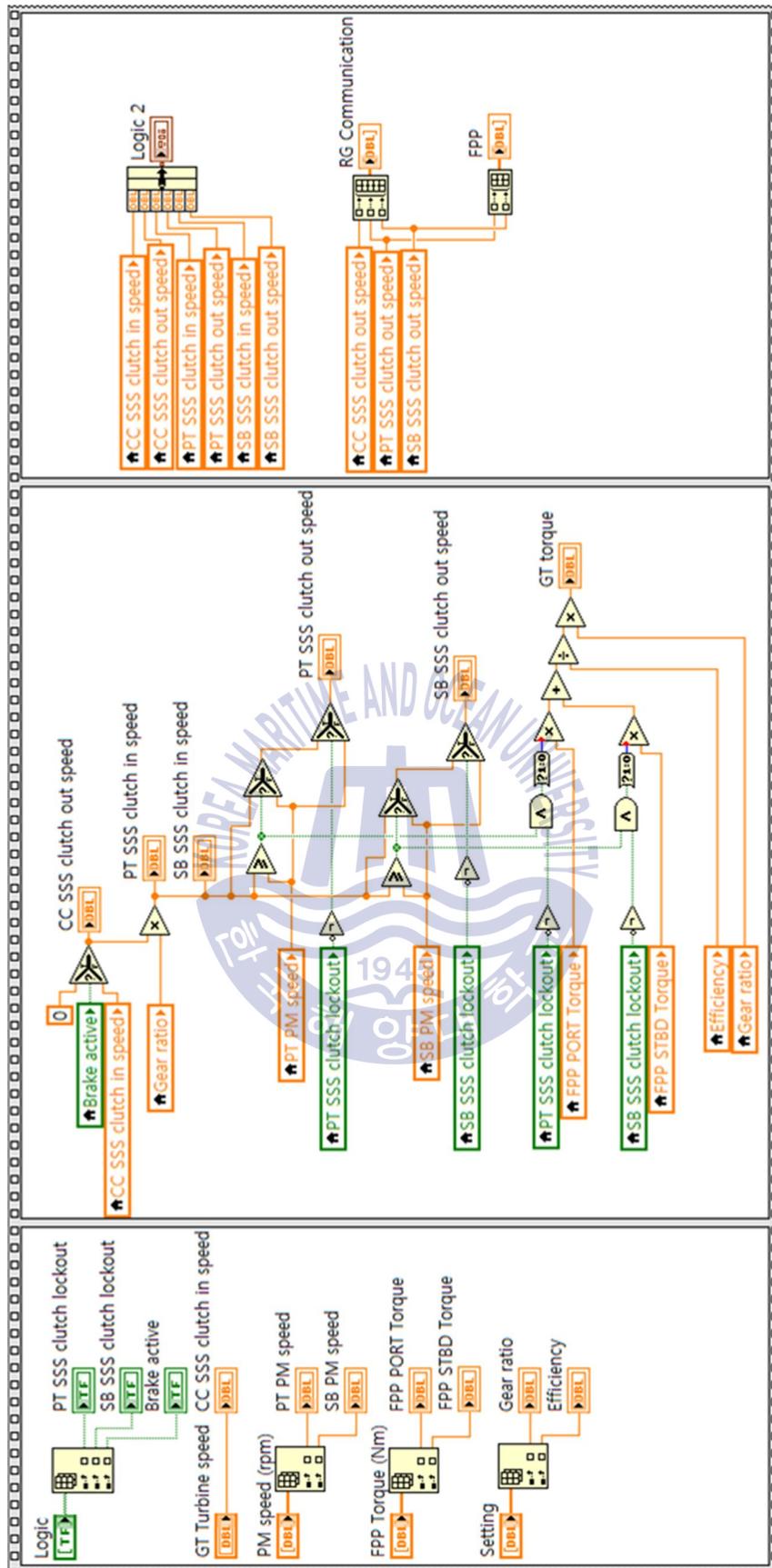


Fig. 44 Program for RG basic model

**Table 20** Input for RG basic model

Signal	Type	From
PT SSS clutch ratcheting	DI	Logic
SB SSS clutch ratcheting	DI	Logic
Brake active	DI	Logic
GT turbine speed [rpm]	AI	Gas Turbine
PT PM speed [rpm]	AI	Propulsion Motor
SB PM speed [rpm]	AI	Propulsion Motor
FPP torque [rpm]	AI	Fixed Pitch Propeller
Gear ratio	AI	Setting
Efficiency	AI	Setting

\* PT : PORT, SB : STBD

**Table 21** Output for RG basic model

Signal	Type	To
RG CC SSS clutch speed [rpm]	AO	Communication
RG PT SSS clutch speed [rpm]	AO	Communication
RG SB SSS clutch speed [rpm]	AO	Communication
CC SSS clutch in speed [rpm]	AO	Logic
CC SSS clutch out speed [rpm]	AO	Logic
PT SSS clutch in speed [rpm]	AO	Logic
PT SSS clutch out speed [rpm]	AO	Logic
SB SSS clutch in speed [rpm]	AO	Logic
SB SSS clutch out speed [rpm]	AO	Logic
GT torque [kNm]	AO	Gas Turbine
PT SSS clutch out speed [rpm]	AO	Fixed Pitch Propeller
SB SSS clutch out speed [rpm]	AO	Fixed Pitch Propeller

### ③ 축계 및 고정피치 프로펠러

축계 및 고정피치 프로펠러에서 축계는 출력을 전달한다고 가정하며 별도의 물리적 모델링을 하지 않는다. 따라서 축계 및 고정피치 프로펠러 모델은 프로펠러에 대한 물리적 모델링만 고려한다.

프로펠러에서 발생하는 추력  $T_p$  [N] 및 토크  $Q_p$  [Nm], 출력  $P_p$  [kW]는 프로펠러 회전속도  $n_p$  [rps]와 프로펠러 직경  $D$  [m] 그리고 해수 밀도  $\rho_w$  [kg/m<sup>3</sup>]로 계산되어 진다[18][19].

다음 식 (3), (4), (5)는 각각 추력, 토크, 출력을 나타낸다.

$$T_p = K_T \rho_w D^4 n^2 \quad (3)$$

$$Q_p = K_Q \rho_w D^5 n^2 \quad (4)$$

$$P_p = 2\pi Q_p = 2\pi K_Q \rho_w D^5 n^2 \quad (5)$$

여기서  $K_T$ 와  $K_Q$ 는 추력과 토크의 계수이며 다음 식 (6), (7)로 나타낼 수 있다.

$$K_T = f_1(J_a, P/D, A_E/A_O, Z) \quad (6)$$

$$K_Q = f_2(J_a, P/D, A_E/A_O, Z, R_n, l/c) \quad (7)$$

여기서  $J_a$ 는 전진계수,  $R_n$ 는 레이놀드 수,  $P/D$ 는 프로펠러의 피치와 직경의 비,  $A_E/A_O$ 는 블레이드 비,  $Z$ 는 프로펠러의 블레이드 개수,  $l$ 은 블레이드의 가장 두꺼운 부분의 두께,  $c$ 는 블레이드의 현의 길이를 나타낸다.

$J_a$ 를 제외한 다른 값은 상수로서 프로펠러 형상에 의해 결정된다.  $J_a$ 는 프로펠러의 회전속도  $n$ [rps]과 함속  $V$ [m/s], 프로펠러 직경  $D$ [m]에 의해 식 (6)처럼 결정된다.

$$J_a = \frac{V}{nD} \quad (6)$$

프로펠러 성능 예측을 위해서는 실제로 위의 식을 고려하여 시뮬레이션을 하지만, 함정 추진체계용 시뮬레이터를 개발하는 경우 실제로 조선소에서 프로펠러 자체에 대하여 제작하기 때문에  $K_T$  및  $K_Q$ ,  $J_a$ 는 프로펠러 단독 실험(open water test)에서 실험적 데이터로 확보할 수 있다.

본 연구에서는  $K_T$ 와  $K_Q$ 는 다음과 같은 데이터를 가정하여 축계 및 고정피치 프로펠러 모델을 개발하였다.

Table 22 Value of coefficients  $K_T$  and  $K_Q$

$J_a$	$K_T$	$K_Q$	$J$	$K_T$	$K_Q$
0.00	0.80980	0.16376	0.70	0.45043	0.09742
0.05	0.78847	0.15982	0.75	0.42611	0.09325
0.10	0.75942	0.15427	0.80	0.40109	0.08894
0.15	0.72923	0.14838	0.85	0.37510	0.08439
0.20	0.70237	0.14310	0.90	0.34846	0.07965
0.25	0.67976	0.13868	0.95	0.32177	0.07481
0.30	0.65939	0.13480	1.00	0.29559	0.06989
0.35	0.63818	0.13088	1.05	0.27018	0.06490
0.40	0.61398	0.12648	1.10	0.24547	0.05981
0.45	0.58653	0.12151	1.15	0.22106	0.05464
0.50	0.55724	0.11622	1.20	0.19642	0.04936
0.55	0.52811	0.11099	1.25	0.17094	0.04397
0.60	0.50055	0.10610	1.30	0.14405	0.03847
0.65	0.47488	0.10162			

위 데이터를 보간하여 모델로 나타내면 식 (8), (9)와 같이 나타낼 수 있다.

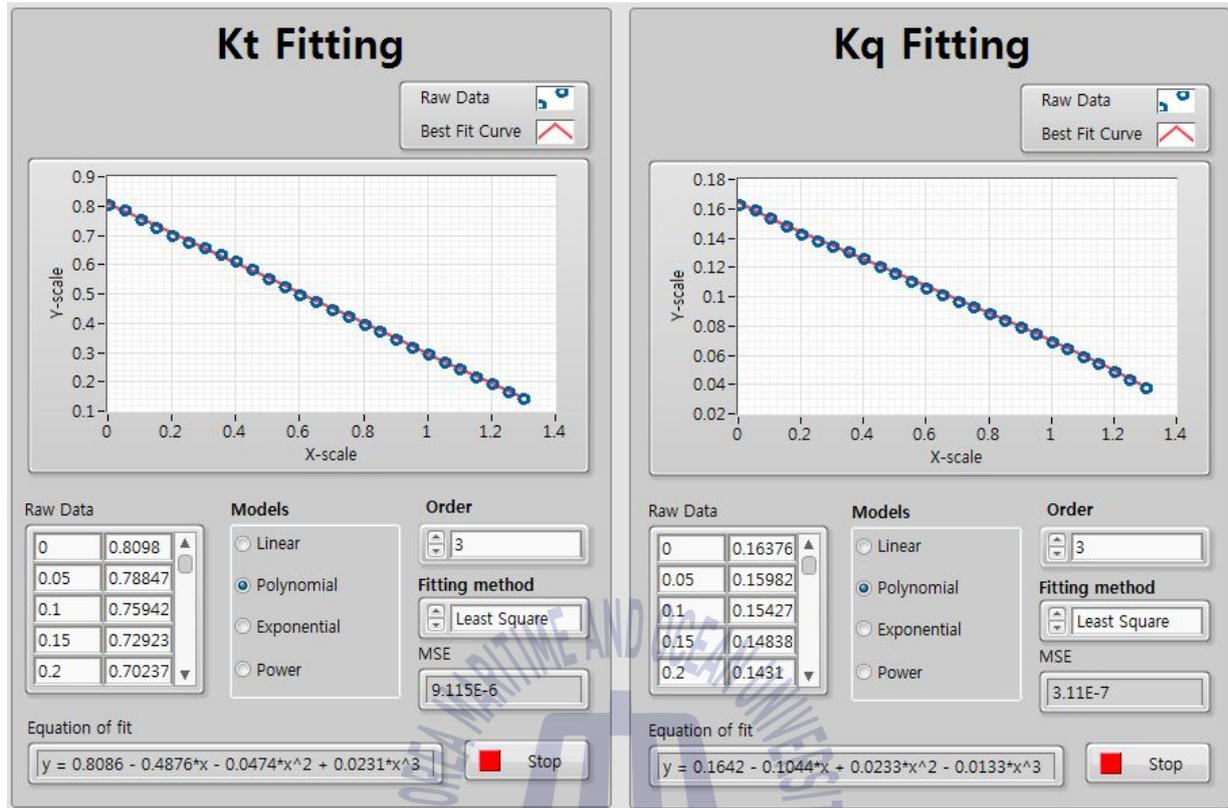


Fig. 45 Fitting of coefficient ( $K_T$  and  $K_Q$ )

$$K_T = 0.8086 - 0.4876J_a - 0.0474J_a^2 + 0.0231J_a^3 \quad (8)$$

$$K_Q = 0.1642 - 0.1044J_a - 0.0233J_a^2 + 0.0133J_a^3 \quad (9)$$

또한 전진계수  $J_a$ 는 회전속도  $n$ 과 함속  $V$ 와의 관계식으로 나타낼 수 있으며 함속은 식 (10)과 같이 나타낼 수 있다.

$$V = -0.2991 + 0.1852n - 9.0232e^{-5}n^2 - 1.949e^{-7}n^3 \quad (10)$$

위 식을 기반으로 프로펠러의 회전속도에 따른  $K_T$ 와  $K_Q$ 를 나타내면 Fig. 46과 같다.

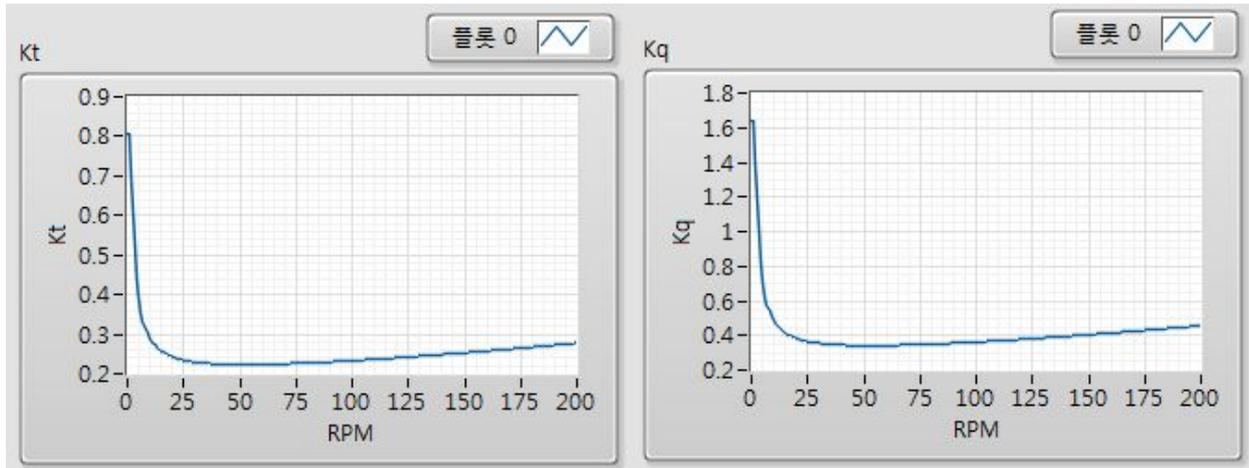


Fig. 46 Coefficient  $K_T$  and  $K_Q$  for propeller RPM

Table 23과 Table 24는 축계 및 프로펠러 관련 식에 따른 입출력이며, Fig. 47과 Fig. 48은 LabVIEW를 이용한 프로그램 구현을 나타낸 것이다.

Table 23 Input for propeller basic model

Signal	Type	From
PORT SSS clutch out speed [rpm]	AI	RG
STBD SSS clutch out speed [Nm]	AI	RG
Propeller diameter [m]	AI	Setting

Table 24 Output for propeller basic model

Signal	Type	To
FPP PORT shaft speed [rpm]	AO	Communication
FPP PORT torque [kNm]	AO	Communication
FPP PORT shaft power [kW]	AO	Communication
FPP PORT shaft speed [rpm]	AO	Communication
FPP PORT torque [kNm]	AO	Communication
FPP PORT shaft power [kW]	AO	Communication
FPP PORT t hrust [kN]	AO	etc
FPP STBD thrust [kN]	AO	etc

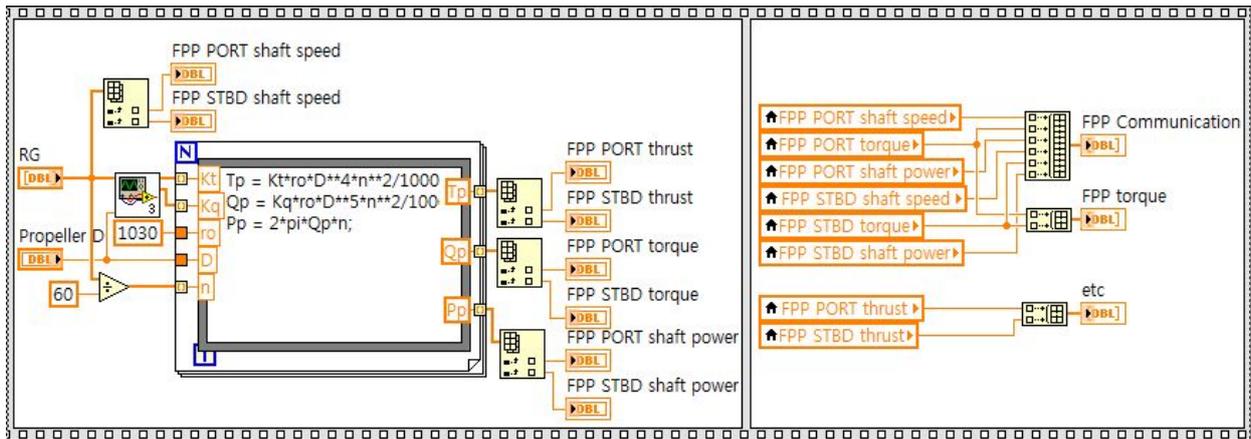


Fig. 47 Program for shaft and FPP basic model (1)

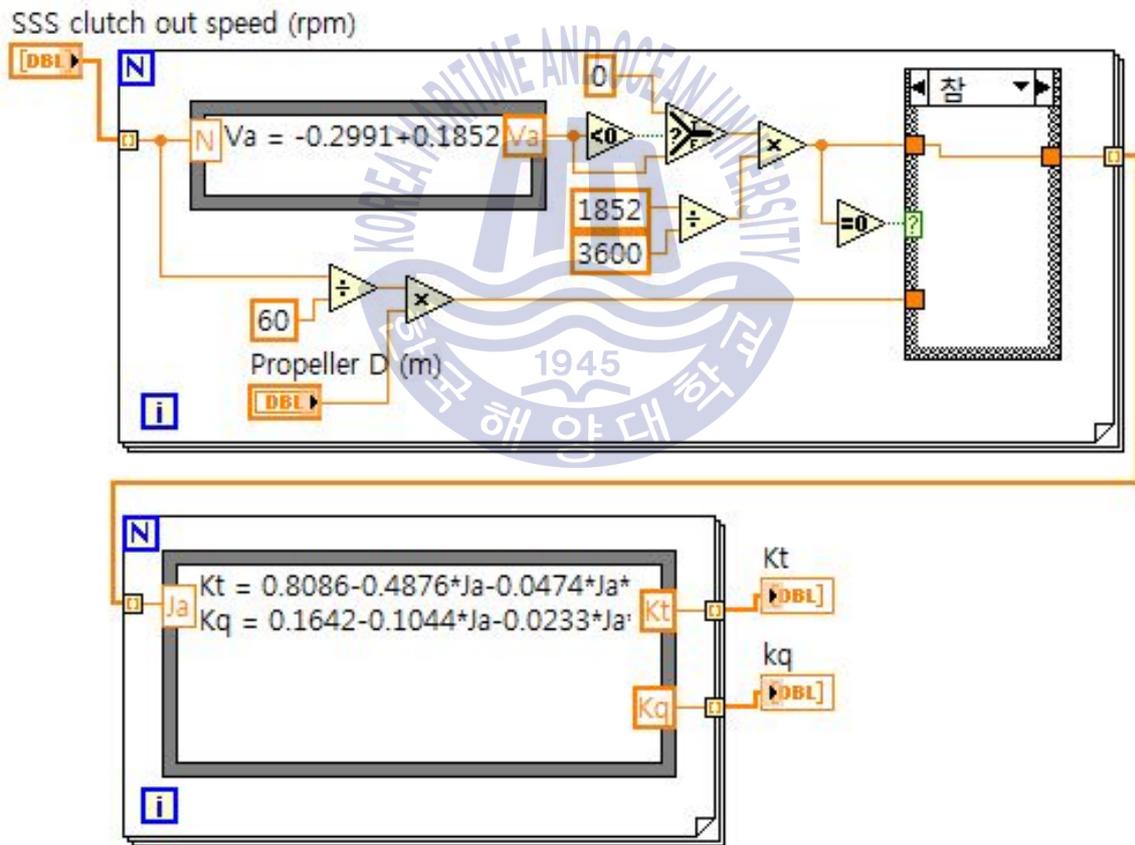


Fig. 48 Program for shaft and FPP basic model (2)

#### 4.4.2 연동장비 제어기 모델

연동장비 제어기 모델은 각각의 연동장비 제어기에서 연동장비를 제어하는 알고리즘을 모델로 나타낸 것이다. 이 제어 알고리즘은 연동장비 제작사가 가지고 있으며, ECS 검증을 위해 연동장비 제작사로부터 제어 알고리즘을 확보하여 모델을 개발해야 한다.

그러나, 이러한 자료는 연동장비 제작사에서 일반인들에게 공개하지 않으며 ECS 개발사 또는 조선소와 같이 계약관계인 경우 필요에 따라 제공함으로써 ECS 검증용 시뮬레이터 개발을 위해서는 합정 건조를 위한 PSI 연동회의에서 이와 같은 점을 명확하게 요구하여 연동장비 제어 알고리즘에 대한 정확한 자료 확보가 필요하다.

본 연구에서는 연동장비 제어 알고리즘을 3.3.2절에서 정의한 ECS 장비 운용 알고리즘에 따른 신호전달 기반으로 유추하여 연동장비 제어기 모델을 개발한다.

##### (1) 가스터빈

가스터빈을 3.3.2절에 따라 기능을 정의하면 다음 3개의 알고리즘으로 구분할 수 있다.

##### a. 인터락 감시 알고리즘

가스터빈 운용 인터락 조건에 대한 알고리즘으로 가스터빈 운용을 위한 보조 기기, 유압 등 신호가 정상이면 ready for start 신호를 1로 만들어서 ECS로 올려보낸다. 실제 가스터빈에는 화재 감시, 윤활유 압력 등 다양한 조건이 있으나 본 연구에서는 팬, 댐퍼, 윤활유 펌프로 단순화하여 구성한다.

Table 25와 Fig. 49는 각각 내부 감시 알고리즘의 입출력 신호와 프로그램을 나타낸 것이다.

Table 25 I/O signal for GT internal check

	Signal	Type	Equipment	Description
Input	Fan start	DI	ECS	0 : stop, 1 : start
	Damper open	DI	ECS	0 : stop, 1 : start
	LO Pump on	DI	ECS	0 : stop, 1 : start
Output	Ready for start	DO	GT	0 : not ready 1 : ready

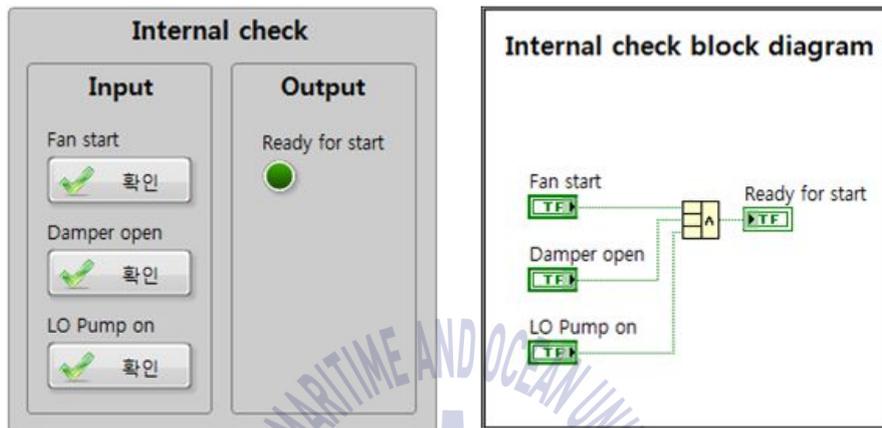


Fig. 49 Program for GT internal check

b. 정상 시동 알고리즘

ECS에서 normal start 신호가 1이 되면 가스터빈은 내부에 정의된 알고리즘에 따라 워업을 5분간 실시한 후, 완료 신호를 ECS로 보내서 ECS에서 가스터빈을 제어할 수 있도록 한다.

Table 26과 Fig. 50은 각각 정상 시동 알고리즘의 입출력 신호와 프로그램을 나타낸 것이다.

Table 26 I/O signal for GT normal start

	Signal	Type	Equipment	Description
Input	Normal start	DI	ECS	0 : not start, 1 : start
Output	Idle warm-up	DO	GT	0 : not warm-up 1 : warming-up (warm-up 5min)
	warm-up time	DO	GT	Present warm-up time (s)

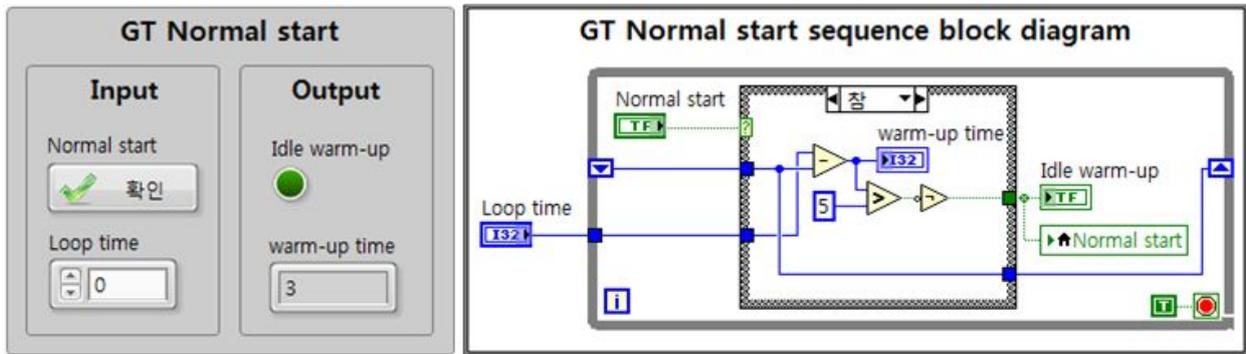


Fig. 50 Program for GT normal start

c. 정상 정지 알고리즘

ECS에서 normal stop 신호가 1이 되면 가스터빈은 내부에 정의된 알고리즘에 따라 워업을 5분간 실시한 후, 완료 신호를 ECS로 보내고 정지한다. Table 27 과 Fig. 51은 각각 정상 정지 알고리즘의 입출력 신호와 프로그램을 나타낸 것이다.

Table 27 I/O signal for GT normal stop

	Signal	Type	Equipment	Description
Input	Normal stop	DI	ECS	0 : not stop 1 : stop
Output	Idle warm-up	DO	GT	0 : not warm-up 1 : warming-up (warm-up 5min)
	warm-up time	DO	GT	Present warm-up time (s)

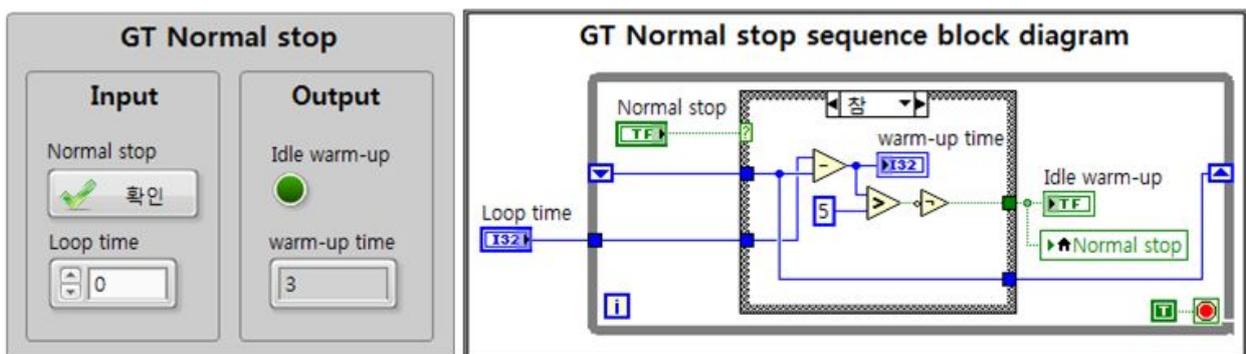


Fig. 51 Program for GT normal stop

d. 긴급 시동 알고리즘

ECS에서 emergency start 신호가 1이 되면 가스터빈은 내부에 정의된 알고리즘에 따라 워업을 수행하지 않고 ECS에서 가스터빈을 바로 제어한다.

Fig. 23과 같이 emergency start 신호가 가스터빈으로 전달되면 가스터빈은 수행하지 않고 공회전 속도로 상승하게 된다. 따라서 별도의 프로그램을 구현하지 않고 ECS에서 공회전 상태가 되면 가스터빈의 출력을 제어한다.

위 4가지 알고리즘을 하나의 프로그램으로 통합하여 가스터빈 제어기 모델을 구성하였다. Fig. 52는 가스터빈 제어기 모델 프로그램을 나타낸 것이다.

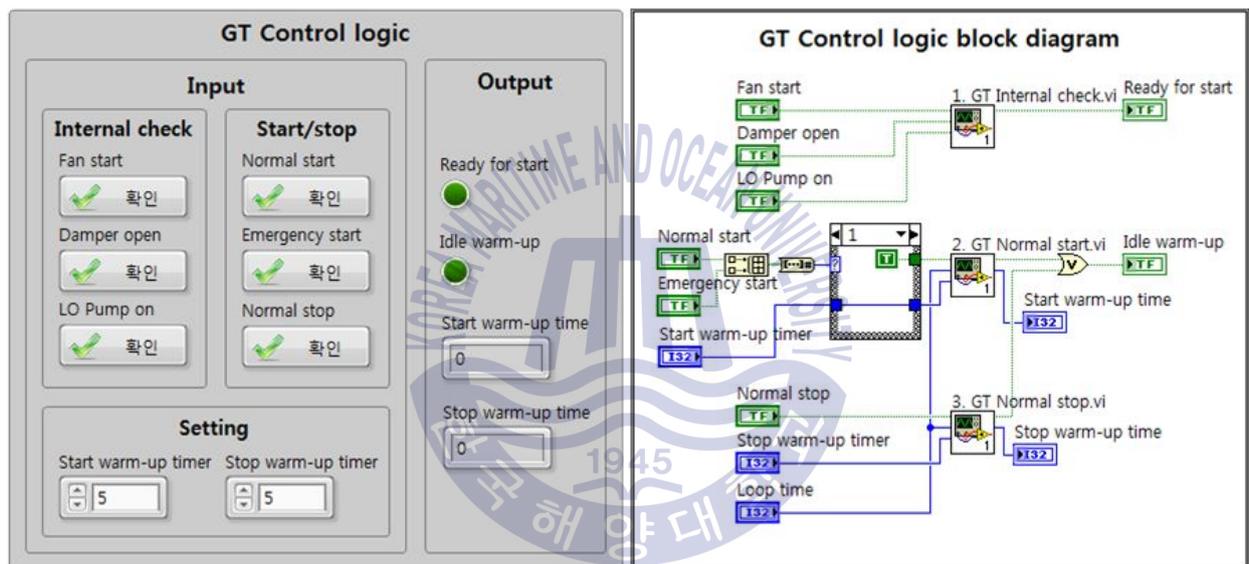


Fig. 52 Overall program for GT control logic

프로그램 제어 화면에 설정값은 가스터빈의 시동과 정지시 워업 시간을 설정하는 기능을 제공하여 시뮬레이터의 워업 시간을 사용자가 임의로 설정할 수 있도록 구성하였다.

(2) 감속기어

감속기어는 각 연동장비의 동작상태와 제어하고자 하는 목적에 따라 3가지 알고리즘으로 구분된다.

a. 내부 점검 알고리즘

감속기어를 기동하기 위해서는 윤활유 압력, 윤활유 펌프 동작 상태, 터닝 기어 분리 등 다양한 조건을 만족시켜야 한다. 이러한 인터락 조건은 ECS에서 제어하는 요소가 아니기 때문에 본 연구에서는 수동으로 신호를 제어할 수 있도록 구성하였다. Table 28과 Fig. 53은 내부 점검 알고리즘에 대한 신호와 프로그램을 나타낸 것이다.

Table 28 I/O signal for RG internal check

	Signal	Type	Equipment	Description
Input	L.O pressure	AI	Aux	pressure [Pa]
	L.O pump running	DI	Aux	0 : off, 1 : on
	Turning gear	DI	Aux	0 : disengaged 1 : engaged
Output	Gear ready	DO	RG	0 : not ready 1 : ready

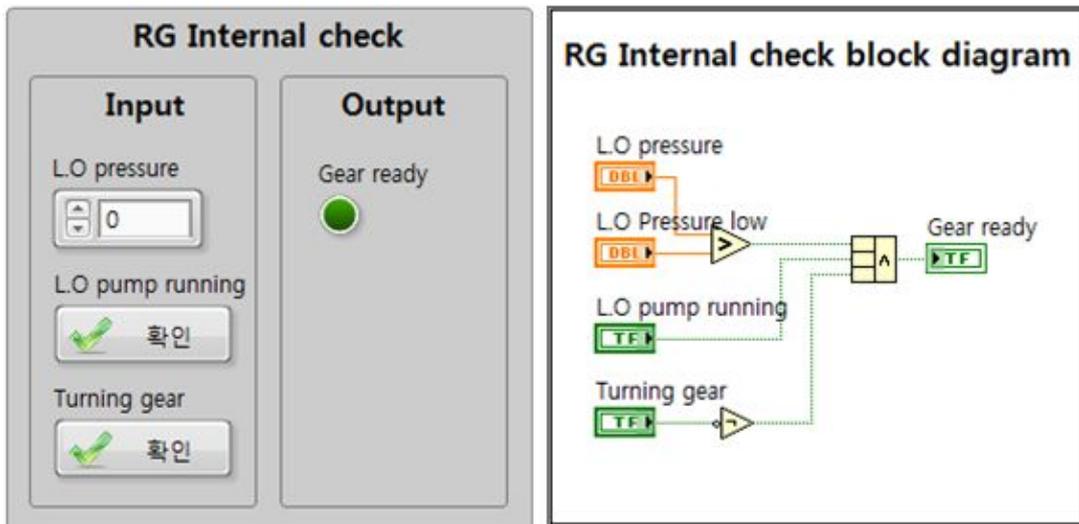


Fig. 53 Program for RG internal check

b. 브레이크 계통 제어 알고리즘

감속기어는 가스터빈 초기 기동 및 정지를 위해 브레이크를 사용한다. 브레이크 시스템은 유압을 이용하여 제어하며, 브레이크 시스템 제어를 위한 초기 설정을 하는 신호와 브레이크 시스템 동작을 위한 신호로 분류할 수 있다.

Table 29와 Fig. 54는 각각 브레이크 계통 제어를 위한 신호와 프로그램을 나타낸 것이다.

Table 29 I/O signal for RG brake system

	Signal	Type	Equipment	Description
Input	Hydraulic brake system	DI	ECS	0 : off, 1 : on
	Brake active	DI	ECS	0 : off, 1 : on
Output	Brake ready	DO	RG	0 : not ready 1 : ready
	Brake engaged	DO	RG	0 : disengaged 1 : engaged

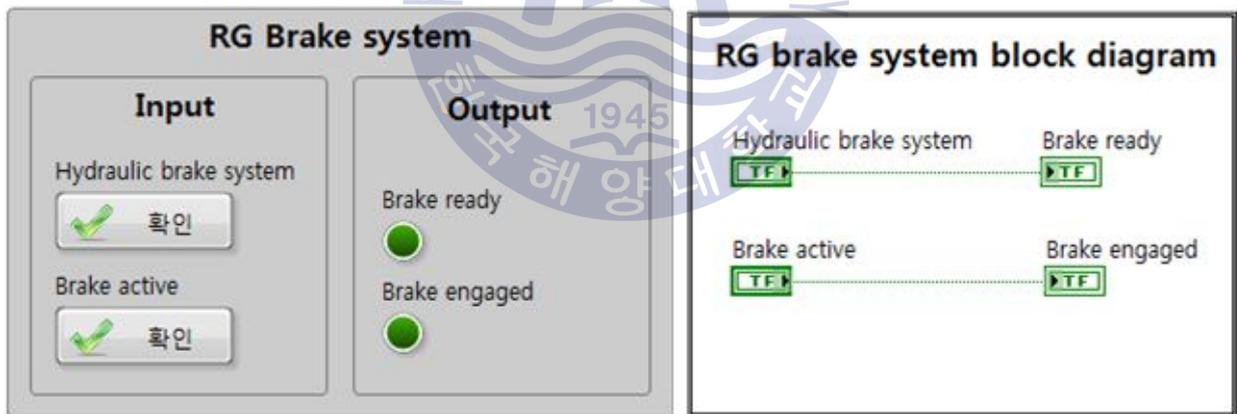


Fig. 54 Program for RG brake system

c. SSS 클러치 제어 알고리즘

감속기어에는 총 3개의 SSS 클러치가 설치되어 있으며 PORT, STBD에 설치된 SSS 클러치는 유압서보모터를 이용하여 제어할 수 있도록 구성이 되어 있다. SSS 클러치는 제어에 따라 ratchet, lockout, engage 상태가 있으며, 이러한

조건들을 고려하여 모델링하였다.

Table 30은 SSS 클러치 제어 알고리즘에 대한 신호를 정리한 것이다.

Table 30 I/O signal for RG SSS clutch control algorithm

	Signal	Type	Equipment	Description
Input	SB SSS clutch lockout	DI	ECS	0 : off, 1 : on
	PT SSS clutch lockout	DI	ECS	0 : off, 1 : on
	CC SSS clutch in speed	AI	GT	RPM
	CC SSS clutch out speed	AI	GT	RPM
	SB SSS clutch in speed	AI	SB PM	RPM
	SB SSS clutch out speed	AI	SB PM	RPM
	PT SSS clutch in speed	AI	PT PM	RPM
	PT SSS clutch out speed	AI	PT PM	RPM
Output	CC SSS clutch engaged	DO	RG	0 : disengaged 1 : engaged
	SB SSS clutch lockout	DO	RG	0 : non lock out 1 : lock out
	SB SSS clutch ratcheting	DO	RG	0 : non ratcheting 1 : ratcheting
	PT SSS clutch engaged	DO	RG	0 : disengaged 1 : engaged
	PT SSS clutch lockout	DO	RG	0 : non lock out 1 : lock out
	PT SSS clutch ratcheting	DO	RG	0 : non ratcheting 1 : ratcheting

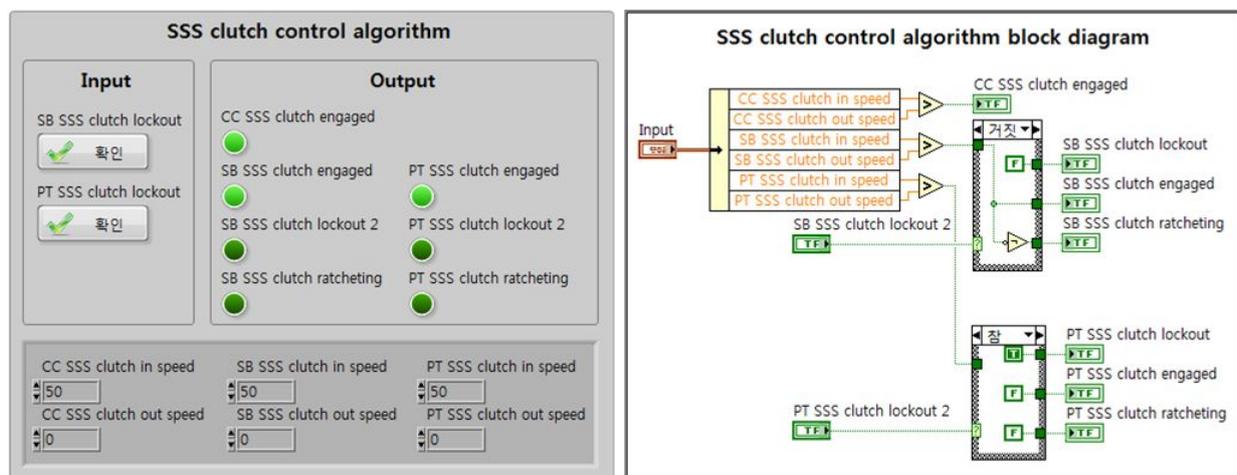


Fig. 55 Program for SSS clutch control algorithm

위 3가지 알고리즘을 하나의 프로그램으로 통합하여 감속기어 제어기 모델을 구성하였다. Fig. 56와 Fig. 57은 감속기어 제어기 모델 프로그램을 나타낸 것이다.

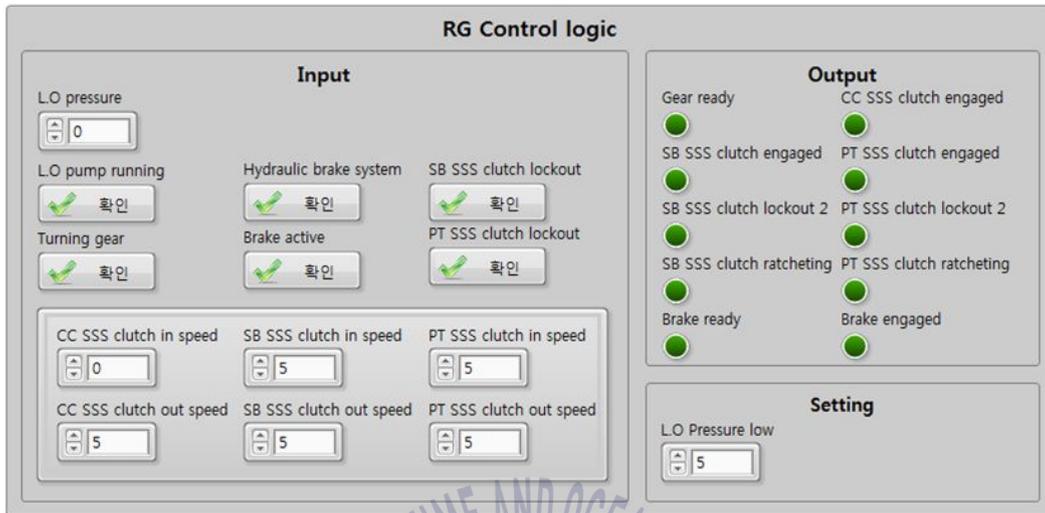


Fig. 56 Overall program for RG control logic (1)

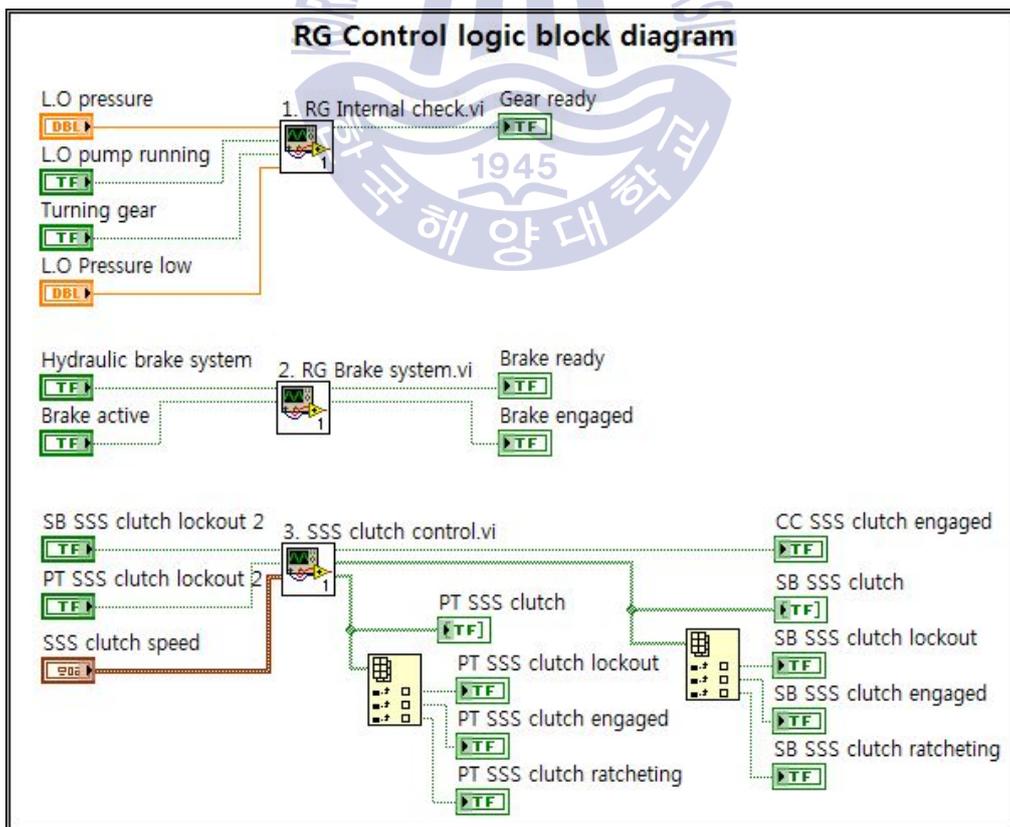


Fig. 57 Overall program for RG control logic (2)

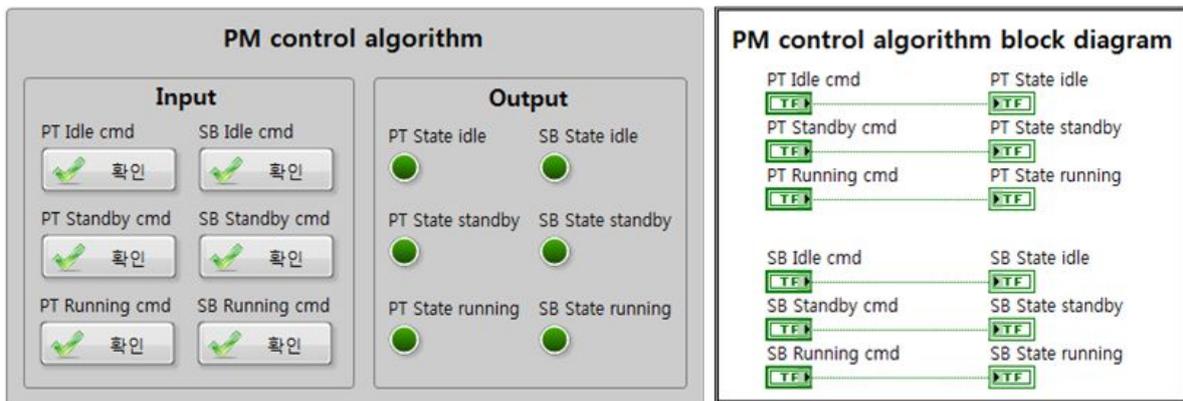
### (3) 추진전동기

추진전동기는 가스터빈이나 감속기어에 비해 보조기기가 적으며, 제어 알고리즘이 비교적 단순하다. 추진전동기는 ECS 명령에 따라 정지, 준비, 동작 3가지 상태를 가지고 제어를 수행한다.

정지 상태는 모든 보조기기가 정지하고 추진전동기도 정지되어 있는 상태이다. 준비 상태는 보조기기를 동작시키고 동작 상태로 넘어갈 수 있는 상태이며, 동작 상태는 모든 시스템이 정상이 되고 ECS를 이용하여 추진전동기를 제어할 수 있는 상태를 의미한다. Table 31과 Fig. 58은 각각 추진전동기 제어 알고리즘에 대한 신호와 프로그램을 나타낸 것이다.

**Table 31** I/O signal for PM control algorithm

	Signal	Type	Equipment	Description
Input	PT idle cmd	DI	ECS	0 : off, 1 : on
	PT standby cmd	DI	ECS	0 : off, 1 : on
	PT running cmd	DI	ECS	0 : off, 1 : on
	SB idle cmd	DI	ECS	0 : off, 1 : on
	SB standby cmd	DI	ECS	0 : off, 1 : on
	SB running cmd	DI	ECS	0 : off, 1 : on
Output	PT state idle	DO	PM	0 : off, 1 : on
	PT state standby	DO	PM	0 : off, 1 : on
	PT state running	DO	PM	0 : off, 1 : on
	SB state idle	DO	PM	0 : off, 1 : on
	SB state standby	DO	PM	0 : off, 1 : on
	SB state running	DO	PM	0 : off, 1 : on



**Fig. 58** Program for PM control algorithm

#### (4) 축계 및 고정피치 프로펠러

실제 축계 및 고정피치 프로펠러는 여러 가지 펌프 등의 제어 대상이 있을 것으로 예상되지만, 정확한 자료를 확보하기 어렵고, 함정 전체적인 추진체계 기준에서 크게 중요하지 않기 때문에 본 연구에서 축계 및 고정피치 프로펠러 제어 알고리즘은 제외한다.

#### (5) 연동장비 제어기 모델 프로그램 통합

각각의 연동장비 제어기 모델을 하나로 통합하여 제어 알고리즘 프로그램을 구성하였다. 각 데이터는 통신 또는 기본 모델의 연산 결과를 받아 각각의 연동장비 제어기 모델로 전달된다.

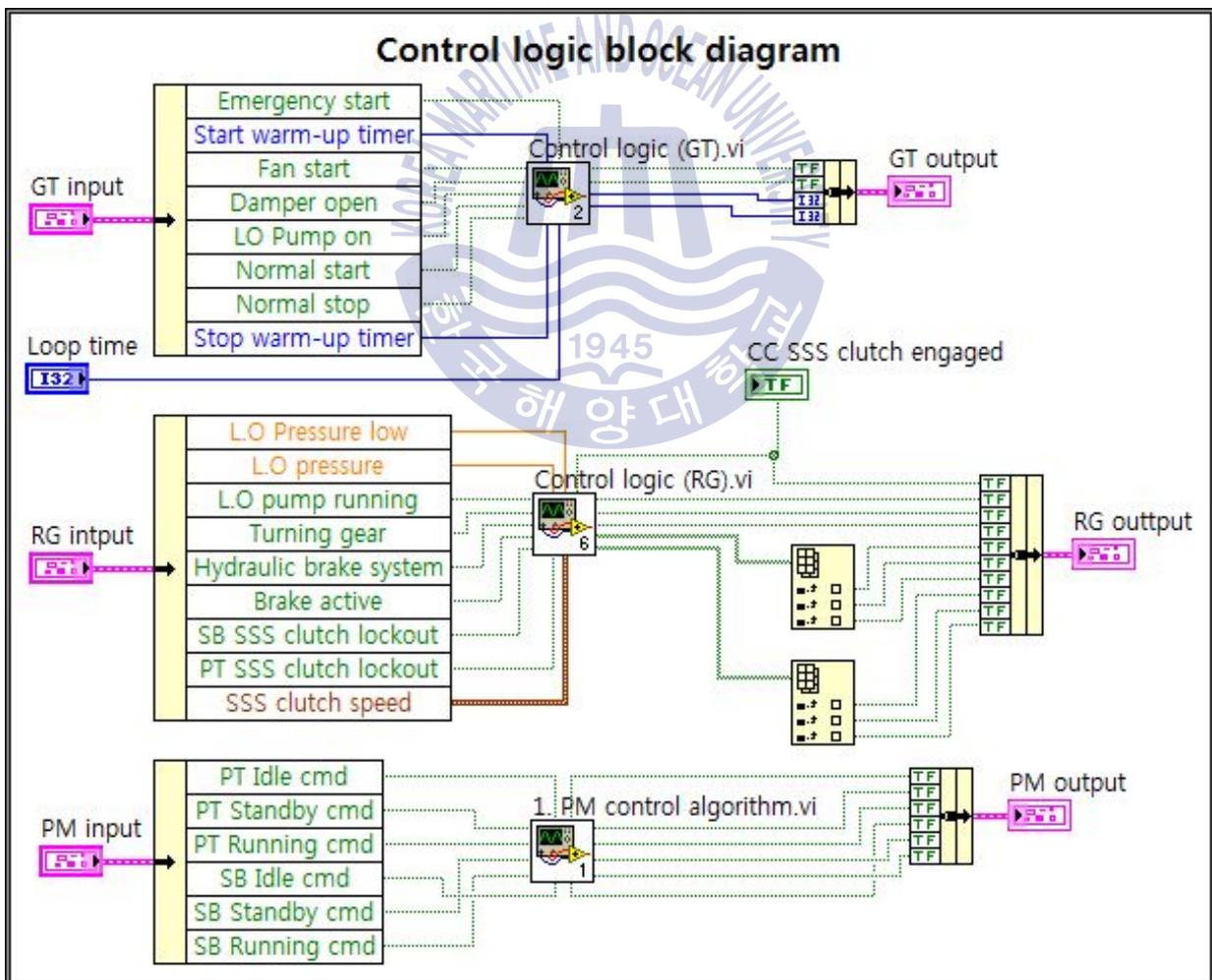


Fig. 59 Overall block program for control logic

### 4.4.3 기타 장비특성 모델

본 연구에서는 기본 모델과 연동장비 제어기 모델에서 각 연동장비의 기능을 모두 구현하였으므로 기타 장비특성 모델은 개발하지 않는다.

### 4.4.4 신호 입출력 모델

각각의 연동장비 제어기는 자신이 가지고 있는 데이터를 상위 제어기인 ECS에 전달하기 위해 여러 가지 통신 방식을 사용한다. 실제 함정 추진체계는 Ethernet, RS232, RS485, CAN 등 다양한 통신방식을 사용하지만 본 연구에서는 일반적으로 많이 사용하는 MODBUS TCP를 이용하여 통합한다.

MODBUS는 산업 현장에서 많이 사용되는 산업용 프로토콜로, 1970년대 Modicon이라는 회사에서 제안한 것이다. MODBUS는 데이터 전송방식에 따라 RTU, ASCII, TCP로 다시 구분되며 기본적으로 마스터-슬레이브 구조를 가지고 있다.

Fig. 60은 MODBUS 통신 구조를 나타내고 있다.

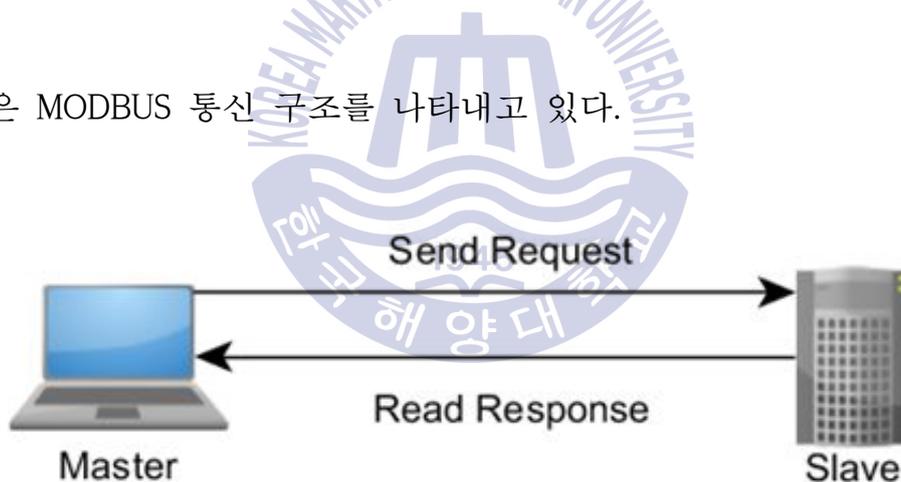


Fig. 60 MODBUS master-slave structure

마스터-슬레이브 구조에서는 마스터가 슬레이브에 요청을 보내고 응답을 기다린다. 이 구조를 통해 마스터는 정보의 흐름을 전적으로 컨트롤할 수 있으므로 네트워크 구성이 용이하다는 장점을 가진다.

MODBUS는 불리언 값과 부호 없는 16비트 정수를 지원하며, Table 32와 같은 메모리 블록을 가지고 있다.

Table 32 Concept of MODBUS memory block

Memory block	Type	Master	Slave
Coils	Boolean	Read/Write	Read/Write
Discrete inputs	Boolean	Read	Read/Write
Holding registers	Unsigned word	Read/Write	Read/Write
Input registers	Unsigned word	Read	Read/Write

MODBUS TCP 기능을 시뮬레이터에 구현하기 위해서는 가장 먼저 마스터와 슬레이브를 정의하여야 한다.

마스터-슬레이브 구조에서 주 제어기가 마스터 역할을 한다. 따라서 본 시뮬레이터에서는 ECS가 마스터가 되고, 시뮬레이터가 슬레이브가 되도록 프로그래밍한다. Table 33은 ECS와 시뮬레이터간 주고받는 신호를 정리하여 해당하는 메모리 블록에 할당한 것이다.

Table 33 Allocation of MODBUS memory block (1)

Name	Type	Memory block
GT power demand	AI	Hardwired
GT normal start	DI	Coils
GT normal stop	DI	Coils
GT emergency start	DI	Hardwired
GT turbine speed	AO	Input registers
GT torque	AO	Input registers
GT power	AO	Input registers
GT ready for start	DO	Coils
GT idle warm-up	DO	Coils
GT start warm-up timer	AO	Input registers
GT stop warm-up timer	AO	Input registers
GT fan start	DI	Coils
GT damper open	DI	Coils
GT LO pump on	DI	Coils
RG LO pump running	DI	Coils
RG turning gear	DI	Coils
RG hydraulic brake system	DI	Coils
RG brake active	DI	Coils
RG SSS clutch speed	AO	Input registers
RG SSS clutch engaged	DO	Coils

RG brake engaged	DO	Coils
RG brake ready	DO	Coils
RG gear ready	DO	Coils
RG SSS clutch lockout	DI	Coils
RG SSS clutch speed	AO	Input registers
RG SSS clutch engaged	DO	Coils
RG SSS clutch ratcheting	DO	Coils
RG SSS clutch locked out	DO	Coils
RG gear ready	DO	Coils
RG SSS clutch lockout	DI	Coils
RG SSS clutch speed	AO	Input registers
RG SSS clutch engaged	DO	Coils
RG SSS clutch ratcheting	DO	Coils
RG SSS clutch locked out	DO	Coils
RG gear ready	DO	Coils
PM speed	AI	Hardwired
PM standby cmd	DI	Coils
PM running cmd	DI	Coils
PM idle cmd	DI	Coils
PM shaft speed	AO	Input registers
PM shaft torque	AO	Input registers
PM shaft power	AO	Input registers
PM state standby	DO	Coils
PM state running	DO	Coils
PM state idle	DO	Coils
PM speed	AI	Hardwired
PM shaft torque	AO	Input registers
PM shaft power	AO	Input registers
PM standby cmd	DI	Coils
PM running cmd	DI	Coils
PM idle cmd	DI	Coils
PM shaft speed	AO	Input registers
PM state standby	DO	Coils
PM state running	DO	Coils
PM state idle	DO	Coils
FPP shaft torque	AO	Hardwired
FPP shaft RPM	AO	Input registers
FPP shaft power	AO	Input registers
FPP shaft torque	AO	Hardwired
FPP shaft RPM	AO	Input registers
FPP shaft power	AO	Input registers

앞의 데이터중 일부는 물리적 신호로 데이터를 전송하며, MODBUS TCP에 포함되지 않는다.

Table 34는 메모리 블록별로 데이터를 정리한 후, 주소를 할당하였다.

**Table 34** Address allocation of MODBUS memory block (2)

Memory block	Name	Address
Coils (Write)	Idle warm-up	00
	Ready for start	01
	Gear ready	02
	Brake ready	03
	Brake engaged	04
	CC SSS clutch engaged	05
	PT SSS clutch engaged	06
	PT SSS clutch lockout	07
	PT SSS clutch ratcheting	08
	SB SSS clutch engaged	09
	SB SSS clutch lockout	10
	SB SSS clutch ratcheting	11
	PT state idle	12
	PT state standby	13
	PT state running	14
	SB state idle	15
	SB state standby	16
SB state running	17	
Coils (Read)	Fan start	20
	Damper open	21
	LO pump on	22
	Normal start	23
	Normal stop	24
	LO pump running	25
	Turning gear	26
	Hydraulic brake system	27
	Brake active	28
	PT SSS clutch locked out	29
	SB SSS clutch locked out	30
	PT idle cmd	31
	PT standby cmd	32
	PT running cmd	33
SB idle cmd	34	

Input registers	SB standby cmd	35
	SB running cmd	36
	Start warm-up timer	00
	Stop warm-up timer	01
	Turbine speed	02
	GT torque	03
	GT power	04
	PT PM shaft speed	05
	PT PM shaft torque	06
	PT PM shaft power	07
	SB PM shaft speed	08
	PM shaft torque	09
	PM shaft power	10
	PT shaft RPM	11
	SB shaft power	12
	PT shaft RPM	13
	SB shaft power	14
CC SSS clutch speed	15	
PT SSS clutch speed	16	
SB SSS clutch speed	17	

LabVIEW에서 MODBUS TCP를 사용하는 방법은 I/O 서버를 이용하여 구축하는 방법과 API(Application Program Interface)를 이용하여 구축하는 방법이 있다. I/O 서버는 일반 공유 변수와 같이 바로 사용하면 되기 때문에 프로그램이 간단하다는 장점이 있으나, 실제 사용해본 결과 데이터 전송에서 손실이나 에러가 자주 발생하였다. 따라서, 본 연구에서는 API를 이용하여 구축한다. Fig. 61은 신호 입출력 모델 프로그램을 나타낸 것이다.

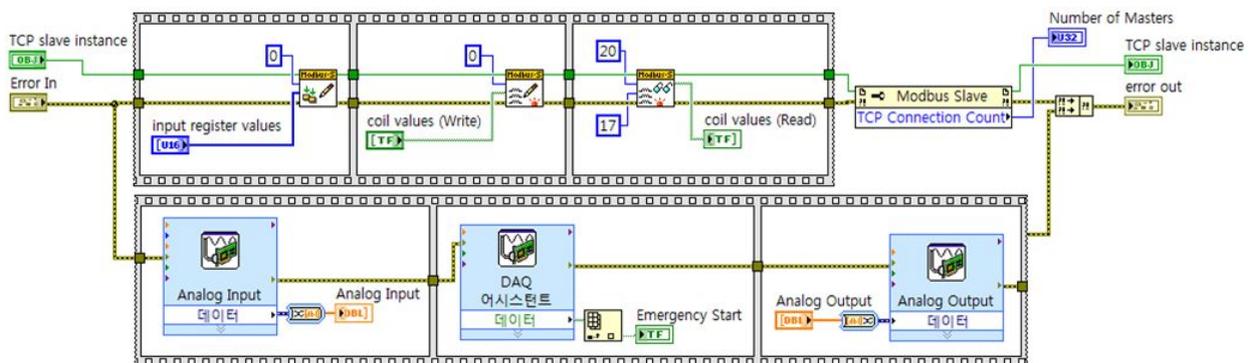


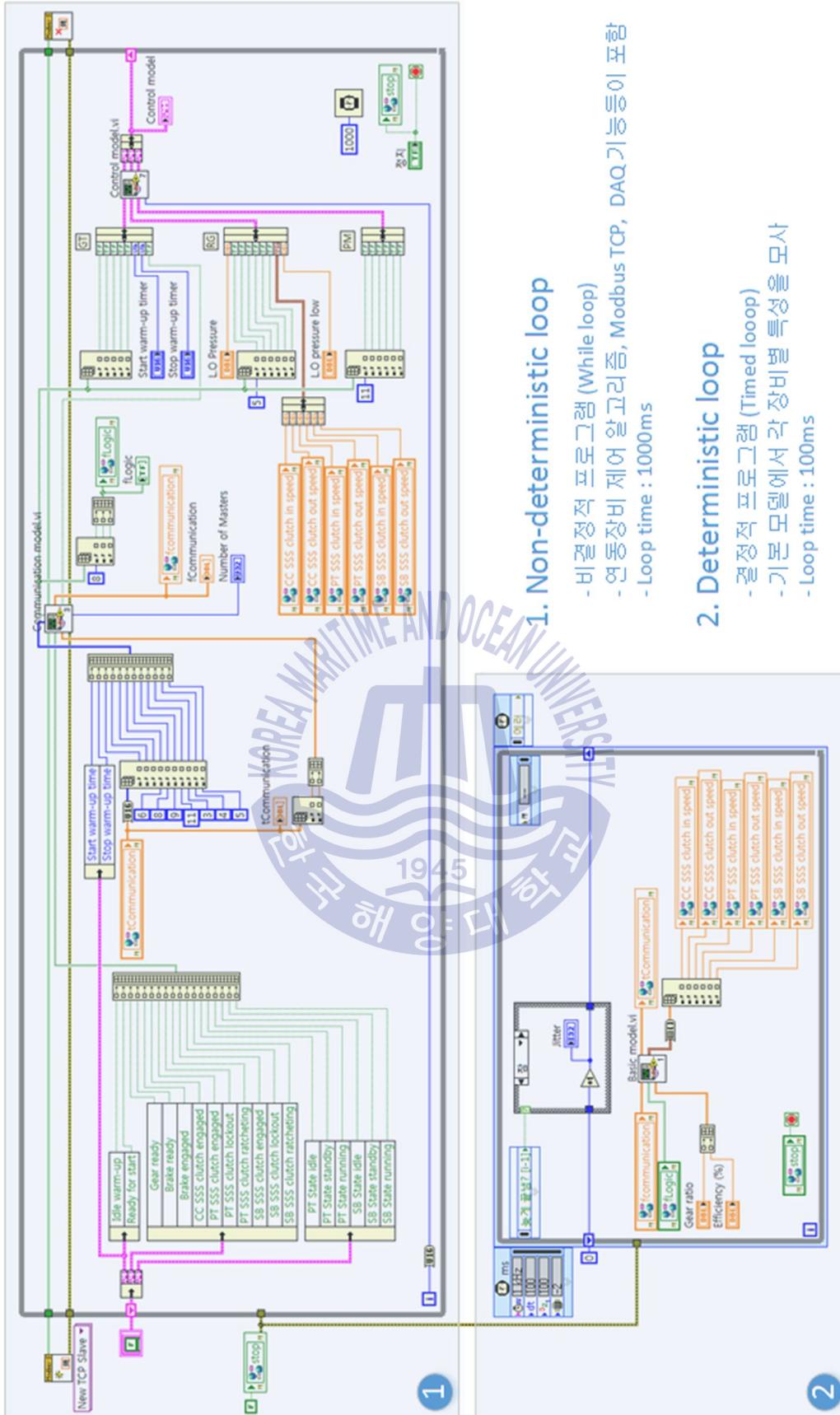
Fig. 61 Signal input/output model program

#### 4.4.5 시스템 통합

앞에서 개발한 기본 모델, 연동장비 제어기 모델, 통신 모델을 하나로 통합하여 중앙 시뮬레이터에서 구동할 수 있도록 구성한다. 이 과정에서 가장 중요한 부분은 프로그램 기능에 대한 우선순위 문제와 시간결정적 프로그램과 비시간 결정적 프로그램간의 데이터 전송에 대한 문제이다. Table 35는 기본 모델의 입출력 신호를 나타내고 있다.

Table 35 I/O signal for basic model

Name	Type	From
Power demand [%]	AI	Communication
PM PORT speed [%]	AI	Communication
PM STBD speed [%]	AI	Communication
Brake active	DI	Logic
PT SSS clutch ratcheting	DI	Logic
SB SSS clutch ratcheting	DI	Logic
Idle warm-up	DI	Logic
Turbine speed [rpm]	AO	Communication
GT torque [kNm]	AO	Communication
GT power [kW]	AO	Communication
PM PORT shaft speed [rpm]	AO	Communication
PM PORT shaft torque [kNm]	AO	Communication
PM PORT shaft power [kW]	AO	Communication
PM STBD shaft speed [rpm]	AO	Communication
PM STBD shaft torque [kNm]	AO	Communication
PM STBD shaft power [kW]	AO	Communication
RG CC SSS clutch speed [rpm]	AO	Communication
RG PT SSS clutch speed [rpm]	AO	Communication
RG SB SSS clutch speed [rpm]	AO	Communication
FPP PORT shaft speed [rpm]	AO	Communication
FPP PORT torque [Nm]	AO	Communication
FPP PORT shaft power [kW]	AO	Communication
FPP STBD shaft speed [rpm]	AO	Communication
FPP STBD torque [Nm]	AO	Communication
FPP STBD shaft power [kW]	AO	Communication
CC SSS clutch in speed [rpm]	AO	Logic
CC SSS clutch out speed [rpm]	AO	Logic
PT SSS clutch in speed [rpm]	AO	Logic
PT SSS clutch out speed [rpm]	AO	Logic
SB SSS clutch in speed [rpm]	AO	Logic
SB SSS clutch out speed [rpm]	AO	Logic



### 1. Non-deterministic loop

- 비결정적 프로그램 (While loop)
- 연동장비 제어 알고리즘, Modbus TCP, DAQ 기능등이 포함
- Loop time : 1000ms

### 2. Deterministic loop

- 결정적 프로그램 (Timed loop)
- 기본 모델에서 각 장비별 특성을 모사
- Loop time : 100ms

Fig. 62 Overall program for PXI

Fig. 62는 PXI에 들어가는 통합 프로그램을 나타낸 것이다.

①번 루프는 비결정적 프로그램이며, 일반 while 루프를 이용하여 구성하였다. 루프 딜레이 타임은 1초이며 연동장비 제어 알고리즘, MODBUS TCP, DAQ (hardwired signal in, out control)기능을 포함하고 있다.

②번 루프는 결정적 프로그램으로 timed 루프를 이용하여 1번 프로그램보다 높은 우선권을 부여함으로써 정시성을 확보하였다. 기본 모델에서 계산을 통해 각 연동장비의 특성을 모사하며, 루프 타임은 100 [ms]으로 설정하였다.

기본 모델의 정밀도는 단위시간( $\Delta t$ )이 작을수록 정밀도는 높아진다. 그러나 단위시간이 작아질수록 시뮬레이터의 CPU에 자원을 많이 차지하기 때문에 일정 수준이상 작게 하면 지터가 발생하며 시뮬레이션의 정밀도가 오히려 떨어지는 현상이 발생한다. 따라서 이와 같은 RT 시뮬레이션에서는 설계자가 테스트를 통해 적절한 단위시간을 정의하는 것이 중요하다.



## 제 5 장 함정 추진체계용 시뮬레이터 검증

### 5.1 가상의 ECS 구축

본 연구에서는 함정 추진체계용 시뮬레이터를 이용하여 ECS를 검증하기 위해 가상의 ECS를 구축한다. 가상의 ECS는 NI사의 cRIO와 DAQ 모듈을 이용하여 구축하며 중앙 시뮬레이터와 MODBUS 통신이 가능하도록 구성한다.

Table 36과 Fig. 63은 가상의 ECS로 활용한 NI사의 cRIO의 구성을 나타낸 것이다.

Table 36 Configuration of cRIO hardware

Name	EA	Description
NI cRIO-9074	1	cRIO, 400MHz, Spartan-32M, 128MB
NI 9215	1	$\pm 10V$ , Analog input, 4Ch
NI 9264	1	$\pm 10V$ , Analog output, 16Ch
NI 9401	1	5V, TTL Digital Input/Output, 32Ch



Fig. 63 cRIO & NI 9215, 9264, 9401

Fig. 64, 65는 각각 중앙 시뮬레이터의 PXI와 cRIO간 물리적 신호 결선도와 설치사진을 나타내며, Fig. 66, 67은 ECS 시뮬레이션 프로그램을 나타낸다.

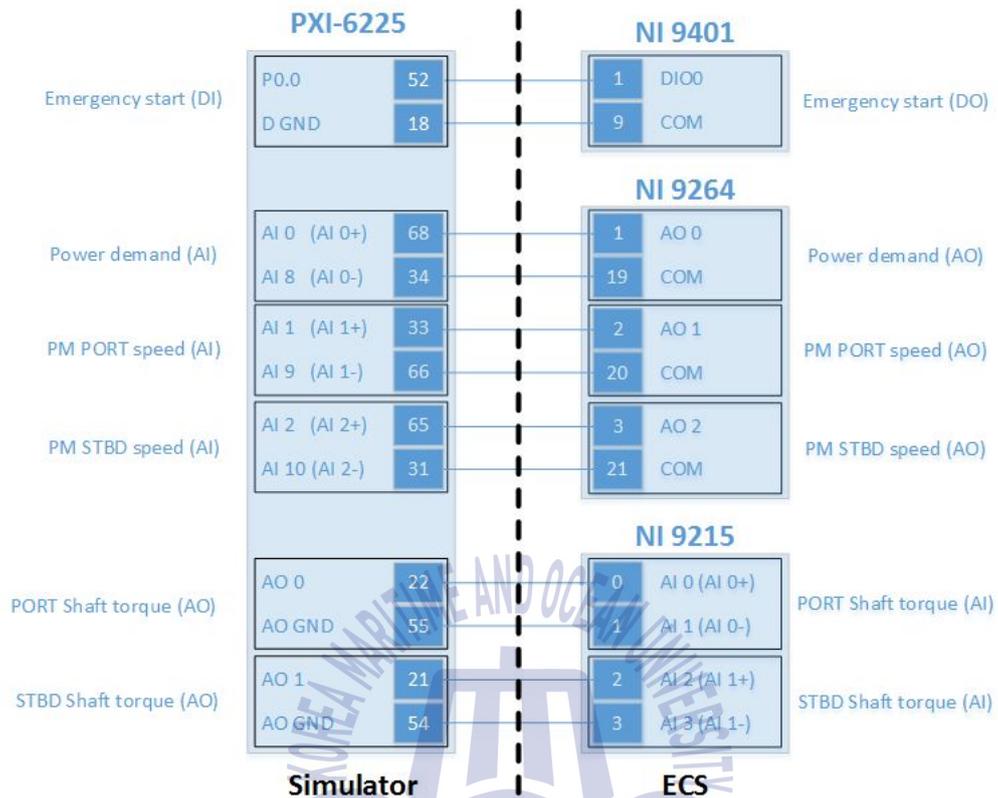


Fig. 64 Hardwired signal diagram between PXI and cRIO

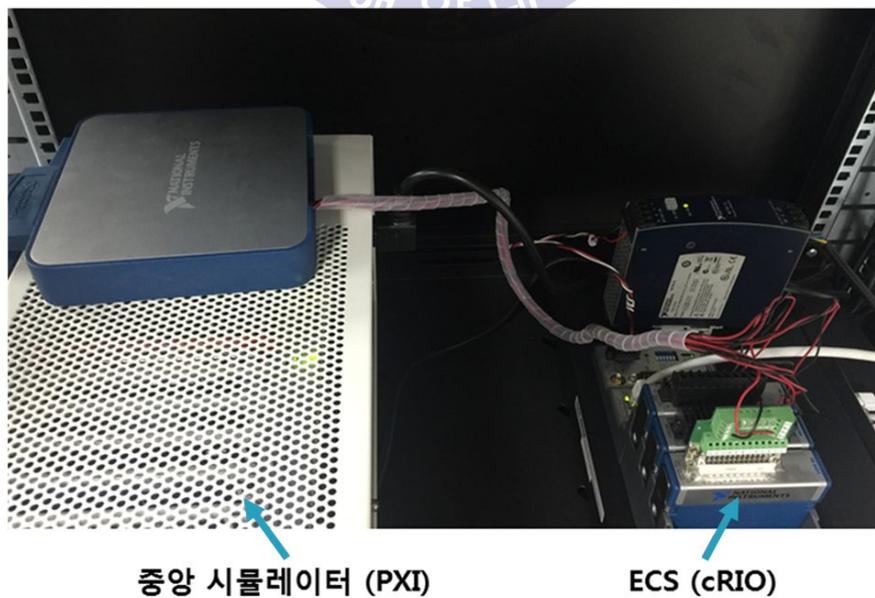


Fig. 65 Photo of PXI and cRIO

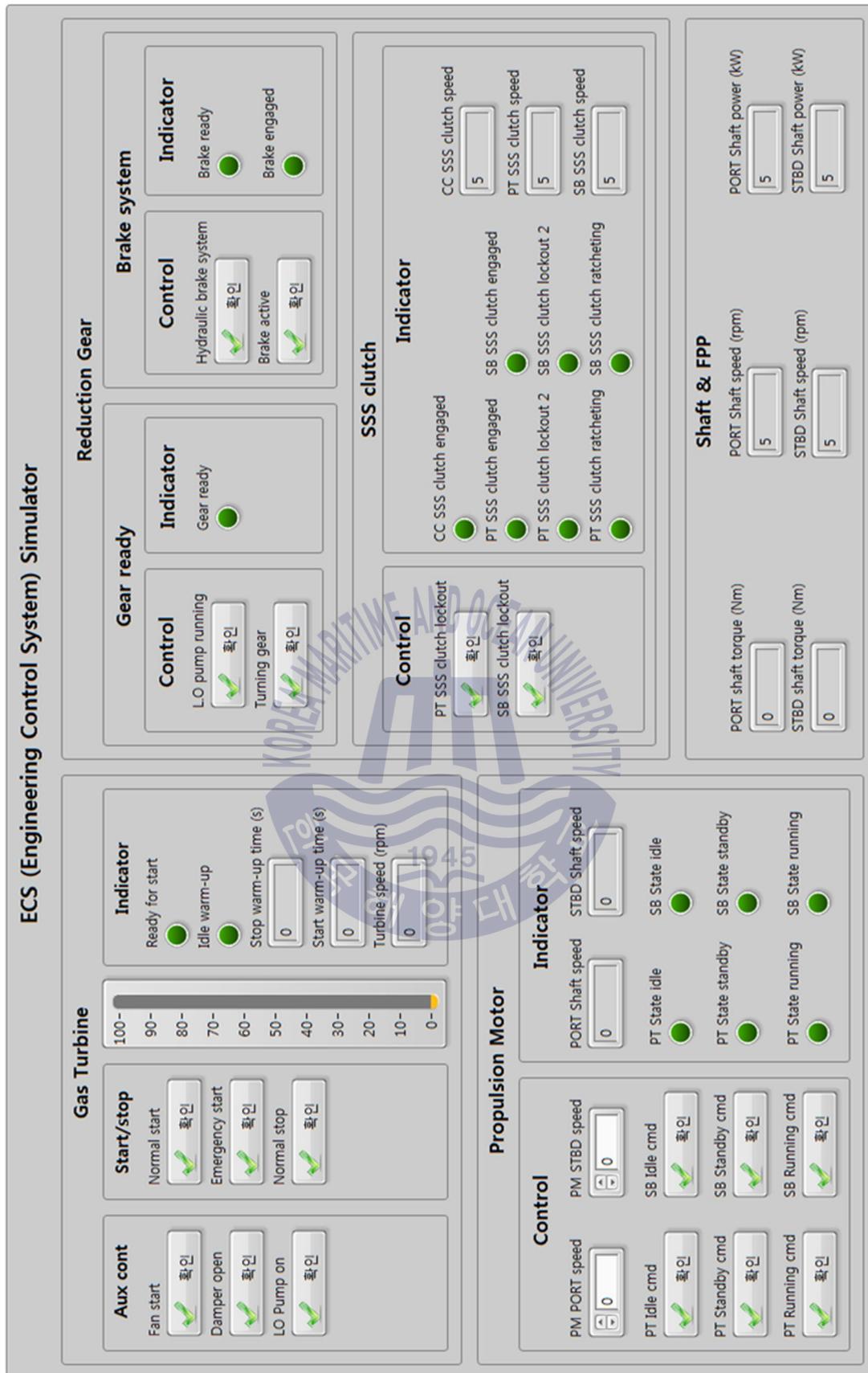


Fig. 66 Front panel for ECS simulation program

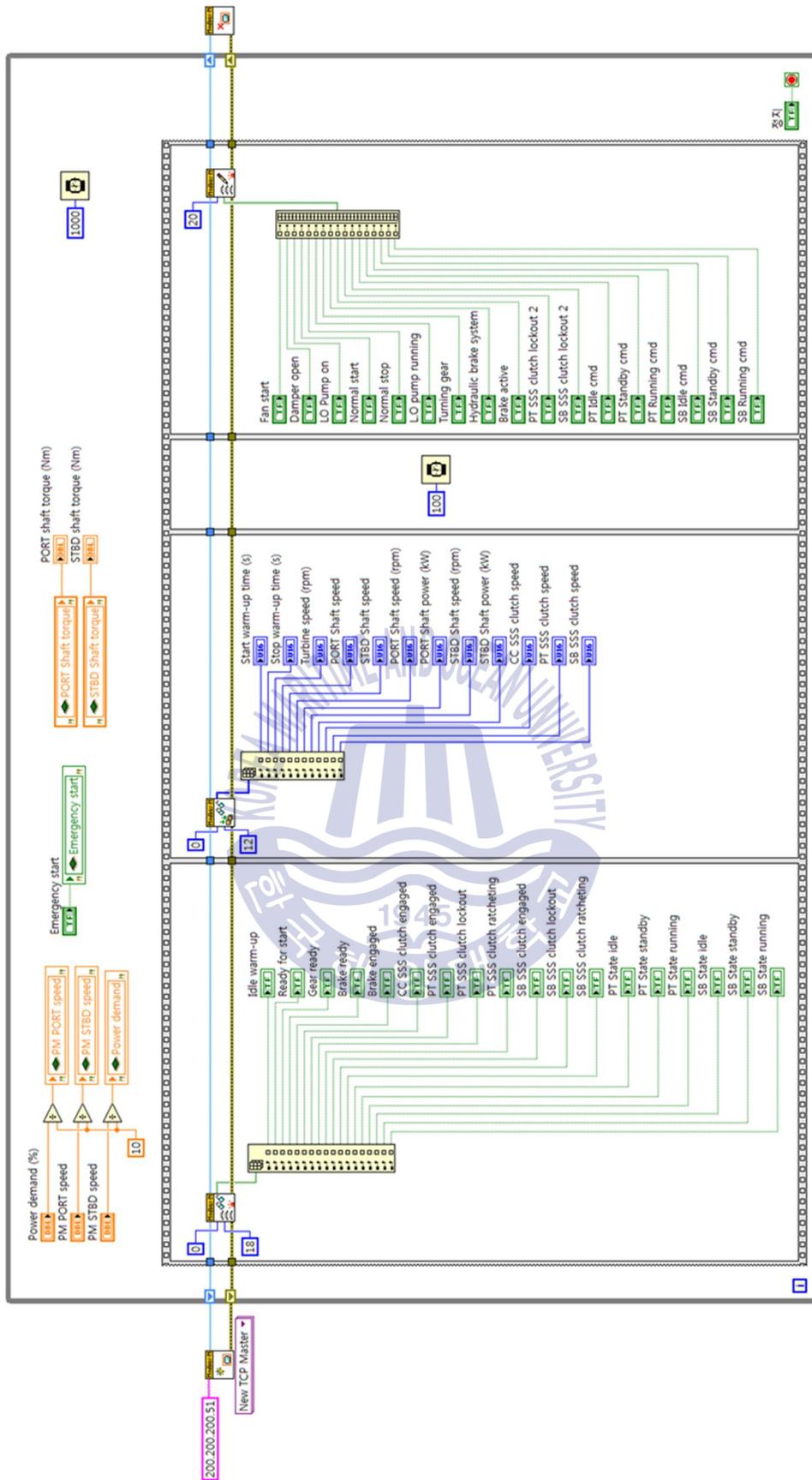


Fig. 67 Block diagram for ECS simulation program

## 5.2 가상의 ECS를 이용한 시뮬레이터 검증

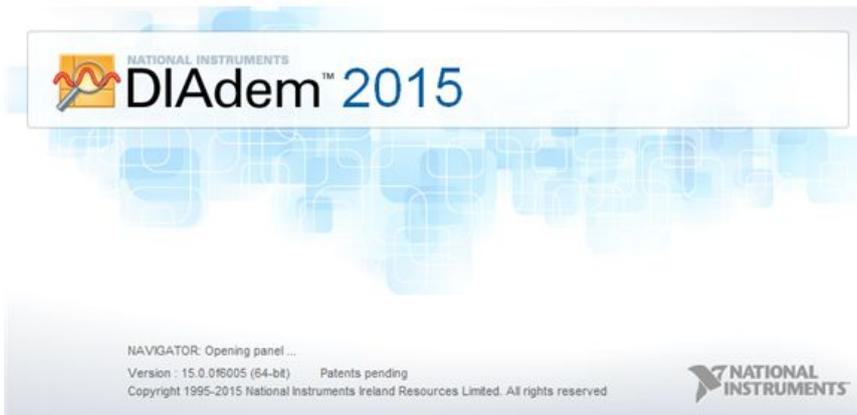
본 절에서는 ECS를 이용하여 3.3.2절에서 정의된 검증항목 알고리즘에 따라 ECS의 제어를 수행하고 그에 따른 시뮬레이터의 출력을 비교 분석한다, 또한, 시뮬레이터 출력을 앞에서 정의한 가상 함정의 예상되는 특성과 비교하여 시뮬레이터의 신뢰성에 대한 검토를 수행한다.

데이터 저장은 ECS에서 TCP/IP를 이용하여 PC로 송신후 TDMS(Technical Data Management Streaming) 포맷을 이용하여 시나리오당 하나의 파일로 저장 하였으며, 데이터 분석은 NI사의 데이터 분석 및 리포트용 소프트웨어인 DIAdem을 이용하여 처리한다.

Fig. 68은 데이터 저장과 DIAdem을 나타내고 있다.

1. GT start	2015-11-13 오전...	TDMS 파일	22KB
1. GT start.tdms_index	2015-11-14 오후...	TDMS_INDEX 파일	4KB
2. GT stop	2015-11-13 오전...	TDMS 파일	21KB
2. GT stop.tdms_index	2015-11-14 오후...	TDMS_INDEX 파일	4KB
3. PM start	2015-11-13 오전...	TDMS 파일	16KB
3. PM start.tdms_index	2015-11-14 오후...	TDMS_INDEX 파일	4KB
4. PM stop	2015-11-13 오전...	TDMS 파일	15KB
4. PM stop.tdms_index	2015-11-14 오후...	TDMS_INDEX 파일	4KB
5. GT to PM	2015-11-17 오전...	TDMS 파일	27KB
5. GT to PM.tdms_index	2015-11-14 오후...	TDMS_INDEX 파일	4KB
6. PM to GT	2015-11-13 오전...	TDMS 파일	28KB
6. PM to GT.tdms_index	2015-11-14 오후...	TDMS_INDEX 파일	4KB
7. Emergency stop	2015-11-13 오전...	TDMS 파일	12KB
7. Emergency stop.tdms_index	2015-11-16 오후...	TDMS_INDEX 파일	4KB

(a) TDMS files



(b) Diadem

Fig. 68 Data storage & DIAdem

(1) GT start

Fig. 69는 가스터빈 시동 시퀀스에 따른 가스터빈 동작 특성을 나타낸 것이다. 25초에 ECS에서 normal start 신호가 가스터빈으로 25초에 전달되었으며 가스터빈의 Start 시퀀스에 따라 워업을 20초간 진행한 것을 확인할 수 있다. 실제 MT30에서는 워업을 5분간 진행하지만 본 시뮬레이션에는 빠른 검증을 위해 40초로 단축 설정하여 진행하였다.

워업 종료후, 53초에 가스터빈 출력 명령을 100%로 상승하였으며 가스터빈은 이에 따라 회전속도가 일정비율로 상승하고 있는 것을 확인할 수 있다.

가스터빈 회전속도는 워업 상태에서 900 [rpm]을 나타내고 있으며, 가스터빈 출력 명령 100%에서는 3,600 [rpm]을 나타내고 있다. 토크와 출력은 워업에서 약 4 [Nm], 4 [kW]이며, 가스터빈 출력 명령 100%에서 97 [Nm], 36,756 [kW]를 내는 것을 확인할 수 있다.

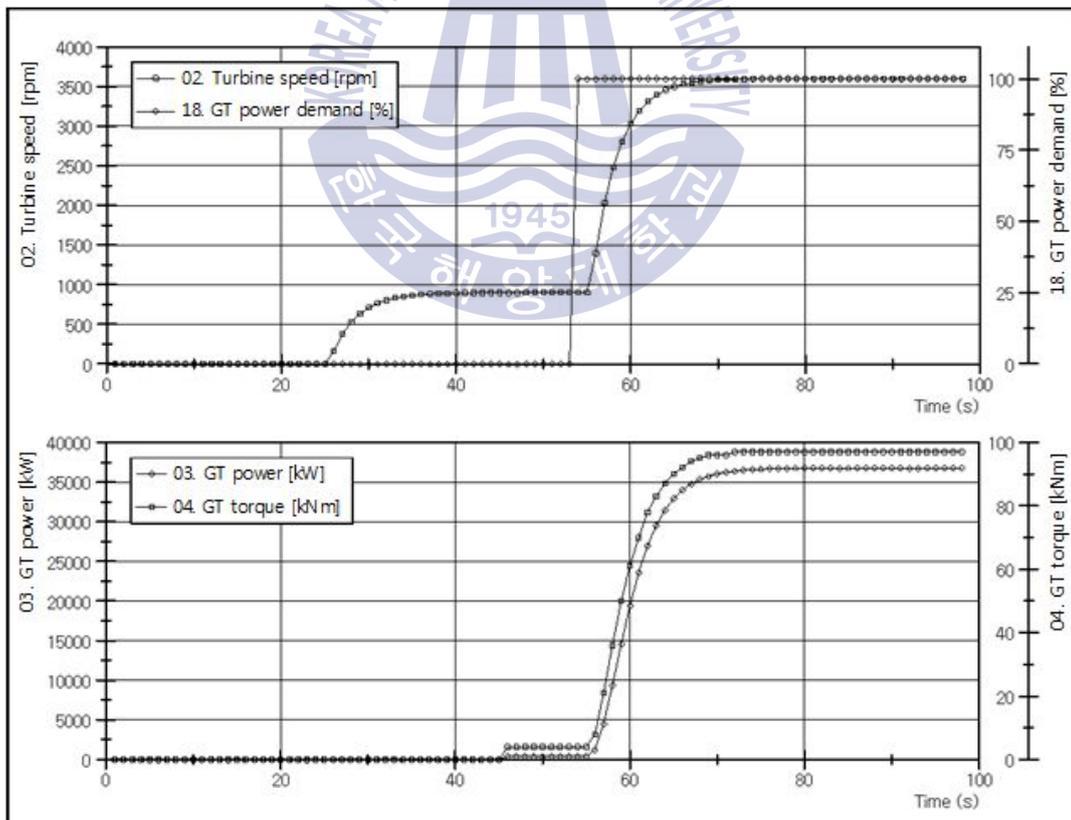


Fig. 69 GT start simulation output - GT

Fig. 70은 가스터빈 시동 시퀀스에 따른 PORT와 STBD 축계의 토크와 감속기어의 SSS 클러치 회전속도 변화 데이터를 나타낸 것이다.

PORT, STBD의 축 토크, 회전속도 데이터를 분석해보면, 0초에서 45초까지는 0의 값을 가지고 있으며, 45초부터는 일정값을 가지는 것을 확인할 수 있다. 이것은 가스터빈 시동시 감속기어에서 브레이크를 이용하여 축을 회전하지 못하도록 정지시키고 있으며, 45초에 브레이크를 해제하였기 때문에 나타나는 현상이다.

또한, 축 토크의 경우, 약간의 노이즈가 발생한 것을 확인할 수 있는데, 이것은 축 토크의 데이터는 다른 신호와 다르게 아날로그 전압 신호를 이용하여 전달하였기 때문에 발생하는 현상으로 실제 시뮬레이터 모델의 연산결과라 보기 어렵다.

CC SSS 클러치 회전속도와 PORT, STBD SSS 클러치 회전속도는 감속기어의 기어비에 따라 감속되어 전달되고 있는 것을 확인할 수 있다.

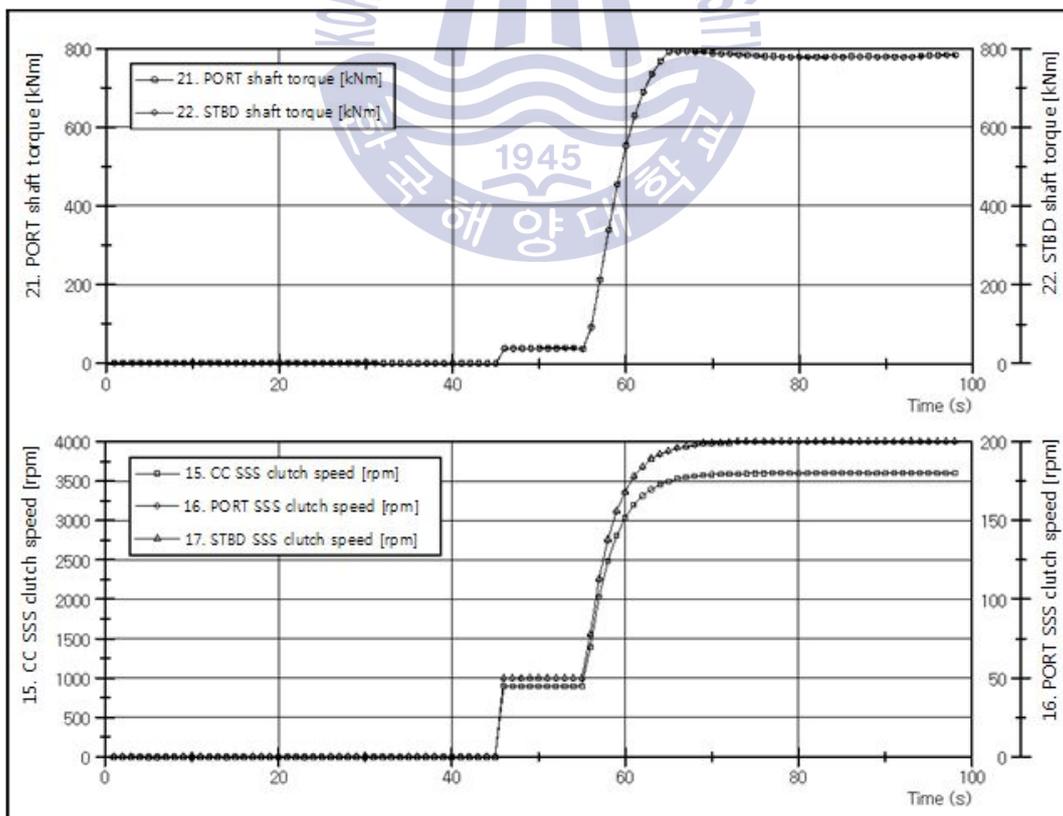


Fig. 70 GT start simulation output - shaft and clutch

Fig. 71은 추진전동기와 관련된 데이터와 FPP의 회전속도, 출력을 나타내고 있다.

가스터빈 시동 시퀀스의 경우, 추진전동기는 동작을 하지 않고 주 전원과 절연상태를 유지하고 있기 때문에 ECS 명령값, 추진전동기 회전속도 모두 0를 나타내고 있다.

FPP는 가스터빈에서 감속기어를 거쳐서 전달된 회전속도와 각각의 축에 대한 출력이 나타나고 있다. 각 FPP별 회전속도는 워업에서 50 [rpm], 가스터빈 출력 명령 100%에서 200 [rpm]을 나타내고 있으며, 출력은 워업에서 199 [kW], 가스터빈 출력 명령 100%에서 18,011 [kW]를 나타내고 있다.

FPP 관련 그래프 또한 브레이크가 해제되는 시점인 45초부터 상승하고 있음을 확인할 수 있다.

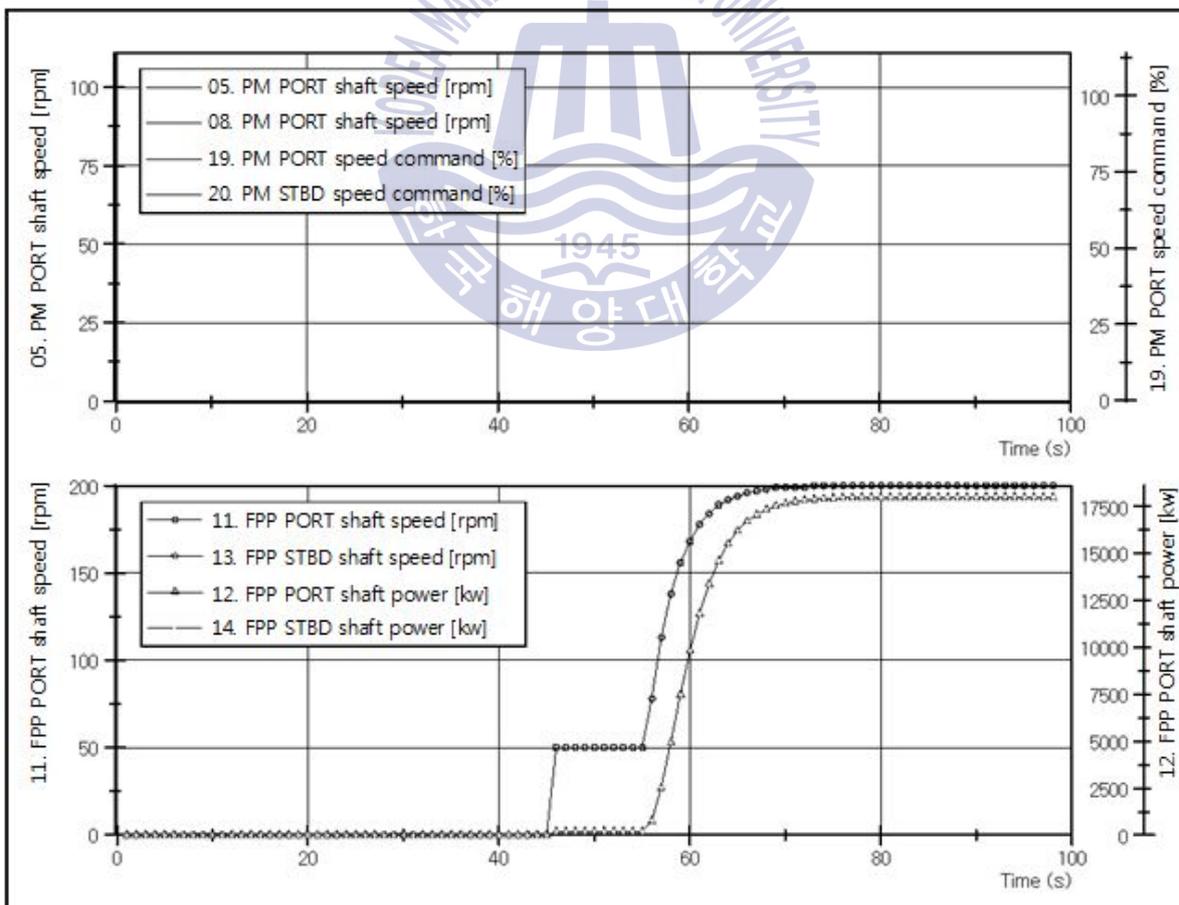


Fig. 71 GT start simulation output - PM and FPP

## (2) GT stop

Fig. 72는 가스터빈 정지 시퀀스에 따른 가스터빈 동작 특성을 나타낸 것이다. 가스터빈 정지 시퀀스는 사전에 정의된 절차에 따라 가스터빈 출력 명령을 0%로 감소시킨다. Fig. 72를 보면 11초에 ECS에서 가스터빈 출력 명령을 0%로 감소시키는 명령을 내렸고, 가스터빈이 공회전 속도에 도착하자 정지 신호를 전달하여 율업 상태로 전환한 것을 확인할 수 있다.

율업 상태로 전환된 시점이 정확하게 나타나 있지는 않지만, 공회전 속도에서 감소하기 시작한 시점이 63초인 것으로 보아 약 43초부터 율업이 시작되었음을 유추할 수 있다.

가스터빈 회전속도, 토크, 출력 등은 가스터빈 시동 시퀀스의 데이터와 동일하다.

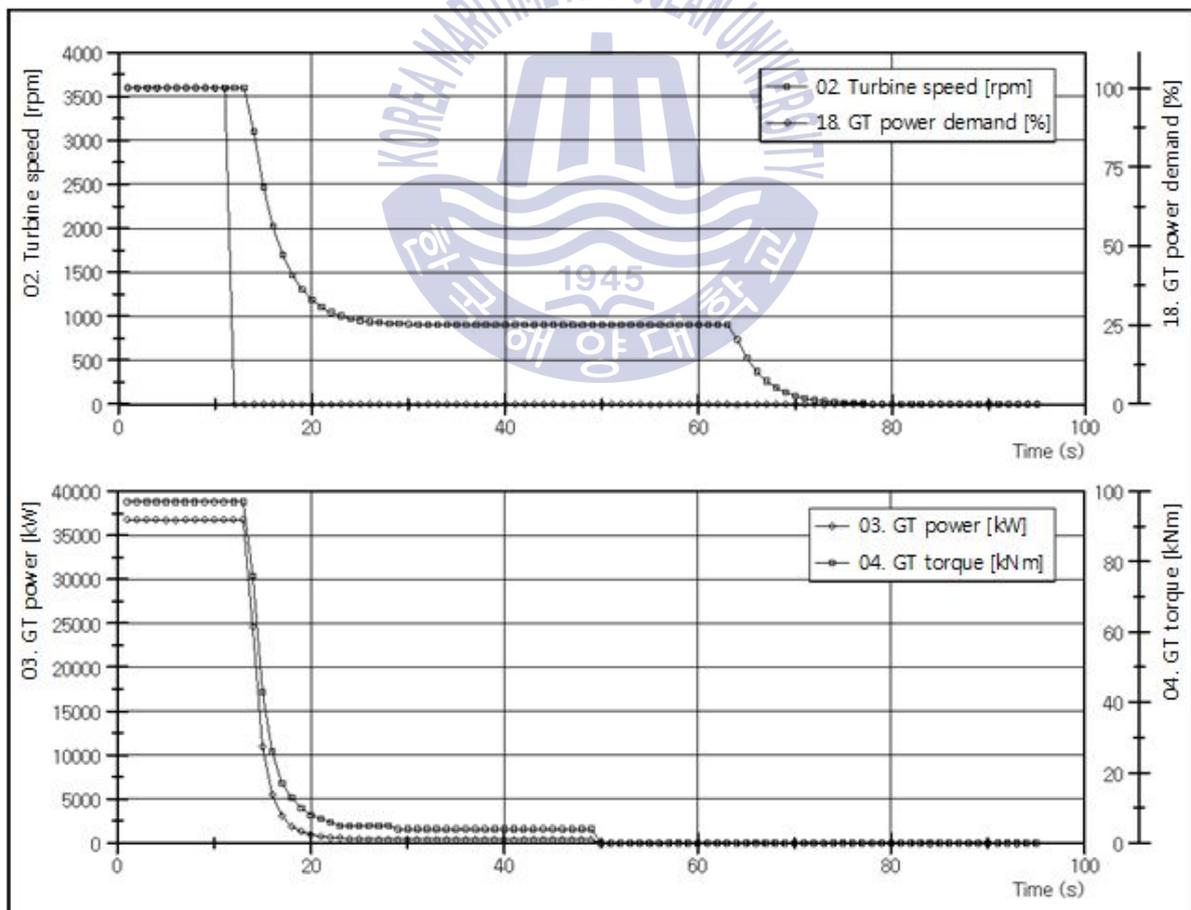


Fig. 72 GT stop simulation output - GT

Fig. 73과 Fig. 74는 가스터빈 정지 시퀀스에 따른 PORT와 STBD 축계 토크와 감속기어의 SSS 클러치 회전속도 변화, 그리고 추진전동기, FPP 관련 데이터를 나타낸 것이다.

가스터빈 정지 시퀀스는 시동 시퀀스와 동일하게 워업 에서 일정 시점에 브레이크를 이용하여 가스터빈의 축을 정지하도록 한다.

CC SSS 클러치 회전속도와 PT, SB SSS 클러치 회전속도를 보면 43초부터 워업을 수행하였으며, 48초에 브레이크가 동작하여 가스터빈 이후 감속기어, 축, FPP를 정지시켰음을 확인할 수 있다.

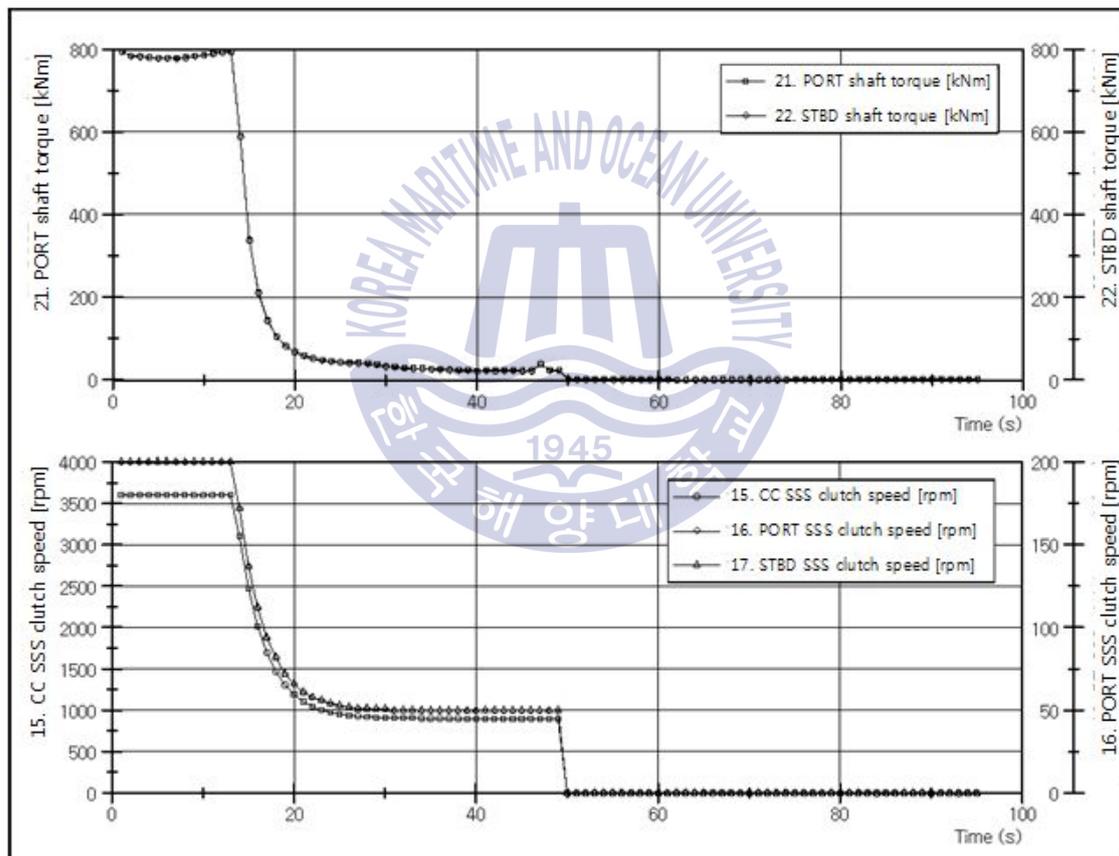


Fig. 73 GT stop simulation output - shaft and clutch

추진전동기는 동작하지 않기 때문에 아래 Fig. 74와 같이 회전속도가 0 [rpm]을 나타내고 있으며, FPP는 브레이크 동작에 따라 회전과 출력이 발생하고 있음을 확인할 수 있다.

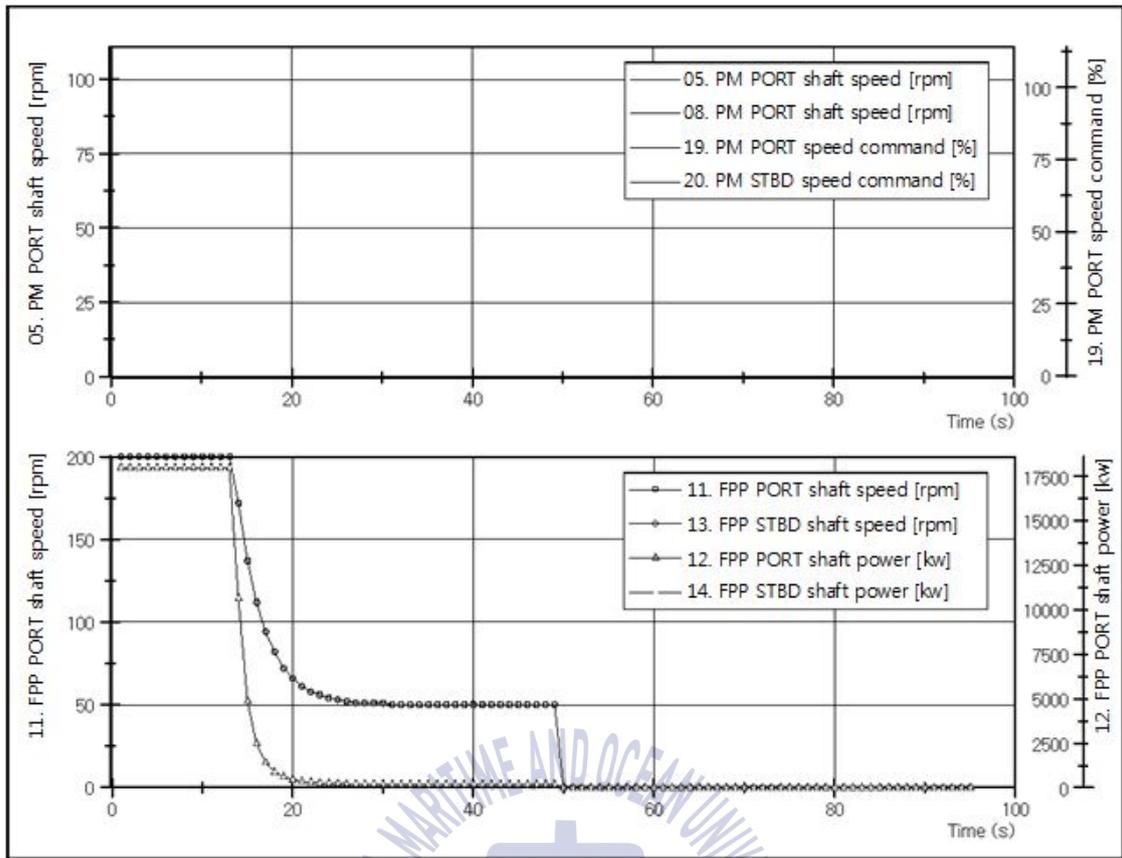
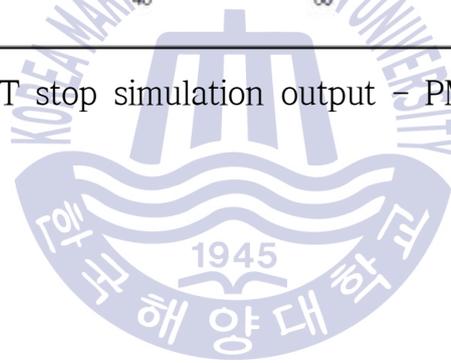


Fig. 74 GT stop simulation output - PM and FPP



### (3) PM start

추진전동기 시동 시퀀스는 가스터빈과 비교했을 때 단순한 편이다. 추진전동기 정지 상태일 때 ECS에서 준비 명령을 추진전동기로 전달한 후, 추진전동기가 준비 상태가 되면 PORT, STBD 추진전동기 속도 제어 명령을 이용하여 회전속도를 제어한다.

Fig. 75는 가스터빈의 상태를 나타내며, CC, PORT, STBD SSS 클러치가 lock out 상태이기 때문에 회전속도가 0 [rpm]으로 나타난다.

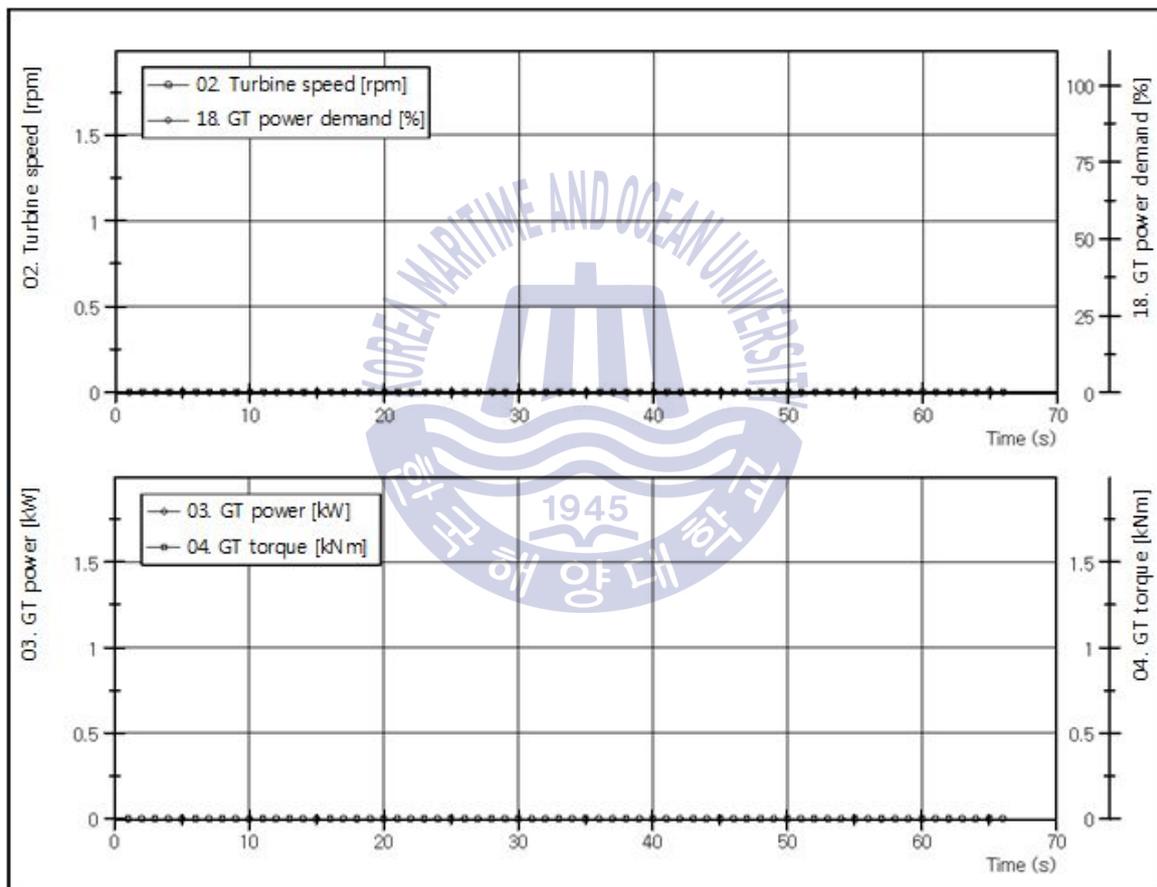


Fig. 75 PM start simulation output - GT

Fig. 76과 Fig. 77을 보면, 추진전동기 속도 제어 명령에 따라 회전속도가 제어되며, 명령값이 100%일 때, PORT, STBD 각각 회전속도 100 [rpm], 토크 164 [kNm], 출력 1,028 [kW]를 나타내는 것을 확인할 수 있다.

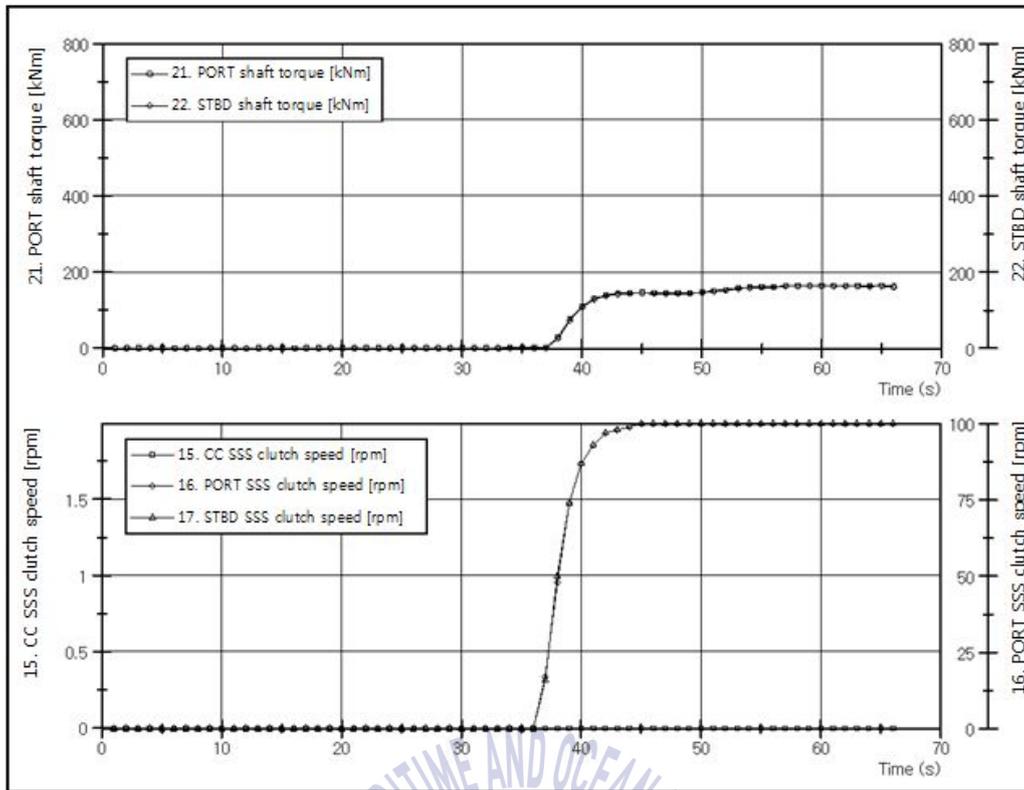


Fig. 76 PM start simulation output - shaft and clutch

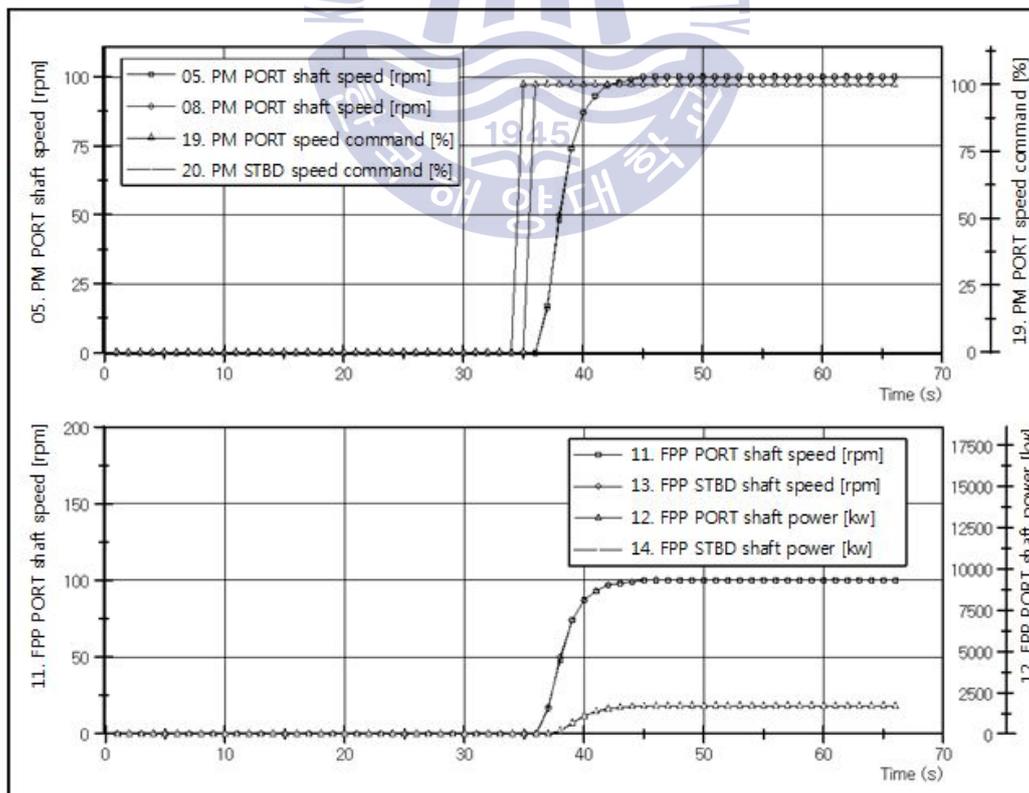


Fig. 77 PM start simulation output - PM and FPP

#### (4) PM stop

추진전동기 정지 시퀀스는 시동 시퀀스와 거의 동일하게 동작한다.

ECS에서 추진전동기 속도 명령(PM speed command)을 0%으로 낮추고 준비 명령을 추진전동기로 전달한 후, 추진전동기가 준비 상태가 되면 정지 명령을 추진전동기로 전달하여 정지한다.

Fig. 78, Fig. 79, Fig. 80은 추진전동기 정지 시퀀스에 따른 가스터빈, 감속기 어, 추진전동기, FPP에 대한 데이터를 시간에 따라 나타낸 것이다.

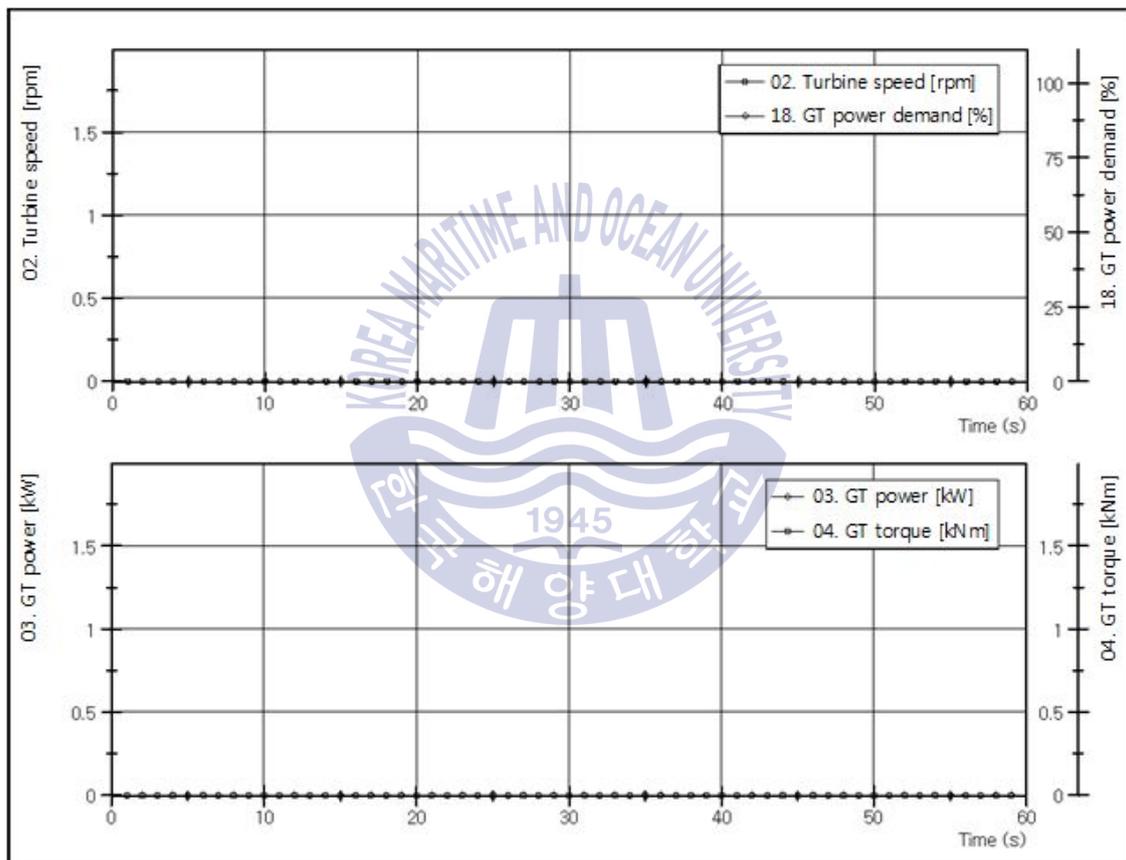


Fig. 78 PM stop simulation output - GT

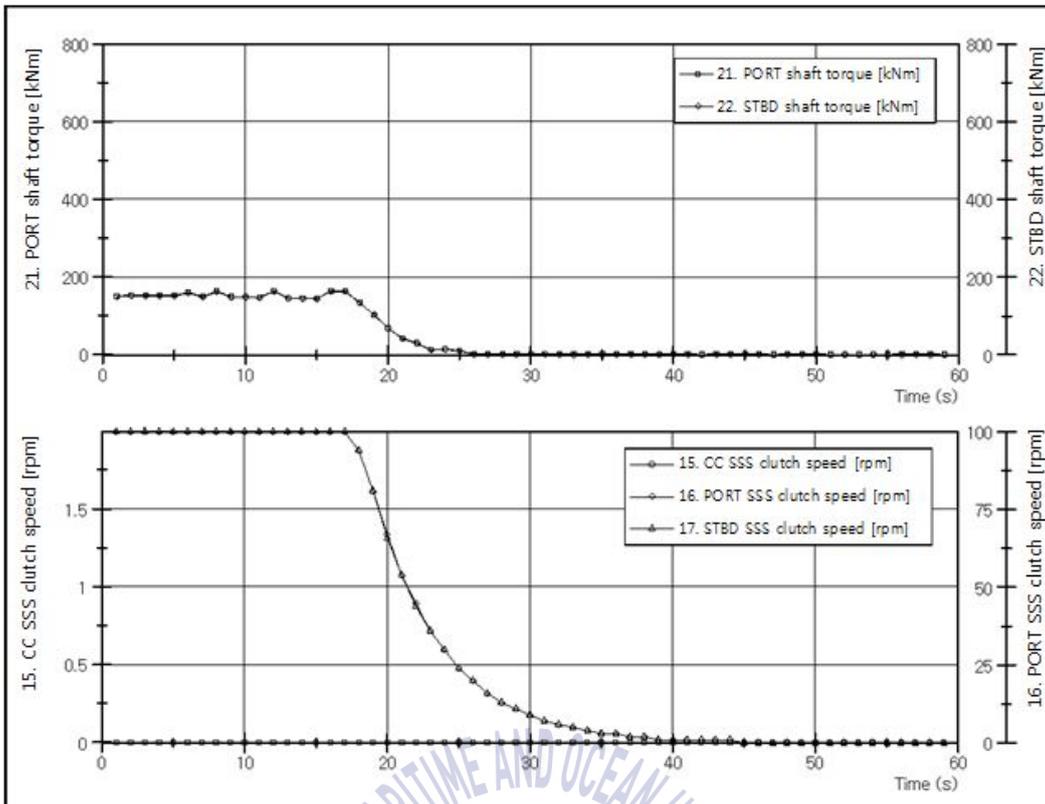


Fig. 79 PM stop simulation output - shaft and clutch

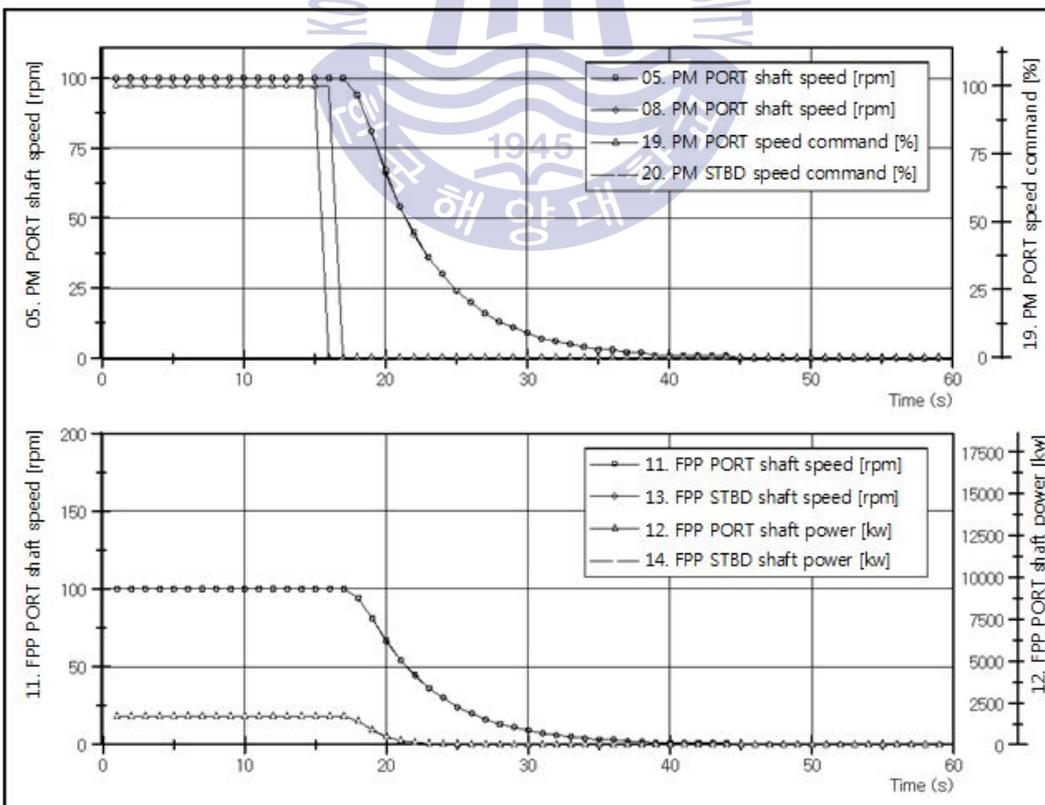


Fig. 80 PM stop simulation output - PM and FPP

(5) GT to PM changeover

가스터빈에서 추진전동기로 전환을 위해서는 추진전동기의 회전속도가 가스터빈의 회전속도보다 높아서 SSS 클러치의 연결이 끊어져야 한다. 따라서 처음에는 가스터빈의 회전속도를 감소시키고, 추진전동기의 회전속도를 증가시킨후, 적절한 시점이 되면 부하를 가스터빈에서 추진전동기로 전달하고, 가스터빈을 정지시키도록 제어한다.

Fig. 81을 보면, 9초에 가스터빈 출력 명령이 0%가 됨에 따라 가스터빈의 회전속도가 공회전 속도까지 감소하였고, 전환 절차에 따라 추진전동기로 부하를 전달한 후, 율입을 수행하고 77초부터 정지하는 것을 볼 수 있다.

가스터빈 출력과 토크를 보면 42초부터 0[rpm]이 되는 것을 확인할 수 있는데, 이것은 추진전동기의 회전속도가 가스터빈의 회전속도보다 높아짐으로 인해 SSS 클러치 연결이 끊어졌기 때문이다.

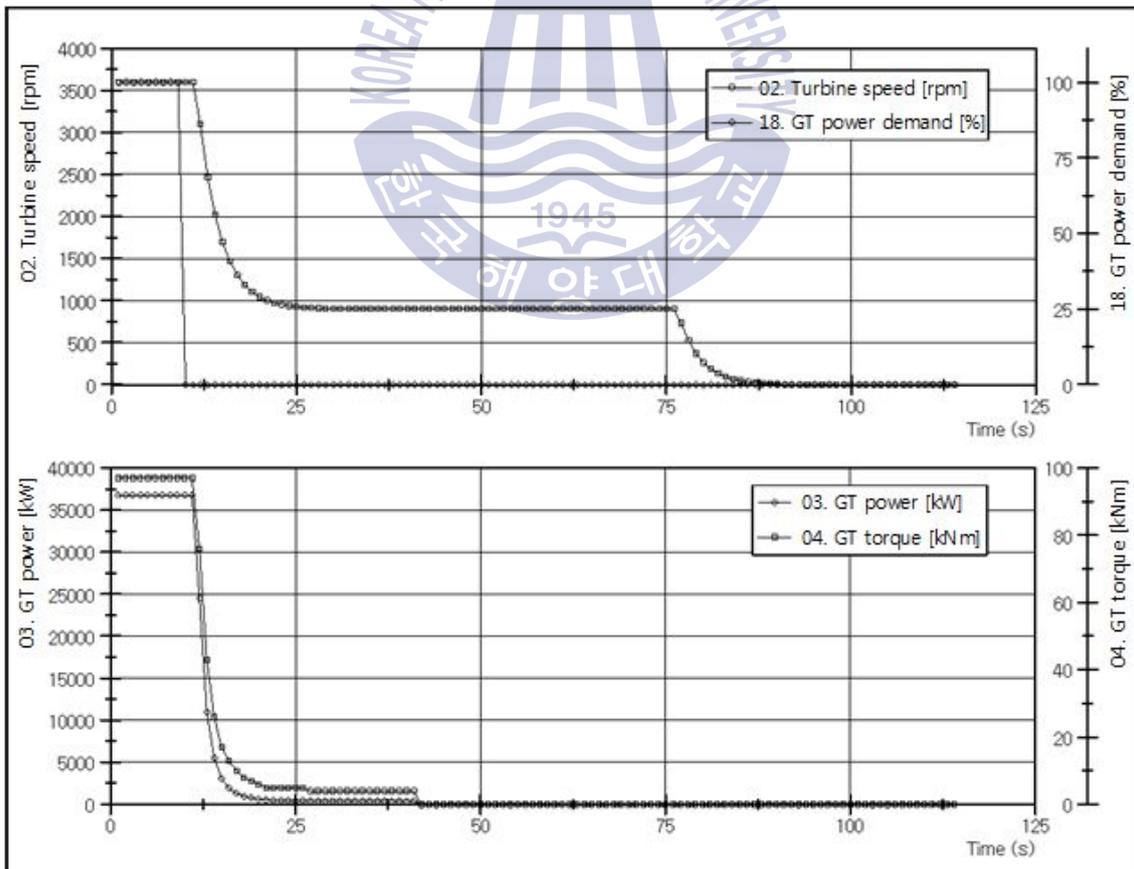


Fig. 81 GT to PM changeover simulation output - GT

Fig. 82는 가스터빈, 추진전동기간 절환절차에 따른 축 토크와 SSS 클러치 회전속도를 나타낸 것이다.

우선 SSS 클러치 회전속도를 보면, 가스터빈 출력 축과 연결되어 있는 CC SSS 클러치는 64초까지 가스터빈과 동일한 그래프를 가지고 있으나, 브레이크가 동작하는 64초 이후부터는 회전속도가 0이 됨을 확인할 수 있다. PORT, STBD SSS 클러치는 추진전동기 속도가 증가함에 따라 서서히 증가하며 축 토크가 약간 변화하는 42초에서 가스터빈 추진전동기간에 전환이 진행되었다.

전환이 완료되고, 가스터빈이 완전히 정지한 84초 이후, PORT, STBD SSS 클러치의 회전속도가 서서히 상승함을 확인할 수 있다.

추진체계 전환시 (46초) 축 토크는 각각 20[kNm]로 나타난다.

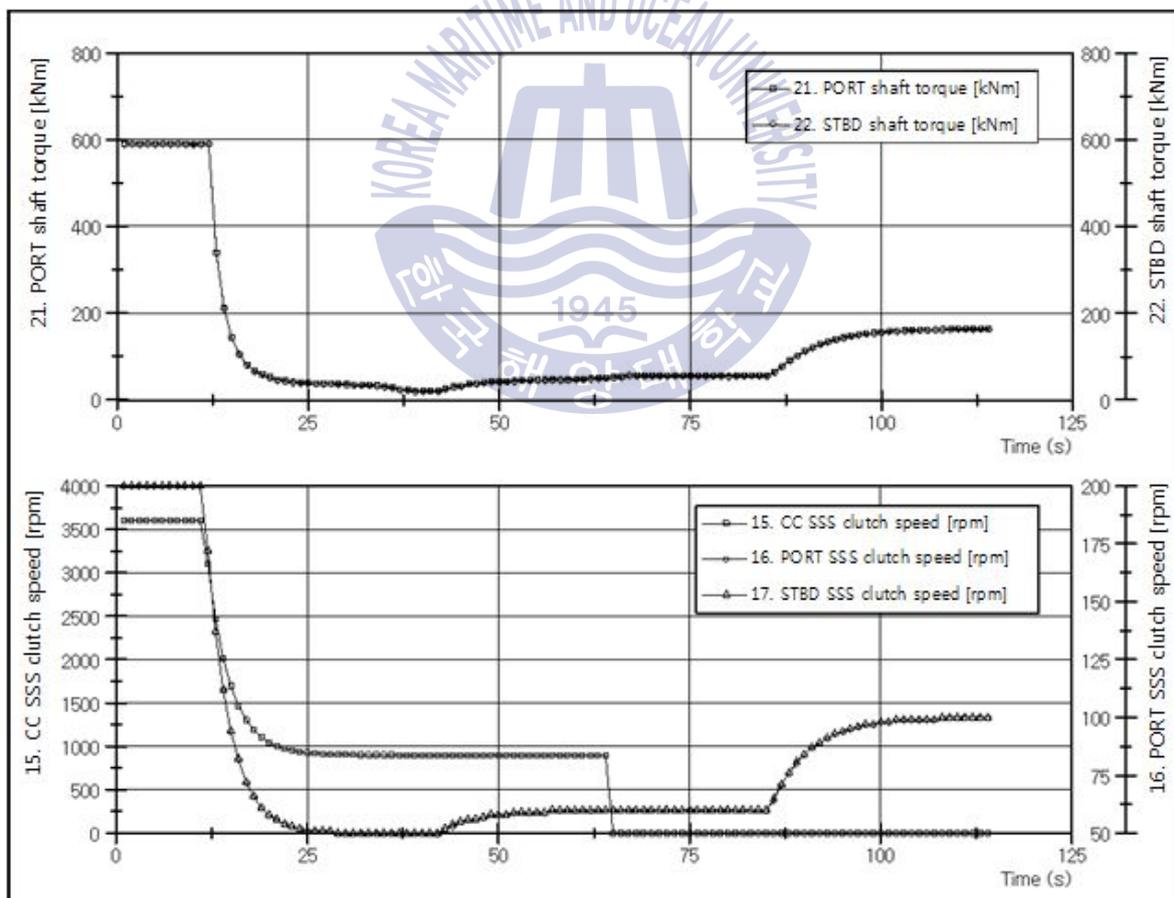


Fig. 82 GT to PM changeover simulation output - shaft and clutch

Fig. 83은 가스터빈, 추진전동기간의 전환 절차에서 추진전동기의 속도 제어 명령에 따른 추진전동기 축계 회전속도와 프로펠러 회전속도, 출력을 나타내고 있다.

추진전동기 속도 명령은 31초와 32초에 0%에서 60%로 변경되었고, 이때 추진전동기 회전속도는 60[rpm]으로 나타났으며, 안정화되는 시점은 31초에서 54초까지 약 23초의 시간이 소모되는 것을 확인할 수 있다.

각각의 프로펠러 출력은 가스터빈으로 운용될 때 18,010[kW] (0~11초), 전환시 약 200[kW] (42초), 추진전동기로 운용될 때 1,706[kW]로 나타났으며, 회전속도는 가스터빈으로 운용될 때 200[rpm] (0~11초), 전환시 약 50[rpm] (42초), 추진전동기로 운용될 때 100[rpm]의 값을 나타낸 것을 확인할 수 있다.

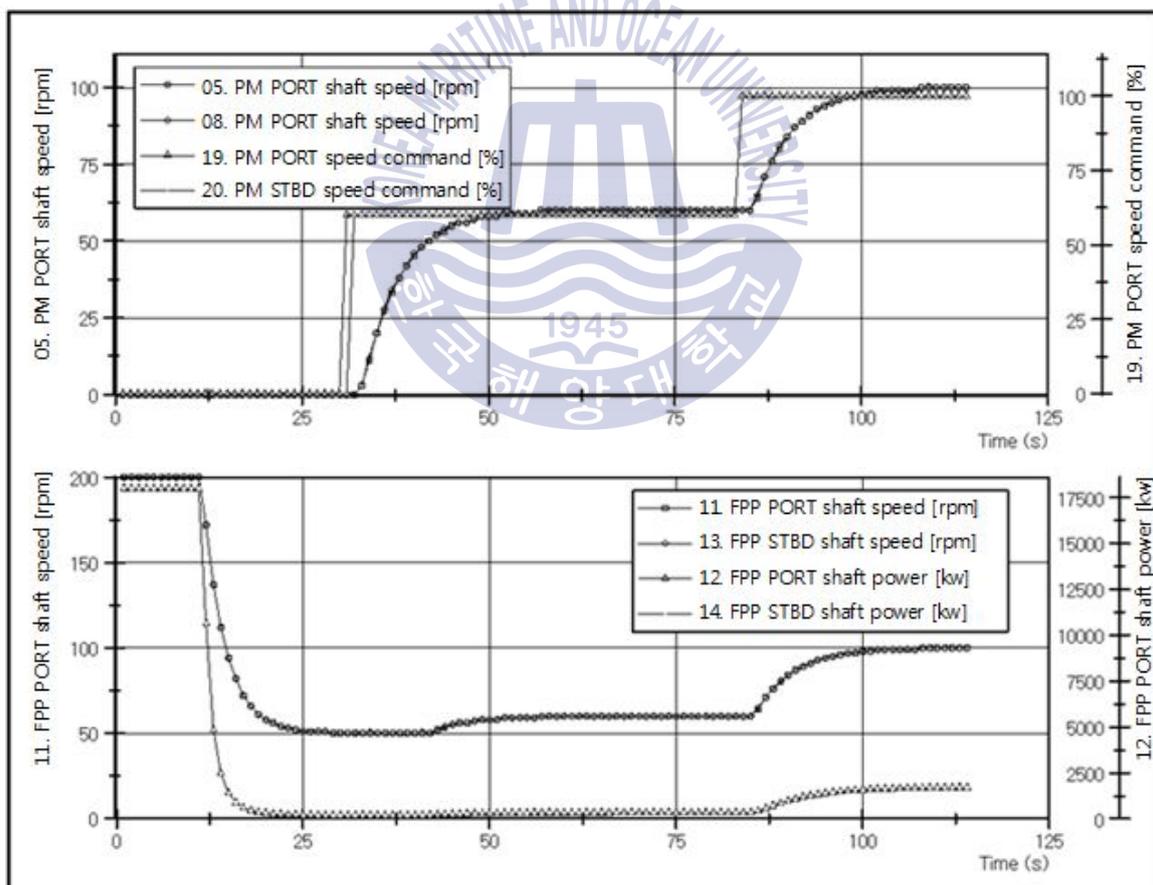


Fig. 83 GT to PM changeover simulation output - PM and FPP

(6) PM to GT changeover

추진전동기에서 가스터빈으로 추진체계 전환을 위해서는 가스터빈의 회전속도가 추진전동기의 회전속도보다 높아서 SSS 클러치가 모두 연결되어야 한다. 특히 CC SSS 클러치는 이 과정이 입출력간의 회전속도에 따라 진행되지만, PT, SB SSS 클러치는 lockout 기능을 가지고 있으며, 추진전동기로 운항시 lockout으로 운용하기 때문에 이와 같은 기능을 ECS에 반영하여 제어해야 한다.

Fig. 84는 추진전동기에서 가스터빈으로 전환하는 과정에서 시간에 따른 가스터빈의 다양한 데이터를 나타내고 있다. 보조기기를 동작시키고, 30초에 가스터빈으로 Start 신호가 전달되면, 워업 과정까지 진행하게 된다. 이후 100초에 가스터빈 출력 명령을 100%로 변화시게 되면 가스터빈 회전속도가 상승하게 된다. 가스터빈 출력과 토크 그래프를 보면, 70초에 브레이크가 해제되어 출력과 토크가 상승하는 것을 확인할 수 있다.

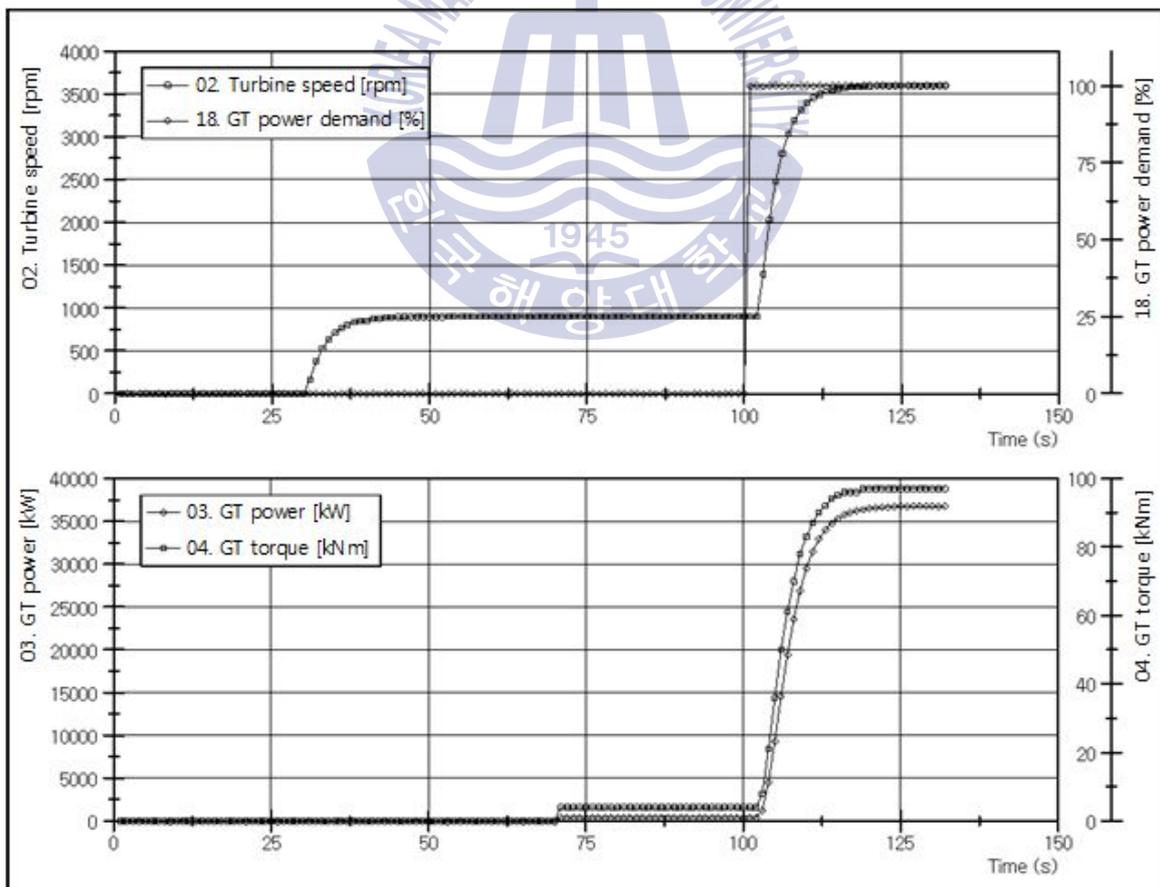


Fig. 84 PM to GT changeover simulation output - GT

Fig. 85는 축 토크, SSS 클러치의 회전속도를 나타내고 있다.

축 토크는 추진전동기의 회전속도를 낮춤에 따라(65~70초) 가스터빈의 공회전(idle running)에 해당하는 30 [kNm]까지 감소하였다가 가스터빈의 출력이 높아짐에 따라(101~132초) 다시 상승하여 119초에는 97 [kNm]에 도달하게 된다.

이때 SSS 클러치의 회전속도를 보면, CC SSS 클러치는 브레이크가 해제되는 60초부터 가스터빈의 공회전 속도인 900 [rpm]로 유지하다가 102초부터 상승하여 최대 회전속도인 3,600 [rpm]에 도달하게 된다.

PORT, STBD SSS 클러치는 60초부터 추진전동기 회전속도가 감소함에 따라 50 [rpm]으로 감소하였다가 CC SSS 클러치와 연결되어 102초부터 상승하여 120초에는 200 [rpm]가 된다.

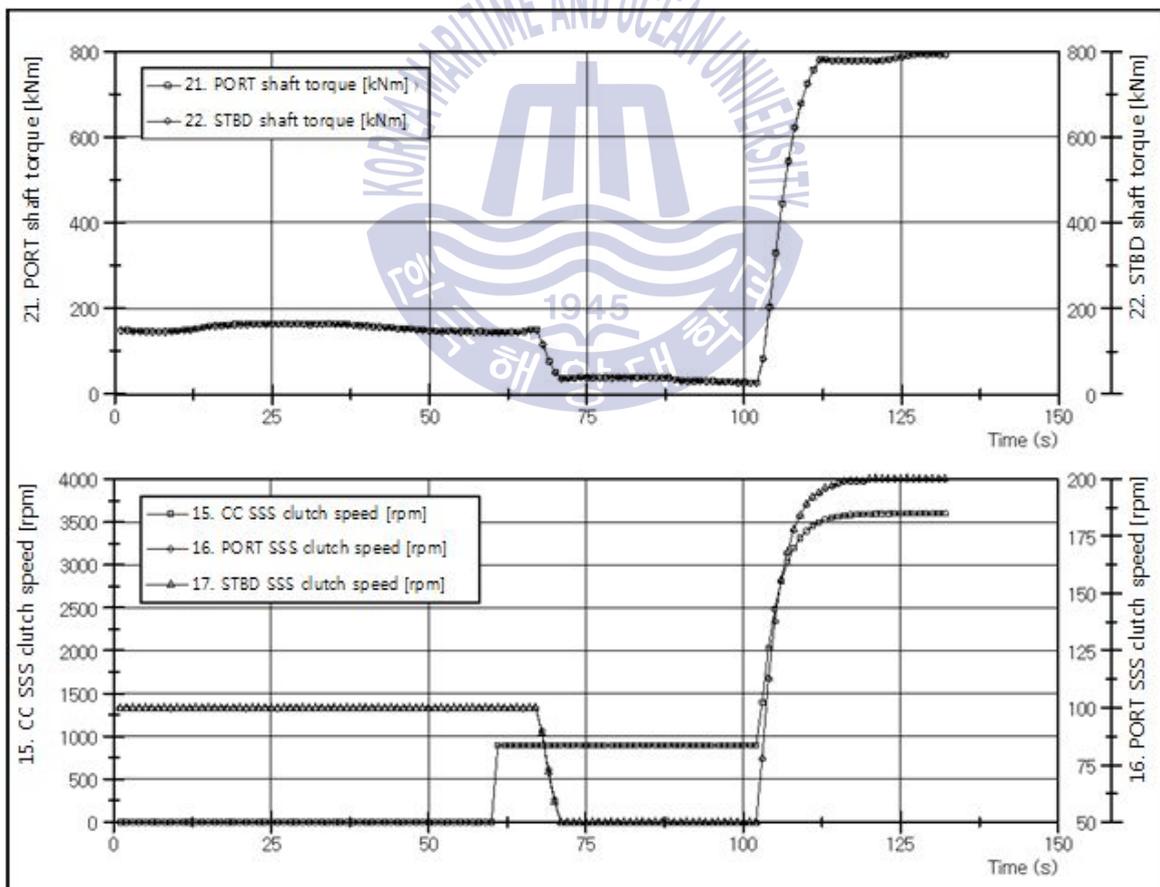


Fig. 85 PM to GT changeover simulation output - shaft and clutch

Fig. 86은 추진전동기 명령값에 따른 회전속도의 변화와 FPP 회전속도, 출력을 나타내고 있다.

추진전동기 속도 명령은 65초에 0%으로 변경되며, 추진전동기는 일정 비율로 감속하게 되어 94초에 추진전동기 축계 회전속도가 0[rpm]이 된다. 이때, 가스 터빈의 회전속도는 900[rpm]이며, 감속기어를 지나게 되면 50[rpm]이 됨으로 PORT, STBD SSS 클러치는 추진전동기의 회전속도가 50[rpm]보다 낮아지는 71초에 연결되어 추진체계가 추진전동기에서 가스터빈으로 전환된다. 이때 축 토크는 약 29[kNm]이며, 출력은 199[kW]로 나타난다.

FPP의 출력은 67초까지 1,713[kW], 102초까지 199[kW], 이후 가스터빈 출력 상승으로 인해 18,008[kW]로 나타난다.

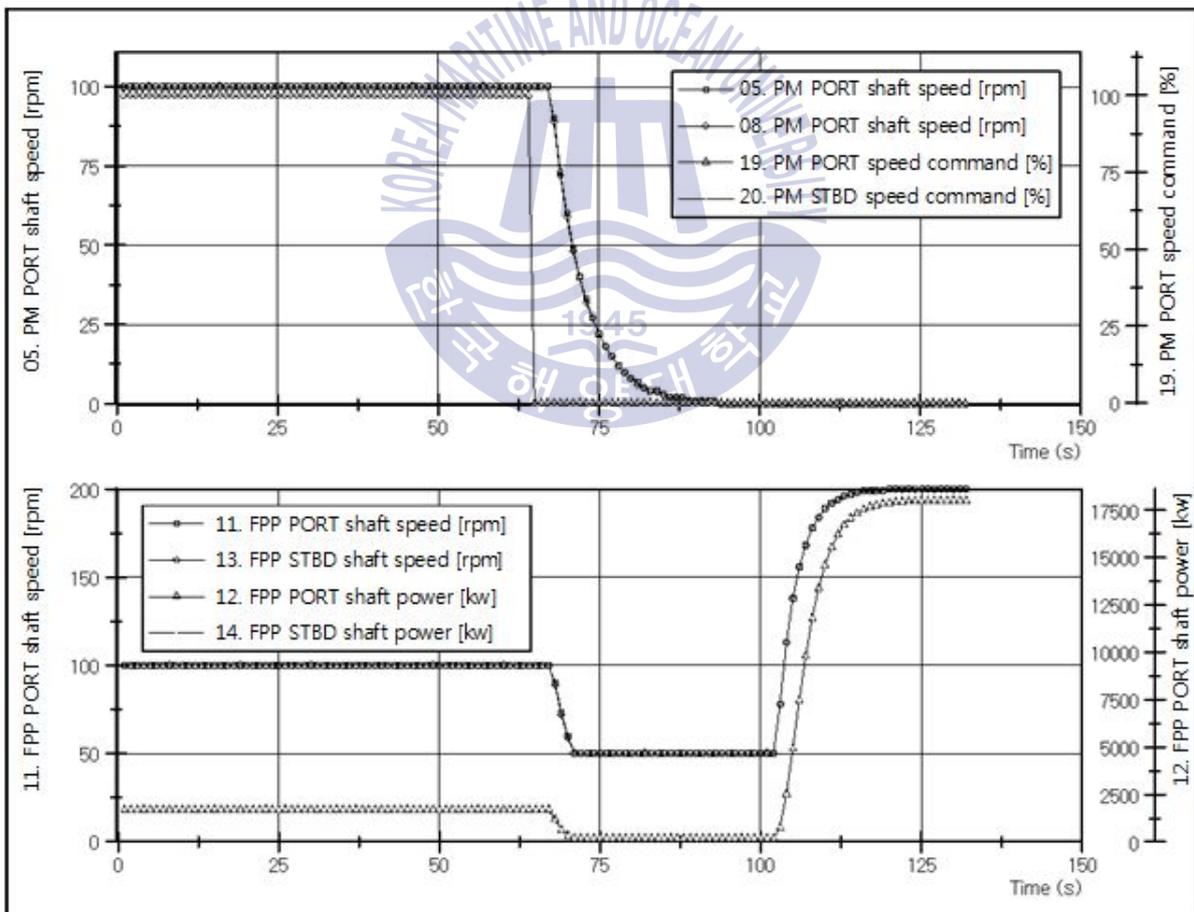


Fig. 86 PM to GT changeover simulation output - PM and FPP

⑦ GT emergency start

비상 시동 시퀀스는 비상시 함정이 빠르게 기동하기 위해 사용하는 비상 모드 중 하나로, 가스터빈 시동 시퀀스와 거의 유사하지만, 워업을 하지 않고 바로 가스터빈의 출력을 상승시킨다.

이렇게 사용하는 경우, 가스터빈의 수명에 좋지 않은 영향을 미치지만, 함정이라는 특수한 목적을 가지기 때문에 함장의 명령에 따라 사용된다.

시뮬레이터 관점에서는 워업을 제외한 가스터빈 시동 시퀀스와 동일하게 나타난다.

Fig. 87 ~ Fig. 89는 비상 시동 시퀀스에 따른 가스터빈, 감속기어, 추진전동기, FPP에 대한 데이터를 그래프로 나타낸 것이다.

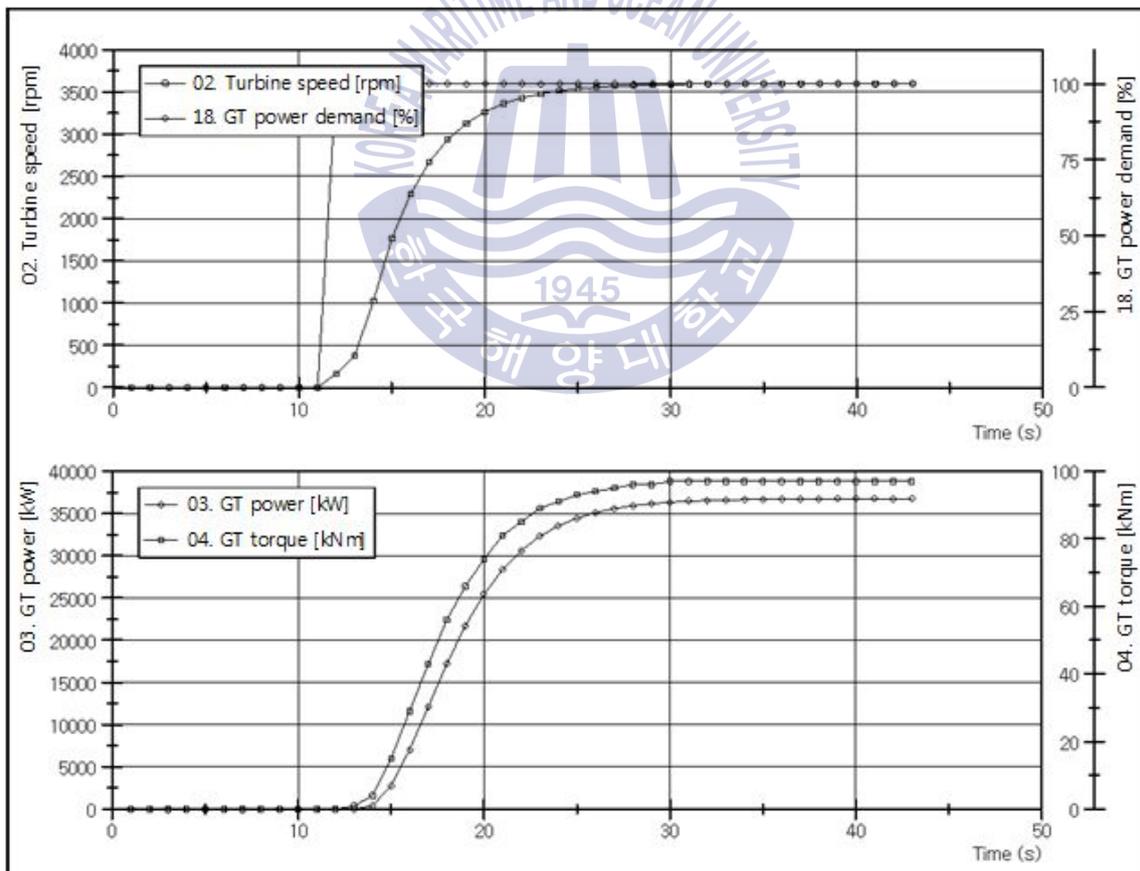


Fig. 87 GT emergency start simulation output - GT

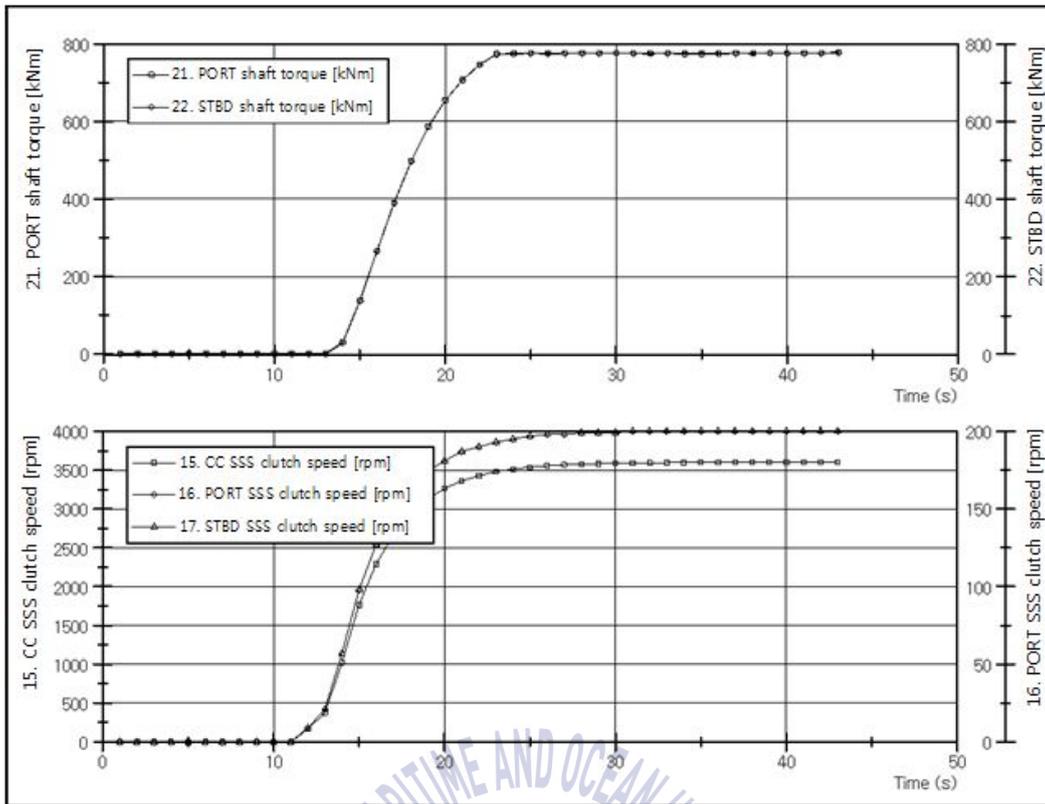


Fig. 88 GT emergency start simulation output – shaft and clutch

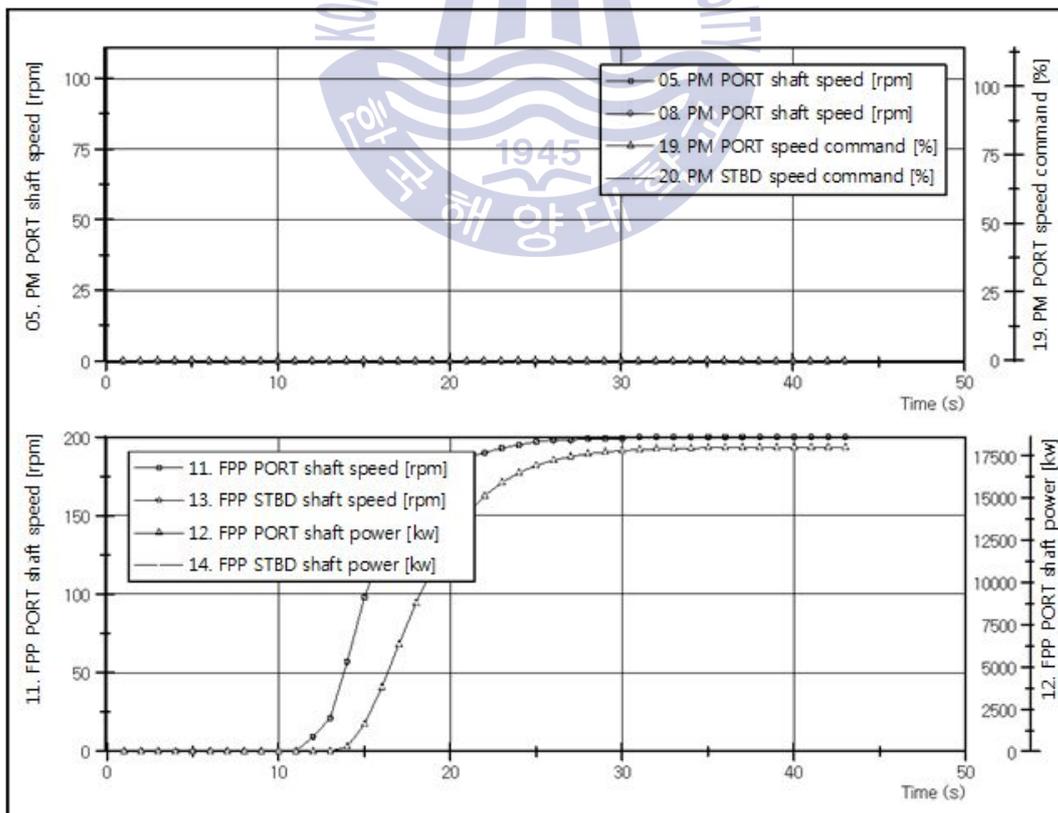


Fig. 89 GT emergency start simulation output – PM and FPP

## 제 6 장 결론

함정 추진체계용 제어기인 ECS는 함정 기술 국산화의 핵심 기술이지만, 동시에 우리나라가 가지고 있는 함정 관련 기술 중 가장 취약한 분야이기도 하다.

ECS의 핵심은 장비 운용 알고리즘의 신뢰도이다. ECS는 함정의 추진체계 전체를 통합 관리 및 제어하기 때문에 함정의 추진성능 확보에 매우 중요한 역할을 수행한다. 따라서 함정용 ECS는 장기간 운용하여 검증된 장비를 사용하려는 경향이 있기 때문에 ECS 국산화를 위해서는 가장 먼저 장비 운용 알고리즘의 신뢰도를 확보해야 한다.

본 논문은 ECS 신뢰도 확보를 위해 활용할 수 있는 함정 추진체계용 시뮬레이터의 개발에 관한 연구이다. 이를 위해 현재 국외 함정 추진체계의 개발 현황과 국외 ECS 개발 업체의 ECS 개발 과정을 조사하고, 핵심 요소를 분석하였다. 또한, 가상 함정의 추진체계와 7개의 주요 운용 알고리즘을 정의하였으며, 이를 토대로 HILS 기반의 함정 추진체계용 시뮬레이터를 개발하였다.

시뮬레이터 개발 과정에서 함정을 대상으로 하였기 때문에 시뮬레이션 모델 구성에 필요한 각 추진장비의 제어 알고리즘, 전체를 운용하기 위한 통합 운용 알고리즘, 각 추진장비별 운동 특성 등과 같은 정확한 데이터 확보와 시뮬레이션 결과에 대한 실제 함정과의 비교 및 검증이 어려운 점이 있었다. 앞으로 이러한 문제점을 고려하여 국가 기관 주도의 연구 개발이 진행되어야 할 것이다.

본 논문은 함정 추진체계용 시뮬레이터와 가상의 ECS를 연동하여 주요 운용 알고리즘에 대해 검증함으로써 ECS 신뢰도 확보를 위한 함정 추진체계용 시뮬레이터 개발 방법과 시뮬레이터를 이용한 ECS 검증 방안에 대해 제시하였으며, 추후 본 연구 결과가 ECS 국산화를 위한 기반자료로 활용 될 수 있을 것으로 생각한다.

## 감사의 글

또다시 아침 7시입니다. 마지막의 마지막까지 이 논문을 보완했고, 그래도 여전히 부족하다 느끼지만, 이제는 떠나보내야 할 때인 것 같습니다.

해양대학교 3학년 겨울에 처음 오진석 교수님을 찾아뵙고 랩에 들어가고 싶다고 했던 날이 아직도 기억납니다. 그로부터 10년이 지난 지금 제 이름 뒤에 박사라는 과분한 수식어를 달고 사회로 나갈 준비를 하고 있습니다.

참으로 오랜 시간이었고, 놀라운 시간들이었습니다. 저는 이 조그마한 사회 안에서 오진석 교수님의 보호하에 다듬어졌고, 또 변화되었습니다. 힘든 일도 있었고, 슬픈 일도 있었습니다. 또, 가슴이 설레어 잠을 이루지 못하는 밤도 있었고, 너무 좋아서 두 주먹을 불끈 쥐고 뛰어오를 것만 같은 날도 있었습니다. 이 모든 경험들이 지금의 저를 만들었고, 또 앞으로 사회에서 살아남기 위한 든든한 버팀목이 되어 줄 것이라 생각합니다. 여기까지 키워주신 오진석 교수님께 진심으로 감사드립니다.

박사학위 논문을 쓰는 과정에서 제 부족함을 절실하게 느꼈습니다. 자신있다고 생각했던 논리적인 사고 능력은 심사위원님들의 한마디에 산산조각났고, 한글의 문법, 용례, 맞춤법까지 얼굴이 부끄러워질 정도로 많은 지적을 받았습니다. 이러한 지적이 있었기에 부족하지만 그래도 박사학위 논문이라는 이름을 달수 있었습니다. 정병건 교수님, 소명옥 교수님, 유희한 교수님, 곽준호 과장님께 진심으로 감사드립니다.

10년 동안 연구실에 있으면서 많은 사람들을 만났습니다. 우선, 정신적 지주이신 이지영 선배님과 이제는 졸업해 나가신 곽준호 선배님, 조관준 선배님, 김연형 선배님, 이종호 선배님, 한성훈 선배님, 박도영 선배님, 배수영 선배님, 강훈에게 감사드립니다. 덕분에 많은 것 들을 배울 수 있었습니다.

그리고 부족한 저 때문에 고생한 김지윤, 이현석, 장재희, 김민욱, 강영민, 손나영 대학원생에게 미안한 마음과 함께 고맙다는 말을 전합니다. 제가 했던 부족한 부분들은 보완하고, 그래도 괜찮았다고 보이는 것들은 잘 받아서 E2E

연구실이 좀 더 나은 곳이 될 수 있도록 노력해주었으면 합니다.

연구실 안에서만 있어서 부족할 수 있었던 시야를 넓혀주신 함연재 사장님, 양향권 사장님, 이창노 님, 최동일 님, 류승현 님, 류동영 님, 유대완 님, 백공구 님, 그 외에도 일일이 거명하지 못하는 많은 분들께 감사의 인사를 드립니다.

그리고, 마지막까지 제 논문의 abstract 번역을 도와준 사촌 형님 권기성과 오랜 친구인 장민석에게 감사드립니다. 조만간 찾아가서 밥 사겠습니다. 또, 오랜 추억이 담겨있는 고등학교 동아리 선후배님들, 고향 친구들 덕분에 힘들 때, 흔들릴 때 쉬어갈수 있었습니다. 고맙습니다.

물심양면으로 도와주신 친가, 외가 친척 모든 분들께 진심으로 감사드립니다. 할머니께서 살아계실 때 제가 박사학위를 받아서 찾아오면 무덤에서도 너울너울 춤을 출거라고 하셨습니다. 곧 논문가지고 찾아 뵙겠습니다 할머니.

항상 저 걱정해주시고, 찾아 뵈 때마다 따뜻한 밥과 응원으로 힘을 북돋아 주신 외할아버지, 외할머니께 감사드립니다. 항상 건강하시고, 앞으로는 좀더 자주 찾아뵙겠습니다.

마지막으로 우리 가족, 사랑하는 어머니, 사랑하는 내 동생. 제 욕심 때문엔 길을 돌아서 여기까지 왔습니다. 기다려줘서 고맙고, 또 함께 기뻐해줘서 고맙습니다. 사랑합니다.

하늘에서 기뻐하고 계실 아버지께 이 논문을 바칩니다. 사랑합니다.

## 참고문헌

- [1] 백현민, 정균식, 이명호, 최재성, 김동함, 2014. 함정의 통합 전력시스템 구성에 관한 연구. *Journal of the Korean Society of Marine Engineering*, 38(9), pp.1070-1074
- [2] 이형민, 조병진, 2011. 고출력 에너지 사용 체계 플랫폼 실현을 위한 해군함정의 통합 동력 시스템 발전 경향 분석, *Journal of the Korean Society of Marine Engineering*, 35(6), pp.796-801
- [3] British Crown, 2009. *Electric Propulsion for UK Warships*
- [4] 국방기술품질원, 2010. *국방과학기술조사서*
- [5] DCNS, 2010. *Simulation and control of power and propulsion systems*
- [6] M Altosole, G Benvenuto, M Figari & U Campora, 2009. Real-time simulation of a COGAG naval ship propulsion system, *Journal of Engineering for the Maritime Environment*, 223, pp.47-61
- [7] M. Altosole & M. Figari, 2011. Effective simple methods for numerical modelling of marine engines in ship propulsion control system design, *Journal of Naval Architecture and Marine Engineering*, 2, pp.129-147
- [8] G. Benvenuto, S. Brizzolara & M. Figari, 2001. Simulation of the propulsion system behaviour during ship standard manoeuvres
- [9] G. Benvenuto & U. Campora, 2006. A gas turbine modular model for ship propulsion studies, *2006' China International Boat Show&HPMV Conference*
- [10] M Figari & M Altosole, 2007. Dynamic behaviour and stability of marine propulsion systems, *Engineering for the Maritime Environment*, 221, pp.

- [11] M. Altosole, G. Benvenuto & M. Martelli, 2011. Advances in automation design for fast vessels propulsion, *IX HSMV Naples*
- [12] U. CAMPORA, M. FIGARI, M. MARTELLI, S. VIGNOLO & M. VIVIANI, 2011. Propulsion plant simulation for fast military vessels, *IX HSMV Naples*
- [13] Christopher Ryan Holsonback, B.S., 2007, Dynamic thermal-mechanical-electrical modeling of the integrated power system of a notional all-electric naval surface ship, master, The University of Texas at Austin: Science in Engineering
- [14] <http://www.sssclutch.com/howitworks/100-2SSSPinciples.pdf>
- [15] Morgan L. Hendry & B. Michael Zekas, 2008. U.S. navy experience with SSS(Synchro-Self-Shifting) clutches. *Proceedings of ASME Turbo Expo 2008: Power for Land, Sea and Air*, June 9-13, 2008, Berlin, Germany
- [16] HED catalog, DRS
- [17] ECS catalog, L3-mapps
- [18] Andreas Torp Karlsen, 2012. *On modeling of a ship propulsion system for control purposes*, NTNU
- [19] MAN Diesel & Turbo, *Basic principles of ship propulsion*
- [20] Saravanamuttoo, H. I. H. & Fawke, A. J. Digital computer methods for the prediction of gas turbine response. *Soc. Auto. Engineers*, 1971, paper no. 710550
- [21] U Campora & M Figari, 2003. Numerical simulation of ship propulsion transients and full-scale validation, *Engineering for the Maritime Environment*, pp.41-52
- [22] Michele Martelli, Massimo Figari, Marco Altosole & Stefano Vignolo, 2014. Controllable pitch propeller actuating mechanism, modelling and simulation,

*Journal of Engineering for the Maritime Environment*, 228, pp.29-43

- [23] Dr. ing., 2009. *Modeling and Control of gas turbine thermoacoustic pulsations*, Ph.D. Technischen University.
- [24] T. PEREZ, O. N. SMOGELI, T. I. FOSSEN & A. J. SORENSEN, Massimo Figari, Marco Altosole and Stefano Vignolo, 2006. An overview of the marine systems simulator (MSS)-a simulink toolbox for marine control systems, *Modeling Identification and Control*, 27(4), pp.259-275
- [25] Andrzej Grzadziela, 2007. Dynamic problems of propulsion system of naval vessels. *Journal of KONES Powertrain and Transport*, 14(3), pp.195-202.
- [26] M.W.C OOSTERVELD & P. VAN OOSSANEN, 1975. FURTHER COMPUTER-ANALYZED DATA OF THE WAGENINGEN B-SCREW SERIES. *INTERNATIONAL SHIPBUILDING PROGRESS*, 22(251), pp.3-14.
- [27] JAN FREDRIK HANSEN, ALF KARE ADNANES & THOR I. FOSSEN, 2001. Mathematical Modelling of Diesel-Electric Propulsion Systems for Marine Vessels. *Mathematical and Computer Modelling of Dynamical Systems*, 7(1), pp.1-33.
- [28] E. Sciberras & A. Grech, 2012. Optimization of Hybrid Propulsion Systems. *International Journal on Marine Navigation and Safety of Sea Transportation*, 6(4), pp.539-546.
- [29] Kyle Schmitt, 2010. *Modeling and Simulation of an All Electric Ship in Random Seas*. Ph.D. University of Wisconsin.
- [30] B. Pinnekamp, F. Hoppe, M.Heger & RENK AG, 2012, Combined Marine Propulsion Systems: Optimization and Validation by Simulation
- [31] JOHN CARLTON, 2007, Marine Propellers and Propulsion, 2nd Ed
- [32] T. Mazzuca & M. Torre, 2008. The FREMM Architecture : a first Step towards Innovation, SPEEDAM 2008 International Symposium on Power Electronics, Electrical Drives, Automation and Motion

- [33] G. Sulligori, D. Bosich, T. Mazzuca & L. Piva, 2012. The Fremm Simulator: a new Software tool to study Electro-Mechanic Dynamics of the Shipboard Integrated Power System. Institute of Electrical and Electronics Engineers, pp.1-6.
- [34] THE SCIENCE AND TECHNOLOGY FOUNDATION FOR FUTURE NAVAL FLEETS, 2011. The Future for Naval Engineering
- [35] Morgan L. Hendry & Matthew G. Hoffman, 2007. Development, Testing, and Implementation of a Gas Turbine Starting Clutch With Manual Turning Feature for U.S. Navy Ships. *Journal of Engineering for Gas Turbines and Power*, 129, pp.785-791.
- [36] Morgan L. Hendry & B. Michael Zekas, 2008. U.S. NAVY EXPERIENCE WITH SSS (SYNCHRO-SELF-SHIFTING) CLUTCHES. *Proceedings of ASME Turbo Expo 2008: Power for Land, Sea and Air*
- [37] Blue Vision New Generation, 2012, Monitoring and remote control for MTU ship propulsion systems
- [38] M.L Parker, P.K. Maeleod & M. Coulson, 1998. Estimation of geometry-based manufacturing cost of complex offshore structures in early design stage. *RTO AVT Symposium on "Gas Turbine Engine Combustion, Emissions and Alternative Fuels"* . pp.1-9.
- [39] Benvenuto G., Carrera G. & Rizzuto E. 1994. *Dynamic Simulation of Marine Propulsion Plants*. NAV94, Rome, 1994.
- [40] Altosole, M., Benvenuto, G., Campora, U. & Figari, M., 2004. Dynamic performance simulation of a naval propulsion system, In: *Proceedings of the Summer Computer Simulation Conference*, San Jose, California.
- [41] Benvenuto, G., Campora, U., Carrera, G. & Casoli, P., 1998. A two-zone diesel engine model for the simulation of marine propulsion plant transients, In: *Proceedings of the Second International Conference on Marine Industry*,

Varna, Bulgaria.

- [42] E, Sciberras & A. Grech, 2012. Optimization of Hybrid Propulsion Systems. International Journal on Marine Navigation and Safety of Sea Transportation, 6(4), pp.539-546.
- [43] Thomas A. Laverghetta, 1975. *Dynamics of naval ship design: A systems approach*. Ph.D. MIT: Department of Ocean Engineering
- [44] Zachary Samuel Zisman, 2011. On the simulation of an all electric ship powertrain utilizing a surface piercing propeller via a modular main propulsion plant model. Master. the Virginia Polytechnic Institute and State University
- [45] James Turso, Abe Boughner & Gianfranco Buonamici, 2010. Propulsion simulator/stimulator development for US navy's newest gas turbine-powered ship, LHD 8 USS MAKIN ISLAND, In: *Proceedings of ASME Turbo Expo 2010: Power for Land, Sea and Air*, Glasgow, UK
- [46] Oliver Rath, 2014. MT30 – The heart of future integrated power and propulsion systems, In: *Proceedings of ASME Turbo Expo 2014: Turbine Technical Conference and Exposition*, Düsseldorf, Germany
- [47] Kao, M. and Moskwa, J. J., 1995. Turbocharged diesel engine modeling for nonlinear engine control and state estimation, Journal of Dynamic Systems, Measurement and Control, 117, pp.20-30
- [48] Altosole, M., Campora, U. and Figari, M., 2009. A proper use of sea trials data for marine propulsion simulations. In: *Proceedings of the 4th Int. Conference on Maritime Transport*, Barcelona, Spain.
- [49] Cohen, H., Rogers, G. F. C., & Saravanamuttoo, H. I. H., 1987. *Gas turbine theory, edition 3*, Longman Scientific & Technical, England
- [50] El-Masri, M. A., 1988. GASCAN – an interactive code for thermal analysis of gas turbine systems, Trans. ASME, J. Engng Gas Turbine Power, 110,

pp.201-209

- [51] Campora, U. and Gioia, F. 1989, Dynamic performance prediction of an aeronautical gas turbine, In: *44th National Congress of the Italian Thermo-technical Association*, Cosenza, Italy.

