

# 스마트선박을 위한 AS-Interface 프로토콜 활용 방안

김리곤<sup>†</sup> · 조영환 · 현기환  
KTE 기술연구소

## AS-Interface Protocol Utilization for Smart Ships

Ri Gon Kim<sup>†</sup> · Young Hwan Cho · Ki Hwan Hyun  
KTE of Technology & Research Institute

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the advent of the Fourth Industrial Revolution, many companies are developing Smart Ships to enable autonomous navigation, unmanned operation, and cost-effective shipping. In order to facilitate the operation of these Smart Ships, significant emphasis is being placed on improving external communication capabilities and developing autonomous navigation systems. Moreover, unmanned ship operation necessitates replacing of all visually observed conditions with sensors, leading to an inevitable increase in the number of sensors and actuators compared to traditional methods. However, in South Korea, the communication network for sensors and actuators primarily relies on the conventional 1:1 hardwired wiring, with the use of higher-level communication networks when necessary. Therefore, in this paper, we introduce the AS-Interface protocol, which is used for monitoring and controlling sensors and actuators, with the expectation that it will greatly contribute to the development of future Smart Ships by reducing production time, costs, and maintenance duration.

**Keywords** : AS-Interface(Actuator and Sensor Interface), Smart ships(스마트선박), Hardwired(1:1 배선)

## 1. 서론

현재의 급변하는 디지털 시대와 4차 산업혁명은 다양한 산업 분야에서 혁신적인 변화를 가져오고 있다. 특히 해운 산업은 이러한 변화에 크게 영향을 받고 있으며, 이에 대응하기 위해 스마트선박(smart ships)이라는 개념이 주목받고 있다. 스마트선박은 자율운항, 무인운항, 경제운항 등의 장점을 갖추어 기존의 선박 운영 방식을 혁신하고 효율성을 극대화하는 것을 목표로 한다.

스마트선박은 선박 운항에 있어서 사람의 개입 없이 또는 최소한의 개입으로 운항하는 선박 및 그 선박의 안정적 운항에 필요한 제반 인프라를 의미한다 (Kim and Jang, 2019). 이를 위해 스마트선박은 다양한 센서와 통신 시스템을 활용하여 외부 환경을 모니터링하고 상황에 맞게 자율적으로 조치를 취할 수 있도록 설계된다. 또한, 선박 내부의 다양한 시스템과 장치들도 스마트한 제어와 감시를 위해 연결되어야 한다.

이러한 스마트선박의 운용을 위해서는 뛰어난 통신 성능과 자율운항 시스템의 개발이 중요한 과제로 떠올랐다. 특히, 스마트

선박의 무인 운항을 위해서는 기존에 육안으로 확인하던 다양한 상태들을 센서로 대체해야 하며, 따라서 센서와 구동기의 수가 증가할 수밖에 없다. 그러나 국내에서는 기존의 1:1 배선 방식인 Hardwired를 주로 사용하고 있으며, 필요에 따라 상위 레벨의 통신망을 도입하는 정도로 제한되어 있다.

본 논문에서는 이러한 문제를 해결하기 위해 AS-Interface 프로토콜을 소개하고자 한다. AS-Interface는 센서 및 구동기의 감시와 제어에 사용되는 효율적인 프로토콜로서, 스마트선박의 제작시간 단축, 비용 절감, 유지보수 시간 감소 등 큰 도움이 될 것으로 기대한다. AS-Interface 프로토콜은 기존의 1:1 배선 방식보다 훨씬 효율적이며, 다양한 센서와 구동기를 단일 통신 네트워크에 연결하여 관리할 수 있다. 이를 통해 스마트선박의 운용 및 유지보수 과정에서 발생할 수 있는 복잡성과 비용을 최소화할 수 있다.

뿐만 아니라 AS-Interface 프로토콜은 신뢰성과 안정성에 대한 높은 수준의 보장을 제공한다. 선박 운항은 매우 까다로운 환경에서 이루어지기 때문에 신뢰할 수 있는 통신 시스템이 필수적

이다. AS-Interface는 강력한 오류 감지 및 오류 처리 기능을 갖추고 있어 장애 발생 시 신속하게 대응할 수 있다.

또한, AS-Interface는 쉬운 설치와 유연한 확장성을 제공한다. 기존의 하드웨어 배선 방식에 비해 간편하게 설치할 수 있으며, 새로운 센서나 구동기를 추가하거나 기존의 구성을 변경하는 작업도 용이하다. 이는 스마트선박의 시스템 업그레이드나 확장에 매우 유용한 장점이다.

따라서, 본 논문은 AS-Interface 프로토콜의 소개와 적용 사례를 통해 스마트선박에 대한 활용 가능성을 탐구함으로써, 선박 산업에서의 혁신과 경쟁력 향상에 기여할 것으로 기대한다.

## 2. 스마트선박

### 2.1 스마트선박 개념

스마트선박은 현대의 디지털 시대와 4차 산업혁명의 도래로 발전된 선박 기술의 새로운 패러다임을 대표하는 개념이다. 기존의 선박 운영 방식을 혁신하고 선박의 자동화와 인공지능 기술을 활용하여 더욱 안전하고 효율적인 운항을 실현하기 위해 개발되었다. 스마트선박은 자율운항, 무인운항, 경제운항 등의 특징을 갖추고 있어 선박 산업의 미래를 모색하고 혁신적인 변화를 가져오고 있다.

스마트선박은 첨단 센서, 통신 기술, 인공지능, 자율운항 등 다양한 혁신적인 기술을 통합하여 구현된다. 선박 내부와 외부의 다양한 환경 요소를 모니터링하고 데이터를 수집함으로써 운항 상황을 실시간으로 파악할 수 있다. 이를 통해 선박 운항 중에 발생할 수 있는 위험 요소를 사전에 예측하고 대응하는 능력을 갖출 수 있다. 또한, 스마트선박은 선박의 성능, 에너지 효율, 유지보수 등을 최적화하기 위한 데이터 분석과 자동화된 제어 시스템을 구축하여 경제운항을 실현한다. 현재도 인공위성을 이용한 선박위치 계산과 항로의 자동제어, 무인자동조타 등 항해의 편의를 도모하는 조선, 기관실 장비의 기능이 고도화되었다 (Ryu, 2018).

자율운항은 스마트선박의 핵심 개념 중 하나로, 인간의 개입을 최소화하고 선박이 자체적으로 항로를 선택하고 목적지에 도달할 수 있는 능력을 의미한다. 선박은 다양한 센서와 인공지능 기술을 활용하여 주변 환경을 감지하고 이에 따라 적절한 항로 및 조치를 선택한다. 이는 선박 운항의 안전성을 향상시키고 인력 및 비용을 절감할 수 있는 장점을 제공한다 (Yoo et al., 2018).

무인운항은 스마트선박의 또 다른 중요한 개념으로, 선박 운항 시 인간의 참여 없이 자동화된 시스템에 의해 운항되는 것을 의미한다. 스마트선박은 자율주행 기술과 자동화된 조종 시스템을 활용하여 인간의 개입 없이 선박을 운항할 수 있다. 이는 인력 비용을 절감하고 운항의 안전성을 높이는 동시에 근무 환경을

개선할 수 있는 장점을 제공한다.

경제운항은 스마트선박의 핵심 가치 중 하나로, 선박 운항 과정에서 발생하는 비용을 최소화하고 수익을 극대화하는 것을 목표로 한다. 스마트선박은 선박의 성능을 최적화하고 연료 효율을 향상시키는 기술을 적용하여 연료 비용을 절감하고 친환경적인 운항을 실현한다. 또한, 데이터 분석과 예측 모델을 활용하여 유지보수 일정을 최적화하고 장애를 예방함으로써 유지보수 비용을 줄일 수 있다.

이러한 개념과 기술들을 통해 스마트선박은 안전성, 생산성, 경제성 등 다양한 측면에서 기존의 선박 운항 방식과 비교하여 큰 변화와 혁신을 가져올 것으로 기대된다. 또한, 스마트선박은 해운 산업의 경쟁력을 강화하고 더욱 지속 가능한 선박 운항을 실현하는 데에 기여할 것으로 예상된다.

### 2.2 스마트선박의 개발

스마트선박의 개발은 현재 4차 산업혁명의 주요 흐름 중 하나로, 많은 기업들이 자율운항, 무인운항, 경제운항을 위해 노력하고 있다. 이러한 개발은 선박 운항의 효율성, 안전성, 경제성 등을 향상시키는 것을 목표로 하며, 다양한 기술과 시스템을 통합하여 구현된다.

스마트선박 개발의 핵심은 외부와의 통신 성능 향상과 자율운항 시스템의 개발에 주력하고 있다. 외부와의 통신 성능은 선박 내부의 센서 데이터 및 운항 정보를 신속하고 정확하게 전달하기 위해 필수적이다. 고성능의 통신 시스템을 개발하여 실시간 데이터 흐름과 원격 모니터링을 가능하게 함으로써 선박 운항의 효율성과 안전성을 향상시킨다.

또한, 스마트선박의 개발을 위해서는 자율운항 시스템을 중심으로 연구하고 있다. 자율운항 시스템은 인공지능, 로봇 공학, 센서 기술 등 다양한 분야의 기술을 융합하여 선박의 운항을 자동화하고 인간의 개입을 최소화하는 시스템이다. 이를 위해 선박의 운항 경로 설정, 충돌 회피, 자동 조종, 항로 추적 등에 필요한 알고리즘과 센서 기술을 개발하고 적용함으로써 자율운항이 가능한 스마트선박을 구현하고자 한다.

스마트선박은 첨단 기술을 활용하여 운항의 효율성과 안전성을 향상시키는 선박 시스템을 말한다. 이를 위해 다양한 센서가 사용된다. 다음은 선박에서 주로 사용되는 센서의 종류이다.

이 외에도 조도 센서, 스트레인 게이지, 초음파 센서, 자기 센서 등 다양한 종류가 있으며, 선박의 운항 조건과 요구 사항에 따라 다양하게 사용된다. 이러한 센서들은 선박의 자율운항, 안전성 향상, 에너지 효율성 등 다양한 측면에서 핵심적인 역할을 수행한다. 스마트선박의 개발을 위해서는 센서 및 구동기의 수증가가 필요한데, 이는 인간이 육안으로 확인하던 상태를 센서로 대체하기 때문이다.

따라서, 본 논문에서는 자동화 하위레벨의 센서와 구동기를 효과적으로 연결하고 제어하기 위한 통신 프로토콜인 AS-

Table 1 Sensor type

Category	Explanation
Temperature sensor	A sensor that detects temperature
Pressure sensor	A sensor that detects pressure changes
Humidity sensor	A sensor that measures the humidity level in the surrounding environment
Accelerometer and gyroscope	Sensors that measure the motion and position
Light sensor	A sensor that detects the ambient light level
Gas and chemical sensors	Sensors that detect the concentration of specific gases or chemicals
Noise and vibration sensors	Sensors that detect surrounding noise or vibrations

Interface를 소개함으로 스마트선박의 제작 시간, 비용, 유지보수 시간 등을 단축하고 스마트선박의 개발을 원활하게 진행할 수 있는 방안을 제시한다.

### 3. AS-Interface

#### 3.1 AS-Interface 개념

AS-Interface(Actuator Sensor Interface)는 자동화 시스템에서 센서와 구동기를 효율적으로 연결하기 위한 산업용 네트워크 프로토콜이다. 주로 제조업체에서 자동화 프로세스를 구현하는데 활용되며, 설치와 유지보수가 용이하면서도 비용을 최소화하는 것을 목표로 한다 (Rolf, 2006).

#### 3.2 AS-Interface 특징

AS-Interface는 Master 모듈과 Slave 모듈로 구성되어 있다. Master 모듈은 PLC 등 상위 장비에 연결되어 Slave 모듈로부터 수집된 데이터를 상위 장비에 전달하고 상위 장비로부터 제어 신호를 Slave 모듈로 전달한다. Slave 모듈은 센서 및 구동기 인접한 위치에 설치되며, 이 때 AS-Interface Master 모듈과 Slave 모듈 간 전력선 통신을 통해서 데이터를 송수신한다. 일반적으로 1:1배선방식인 Hardwired로 구성 시 Fig. 1 과 같이 센서 및 구동기까지 개별적으로 선이 가설되어야 한다.

반면 AS-Interface를 적용할 경우 Fig. 2와 같이 PLC와 센서 인접까지 Master 모듈과 Slave 모듈이 전력선 통신으로 구성된다.

따라서 AS-Interface는 Fig. 2와 같이 단순하고 경제적인 설

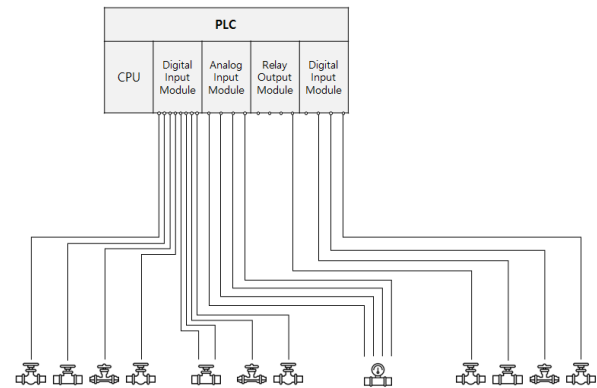


Fig. 1 Schematic diagram before applying the AS-Interface

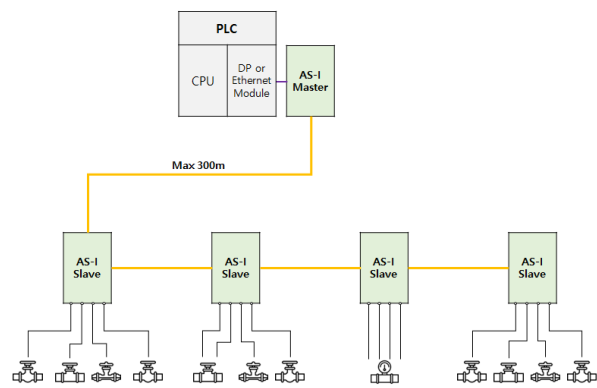


Fig. 2 Schematic diagram after applying the AS-Interface

치와 연결 방법을 제공하며, 센서와 구동기의 감시 및 제어를 단일 통신 라인을 통해 수행할 수 있다. 기존의 1:1 배선 방식에 비해 간편하고 효율적이다. 이를 통해 복잡한 배선 작업을 최소화하고 설치 시간과 비용을 절감할 수 있다.

AS-Interface는 다양한 종류의 센서와 구동기를 지원하며, 디지털 및 아날로그 신호를 처리할 수 있다. 이를 통해 선박 내부의 다양한 장치와 시스템을 통합하고, 센서로부터 수집한 데이터를 중앙 제어 시스템으로 전달하여 선박 운항 상황을 모니터링하고 제어할 수 있다.

AS-Interface는 신뢰성과 안정성에 대한 높은 수준의 보장을 제공한다. 오류 감지와 오류 처리 기능을 내장하고 있어 장애 발생 시 신속하게 대응할 수 있으며, 프로토콜은 간섭 방지 기능을 갖추고 있어 선박 운항 중에 발생할 수 있는 외부 요인으로부터의 영향을 최소화한다.

AS-Interface는 유연한 확장성을 제공하며, 기존 시스템에 쉽게 통합될 수 있다. 새로운 센서나 구동기를 추가하거나 기존의 구성을 변경하는 작업이 용이하며, 선박의 시스템 업그레이드나 확장에 유용하다.

따라서, AS-Interface는 선박 산업에서 스마트선박의 센서 및 구동기 감시 및 제어에 활용되어 제작시간, 비용, 유지보수 시간 등 다양한 측면에서 혜택을 제공하는 효율적인 프로토콜이다.

### 3.3 AS-Interface 장단점

앞서 기존 1:1배선방식인 Hardwired를 적용한 구성도 Fig. 1과 AS-Interface를 적용한 구성도 Fig. 2를 비교하였을 때 케이블 경감이 효과적임을 알 수 있다. Master모듈 당 최대 62개의 Slave모듈을 장착할 수 있는데, 이는 Slave모듈이 300m에 위치하고 모두 디지털 입력 모듈일 경우 300m의 선로의 개수가 496가닥에서 27가닥으로 기존대비 최대 0.4%까지 경감됨을 알 수 있다.

또한, AS-Interface는 케이블 선택 시 많은 장점을 가지고 있다. 기존의 통신 방식에서는 데이터 전송률에 따라 케이블에 높은 요구사항이 필요했으며, 실딩(shielding) 기능을 통해 전송 특성과 데이터 전송 간 방사되는 노이즈를 차단해야 했다. 그러나 AS-Interface는 PELV(Protective Extra-Low Voltage) 시스템의 규격에 따라 실딩이 필요하지 않다. 이는 케이블 비용 절감과 다양한 위치에 효율적으로 사용할 수 있는 장점을 제공한다.

AS-Interface 제품은 전자파 적합성(EMC) 검사 기준을 충족하며, Table 1과 같이 규정된 표준을 따른다. 따라서 운용자는 전자파 영향에 안정적으로 운용할 수 있다. AS-Interface는 EN 50295와 IEC 62026-2 표준규격으로 개발되었으며, 이는 개방형 시스템 이점을 가지고 있다. 다양한 제조사 간에 상호운용성이 보장되므로 필요에 따라 타 제조사의 장비를 혼용하여 최대 효율을 이끌어낼 수 있다.

그러나 AS-Interface의 데이터 전송량은 한 주기에 Slave별 4비트로 제한되어 있다. 따라서 4비트를 초과하는 메시지는 여러 주기를 걸쳐 분할하여 전송된다. 이는 온도나 레벨과 같이 느리게 변화하는 파라미터를 설정할 때에는 적합하지만, 고도의 동적인 제어 루프에는 적합하지 않을 수 있다.

Table 2 AS-Interface EMC inspection criteria

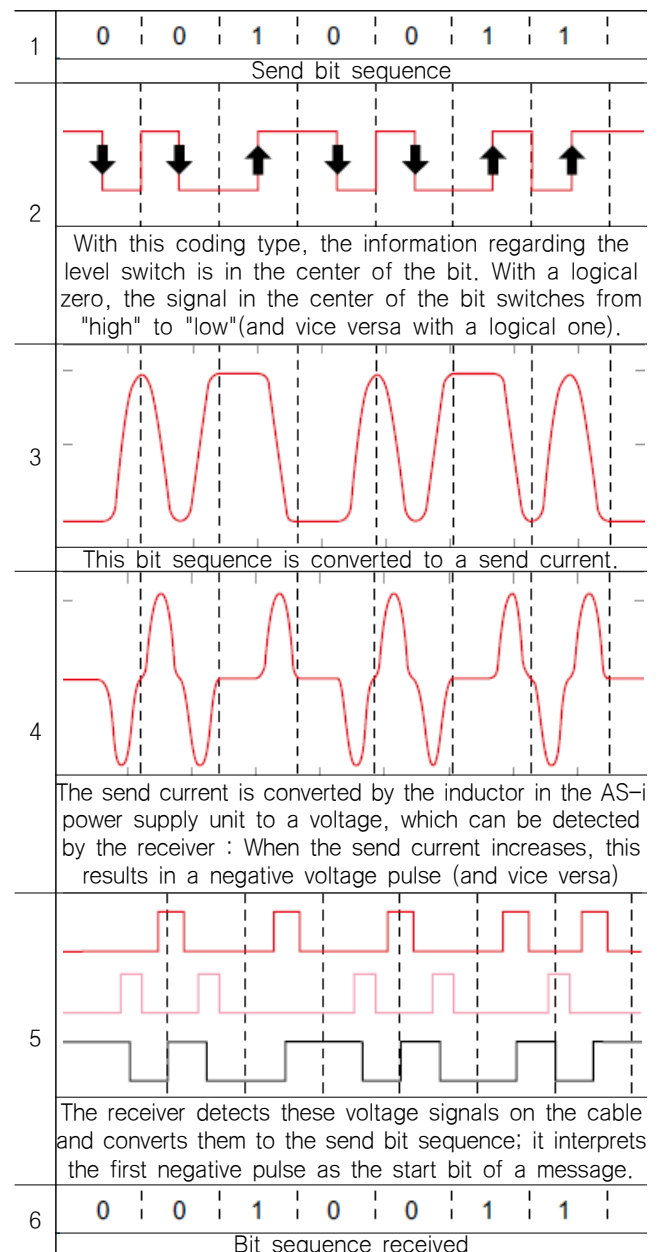
Environmental impact	Test level	Reference Rating	Standard Specifications
Electrostatic emissions	Contact 4kV Contactless 8kV	B	EN 61000-4-2
High frequency (HF) field, amplitude modulation	80 ~ 1000MHz 10V/m 80% Amplitude modulation	A	EN 61000-4-3
High frequency (HF) field, pulse modulation	900±5MHz 10V/m 50% ED 200Hz Repeating frequency	A	EN 61000-4-3
Asymmetric high-speed overcurrent	2kV(Peak) 5/50ns tr/tn 5kHz Repeating frequency	B	EN 61000-4-4
Asymmetric high frequency	0, 15~80MHz 10V 80% Amplitude modulation(1kHz) 150Ω source impedance	A	EN61000-4-6

또한, AS-Interface의 네트워크 길이는 최대 100m로 제한된다. 이는 물리적인 한계로 개발 과정에서 타협된 부분이다. 네트워크를 확장하기 위해서는 리피터(repeater)가 필요하며, 최대 2개의 리피터를 연속하여 Master 장비로부터 300m까지 연결할 수 있다.

### 3.4 AS-Interface 통신 시스템

AS-Interface는 베이스 밴드 내 직렬 전송을 위해 자체 개발한 교류 펄스 변조(APM)를 사용하고 있다.(Table 2 참고) 비트를

Table 3 The individual steps involved in AMP



송신하는 과정에서 비트 신호가 변할 때마다 위상 변화를 수행하여 원하는 신호 전압 레벨을 생성한다. 이를 위해 송신 전류는 시스템 내의 단일 도체와 미분을 사용하여 AS-Interface 케이블 상에 신호를 생성한다. 이러한 과정으로 송신 전류의 상승 및 하강 시 음전압 및 양전압 펄스가 발생하게 된다. 이 방식을 통해 Slave 모듈 내에서는 실제 공급 전압보다 더 높은 전압을 가진 신호를 생성하게 된다.

수신기 측에서는 라인 상에서 검출되는 전압 펄스 중 최초 음전압 펄스를 기준으로 메시지 시작 비트를 동기화하고, 나머지 전압 펄스를 통해 비트를 해석한다. 이 변조 과정을 통해 6μs의 비트 타임을 얻을 수 있으며, 전체 전송률은 167kBit/s를 지원할 수 있다. 또한, 시스템 내의 단일 도체를 이용한 신호 발생으로 Slave 내에 별도의 도체가 필요하지 않아지며, 이는 센서나 구동기에 통합된 전자 장치의 소형화와 가격 경쟁력을 향상시킬 수 있다.

AS-Interface Slave의 응답은 전원 공급 장치간의 거리에 따라 진폭의 크기가 달라진다. 전원 공급 장치 간 거리가 가까울수록 최소 진폭을 가지며, 거리가 멀어질수록 진폭이 커지는 것을 Fig. 3과 Fig. 4에서 확인할 수 있다.

다양한 Slave 응답 메시지의 진폭을 처리하기 위해 수신기는 최대 전압의 35%의 진폭 오차와 최대 전압의 30% 이내의 오버슈트를 처리할 수 있다. 이러한 진폭 오차 처리는 비종단 케이블을 사용함으로써 발생하는 반사 현상에도 메시지 처리가 가능하게 한다.

AS-Interface 메시지는 Fig. 5에 나타난 바와 같이 Master 요청, Master 휴지기, Slave 응답, Slave 휴지기로 구성된다. 모든 Master 요청의 길이는 14비트 타임이고 모든 Slave 응답은 7비트 타임이다(1비트 타임 = 6μs).

Master 휴지기는 최소 2비트 타임에서 최대 10비트 타임까지 가질 수 있으며, Slave는 동기화 후 2비트 타임이 끝나면 응답 전송을 시작할 수 있다. 만약 Slave 응답 시작 비트가 Master 요청 후 10비트 타임까지 응답이 없을 경우 Master는 다시 요청

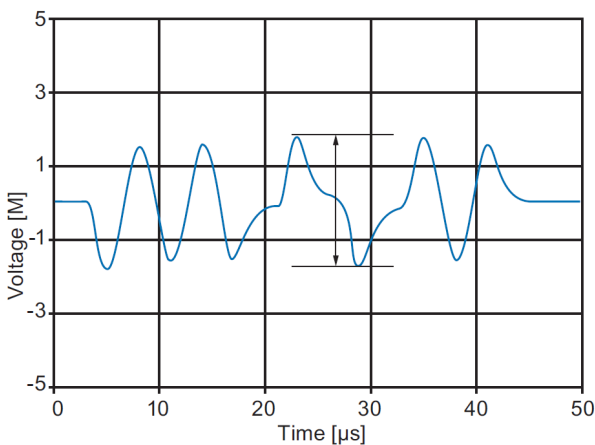


Fig. 3 Response from slave near power supply unit

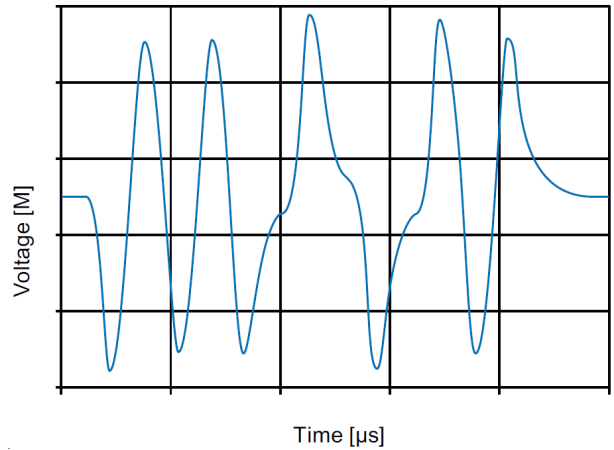


Fig. 4 Response from slave far from power supply unit

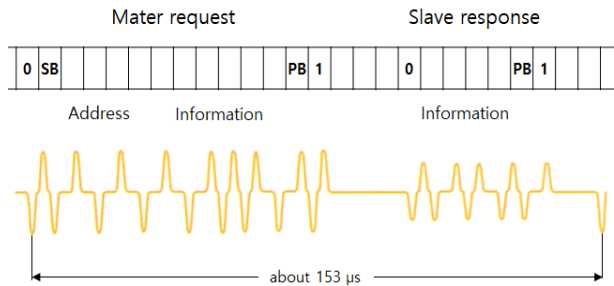


Fig. 5 AS-Interface message structure

신호를 보낸다. Slave 응답 종료와 그 다음 Master 요청 사이의 휴지기의 길이는 1.5~2비트 시간을 초과하지 않는다.

Fig. 5에 나타난 신호 시퀀스를 분석하면 비트 시퀀스 00101010111011 [bin]을 나타낸다. 이는 주소 21과 사용자 데이터 E[hex]를 가지는 Slave에 대한 데이터 요청을 의미한다. Slave는 비트 시퀀스 0011001[bin]로 응답하며 이에는 사용자 데이터 6[hex]가 포함된다.

관련된 Master 요청 규칙은 Table 3과 같다.

Table 4 Master request data map

Category	Explanation
Start bit(ST)	Start master request(0:Valid, 1:Invalid)
Control bit(CB)	0:Data requests/parameter requests/address assignments, 1:control request
Address(A0~A4)	Address of the contacted slave
Information(I0~I4)	Information to be sent to the slave
Parity bit(PB)	The number of 1s in the mast request signal must be even.
End bit(EB)	End of master request(0:Invalid, 1:Valid)

Master 요청 신호는 Table 3에 따라 결정되며, 정보 비트 (I0~I4)에 따라 Master 요청 정보가 결정된다. Table 4를 통해 Master 요청 비트에 따라 어떤 정보를 요청하는지 확인할 수 있다. 다만, 정보 비트 I3의 경우 선택 비트(S)에 관련이 있다. AS-Interface 규격 2.1(1998) 이전의 Master는 선택 비트를 0으로 고정하였으며(S=0, S=1), 확장 주소 이후 AHEX 주소인 경우 선택 비트를 0으로 지정하고 BHEX 주소인 경우 선택 비트를 1로 지정하여 구분하였다. 이는 확장 주소 AHEX가 기본 주소와 동일한 Master 요청을 가지도록 설계되었음을 의미한다.

Slave의 응답 구조는 Table 5 과 같다. Master 요청에 따라 4 비트의 정보(I0~I3)를 회신한다.

Table 4을 통해서 Master 요청별 비트 구성 방식을 알아보았다. 해당 비트 조합을 통해서 Slave에 명령을 요청하는데, 명령에 관한 상세내용을 아래와 같다.

Table 5 Master request data map(detail)

Category	ST	CB	Address						Information					PB	EB
			A4	A3	A2	A1	A0	I4	I3	I2	I1	I0			
Data request	0	0	A4	A3	A2	A1	A0	0	S	D2	D1	D0	PB	1	
Parameter request	0	0	A4	A3	A2	A1	A0	1	S	D2	D1	D0	PB	1	
Address assignment	0	0	0	0	0	0	0	A4	A3	A2	A1	A0	PB	1	
Command req: Write ID code1	0	1	A4	A3	A2	A1	A0	0	D3	D2	D1	D0	PB	1	
Command req: Delete address	0	1	A4	A3	A2	A1	A0	0	S	0	0	0	PB	1	
Command req: Slave reset	0	1	A4	A3	A2	A1	A0	1	S	1	0	0	PB	1	
Command req: Read I/O config	0	1	A4	A3	A2	A1	A0	1	S	0	0	0	PB	1	
Command req: Read ID code	0	1	A4	A3	A2	A1	A0	1	S	0	0	1	PB	1	
Command req: Read ID code1	0	1	A4	A3	A2	A1	A0	1	S	0	1	0	PB	1	
Command req: Read ID code2	0	1	A4	A3	A2	A1	A0	1	S	0	1	1	PB	1	
Command req: Read status	0	1	A4	A3	A2	A1	A0	1	S	1	1	0	PB	1	
Broadcast comm.(reset)	0	1	1	1	1	1	1	1	0	1	0	1	PB	1	

Table 6 Slave data map

Category	Explanation
Start bit(ST)	Start slave response(0:Valid, 1:Invalid)
Information(I0~I3)	4bit information
Parity bit(PB)	The number of 1s in the slave response signal must be even.
End bit(EB)	End of slave response(0:Invalid, 1:Valid)

가) 데이터 요청

데이터 요청은 평상시 Master와 Slave 구성을 마친 상태에서 Slave의 데이터를 읽거나 제어하는 명령이다. 이를 통해서 센서 및 구동기 제어를 수행한다. Slave는 Master와의 통신이 50ms 이상 중단될 경우 출력을 리셋하는 워치독 기능을 가지고 있으며, Slave의 주소가 0일 경우 데이터 요청 명령은 불가하다.

나) 파라미터 요청

파라미터 요청은 비주기적으로 출력 신호에 대한 명령을 전송하여 일정 기능을 원격 제어할 수 있다. 예를 들어 타이밍 기능 활성화, 센서 스위칭 거리 변경, 측정 절차 변경, 출력 신호 확인 등 다양한 기능을 원격으로 제어할 수 있으며, 단 Slave 주소가 0일 경우 해당 명령은 수행되지 않는다.

다) 주소 지정

주소 지정은 Master가 기본주소(0)을 가지고 있는 Slave의 주소를 새로운 주소로 지정 하는 기능이다.

라) ID 코드1 읽기/쓰기

해당 ID 코드1의 경우 운용자가 동일한 모듈과 주소를 사용할 때 Slave들간 식별을 위해 제공된다. 따라서 Master 명령에 의해 ID 코드1을 쓰고 읽을 수 있다.

마) 주소 삭제

주소 삭제 명령은 Slave의 주소를 일시적으로 삭제하는데 사용된다. 기본주소가 아닌 동작주소의 주소를 변경을 위해 해당 기능을 사용하는데, 해당 명령을 통해 동작주소의 주소를 삭제하여 기본주소(0)으로 변경하고 Master 명령 주소 지정을 통해 새로운 주소를 지정한다.

바) Slave 리셋

Slave 리셋 명령은 Slave를 기본 상태로 설정하는 기능이다.

사) I/O 구성 읽기

I/O 구성 읽기는 Slave의 설정된 I/O 구성을 읽는 기능이다. Table 6와 같이 응답 코드를 통해서 Slave의 구성을 확인할 수 있으며 IN은 입력, OUT은 출력, I/O는 입력/출력, TRI는 기능 없음을 의미한다. TRI의 경우 I/O 구성을 찾을 수 없을 때 발생한다.

Table 7 I/O configuration code

Code	D0	D1	D2	D3
0[HEX]	IN	IN	IN	IN
1[HEX]	IN	IN	IN	OUT
2[HEX]	IN	IN	IN	I/O
3[HEX]	IN	IN	OUT	OUT
4[HEX]	IN	IN	I/O	I/O
5[HEX]	IN	OUT	OUT	OUT
6[HEX]	IN	I/O	I/O	I/O
7[HEX]	I/O	I/O	I/O	I/O
8[HEX]	OUT	OUT	OUT	OUT
9[HEX]	OUT	OUT	OUT	IN
A[HEX]	OUT	OUT	OUT	I/O
B[HEX]	OUT	OUT	IN	IN
C[HEX]	OUT	OUT	I/O	I/O
D[HEX]	OUT	IN	IN	IN
E[HEX]	OUT	I/O	I/O	I/O
F[HEX]	TRI	TRI	TRI	TRI

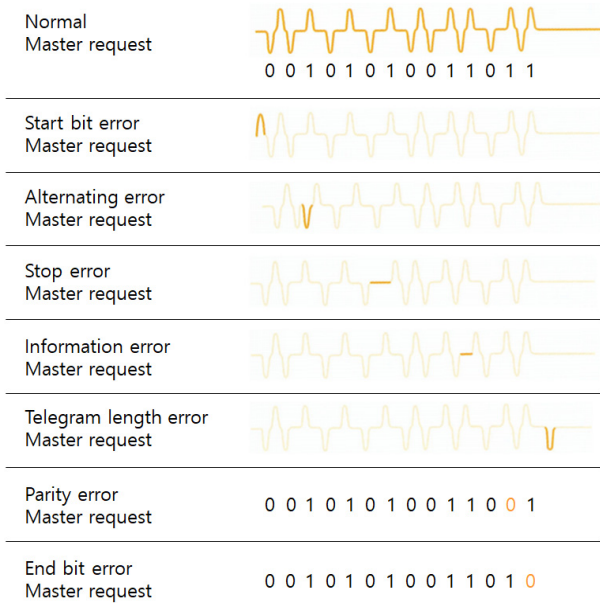


Fig. 6 Error detection at bit level

AS-Interface에서는 Master와 Slave 간 신호 길이가 매우 짧기 때문에 일반적인 버스 시스템에서 사용되는 오류 검출 방법과는 다른 방법을 사용하여 오류를 검출한다. 일반적인 버스 시스템에서는 4해밍 거리(HD=4)를 사용하여 오류를 확인한다. 이 경우 Master 송신 신호를 위해 7비트와 Slave 응답 신호를 위해 4비트를 추가적으로 전송해야 한다. 그러나 AS-Interface에 적용 시 신호 길이가 기존 대비 75% 증가하여 네트워크 전송 속도가 크게 감소하는 단점이 있다. 따라서 AS-Interface에서는 비트 전송 자체에 오류 확인 작업을 수행하며, 코드의 중복과 신호 길이에 대한 정보를 활용한다. Fig. 6 과 같이 코드의 중복과 길이에 따라 시작 비트, 교번, 중지, 정보, 패리티, 종료 비트, 길이 오류를 구별할 수 있다. 모든 Master 요청과 Slave 응답은 이 확인 과정을 거치게 된다. 오류 확인 작업은 처리 시간에 거의 영향을 주지 않는 간단한 논리 연산을 통해 수행되며, 위에서 언급된 오류 중 하나라도 검출되면 요청은 오류가 있는 것으로 판단한다.

### 3.5 AS-Interface 적용 사례

#### 가) 식품 가공(발효)

Cerestar Deutschland GmbH는 발효 기능을 갖춘 식품 가공 시스템을 설계 및 제조업체이며, 400개 이상의 격판 밸브를 사용하고 있는 GEMU의 발효조는 해당 시스템의 핵심이다. 해당 시스템에는 Fig. 7 과 같이 AS-Interface를 적용하였다. 밸브와 리프트 드라이브뿐만 아니라 모든 파트-턴 밸브 액추에이터들에도 GEMU-AS-Interface 제어 헤드가 장착되었다. 간편한 설치 방식으로 설치비용 절감, 설계 및 문서화 작업의 부담이 현저히 낮아지는 이점을 가질 수 있었다.

#### 나) 안전문(Safety at work)

Lemforder Fahrwerktechnik AG & Co. KG는 자동차 제조업

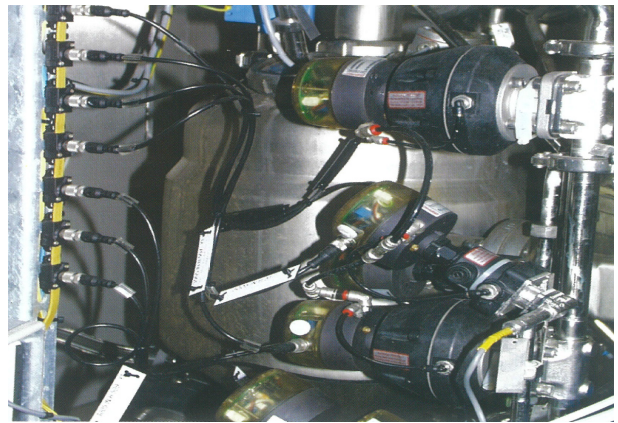


Fig. 7 Simple valve connection on the AS-Interface string



Fig. 8 Assembly cell with AS-Interface

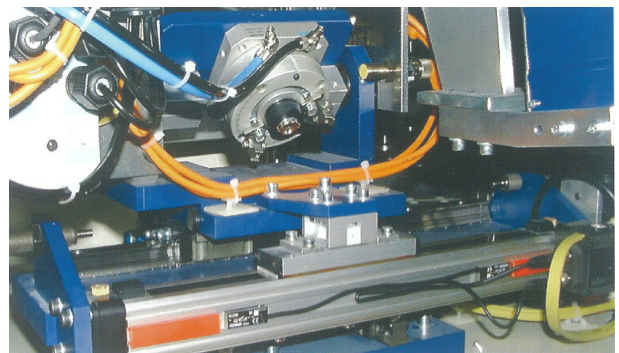


Fig. 9 AS-Interface actuator on rotary index table

체이며, EUCHNER GmbH+Co,에 의한 버스 안전기술, 조립 라인 안전 관련 기술을 AS-Interface를 통해 구현하였다. Fig. 8 과 같이 AS-Interface 적용으로 안전 관련 센서 및 구동기간 배선 관련 비용 절감하였고, 복잡성의 감소에 따른 부품 수 감소 및 오류 발생 시 빠른 처리가 가능해졌다. 이를 통해 배선 비용이 약 70% 감소하였고, 제어 케비닛 소형화 및 신뢰성이 높은 시스템을 개발하였다.

#### 다) 자동차 부품 조립 시스템

Horbigler-Origa는 승용차 냉각 시스템 부품 조립을 위해 회전

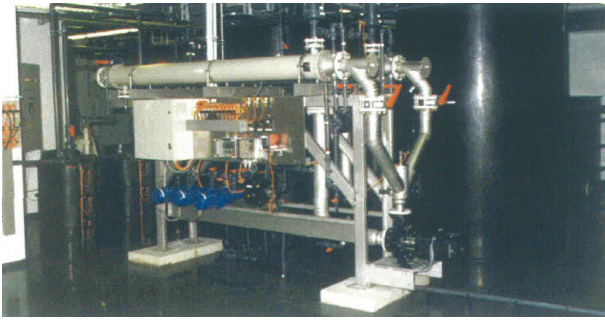


Fig. 10 AS-Interface in micro filtration systems

인덱스 테이블을 이용한 자동차 부품 조립 시스템 제어를 Fig. 9와 같이 AS-Interface를 통해 구현했다. 조립 시스템 내에는 총 10개의 쌍방향 드라이브 유닛과 추가적인 20개 공압식 에어박스 시스템이 AS-Interface를 통해 제어된다. 이를 통해 설치 시간 감소 및 공간 절약하였으며, 간편한 문제 해결 방식으로 빠른 유지보수가 가능하다.

#### 라) 생물학적 자동화 폐수 처리설비

Lake Constance, Wasserburg의 생물학적 폐수 처리 설비는 ifm 센서의 포일 생상간 발생하는 폐수 내 구리를 비롯한 중금속과 기타 위험 물질을 제거하기 위한 설비이다. 본 시스템의 제어는 Fig. 10과 같이 AS-Interface를 이용하여 구현되었으며, PROFIBUS-DP를 통해서 빌딩 서비스 관리 시스템에 연결되어 시스템의 모든 신호를 전력선을 통한 중앙 제어 유닛에 의해 처리된다. Bio-Substrate 공정은 기존의 물리 화학적 설비와 비교했을 때, 폐수 처리 비용이 10~20%정도 절약할 수 있었고, 설비 운전 및 유지보수를 위한 인력 투입 시간도 하루 평균 20분 정도로 줄었다.

#### 마) 자동 운할 시스템

Lincoln GmbH은 인쇄기의 잉크 이송을 위한 중앙 운할 시스템의 효율성 증가를 위해 구식 압축 공기식 운할 시스템을 AS-Interface 밸브 커넥터를 이용하여 유압식 밸브 기반의 단일 및 이중 고압력 시스템으로 대체하였다. 기존 공압식 밸브는 최대 압력이 100bar에 불과했으나, 유압식 밸브로 최대 250bar까지 압력의 범위를 늘렸고 별도의 운할 스테이션 없이 추가로 시스템을 확장시킬 수 있어서 더 많은 운할유를 더 먼 거리까지 이송시킬 수 있었다. 유압식 밸브를 AS-Interface 밸브 커넥터와 함께 사용하여 시스템의 단순화 및 확장성을 증대하였다.

#### 사) 잠수함 통합플랫폼관리체계

(주)KTE에서 개발한 잠수함용 통합플랫폼관리체계에는 잠수함 내 각종 밸브 감시 및 제어를 위해 AS-Interface를 적용하였다. 설계 시 잠수함이라는 특수하고 협소한 공간에서 밸브 제어를 위해 Hardwired와 비슷한 수준의 신뢰성과 잠수함 내 케이블 수량 감축에 따른 공간 활용을 위해 AS-Interface를 선정하였다. 000급 잠수함의 경우 14개의 Master 모듈, 190개의 Slave 모듈, 5개의 리피터를 사용하였고, 이를 통해 약 200개소 이상의 밸브를 제어 및 감시를 하고 있다. 따라서 Hardwired 결선 시 최대 380

가닥의 케이블을 28가닥의 케이블로 약 0.7% 수준으로 경감됨을 알 수 있다. 또한 설계 시 AS-Interface의 확장성 덕분에 설계 수정사항을 비교적 쉽게 반영할 수 있어 변경에 따른 시간과 비용을 단축할 수 있었다.

### 3.6 AS-Interface 적용 방안

스마트선박에 AS-Interface 적용 가능한 다양한 방안이 있다. 주로 자동화 하위레벨에서 사용되는 디지털 및 아날로그 입력/출력에 AS-Interface를 적용하는 경우가 대부분이며, 디지털 입력은 스마트선박 내의 밸브, 플랩 등 다양한 장비에 활용 가능하고, 아날로그 센서는 온도, 압력, 수위 등 다양한 환경에서 사용된다. AS-Interface는 선박 환경에서 요구되는 전자파 내성과 오류 검출 및 처리 기능을 제공하여 다양한 환경에서 유용하게 활용 가능하며, 선박 운용 중에 안정적인 제어와 감시를 수행할 수 있다.

스마트선박의 경우 승조원의 숫자는 감소하고 시스템의 복잡성은 증가하는 경향이 있다. 이에 따라 다양한 센서와 구동기가 늘어남에 따라 케이블 개수도 증가하며, 이로 인해 케이블이 차지하는 공간과 유지보수가 어려움을 겪는다. 하지만 AS-Interface를 적용하면 케이블 개수를 줄이고 케이블 사양을 낮춤으로써 비용과 설치시간을 절감하며 시스템의 복잡성을 간소화할 수 있다.

AS-Interface를 적용하면 Master 모듈 당 최대 62개의 Slave 모듈을 적용할 수 있으며, 디지털 센서의 경우 1개의 모듈당 4개의 입력과 출력을 가질 수 있어 1개의 Master 모듈로 248개의 디지털 센서와 구동기를 관리할 수 있다. 이를 통해 원거리의 많은 센서를 필요한 숫자만큼 모듈을 추가하여 감시 및 제어가 가능해진다.

또한 AS-Interface를 활용하면 기존 상위 프로토콜을 이용한 별도의 제어반 운용 개수를 줄일 수 있다. 따라서 여러 제어반을 하나의 제어반으로 통합하여 스마트선박의 시스템을 단순화할 수 있다. AS-Interface의 다양한 활용 방안을 통해 스마트선박의 자동화와 효율성을 향상시킬 수 있으며, 더 나아가 산업 자동화 분야에서도 그 잠재력을 인식하여 새로운 응용 분야로의 확장이 가능할 것으로 기대된다.

## 4. 결론

본 논문에서는 스마트선박에 적용 가능한 AS-Interface 프로토콜에 대해 탐구하였다. 스마트선박은 선박의 자동화와 효율성 증대를 목표로 하며, 다양한 센서와 구동기를 통합하여 선박 운영 및 유지보수를 최적화하고 안전성을 향상시키는 것을 목표로 한다. 하지만 기존의 1:1 배선방식인 Hardwired는 설치와 유지보수에 많은 시간과 노력을 요구하여 스마트선박의 구축에 어려움을 겪고 있다.



AS-Interface 프로토콜은 하위레벨 구성에 특화된 통신 프로토콜로, 센서와 구동기를 효과적으로 연결하고 제어할 수 있다. 이를 통해 스마트선박의 제작 시간과 비용, 유지보수 시간 등을 단축하여 스마트선박의 개발을 원활하게 진행할 수 있다. 논문에서는 AS-Interface를 사용하여 선박 내 다양한 센서와 구동기의 네트워킹을 구현하는 방안과 활용 사례를 제시하였다. 또한 AS-Interface가 산업 자동화 분야에서도 유용하게 활용될 수 있는 가능성을 밝혀냈다.

본 연구 결과, AS-Interface 프로토콜을 스마트선박에 적용함으로써 기존의 Hardwired 방식의 단점을 극복하고 효율적인 자동화 시스템을 구축하는데 기여할 수 있다는 결론을 얻었다. AS-Interface의 활용으로 선박의 운영을 최적화하고 안전성을 향상시키는데 도움이 될 것으로 기대된다. 또한 산업 자동화 분야에서도 AS-Interface의 잠재력을 인식하여 새로운 응용 분야로의 확장이 가능할 것으로 기대된다.

최종적으로, AS-Interface를 스마트선박에 적용하여 자동화와 효율성을 극대화하는 것은 선박 산업의 혁신과 발전을 이끌어낼 수 있는 기술적인 도약임을 확인하였다. 미래의 스마트선박 개발과 산업 자동화 분야의 성공적인 구축을 위해 AS-Interface의 활용 가능성을 고려해볼 필요가 있으며, 추가적인 연구와 기술 개발을 통해 더욱 높은 수준의 스마트선박 시스템을 구현할 수 있을 것으로 기대한다.

## References

- Kim, J. and Jang, H.S., 2019. Autonomous ship technology trends and preparations. *Bulletin of the Society of Naval Architects of Korea*, 56(4), pp.4-7.
- Rolf, B., 2006. *AS-Interface automation solution*. Siemens.
- Ryu, K.S., 2018. Our needs for future ships. *Bulletin of the Society of Naval Architects of Korea*, 55(1), pp.56-59.
- Yoo, Y.H, Shin, I.S. and Hwang, H.G., 2018. Smart autonomous shipping technology trends. *TTA Journal*, 178, pp.27-35.

