

## DDS 미들웨어 기반의 선박 통합기관감시제어체계 개발 가능성 연구

오성원\*\*

\* 국립목포해양대학교 해군사관학부 교수

## Feasibility Study of Developing Ship Engineering Control System based on DDS Middle-ware

Seongwon Oh\*\*

\* Professor, Division of Naval Officer Science, Mokpo National Maritime University, Mokpo 58628, Korea

**요 약** : 함정의 전투관리체계나 민간의 스마트 시티와 같은 많은 센서와 동작기기가 연결된 체계에서는 주로 미들웨어인 DDS(Data Distribution Service)를 사용하여 대규모의 데이터를 전송하고 향후 센서나 장비 증가에 대해 효과적으로 대응하고 있다. 선박에서 함정의 전투관리체계(Combat Management System, CMS)와 같은 역할을 하는 중요한 제어체계인 통합기관감시제어체계(Engineering Control System, ECS)는 여전히 Server-Client 모델을 기반으로 산업용 프로토콜(Modbus, CAN(Controller Area Network) bus 등)을 사용하여 데이터를 전송하고 있어 확장성 측면에서 불리하다. 따라서 향후 자동화나 무인화가 진행될 경우에 많은 센서류가 추가될 것이며 이는 많은 프로그램 수정 수요가 발생하게 되며, DDS는 이러한 상황에서 좋은 해결책이 될 수 있다. 본 연구에서는 전투관리체계에서 사용 중인 미들웨어인 DDS 중에서 OMG(Object Management Group) 표준을 따르는 OpenDDS를 활용하여 선박의 통합기관감시제어체계의 개발 가능성을 확인하였다. 이를 위해 필드장비 시뮬레이터 및 ECS 서버를 구성하여 DDS를 이용한 필드장비 데이터 입출력 시뮬레이션을 수행하였다. 개발한 ECS 축소모형은 데이터를 발간-구독하는데 문제가 없으며, DDS가 선박 ECS의 미들웨어로 충분히 사용할 수 있을 것으로 판단된다.

**핵심용어** : 미들웨어, DDS, OMG, 통합기관감시제어체계, 시뮬레이터

**Abstract** : In systems like the combat management system of a naval ship or smart city of civilians, where many sensors and actuators are connected, the middle-ware DDS (Data Distribution Service) is mainly used to transmit large amounts of data. It is scalable and can effectively respond to the increase in sensors or equipment connected to the system in the future. The engineering control system (ECS), which plays an important role similar to the combat management system of a naval ship, still uses Server-Client model with industrial protocols such as Modbus and CAN (Controller Area Network) bus, to transmit data, which is unfavorable in terms of scalability. However, as automation and unmanned systems advance, more sensors and actuators are expected to be added, necessitating substantial program modification. DDS can effectively address such situations. The purpose of this study is to confirm the development possibility of an integrated monitoring and control system of a ship by using OpenDDS, which follows the OMG (Object Management Group) standard among the middle-ware DDS used in the combat management system. To achieve this goal, field equipment simulators and an ECS server were configured to perform field equipment data input/output and simulation using DDS was performed. The ECS prototype successfully handled data transmission, confirming that DDS is capable of serving as the middle-ware for the ECS of a ship.

**Key Words** : Middle-ware, Distributed data service, Object management group, Engineering control system, Simulator

## 1. 서 론

선박의 엔진과 펌프와 같은 장비류를 감시하고 제어하기 위해서 통합기관감시제어체계(Engineering Control System, ECS)를 이용하고 있으며, 이는 IPMS(Integrated Platform Management

System), ICMS(Integrated Control and Monitoring System), IMCS(Integrated Monitoring and Control System), IACS(Integrated Automation and Control System) 등으로도 통칭된다(Kwak, 2021; Liu et al., 2014; Peplinski, 2021; Warżala, 2020에서 발췌).

선박의 ECS는 선박의 주요 장비와 체계에 대한 전반적인 감시와 제어를 수행하는 장비로써, 크게 추진계통(디젤엔진, 감속기어, 조타기 등), 전력계통(발전기, 변압기, 차단기 등)

† osw123@mmu.ac.kr, 061-240-7123

및 보조기기(펌프, 밸브, 팬, 도어 등) 등 제반 장비의 시동과 정지, 상태 감시 및 제어 명령 수행과 장비 간의 인터록 등을 원격으로 수행한다. 최근에는 자동화가 더욱 진행됨에 따라 많은 센서와 장비가 통합되고 있으며, 통합선교시스템(Integrated Bridge System, IBS) 또는 통합항해시스템(Integrated Navigation System, INS) 등과 연동되거나 이를 ECS 내로 통합하고 있다(Plinski, 2021). 이러한 통합은 최근 들어 자율운항선박이나 원격제어선박 개발이 활발해짐에 따라 ECS에 기존의 장비나 센서류 뿐만 아니라, 항해 레이더나 통신장비까지 선박 내의 대부분의 장비들이 ECS와 연동되거나 통합될 것으로 예상된다.

Fig. 1은 일반적인 선박의 ECS 구성으로써, 하드웨어는 크게 서버, 네트워크, 신호 입출력장치(Data Acquisition Unit, DAU) 및 필드장비로 구분할 수 있다. 선박 내에 설치된 엔진, 펌프와 같은 장비들은 지정된 프로토콜로 DAU에 정보를 전송하고, DAU는 ECS 네트워크를 통해 데이터를 서버로 전달하는 구조이다. 네트워크 및 관련장비와 서버는 이중으로 구성되며 선박 내의 다른 네트워크와 연동될 경우에도 이중으로 연결된다(Plinski, 2021).

현재 선박의 ECS는 필드장비와 DAU 간에는 CAN, Profibus, Modbus 등과 같은 산업용 프로토콜과 디지털 및 아날로그 Hardwired 신호를 이용하고, 서버 간에는 정적 주소를 사용하는 이더넷을 이용한다. 그러나 위에서 언급한 바와 같이 자동화 수준이 증가하고, IoT(Internet of Thing) 기술 등 다양한 기술이 점차 선박에 적용됨에 따라 ECS에 연결되

는 노드가 증가하고, 시간에 따라 연결되는 노드가 변경됨에 따라 기존의 방식으로는 이러한 변화에 적응하기 위한 유연성, 확장성 및 상호운용성을 지원하기 어렵다. 또한 위에서 언급한 선박 내의 다른 네트워크와 연동되는 경우에는 기존의 시스템을 대폭 수정해야 하는 단점이 있다.

DDS(Data Distribution Service)는 OMG(Object Management Group)에서 표준화한 데이터 중심의 통신 미들웨어이다(OMG, 2015). DDS는 분산 환경에서 실시간 통신을 위한 발간/구독 방식을 지원하며, 성능, 확장성 그리고 가용성 측면에서 강점을 가진다. 이러한 장점 때문에 국방, 교통, 의료 분야 등 다양한 산업시스템에서 사용되고 있다(Im, 2021).

민간분야에서는 철도 안전관제 시스템과 철도 현장의 인터페이스 장치 간 DDS가 표준으로 채택됨에 따라, 기존 인터페이스 장치 데이터를 DDS 표준 형식으로 변환하는 방법에 대한 연구가 진행되었다(Park and Kim, 2016).

국방분야에서는 함정의 경우 레이더, 소나 및 무장이 통합된 CMS를 DDS로 개발하거나, CMS의 토픽을 정의하고 관리하기 위한 시스템에 대한 연구가 주로 수행되었다(Lee, 2017). 이외에도 CMS와 ECS와 같은 여러 통제체계의 통합에 관한 연구가 수행되었으나(Oh, 2018; Park et al., 2020), 타 기종의 네트워크 간의 통합에 대한 아키텍처 수준의 연구에 국한되었다. 또한 ECS에 대한 연구가 수행되었으나 ECS의 국산화에 따른 제어로직과 성능시험을 위한 시뮬레이터 개발에 국한되었다(Jang et al., 2017; Kim et al., 2021)

무인체계의 중요성이 대두되면서 다수의 무인체계와 통

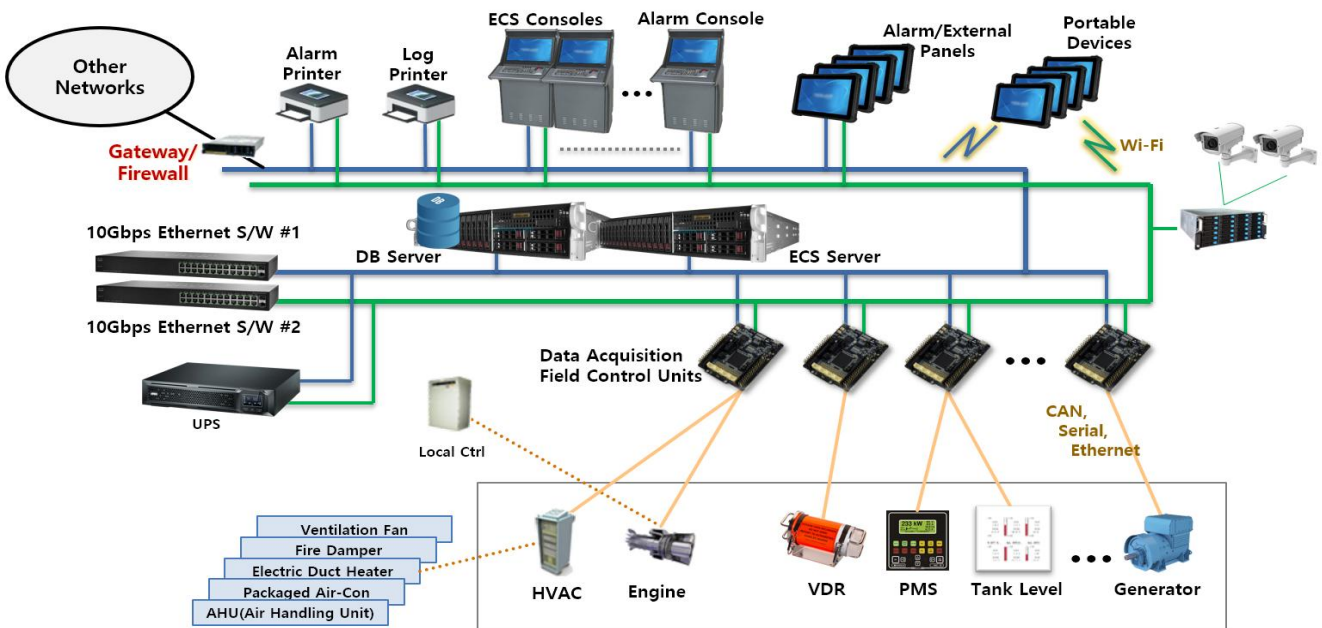


Fig. 1. General ECS configuration.

제 스테이션 간에 DDS를 이용하여 데이터를 전송하는 연구가 진행되었다(Molina et al., 2017). 또한, 미 해군은 무인체계 내에서 센서와 장비간 통합을 위한 아키텍처로 UMAA (Unmanned Maritime Autonomy Architecture)를 제시하였으며, 이의 근간이 되는 미들웨어로 DDS를 적용하였다(PMS 406, 2021).

이와 같이 DDS는 실시간 데이터 처리가 필요하고 연결된 노드가 많은 분산 환경에서 효율적으로 데이터를 배포하기 위해 민간과 군용으로 사용이 증가하고 있다. 또한 자율운항 선박의 경우에는 다양한 필드장비, 센서류 및 시스템이 증가하고, 다른 네트워크와의 연동 또는 통합이 필수적이다. 이에 따라 선박의 ECS는 기관장비 뿐만 아니라 통신, 항해 센서와 같은 다양한 노드들이 연결될 수 있으며, 다른 네트워크와 연결될 수 있는 환경에 놓여있으므로, DDS 기반으로 개발하는 것이 유리하지만 이에 관한 연구는 진행되지 않았다. 따라서 본 연구에서는 이러한 상황에서 최초로 DDS를 이용하여 선박용 ECS를 개발하여 성능을 확인하기 위해 수행하였다.

## 2. 데이터통신 미들웨어 분석

대부분의 분산 네트워크 애플리케이션은 데이터나 서비스를 제공하는 Server와 이를 요청하는 Client로 구성된 Client-Server 모델을 적용하고 있다(Fall and Stevens, 2012). 데이터를 얻기 위해서 Client와 Server는 동기화된 Request와 Reply 절차를 거치므로 강한 연결성을 가진 분산 시스템에 주로 적용된다.

DDS는 데이터 중심의 Publish-Subscribe 방식의 통신 미들웨어이다. Publish-Subscribe 모델은 동기화되지 않아 낮은 연결성을 가지고 있으며 참여 노드가 증가하는 경우에도 관리가 용이하고 복잡도가 낮다. 반면 Client-Server 모델은 참여 노드가 증가하는 경우에 복잡도가 기하급수적으로 증가한다. 따라서 Publish-Subscribe 모델을 적용하는 데이터 중심의 미들웨어인 DDS는 대규모 분산 시스템을 위한 미들웨어로, 컴포넌트 간에 확장 가능성(Scalable)과 효율적인 발간 및 구독(Publish-Subscribe) 통신을 제공한다. 또한 유용성(Usability), 관리성(Administrability), 상호 운용성(Interoperability) 및 확장성(Extensibility)과 같은 기존 미들웨어 요구사항을 해결할 수 있다(Pietzuch, 2004). 데이터 기반 미들웨어에서는 많은 토픽이 발생하지만 구독자(Subscriber)는 필요한 토픽들만 수신한다. 여기서 토픽은 DDS 도메인에서 유일한 이름을 가지고 수신 또는 송신하고자 하는 사용자 데이터의 구조이다(Jeong, 2018). 데이터 구독자, 즉, 데이터를 수신하는 시스템은 특정 데이터를 구독의 형태로 수신하게 된다. 데이터 발

간자(Publisher) 즉, 데이터를 생성하는 서버들은 구독자들에게 전달될 데이터를 발간한다. 이러한 모델은 자연스럽게 발간자와 구독자 사이를 분리하게 되고 1:N 및 N:N 통신을 지원하게 된다. Fig. 2에서 보는 바와 같이 구독자는 토픽을 제공하는 발간자에 대해 알 필요가 없으며, 마찬가지로 발간자도 발간된 토픽을 받을 구독자들에 대해 알 필요가 없다(Lee and Ko, 2020).

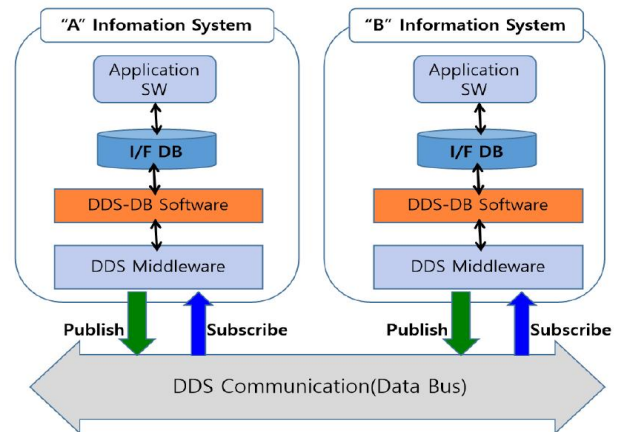


Fig. 2. Data communication structure of DDS system(Lee and Ko, 2020).

## 3. 실험 및 고찰

DDS 미들웨어 기반의 ECS 성능 실험은 구성된 ECS 축소 모형에서 토픽 크기에 따른 토픽 수신 오류를 측정하는 데 중점을 두었다. 실험을 위해 토픽 크기를 500 Byte부터 약 2 배씩 증가시켜 181 KByte까지 전송하고 오류를 측정하였다. 이는 DDS 분야에서 최대의 시장 점유율을 차지하고 있는 RTI사 Connex DDS의 최대 패킷 크기인 64 KByte의 3배 수준이다. 여기서 오류는 토픽의 손실 또는 수신 순서의 에러로 정의하였다.

성능 실험의 중심이 되는 DDS는 RTI사 Connex DDS 또는 PrismTech사 OpenSplice DDS 등이 많이 이용되고 있으며 무인체계와 CMS와 같이 주로 고성능을 요구하는 국방분야에서 주로 이용되고 있다. 선박의 ECS는 이에 비해 요구되는 성능이 상대적으로 낮고 라이선스 비용 등을 고려하여 OMG 표준을 따르는 오픈소스인 OpenDDS를 미들웨어로 선정하였다.

Fig. 3은 OpenDDS를 이용하여 함정의 ECS를 간략하게 구현한 ECS 축소모형 구성도이다. 축소모형은 Fig. 1에서 필드장비-DAU-ECS 서버 간의 연결 계통을 모사하기 위해 구성하였다. 하드웨어는 ECS 서버와 시뮬레이터 역할을 하는 랩

탑 3대와 이더넷 스위치 등으로 구성되어 있다. 필드장비 시뮬레이터는 ARM Cortex-M4 아키텍처를 적용한 ST사의 STM32F446 마이크로프로세서를 기반으로 제작하였으며, 축소모형 전체에 대한 구체적인 하드웨어 및 소프트웨어 사양은 Table 1과 같다.

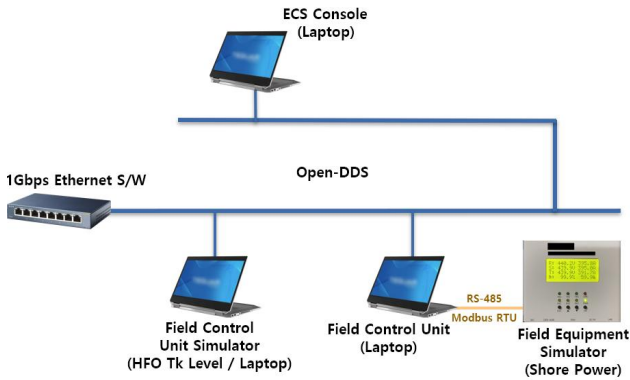


Fig. 3. Simulation setup for DDS performance test.

Table 1. Hardware and software configuration

Item	Specification
ECS console (Subscriber)	Laptop 1 EA (Intel Core i5-1135G7 @ 2.4 GHz)
Field control unit simulator (Publisher)	Laptop 1 EA (Intel Core i5-1135G7 @2.4 GHz)
Field control unit (Pulisher)	Laptop 1 EA (Intel Core i5-1135G7 @2.4 GHz)
Field equipment simulator	MICOM 1 EA (STM32F446 ARM, 180 MHz)
Ethernet switch	1 Gbps 8 port
Middle-ware	OpenDDS 3.20
OS	Windows 10 64 bit

OpenDDS는 중앙에 브로커가 있어 중앙화된 디스커버리를 지원하는 DCPS(Data-Centric Publish-Subscribe) 계층과 탈중앙화된 디스커버리 메커니즘을 운용하는 RTPS(Real-Time Publish Subscribe) 계층을 모두 지원한다. 이번 논문에서는 RTPS 계층을 이용하여 시험적으로 ECS를 구현하였다.

개발한 ECS 축소모형은 선박에서의 실제 상황을 모사하기 위해 두 가지 종류의 토픽(6개의 탱크 레벨 및 좌우현 육상전원 데이터)을 두 가지 방식으로 발간 및 수신하도록 구성하였다. 첫 번째로 탱크 레벨은 시뮬레이터(랩탑)에서 DDS상으로 직접 토픽을 발간하고, 두 번째로 육상전원 데이터(전압, 전류)는 시뮬레이터(MICOM)에서 RS-485 버스를 통해 Modbus RTU 프로토콜로 전송한 후, FCU(Field Control

Unit)인 랩탑과 통신한다. FCU 랩탑은 이를 토픽으로 발간하고 ECS 서버인 랩탑에서 수신하여 전시하는 구조로 구성되어 있다.

Fig. 4와 Fig. 5는 탱크레벨 데이터와 육상전원 데이터를 생성하는 시뮬레이터의 토픽 Publisher GUI(Graphic User Interface) 화면이다. 시뮬레이터는 설정된 범위 내에서 임의로 데이터를 생성하여 125 msec 간격으로 토픽을 발간한다.

Fig. 6은 ECS 콘솔에서 전시하는 Subscriber GUI 구성 화면으로 운용자가 실제로 보는 감시/제어 화면이다. 2대의 랩탑에서 발간한 토픽(탱크 레벨과 육상전원의 각 상별 전압과 전력)을 실시간으로 수신하여 Bar 차트와 수치로 전시토록 구성하였으며, 하단부분에는 토픽 발간, 구독 상황을 전시하였다.

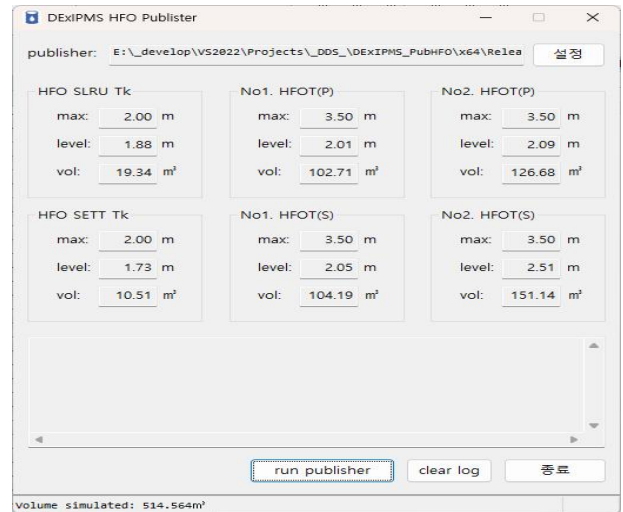


Fig. 4. Graphic user interface for HFO level publisher.

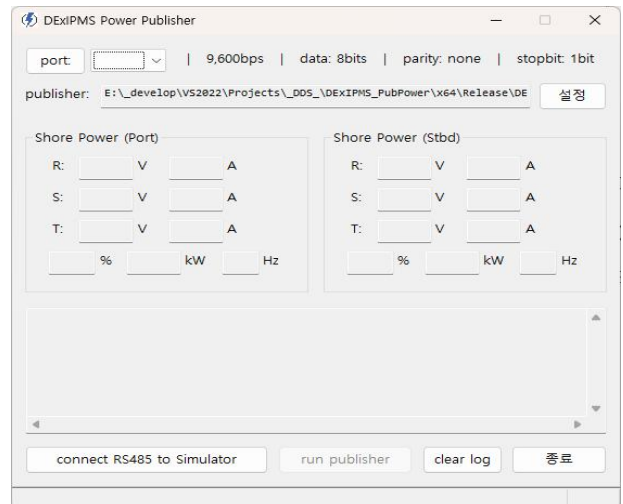


Fig. 5. Graphic user interface for shore power data publisher.

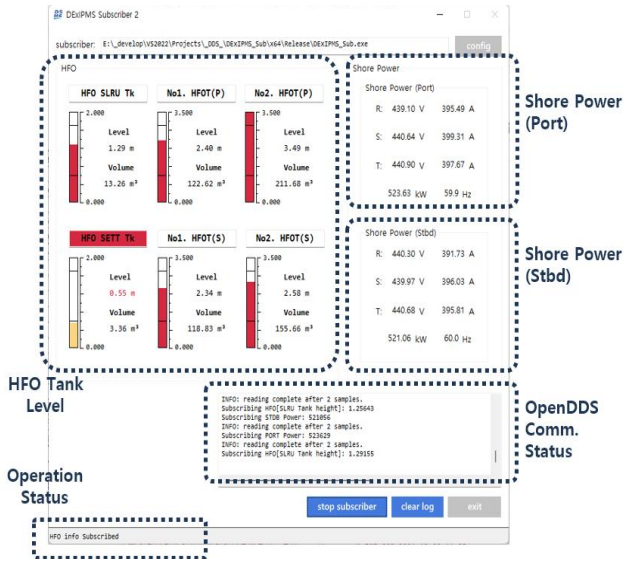


Fig. 6. Graphic user interface for ECS subscriber.

개발한 선박용 ECS의 성능 실험은 토픽 크기에 따른 토픽 오류를 측정하여 평가하였다. 토픽 크기는 500 Byte부터 약 2배씩 증가시켜 16 KByte, 64 KByte, 128 KByte, 152 KByte, 181 KByte까지 임의로 정하고 125 ms 간격으로 무작위로 각각 10 세트씩 반복하여 전송하였다.

단일 토픽의 최대 크기에 대해서 OMG에서 명시하는 제한은 없으나, 상용 제품군에서는 최대 크기를 64 KByte로 규정하고 이를 초과할 경우는 패킷을 분할하여 송신토록 규정하고 있다.

실험은 1:N 통신상황을 모사하고 있으며, 토픽 오류 발생을 확인하기 위해 각 토픽 별로 일련번호와 CRC(Cyclic Redundancy Check)를 기록한 뒤, 수신 측에서 이를 확인하였다. Fig. 7은 실험 결과를 기록한 분석용 엑셀파일 중의 일부이미지이다. 16 KByte 토픽에 대한 발간-구독 시간, CRC, 수신한 토픽 크기 등을 매 토픽마다 기록하여 분석하였으며, 송수신 시간차, CRC를 비교한 결과 모두 정상적으로 토픽을 송수신한 것으로 알 수 있다.

topic size	serial	serial max	timestamp(recv)	timestamp(send)	diff(send_recv)	CRC	total recv
16000	1	100	3858749997602	3858749995790	-1812	1	16000
16000	2	100	3858749997707	3858749995904	-1803	1	32000
16000	3	100	3858749997915	3858749996119	-1796	1	48000
16000	4	100	3858749998022	3858749996232	-1790	1	64000
16000	5	100	3858749998127	3858749996352	-1775	1	80000
16000	6	100	3858749998234	3858749996467	-1767	1	96000
16000	7	100	3858749998338	3858749996568	-1870	1	112000
16000	8	100	3858749998546	3858749996699	-1847	1	128000
16000	9	100	3858749998654	3858749996810	-1844	1	144000
16000	10	100	3858749998759	3858749996917	-1842	1	160000
16000	11	100	3858749998865	3858749997029	-1836	1	176000
16000	12	100	3858749998969	3858749997137	-1832	1	192000

Fig. 7. Image of test result for topic size of 16 KByte.

Fig. 8은 토픽크기를 500 Byte에서 181 KByte까지 증가시켰을 때의 결과(정상적으로 수신한 패킷 비율)를 종합한 것이다. 오픈소스인 OpenDDS를 기반으로 한 ECS 축소모형에서는 128 KByte까지는 정상적으로 발간-구독이 가능하며, 이를 넘어서부터는 급격하게 토픽 오류가 증가함을 알 수 있었다. 실험에서 토픽 손실은 발생하였으나 토픽 순서의 에러는 발생하지 않았다.

안정적으로 발간-구독하는 토픽 크기인 128 KByte는 ECS에 연결된 필드장비(예, 엔진, 발전기, 펌프 등)에서 보내는 데이터 종류, 크기와 형식을 고려하였을 때 충분한 크기이다. 또한 무기체계분야에서 독보적인 지위를 차지하는 RTI사 Connext DDS에서 권장하는 한 토픽 크기인 64 KByte에 비해 2배 크기이다. 따라서 OpenDDS 미들웨어를 기반으로 선박 ECS를 개발하는 것은 충분히 가능한 것으로 판단된다.

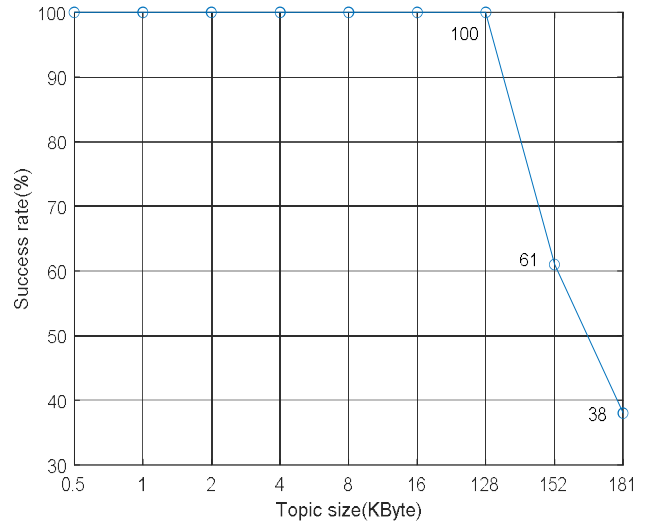


Fig. 8. Success rate of topic between publisher and subscriber.

#### 4. 결론 및 향후 과제

선박의 ECS는 엔진이나 필드장비를 총괄하여 제어하는 핵심 통제장치이다. ECS에는 많은 필드장비들이 연결되어 있고 각각 비동기적으로 데이터를 생성하여 송신하나, 데이터는 비교적 범위나 형식이 정형화(예, 온도, RPM, 전압 등)되어 있으며, 군용 무기체계와 같이 빠른 응답이 요구되지 않는다. 그러나, ECS는 민간에서 항해센서, 통신체계, 기관체계까지 모든 정보를 통합해야 하는 자율운항선박이 개발되고 있고, 함정에서는 CMS와의 통합과 같은 기술적인 요구사항의 변화에도 불구하고 기존의 Client-Server 모델에 기반하고 있다.

본 연구에서는 기존 방식에서 벗어나 최초로 DDS를 기반으로 선박의 ECS를 개발하였으며, 이를 축소모형을 통해 성능을 시험하였다. 선박의 ECS에 연결된 다수의 필드장비에서 비동기적으로 데이터를 송신하는 상황과 동일하게 시뮬레이션 환경을 구성하고 시험하였다. 개발한 ECS 축소모형은 오픈소스를 기반함에도 불구하고 ECS상에서 상용 DDS의 2배 수준인 128 KByte 크기의 토픽을 비동기적으로 송수신하는 데 문제가 없음을 확인하였다. 따라서 실제로 자율운항선박이나 다수의 체계 통합을 위해 DDS를 활용하거나 토픽을 설계할 경우에 도움이 될 수 있을 것이다.

향후에는 실제 선박의 ECS 상황을 최대한 고려하여 수천개의 토픽을 발간하고, 연결된 노드의 수가 변하는 상황에서의 데이터의 교환과 이종의 DDS 간의 데이터 교환에 대한 연구를 진행할 예정이다.

## References

- [1] Fall, K. R. and W. R. Stevens(2012), TCP/IP Illustrated, Volume 1, 2<sup>nd</sup> Edition, Addison-Wesley.
- [2] Im, J. Y.(2021), Performance Evaluation of Discovery and Message Transmission of DDS (Data Distribution Service) Security, Journal of the Korea Institute of Information and Communication Engineering, Vol. 25, No. 5, pp. 701-708.
- [3] Jang, J. H., D. J. Kim, M. G. Kim, and J. S. Oh(2017), Development of Naval Ship Propulsion System Simulator for CODLOG based ECS Verification, Journal of the Korea Institute of Information and Communication Engineering, Vol. 21, No. 9, pp. 1976-1807.
- [4] Jeong, Y. W.(2018), A Study on the Usages of DDS Middleware for Efficient Data Transmission and Reception, Journal of the Korea Society of Computer and Information, Vol. 23, No. 11, pp. 59-66.
- [5] Kim, J. H., S. H. Cha, and J. S. Shim(2021), Study for Component based ECS Control Monitoring SW Generation Tool, Journal of the Institute of Electronics and Information Engineers, Vol. 58, No. 11, pp. 1094-1100.
- [6] Kwak, S. C.(2021), A Study on the Standardization of On-Board Training System Software for Naval Ship Engineering Control System, Journal of the Korea Society of Computer and Information, Vol. 26, No. 9, pp. 97-104.
- [7] Lee, H. H. and K. S. Ko(2020), Comparative Analysis of Data Transmission Efficiency for JMS and DDS based Middlewares, Journal of Korean Institute of Information Technology, Vol. 18, No. 8, pp. 73-80.
- [8] Lee, J. W.(2017), Development of Message Define and Management System based on Distributed Processing Environment for Naval Combat Systems, KIISE Transactions on Computing Practices, Vol. 23, No. 12, pp. 670-676.
- [9] Liu, S., B. Xing, B. Li, and M. Gu(2014), Ship Information System: Overview and Research Trends, International Journal of Naval Architecture and Ocean Engineering, Vol. 6, No. 3, pp. 670-684.
- [10] Molina, J. R., S. Bilbao, B. Martinez, M. Frasher, and B. Cürüklü(2017), An Optimized, Data Distribution Service-Based Solution for Reliable Data Exchange Among Autonomous Underwater Vehicles, Sensors, Vol. 17, No. 8, pp. 1-32.
- [11] Oh, S. W.(2018), An Integrated Architecture for Control and Monitoring Systems on Naval Surface Combatants, Journal of the Korea Institute of Military Science and Technology, Vol. 21, No. 1, pp. 103-114.
- [12] OMG(2015), Data Distribution Service for Real-time Systems Version 1.4.
- [13] Park, G. S., B. C. Yoo, K. T. Kim, and B. W. Choi(2020), A Methodology for the Ship System Integration with Open Architecture : Focusing on the Total Ship Computing Environment based Architecture Building and Validation, Journal of Korean Society of Industrial and Systems Engineering, Vol. 43, No. 3, pp. 68-79
- [14] Park, Y. J. and S. A. Kim(2016), Research on Railway Safety Common Data Model and DDS Topic for Real-time Railway Safety Data Transmission, Journal of The Korea Society of Computer and Information, Vol. 21, No. 5, pp. 57-64
- [15] Peplinski, H.(2021), Ship and Mobile Offshore Unit Automation, Gulf Professional Publishing.
- [16] Pietzuch, P. R.(2004), Hermes: A Scalable Event-based Middleware, University of Cambridge, pp. 53-58.
- [17] PMS 406(2021), Unmanned Maritime Autonomy Architecture (UMAA) Sensor and Effector Management (SEM) Interface Control Document (ICD), Version 3.0.1.
- [18] Warzala, R.(2020), Modern Integrated Platform Management System Laboratory for Polish Naval Academy: Design And Implementation, Scientific Journal of Polish Naval Academy, Vol. 1, No. 2, pp. 59-69.

---

Received : 2023. 08. 02.

Revised : 2023. 09. 26.

Accepted : 2023. 10. 27.