



ISSN ONLINE: 2447-0228

ITEGAM-JETIA

Manaus, v.12 n.60, p.1410-1421. July/August, 2026.

DOI: <https://doi.org/10.5935/jetia.v12i60.4179>

RESEARCH ARTICLE

OPEN ACCESS

DESIGN AND IMPLEMENTATION OF AN INTELLIGENT SCADA SYSTEM FOR MARINE ENGINE PREDICTIVE MAINTENANCE USING OT-IT CONVERGENCE

Huu-Khanh Doan¹, Trung-Phuc Mac^{2*}, Hoang-Binh Trong Nguyen³, Hoang-Nam Nguyen⁴,
Duy-Anh Nguyen⁵

^{1, 4, 5}Faculty of Marine Engineering, Vietnam Maritime University^{2, 3}Faculty of Electrical - Electronics Engineering, Vietnam Maritime University¹<https://orcid.org/0009-0005-1414-5280>, ²<https://orcid.org/0009-0003-7813-368X>, ³<https://orcid.org/0009-0001-6345-4232>⁴<https://orcid.org/0009-0009-4095-6035>, ⁵<https://orcid.org/0009-0008-6364-8556>

E-mail: khanhhd.dtt@vimaru.edu.vn, phuc102499@st.vimaru.edu.vn, binh102425@st.vimaru.edu.vn, nam102489@st.vimaru.edu.vn,
anh102416@st.vimaru.edu.vn

ARTICLE INFO

Article History

Received: March 19, 2026

Reviewed: April 19, 2026

Accepted: July 19, 2026

Published: August 31, 2026

Keywords:

Marine Predictive Maintenance,
Intelligent SCADA,
OT-IT Convergence,
Hybrid Artificial Neural Network,
Edge Computing.

ABSTRACT

In the maritime industry, the transition from Time-Based Maintenance (TBM) to Predictive Maintenance (PdM) is increasingly critical to meet stringent environmental regulations and minimize operational downtime. However, implementing data-driven diagnostics on marine vessels faces significant challenges due to the "OT-IT gap" and the hardware constraints of legacy shipboard systems. This study proposes a cost-effective, Edge-centric Intelligent SCADA architecture that integrates a Siemens S7-1200 PLC with a Hybrid Artificial Neural Network (ANN) via an SQL-based middleware. To ensure compatibility with standard low-configuration Industrial PCs (IPCs), the system employs a deliberate asynchronous sampling strategy ($T_{scan} = 3$ s), decoupling heavy predictive analysis from real-time monitoring. This design allows the diagnostic module to operate seamlessly as a background service without overburdening the control loop. Additionally, an optimized Bit-wise State Encoding (Bit-packing) protocol is implemented to address bandwidth constraints. This method compresses diagnostic vectors into atomic integer payloads, ensuring data integrity during transmission. The system was validated through Hardware-in-the-Loop (HIL) simulations and field trials on a Daihatsu 6PSHTb-26D training ship. Experimental results demonstrate that the architecture maintains a standard supervision latency of < 450 ms. Meanwhile, the AI-based fault detection operates with a stable latency of 3.1 s to 4.9 s—a designed safety margin that is sufficient for marine propulsion dynamics while optimizing computational resources. Furthermore, the Bit-packing protocol reduces communication overhead by approximately 90% compared to traditional tag-based methods, confirming the system's feasibility as a robust retrofit solution.



Copyright ©2026 by authors and Galileo Institute of Technology and Education of the Amazon (ITEGAM). This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

I. INTRODUCTION

1.1 MOTIVATION: THE EVOLVING PARADIGM OF MARITIME MAINTENANCE

The global maritime industry is currently navigating a profound structural transformation, often characterized in academic discourse as "Maritime 4.0" [1]. This digitalization wave is driven not merely by technological availability but by an urgent necessity to enhance operational efficiency and environmental compliance.

The International Maritime Organization (IMO) has instituted rigorous regulations regarding greenhouse gas emissions, most notably the Energy Efficiency Existing Ship Index (EEXI) and the Carbon Intensity Indicator (CII), which entered into force in 2023 [2]. These regulations create substantial pressure on vessel operators to optimize the performance of marine diesel engines, as inefficient operation directly impacts a vessel's Carbon Intensity rating and commercial viability [3]. Historically, marine engine maintenance has relied heavily on Time-Based Maintenance (TBM) strategies, which mandate the replacement of components at fixed intervals regardless of their actual physical condition [4]. While structured, TBM is inherently inefficient; components with remaining useful life are often replaced prematurely, incurring unnecessary costs, while unforeseen failures can still occur between scheduled intervals. Consequently, the industry is witnessing a strategic migration towards Condition-Based Maintenance (CBM) and Predictive Maintenance (PdM), leveraging real-time data to forecast equipment health and minimize unplanned downtime [5].

1.2 PROBLEM STATEMENT: THE OT-IT CONVERGENCE CHALLENGE

Despite the theoretical benefits of PdM, its practical deployment on seafaring vessels faces significant technical hurdles known as the "OT-IT gap" [6]. Operational Technology (OT) systems, such as Programmable Logic Controllers (PLCs), operate in a deterministic real-time environment focused on immediate machinery control. Conversely, Information Technology (IT) systems, required for complex data analytics, typically operate on probabilistic timing. A critical "disconnect" often exists between these layers. Operational data is frequently "locked" within low-level PLC registers, inaccessible to high-level analytical tools without complex interfaces. Furthermore, recent trends in PdM research often favor deep learning models (e.g., Transformers, LSTMs) deployed via cloud-based architectures [7]. However, such approaches are often unsuitable for the majority of the existing merchant fleet ("brownfield" vessels) due to two main reasons:

- **Hardware Constraints:** Deep learning models require high-performance computing resources (often GPUs) that are not standard on legacy Industrial PCs (IPCs).
- **Connectivity Limitations:** Cloud-centric solutions suffer from latency and instability due to reliance on satellite communication, which can compromise the availability of safety-critical monitoring systems [8].

1.3 CONTRIBUTIONS

Addressing these limitations, this research focuses on the design and implementation of a cost-effective, Edge-centric Intelligent SCADA system. The primary objective is to engineer a unified architecture that bridges the OT-IT gap directly on-board, enabling predictive intelligence without relying on unstable cloud connectivity or expensive hardware upgrades. The specific contributions of this study are:

- **Architecture of Convergence:** We propose a robust OT-IT framework utilizing a middleware bridge (Kepware) and a central data warehouse (SQL Server). This architecture creates a "Local Digital Twin," allowing the seamless bidirectional flow of data between the deterministic control layer and the analytical layer.
- **Resource-Optimized Asynchronous Strategy:** To ensure compatibility with standard low-specification IPCs, we introduce a deliberate asynchronous sampling strategy ($T_{scan} = 3$ s). This design decouples the heavy computational tasks of the Hybrid ANN model from the critical real-time scanning of the PLC, ensuring that predictive analysis operates smoothly as a background service without inducing system latency.
- **Optimized Communication Protocol:** We develop an optimized Bit-wise State Encoding (Bit-packing) algorithm. This technique addresses the bandwidth constraints of shipboard networks by compressing complex diagnostic vectors into atomic integer payloads, reducing communication overhead by approximately 90% and ensuring data integrity during transmission.

II. RELATED WORK

II.1 CLOUD-CENTRIC DIGITAL TWINS VS. EDGE COMPUTING

In the era of Maritime 4.0, the dominant trend in predictive maintenance research involves the development of Cloud-based Digital Twins. High-fidelity models, such as those proposed by Ritou et al. [9], continuously stream sensor data to onshore servers for processing. This centralized approach leverages virtually unlimited computing power, enabling complex simulations that are impossible to run locally. Recent frameworks leveraging IoT and cloud integration have demonstrated significant potential in revolutionizing predictive maintenance across various industrial sectors [10]. A comprehensive overview of these advancements highlights the industry's shift from basic component monitoring to complex, data-driven ecosystem management [11]. However, the practical deployment of these cloud-centric solutions on ocean-going vessels faces critical hurdles. Beyond the high implementation costs for legacy fleets [11], the reliance on continuous ship-to-shore connectivity presents a severe vulnerability. According to a comparative analysis by Shi et al. [12], cloud systems suffer from significant latency and jitter, particularly when relying on VSAT satellite links susceptible to weather interference. Furthermore, a 2025 analysis of maritime cybersecurity warns that the direct integration of OT and IT systems for cloud uploading exposes vessel infrastructures to escalating cyber threats, including Advanced Persistent Threats (APTs) targeting satellite communications [13]. Consequently, recent scholarship argues for a strategic shift towards "Edge Computing," where data processing occurs near the source to ensure autonomy and security. Our research aligns with this paradigm, prioritizing deterministic response times over the expansive but vulnerable processing capabilities of the cloud.

II.2 DEEP LEARNING MODELS VS. EXPLAINABLE HYBRID DIAGNOSTICS

Regarding diagnostic algorithms, Deep Learning (DL) has gained traction. Cheliotis et al. [14] applied Long Short-Term Memory (LSTM) networks, while others have explored Convolutional Neural Networks (CNNs) for vibration analysis [15]. Recent studies in 2025 have further advanced this field by proposing deep hybrid models capable of capturing both temporal and spatial features of marine engine data with high accuracy [16]. Despite their performance, these complex models often impose computational burdens exceeding the capabilities of standard marine IPCs [17]. Furthermore, they require massive datasets, which are rare due to the "data scarcity" of maritime fault events. While unsupervised anomaly detection techniques have emerged as a potential solution [18], there is an urgent need for "lightweight" architectures that reduce parameter overhead to ensure adaptability in resource-constrained environments, as emphasized in recent reviews of port machinery diagnostics [19]. Addressing this trade-off, hybrid approaches combining physics-based rules with shallow neural networks offer a better balance between accuracy and efficiency [20]. This direction is supported by recent findings in [21], which demonstrate that lightweight adaptive models can achieve superior diagnostic performance on rotating machinery without the heavy computational cost of deep transfer learning.

II.3 DATA SYNCHRONIZATION AND SEMANTIC COMPRESSION

A significant yet often overlooked challenge in OT-IT convergence is ensuring data integrity during the transmission of diagnostic results. Conventional research on industrial data compression has largely focused on time-series reduction techniques, such as the Gorilla algorithm or Delta encoding, to minimize storage costs [22]. However, these techniques prioritize storage efficiency over real-time "state atomicity." In a SCADA environment, if diagnostic states (e.g., Warning vs. Alarm) are transmitted sequentially as separate tags, network latency can lead to "data tearing," where the operator sees an inconsistent system state. There is a lack of research focusing on semantic state compression for ensuring atomic updates in control loops. This study addresses this gap by introducing the Bit-wise State Encoding technique, which packages the entire diagnostic vector into a single atomic payload, ensuring that the AI's inference is perfectly synchronized with the HMI display.

III. PROPOSED SYSTEM ARCHITECTURE

III.1 THE OT-IT CONVERGENCE FRAMEWORK

The overall architecture of the proposed Intelligent SCADA system is illustrated in Figure 1. Predicated on a centralized data brokerage model, the system utilizes Kepware Server as the primary gateway to facilitate a seamless, closed-loop data exchange between the Operational Technology (OT) and Information Technology (IT) layers. To reconcile the conflicting requirements of real-time operational safety and deep-learning analytical depth, the communication topology between the middleware and the visualization layer is bifurcated into two distinct operational streams.

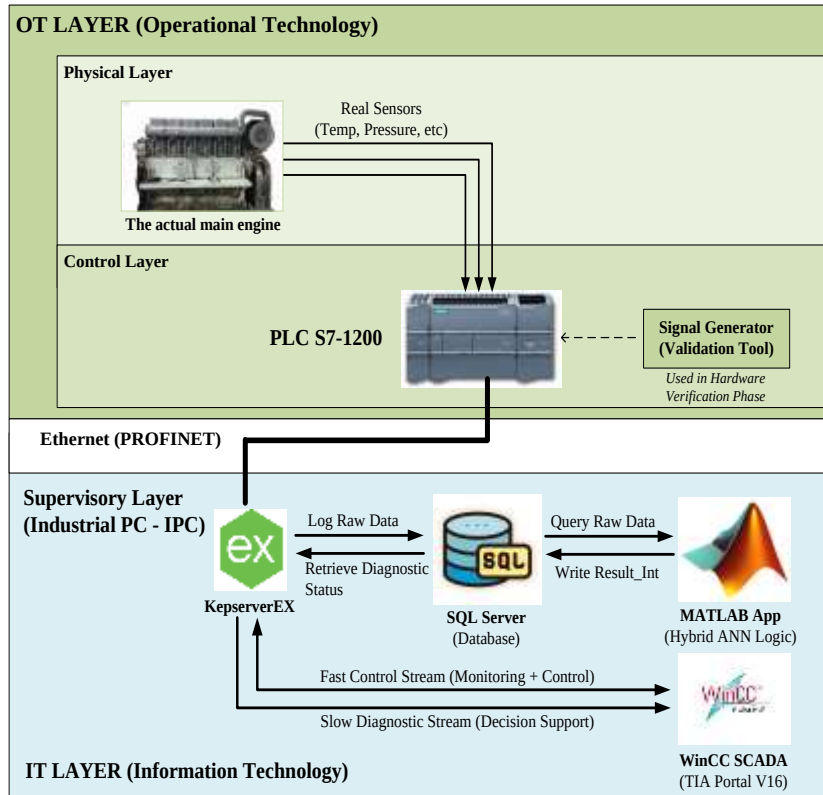


Figure 1: The proposed OT-IT Convergence Architecture.

Note: The separation between the Fast Control Stream (Bidirectional) for real-time operation and the Slow Diagnostic Stream (Unidirectional) for ANN-based decision support

Source: Authors, (2026).

The first channel, designated as the Fast Control Stream, establishes a high-speed, bidirectional link ($T < 500\text{ms}$) connecting WinCC directly to the PLC tags. This deterministic path ensures immediate visualization of critical telemetry—such as RPM, pressure, and temperature—while simultaneously enabling the operator to execute control commands with instant responsiveness. In parallel, the Slow Diagnostic Stream functions as a dedicated unidirectional channel (T approx. 3 s) that delivers the ANN-predicted health status (*Result_Int*). This stream is strictly architected for decision support, thereby preventing the probabilistic nature of the neural network from interfering with the safety-critical control logic. Underpinning this dual-stream topology is a robust closed-loop analytics cycle. Kepware utilizes the ODBC Data Logger to continuously execute the "Log Raw Data" action, pushing sensor data from the PLC into the SQL Server archive. The MATLAB Engine then asynchronously fetches these records to perform inference and writes the computed diagnostic result back to a feedback table. Finally, Kepware executes the "Retrieve Diagnostic Status" action via its ODBC Client driver, polling the ANN output and mapping it back to an OPC tag for the Slow Stream visualization.

III.2 ASYNCHRONOUS CLOSED-LOOP SYNCHRONIZATION STRATEGY

Integrating deterministic OT systems with probabilistic IT environments introduces a fundamental "impedance mismatch" in processing speeds. While the PLC operates on a rigid millisecond-level scan cycle, the matrix operations required for the Hybrid ANN model necessitate a variable computation time. To resolve this, the system employs an asynchronous "Producer-Consumer" architecture, visualized in the flowchart in Figure 2.

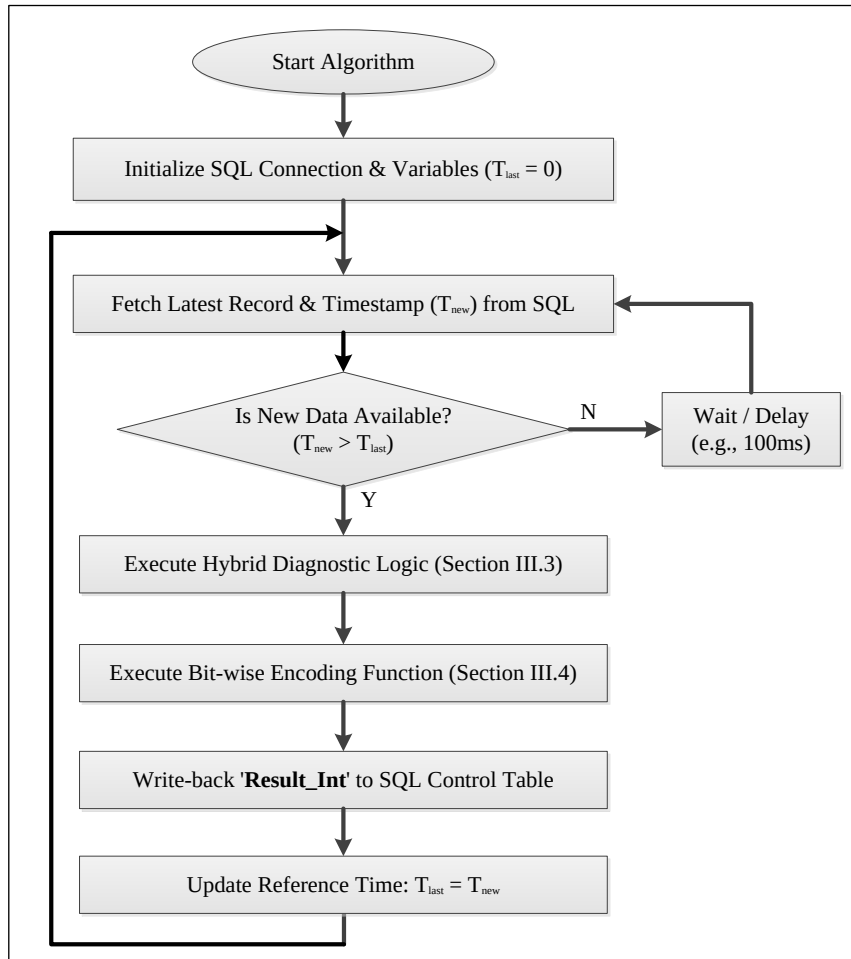


Figure 2: Flowchart of the Asynchronous Synchronization Service acting as a safety gatekeeper between OT and IT layers.
Source: Authors, (2026).

This synchronization logic acts as a strict gatekeeper to manage the interaction between the two data streams without conflict. The MATLAB service operates in a background loop, triggering the heavy diagnostic computation only when a temporal handshake confirms the arrival of new data ($T_{new} > T_{last}$). This decoupled design ensures that the latency incurred by the AI inference loop—typically ranging from 3 s to 5 s—is effectively isolated within the Slow Diagnostic Stream, guaranteeing that the operator maintains full, real-time situational awareness.

III.3 HYBRID DIAGNOSTIC ALGORITHMS IMPLEMENTATION

To ensure comprehensive fault coverage across the engine's operating range, the system employs a "Hybrid" diagnostic strategy that combines deterministic physics-based rules with a probabilistic Artificial Neural Network (ANN).

III.3.1 Rule-based Logic (Linear Parameters)

For failure modes exhibiting linear characteristics or immediate threshold violations, we implemented a deterministic rule-based algorithm directly within the MATLAB processing loop. This module calculates Diagnostic Indices (DI) based on deviations from the manufacturer's baseline specifications [23].

Exhaust Gas Temperature Deviation (DI1): To detect cylinder load imbalance or injector fouling, the deviation of each cylinder's temperature (T_i) from the average temperature of all cylinders (T_{avg}) is monitored:

$$DI_{EGT} = \frac{|T_i - T_{avg}|}{T_{avg}} \times 100 \quad (\%) \quad (1)$$

Threshold: An alarm is triggered if the deviation exceeds 5%. **Governor Hunting Detection (DI4):** Instability in the fuel rack or governor mechanism is identified by analyzing the oscillation amplitude of the engine speed (N_{act}) relative to the steady-state setpoint (N_{set}) over a sliding time window:

$$DI_{Gov} = |N_{set} - N_{act}| \quad (2)$$

Threshold: A "Hunting" warning is issued if the RPM fluctuation exceeds 15 rpm for a duration longer than 5 s. **Pneumatic Leakage Detection (DI5):** To identify sealing failures in the starting air system, the pressure decay rate is calculated during intervals when the air compressor is in the "OFF" state. The leakage index is defined as the pressure drop (ΔP) over a fixed observation period (Δt):

$$DI_{Leak} = \frac{P(t) - P(t + \Delta t)}{\Delta t} \quad (3)$$

Threshold: A leakage alarm is triggered if the pressure drop rate exceeds 0.2 bar/minute, indicating a potential valve failure or pipe rupture.

III.3.2 Residual-based ANN Logic (Non-linear Parameters)

While linear rules suffice for DI1, DI4, and DI5, the thermodynamic behavior of the lubrication and cooling systems exhibits complex non-linearities that cannot be accurately modeled by static thresholds. Therefore, the Lubricating Oil Pressure (DI2) and Cooling Water Temperature (DI3) are monitored using a Residual-based approach driven by the ANN. **Network Topology and Inputs:** The core predictive engine is built upon the architecture previously developed and validated in our accompanying study [24]. The model utilizes a Multilayer Perceptron (MLP) with an optimized [3] topology. This specific configuration was determined through an empirical grid search to balance learning capability (RMSE $\approx 0.0737^\circ\text{C}$) and computational efficiency for Edge deployment [24]. The network is trained using the Levenberg-Marquardt algorithm on a high-fidelity dataset generated via Hardware-in-the-Loop (HIL) simulation. The input vector X_t comprises three correlated operational variables: Engine Speed (n), Load Factor (L), and Lube Oil Temperature (T_{LO}). Mathematically, the ANN functions as a non-linear mapping operator $f_{ANN}(\cdot)$ that correlates the operational state vector with the expected thermodynamic outputs. The estimation model for Lube Oil Pressure and Jacket Water Temperature is defined as follows:

$$Val_{predict} = f_{ANN}(n, L, T_{LO}) = \Phi_{out} \left(\sum_{j=1}^M w_j \cdot \Phi_{hidden} \left(\sum_{i=1}^3 w_{ij} x_i + b_j \right) + b_{out} \right) \quad (4)$$

where $Val_{predict}$ is the ideal healthy value, $x_i \in \{n, L, T_{LO}\}$ represents the input features, w and b denote the trained weights and biases, and Φ represents the activation functions (typically Sigmoid for hidden layers and Linear for the output layer [24]).

Residual Calculation Logic: The ANN predicts $Val_{predict}$ for the oil pressure and cooling temperature under current load conditions. The diagnostic indices are then derived from the residual error between the prediction and the actual sensor measurement ($Val_{Measure}$):

$$DI_2 \text{ (or } DI_3) = |Val_{Measure} - Val_{predict}| \quad (5)$$

Threshold Comparison: Unlike fixed limits, the thresholds for DI2 and DI3 are dynamic. The system flags a fault only if the residual exceeds a statistically determined boundary derived from Receiver Operating Characteristic (ROC) analysis [25]. As established in [24], this adaptive thresholding mechanism incorporates both the algorithm's noise floor and the sensor's uncertainty margin (δ_{sensor}), effectively filtering out false alarms caused by transient load changes.

III.4 OPTIMIZED BIT-WISE STATE ENCODING PROTOCOL

Following the diagnostic process, transmitting the results from the IT layer (MATLAB) back to the HMI (WinCC) presents a bandwidth challenge. To address this, we implemented a Base-4 Weighted Sum encoding algorithm, specifically designed to encapsulate multi-level fault severities into a single atomic payload.

III.4.1 Mathematical formulation (base-4 encoding)

Unlike traditional binary systems that only indicate the presence of a fault (0/1), our system requires the transmission of three distinct operational states for each diagnostic indicator (DI):

- 0 (Normal): The parameter is within safety limits.
- 1 (Warning): Significant deviation detected (Pre-alarm).
- 2 (Alarm): Critical deviation breaching safety thresholds.

The encoding process is visualized in the flowchart in Figure 3. The algorithm iterates through the state vector $S = \{S_1, S_2, S_5\}$ and computes a weighted sum to compress all states into a single integer variable named *Result_Int*.

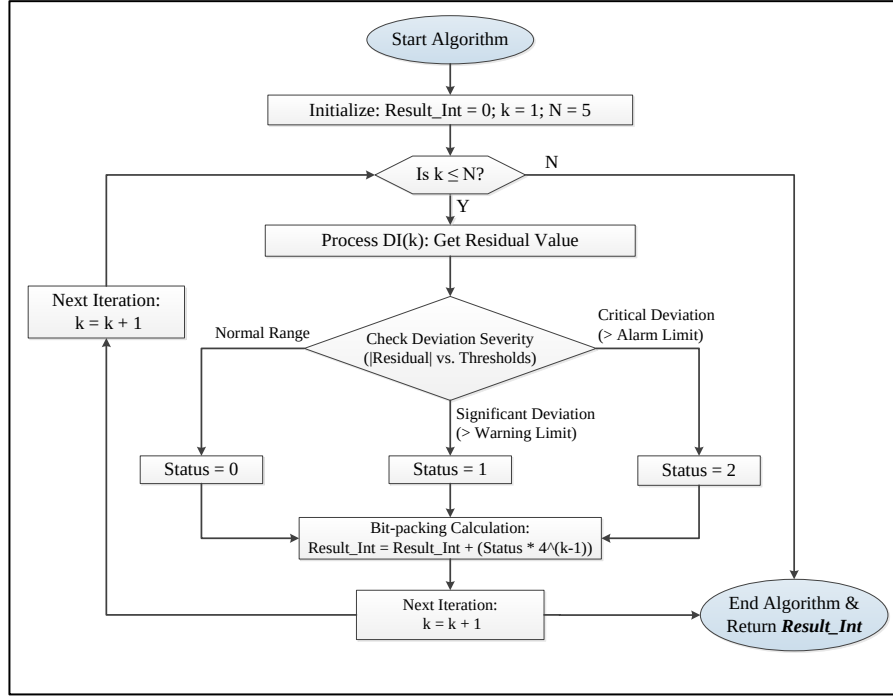


Figure 3: Flowchart of the Base-4 Weighted Sum encoding algorithm implemented in the MATLAB script.
Source: Authors, (2026).

The mathematical definition is as follows:

$$Result_Int = \sum_{k=1}^5 S_k \cdot 4^{(k-1)} \quad (6)$$

Where S_k is the status (0, 1, or 2) of the k -th diagnostic indicator, $4^{(k-1)}$ represents the positional weight, equivalent to allocating a 2-bit memory block for each indicator. For example, if the system detects that DI1 is Normal ($S_1=0$), DI2 is at Warning ($S_2=1$), and DI3 is at Alarm ($S_3=2$), the payload is calculated as:

$$Result_Int = (0 \times 1) + (1 \times 4) + (2 \times 16) = 36 \quad (7)$$

This compact integer is transmitted via a single Modbus register (Word) [26], ensuring synchronization across all subsystems.

III.4.2 Memory Mapping Strategy

At the Visualization Layer (WinCC SCADA), a Direct Bit-Mapping Strategy is employed to decode the *Result_Int* payload. Since each indicator occupies exactly 2 bits, the status can be extracted by checking specific bit pairs in the memory tag. Table 1 details the memory allocation schema used in the experimental setup to map the payload to specific HMI alarms. By utilizing this protocol, the system effectively reduces the communication overhead by approximately 80% compared to traditional tag-based methods, ensuring that the synchronization between the ANN model and the HMI remains real-time.

Table 1: Proposed Bit-wise Encoding Schema for Real-time Status Transmission.

Bit Index (i)	Weight (2i)	Target Subsystem (Diagnostic Indicator)	Semantic Status	Trigger Logic (in WinCC ANN_Class)
0	1	DI1 (Exhaust Gas Deviation)	Warning	Bit 0 = 1 AND Bit 1 = 0
1	2	DI1 (Exhaust Gas Deviation)	Alarm	Bit 1 = 1
2	4	DI2 (Lube Oil Pressure)	Warning	Bit 2 = 1 AND Bit 3 = 0
3	8	DI2 (Lube Oil Pressure)	Alarm	Bit 3 = 1
4	16	DI3 (Jacket Water Temp)	Warning	Bit 4 = 1 AND Bit 5 = 0
5	32	DI3 (Jacket Water Temp)	Alarm	Bit 5 = 1
6	64	DI4 (Governor Stability)	Warning	Bit 6 = 1 AND Bit 7 = 0

7	128	DI4 (Governor Stability)	Alarm	Bit 7 = 1
8	256	DI5 (Start Air Leakage)	Warning	Bit 8 = 1 AND Bit 9 = 0
9	512	DI5 (Start Air Leakage)	Alarm	Bit 9 = 1

Source: Authors, (2026).

IV. SYSTEM IMPLEMENTATION AND EXPERIMENTAL SETUP

To validate the feasibility and robustness of the proposed Intelligent SCADA architecture, a two-phase implementation strategy was adopted: (1) Laboratory validation using a Hardware-in-the-Loop (HIL) configuration, and (2) Field deployment on an actual marine diesel engine.

IV.1 LABORATORY IMPLEMENTATION: HARDWARE-IN-THE-LOOP (HIL) CONFIGURATION

Before field deployment, the system was prototyped in a laboratory environment to verify the stability of the communication stack and the inference speed of the ANN model. The experimental setup, as illustrated in Figure 4, comprises three main layers:

- OT Domain (Physical Emulation): A simplified physical simulation kit representing the marine engine's signals. It generates analog signals (0-10V, 4-20mA) simulating temperature and pressure variations.
- Control Layer: An Industrial PC (IPC) acting as the central processing unit, running the SCADA software (WinCC) and the Intelligent Fault Detection module.
- IT Domain (Data Processing): The software stack integrates KepserverEX (OPC Server) for data acquisition, Microsoft SQL Server for time-series logging, and the trained ANN model for predictive analytics.

This HIL setup ensures that the Dual-Rate Sampling mechanism (Fast monitoring vs. Slow analysis) operates without conflict before integration with real industrial hardware.

IV.2 FIELD DEPLOYMENT ON MARINE DIESEL ENGINE

IV.2.1 Testbed Specifications

The complete system was deployed on the main engine of a training ship, with the Intelligent SCADA station installed in the Engine Control Room (ECR) as shown in Figure 5a. The testbed is a Daihatsu 6PSHTb-26D, a 4-stroke, turbocharged, intercooled marine diesel engine. The detailed technical specifications of the engine are listed in Table 2.

Table 2: Technical specifications of the experimental main engine.

Parameter	Specification
Manufacturer	Daihatsu Diesel Mfg. Co., Ltd.
Model Type	6PSHTb-26D
Engine Type	4-stroke, Single acting, Trunk piston, Turbocharged
Number of Cylinders	6 (In-line)
Cylinder Bore	260 mm
Piston Stroke	320 mm
Rated Output (MCR)	1000 PS (735 kW)
Rated Speed	720 rpm
Combustion System	Direct Injection
Cooling System	Fresh Water (Jacket) / Sea Water (Intercooler)
Starting System	Compressed Air
Turbocharger	Axial Turbine type (VTR Series)

Source: Authors, (2026).

IV.2.2 Fault Injection Methodology

Conducting destructive fault tests on an operational marine engine is strictly prohibited due to safety regulations. Therefore, to evaluate the Intelligent SCADA system's fault detection capability under realistic conditions, two non-destructive fault injection strategies were employed. First, Signal Injection was utilized for critical safety parameters, specifically Lube Oil Pressure (P_{LO}) and Jacket Water Temperature (T_{JW}). Instead of physically overheating the engine, a precision 4-20 mA current loop calibrator was connected in parallel to the PLC analog input module (Figure 5b). This technique allows for the precise simulation of sudden sensor failure scenarios without risking the engine's integrity.

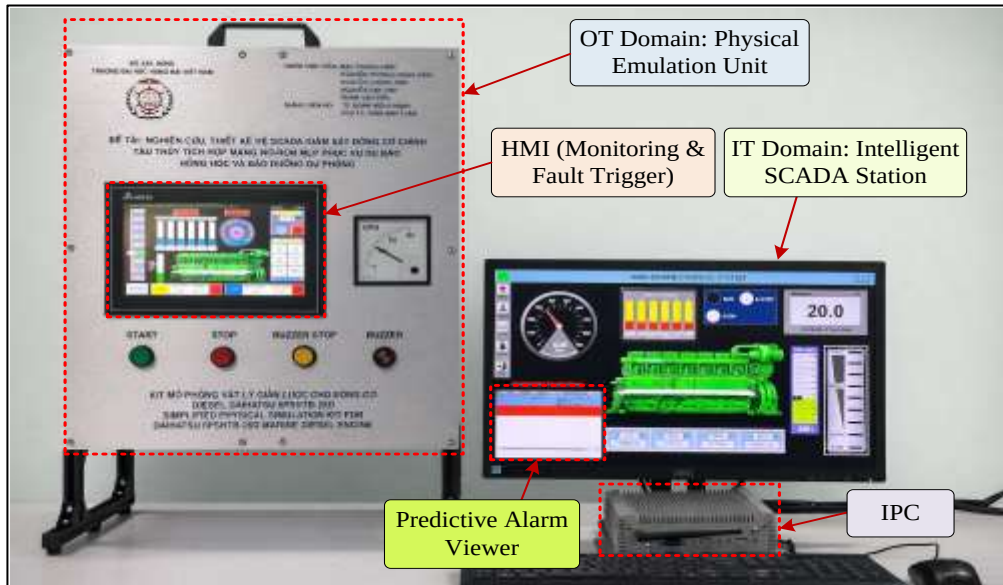


Figure 4: Experimental setup of the Intelligent SCADA station in the laboratory environment (HIL configuration).
Source: Authors, (2026).

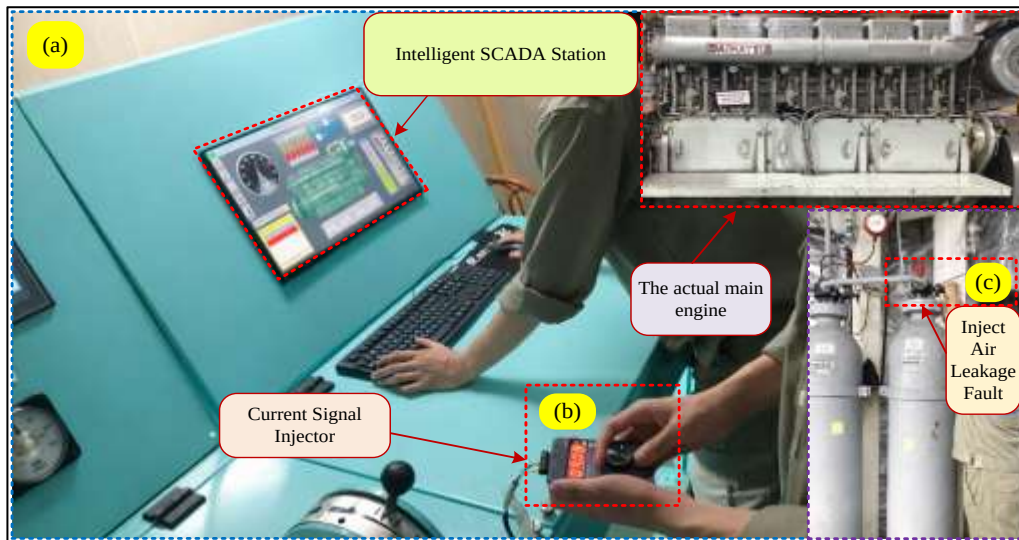


Figure 5: Field trial deployment on the Daihatsu 6PSHTb-26D main engine: (a) Overview of the monitoring station in the engine control room; (b) Signal injection using a 4-20mA loop calibrator to simulate sensor faults; (c) Physical fault injection via manual manipulation of the air receiver valve.

Source: Authors, (2026).

Second, Physical Fault Injection was applied to the pneumatic starting system to simulate mechanical anomalies. By manually manipulating the air receiver service valve (Figure 5c), a gradual pressure drop (Drift Fault) was introduced. This method tests the sensors' sensitivity and the algorithm's ability to detect slow-evolving deviations ($P_{Air} < P_{limit}$) amidst the engine's physical vibration.

IV.3 EXPERIMENTAL FAULT SCENARIOS DEFINITION

To ensure a consistent evaluation across both the HIL simulation (Section V.1) and the field deployment (Section V.2), standardized fault scenarios were established. As detailed in Table 3, specific severity levels were defined for the Lube Oil Pressure (P_{LO}) and Jacket Water Temperature (T_{JW}) parameters. These scenarios correspond to the "Warning" and "Alarm" thresholds set in the SCADA system, representing the progression from an incipient fault to a critical failure.

Table 3: Definition of experimental fault scenarios and severity levels.

Parameter	Fault Type	Severity Level	Physical/Simulated Condition	Target Threshold	Expected System Status
Lube Oil Pressure (PLO)	Filter Clogging / Pump Failure	Level 1 (Moderate)	50% Blockage	$\Delta P = -0.25$ bar	WARNING
		Level 2 (Severe)	73% Blockage	$\Delta P = -0.60$ bar	ALARM
Jacket Water Temp (TJW)	Radiator Fouling / Cooling Loss	Level 1 (Moderate)	46% Fouling	$\Delta T = +1.2$ °C	WARNING
		Level 2 (Severe)	87% Fouling	$\Delta T = +4.5$ °C	ALARM

Note: The severity percentages representing filter clogging and radiator fouling are derived from the system's hydraulic and thermal models used in the HIL configuration and replicated via signal injection/physical adjustment in the field trial.

Source: Authors, (2026).

V. EXPERIMENTAL RESULTS AND DISCUSSION

To rigorously evaluate the proposed Intelligent SCADA system, a comprehensive validation process was conducted in two phases: (1) Laboratory validation using the HIL framework to establish a performance baseline, and (2) Field trials on the Daihatsu 6PSHTb-26D marine diesel engine to assess real-world applicability.

Table 4: Performance verification in HIL simulation.

Monitored Parameter Group	Specific Conditions / Examples	Standard SCADA Latency	ANN Detection Latency	Note
I. General Operational Parameters (Real-time Control & Monitoring)	Engine Speed (RPM), Fuel Rack Position, Air Pressure, etc.	< 350 ms	-	Instant Feedback
II. Lube Oil Pressure (PLO)	Level 1 (50% Clogging)	< 350 ms	3.8 s	ANN Detected
	Level 2 (73% Clogging)	< 350 ms	4.2 s	ANN Detected
III. Jacket Water Temp (TJW)	Level 1 (46% Fouling)	< 350 ms	2.5 s	ANN Detected
	Level 2 (87% Fouling)	< 350 ms	4.5 s	ANN Detected

Source: Authors, (2026).

V.1 LABORATORY VALIDATION (HIL FRAMEWORK)

The HIL simulation was configured to replicate the exact fault scenarios defined in Table 4. By injecting the specific fault vectors—Level 1 (Warning) and Level 2 (Alarm)—into the engine model, the system's baseline latency was measured under ideal conditions. As summarized in Table 4, the system demonstrates a clear distinction between the two monitoring channels. The standard monitoring path (PLC → WinCC) maintains a near-instantaneous response of < 350 ms. In contrast, the ANN detection module exhibits a variable latency ranging from 2.5 s to 4.5 s. Since the HIL environment assumes ideal sensors with zero response time, this delay is attributed strictly to the asynchronous sampling interval of the ANN application ($T_{scan} = 3$ s).

V.2 FIELD TRIAL ON REAL MARINE DIESEL ENGINE

Following the HIL validation, the system was deployed on the actual engine to evaluate its performance under realistic operating conditions. The experiments focused on the two contrasting dynamic behaviors defined in Table 4: the fast-response Lube Oil Pressure (P_{LO}) and the slow-response Jacket Water Temperature (T_{JW}).

V.2.1 Lube Oil Pressure Response (P_{LO})

In the first scenario, a signal injection technique was employed to simulate a critical drop in Lube Oil Pressure. As depicted in Figure 6, while the standard HMI monitoring updated the values near-instantly (< 450 ms), the Intelligent ANN module processed the data according to its specific scanning interval. The module successfully identified the anomaly, triggering the 'Warning' state with a measured latency of $\Delta t_1 = 3.2$ s. Subsequently, as the signal continued to degrade below the critical limit, the system escalated to the 'Alarm' state with a latency of $\Delta t_2 = 4.9$ s. This response time is consistent with the designed operation of the predictive algorithm.

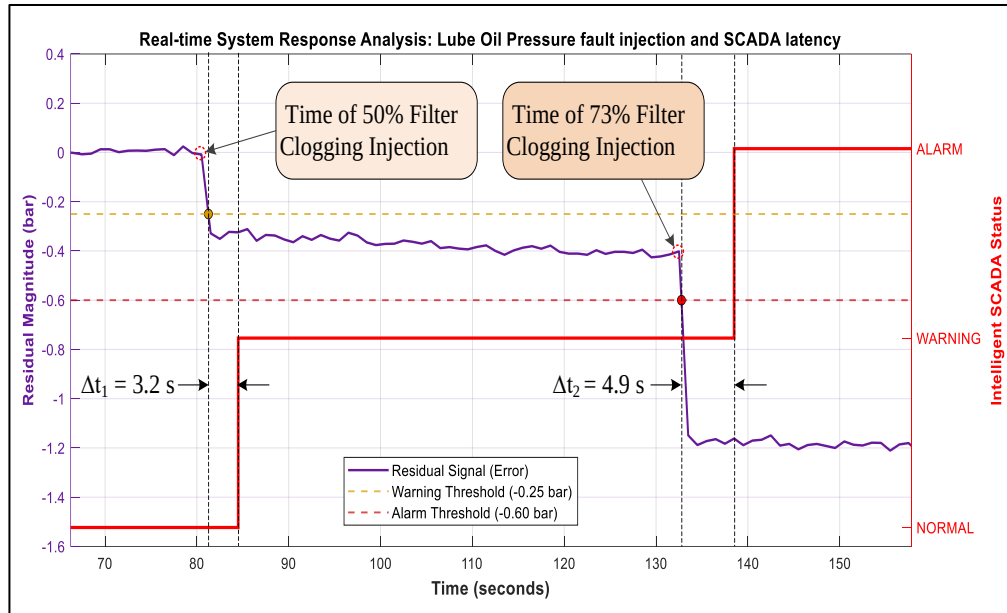


Figure 6: Real-time System Response (Fast Dynamics): Fault detection analysis for Lube Oil Pressure (P_{LO}) under signal injection scenario.

Source: Authors, (2026).

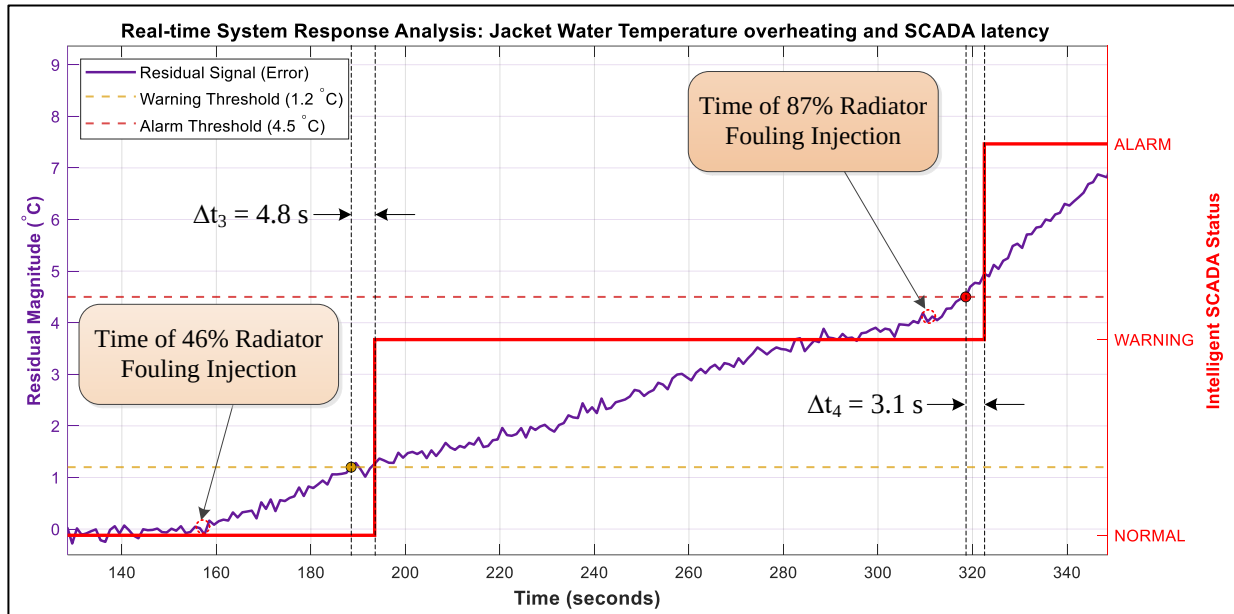


Figure 7: Real-time System Response (Slow Dynamics): Fault detection analysis for Jacket Water Temperature (T_{JW}) under physical overheating scenario.

Source: Authors, (2026).

V.2.2 Jacket Water Temperature Response (T_{JW})

Conversely, the second scenario focused on the Jacket Water Temperature, a parameter characterized by high thermal inertia. A physical restriction of the radiator airflow was introduced to induce a gradual overheating condition. Figure 7 illustrates the detection process. The intelligent algorithm detected the initial deviation and activated the 'Warning' status with a latency of $\Delta t_3 = 4.8$ s. Notably, the critical 'Alarm' condition was detected with a faster response time of $\Delta t_4 = 3.1$ s. This variation in detection time (from 3.1 s to 4.8 s) confirms the stochastic nature of the ANN application's asynchronous scanning cycle ($T_{scan} = 3$ s), independent of the faster real-time monitoring layer.

V.2.3 Overall Performance Summary

The quantitative results from the field trials are consolidated in Table 5. The data confirms that the Standard SCADA Latency for general monitoring (Speed, Rack, Pressure, etc.) is consistently < 450 ms, which aligns with standard industrial HMI update rates and ensures real-time feedback for operators. In contrast, the ANN Detection Latency for fault scenarios ranges between 3.1 s and 4.9 s, which accounts for the computational overhead of the intelligent algorithm but remains well within the safety margin.

Table 5: Real-world system performance.

Monitored Parameter Group	Specific Conditions / Examples	Standard SCADA Latency	ANN Detection Latency	Note
I. General Operational Parameters (Real-time Control & Monitoring)	Engine Speed (RPM), Fuel Rack Position, Air Pressure, etc.	< 450 ms	-	Instant Feedback
II. Lube Oil Pressure (PLO)	Level 1 (50% Clogging)	< 450 ms	3.2 s	ANN Detected
	Level 2 (73% Clogging)	< 450 ms	4.9 s	ANN Detected
III. Jacket Water Temp (T_{JW})	Level 1 (46% Fouling)	< 450 ms	4.8 s	ANN Detected
	Level 2 (87% Fouling)	< 450 ms	3.1 s	ANN Detected

Source: Authors, (2026).

V.3 DISCUSSION

The experimental results validate the efficacy of the proposed Dual-Rate Sampling Architecture, which successfully decouples real-time monitoring from complex predictive analysis. By maintaining a low latency of approximately 450 ms on the standard HMI channel (PLC $\rightarrow$ WinCC), the system ensures that critical operational parameters, such as engine speed and exhaust gas temperature, are updated instantaneously for human operators. This design satisfies the ergonomic requirements for marine supervision, ensuring that the crew retains immediate control over the engine's status regardless of the computational load on the intelligent module. Regarding the intelligent fault detection capability, a comparative analysis between the HIL simulation (Table 4) and the field trial (Table 5) reveals a consistent latency pattern. The detection time in the real-world environment fluctuates between 3.1 s and 4.9 s. This latency is primarily governed by the asynchronous interaction between the industrial communication stack (PLC-to-Kepserver polling) and the ANN application's scanning interval ($T_{scan} = 3$ s). Although this introduces a slight delay compared to the ideal HIL baseline, it remains well within the safety margins required for marine propulsion systems.

Given the large thermal and mechanical time constants of a marine diesel engine (where temperature changes occur over several seconds), a maximum processing delay of 5 s is negligible and does not compromise the safety of the vessel. Furthermore, the system demonstrates significant robustness against environmental noise and sensor fluctuations. As evidenced in the response curves of Figure 6 and Figure 7, the decision-making algorithm effectively filters out high-frequency noise without triggering false alarms. The stability of the "Intelligent SCADA Status" signal confirms that the system is not only capable of early fault detection but also reliable enough to operate in the harsh vibration environment of a ship's engine room. Finally, Table 6 benchmarks the proposed system against prevalent Cloud-centric [9, 12] and Deep Learning Edge [14, 15] approaches. While Deep Learning models utilizing GPU acceleration offer rapid inference times (< 100 ms), such speed is often excessive for marine applications characterized by large thermal and mechanical inertia. A response time of milliseconds risks capturing transient noise rather than genuine trend deviations, potentially leading to false alarms. In contrast, our system's asynchronous sampling (3.1–4.9 s) acts as a temporal filter. This design represents an optimal engineering trade-off: it provides a stable, noise-robust detection window that aligns with the engine's physical time constants (), while allowing deployment on cost-effective, standard IPCs without the need for specialized GPU hardware.

Table 6: Comparative analysis: Proposed System vs. Cloud-centric and Deep Learning approaches.

Feature	Cloud-centric Digital Twins [9, 12]	Deep Learning Edge [14, 15]	Proposed Lightweight Edge
Computing Hardware	Remote Server (Unlimited)	High-End IPC (Requires GPU)	Standard IPC (CPU only)
Diagnostic Latency	Stochastic (Unstable)	Instant (< 100 ms)	Deterministic (3.1s–4.9s)
Suitability for Inertia	Risk of Disconnection	Noise-Sensitive (Overkill)	Optimal (Inertia-Matched)
Retrofit Feasibility	Low (High recurring costs)	Low (Expensive Hardware)	High (Software-based)

Source: Authors, (2026).

VI. CONCLUSION

This study successfully developed and validated a cost-effective Intelligent SCADA system for marine diesel engines, integrating industrial automation standards with advanced predictive maintenance capabilities. By employing a "Dual-Rate Sampling Architecture," the system overcomes the computational limitations of standard PLCs, allowing strictly real-time monitoring to coexist with complex AI-based fault diagnosis. The experimental validation, conducted through both Hardware-in-the-Loop (HIL) simulations and field trials on a Daihatsu 6PSHTb-26D training ship, yielded the following key conclusions:

- **Architecture Efficiency:** The proposed decoupling strategy proved effective in a real-world marine environment. The standard monitoring channel (HMI path) maintained a stable latency of < 450 ms for all critical operational parameters, complying with industrial safety standards for real-time supervision.
- **Diagnostic Performance:** The Intelligent ANN module successfully detected both fast-dynamic faults (Lube Oil Pressure anomalies) and slow-dynamic faults (Cooling Water thermal drift). Despite the asynchronous scanning interval, the system achieved a detection latency ranging from 3.1 s to 4.9 s. Given the large thermal and mechanical time constants of marine propulsion systems, this delay is negligible and provides a sufficient safety margin for early warning.
- **Robustness and Applicability:** The field trials confirmed that the system is robust against the harsh conditions of an engine room, including significant vibration and electromagnetic noise. By utilizing standard industrial protocols (OPC UA, Modbus TCP) and commercial hardware (S7-1200 PLC), the proposed solution offers a scalable and low-cost upgrade path for existing legacy vessels, bridging the gap between traditional automation and the era of autonomous shipping.

Future Work: Future research will focus on expanding the fault dataset to include combustion-related anomalies (via cylinder pressure analysis) and implementing an Edge-to-Cloud synchronization mechanism. This will allow the local Intelligent SCADA models to be updated remotely, further enhancing the system's adaptability to different engine types and operating conditions.

VIII. REFERENCES

- [1] M. Lambrou, D. Watanabe, and J. Iida, "Shipping digitalization management: conceptualization, typology and antecedents," *Journal of Shipping and Trade*, vol. 4, no. 1, 2019.
- [2] I. M. O. (IMO), "Amendments to the Protocol of 1997 to Amend the International Convention for the Prevention of Pollution from Ships, 1973, as Modified by the Protocol of 1978 Relating Thereto (2021 Revised MARPOL Annex VI)," London, UK, Res. MEPC.328(76), Jun. 2021.
- [3] H. N. Psaraftis, "Decarbonization of maritime transport: to be or not to be?," *Maritime Economics & Logistics*, vol. 21, no. 3, pp. 353-371, 2018.
- [4] R. K. Mobley, "An Introduction to Predictive Maintenance," 2nd ed. Woburn, MA, USA: Elsevier, 2002.
- [5] A. K. S. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical Systems and Signal Processing*, vol. 20, no. 7, pp. 1483-1510, 2006.
- [6] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial Internet of Things: Challenges, Opportunities, and Directions," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4724-4734, 2018.
- [7] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, vol. 115, pp. 213-237, 2019.
- [8] F. S. Alqurashi, A. Trichili, N. Saeed, B. S. Ooi, and M.-S. Alouini, "Maritime Communications: A Survey on Enabling Technologies, Opportunities, and Challenges," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 3525-3547, 2023.

- [9] M. Soori, B. Arezoo, and R. Dastres, "Digital twin for smart manufacturing, A review," *Sustainable Manufacturing and Service Economics*, vol. 2, 2023.
- [10] P. V. S. Anusha, P. Swapna, and D. V. R. K. Reddy, "Revolutionizing Conveyor Belt Systems: Empowering Predictive Maintenance with IoT, Cloud, and Machine Learning," *International Journal of Electrical and Electronics Engineering*, vol. 11, no. 6, pp. 224-233, 2024.
- [11] A. S. Kalafatelis, N. Nomikos, A. Giannopoulos, G. Alexandridis, A. Karditsa, and P. Trakadas, "Towards Predictive Maintenance in the Maritime Industry: A Component-Based Overview," *Journal of Marine Science and Engineering*, vol. 13, no. 3, 2025.
- [12] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637-646, 2016.
- [13] M. Badea, O. Bucovețchi, A. V. Gheorghe, M. Hnatiuc, and G. Raicu, "Maritime Industry Cybersecurity Threats in 2025: Advanced Persistent Threats (APTs), Hacktivism and Vulnerabilities," *Logistics*, vol. 9, no. 4, 2025.
- [14] M. Cheliotis, I. Lazakis, and G. Theotokatos, "Machine learning and data-driven fault detection for ship systems operations," *Ocean Engineering*, vol. 216, 2020.
- [15] O. Janssens et al., "Convolutional Neural Network Based Fault Detection for Rotating Machinery," *Journal of Sound and Vibration*, vol. 377, pp. 331-345, 2016.
- [16] S.-H. Kim, T.-G. Kim, J. Lee, H.-K. Song, H. Moon, and C.-J. Chun, "Deep Hybrid Model for Fault Diagnosis of Ship's Main Engine," *Journal of Marine Science and Engineering*, vol. 13, no. 8, 2025.
- [17] J. Chen and X. Ran, "Deep Learning With Edge Computing: A Review," *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1655-1674, 2019.
- [18] M. N. Alam, V. Laxmi, N. Kumar, R. Kumari, and S. Sharma, "Unsupervised Machine Learning for Anomaly Detection: A Systematic Review," *International Journal of Electrical and Electronics Engineering*, vol. 12, no. 7, 2025.
- [19] H. Wang, H. Wang, and X. Tang, "A Review of Deep Learning in Rotating Machinery Fault Diagnosis and Its Prospects for Port Applications," *Applied Sciences*, vol. 15, no. 21, 2025.
- [20] A. Coraddu, L. Oneto, F. Cipollini, M. Kalikatzarakis, G.-J. Meijn, and R. Geertsma, "Physical, data-driven and hybrid approaches to model engine exhaust gas temperatures in operational conditions," *Ships and Offshore Structures*, vol. 17, no. 6, pp. 1360-1381, 2021.
- [21] Z. Wang, X. Yang, T. Li, L. She, X. Guo, and F. Yang, "Intelligent Fault Diagnosis for Rotating Machinery via Transfer Learning and Attention Mechanisms: A Lightweight and Adaptive Approach," *Actuators*, vol. 14, no. 9, 2025.
- [22] T. Pelkonen et al., "Gorilla: A fast, scalable, in-memory time series database," *Proceedings of the VLDB Endowment*, vol. 8, no. 12, pp. 1816-1827, 2015.
- [23] L. Daihatsu Diesel Mfg. Co., "Instruction Manual: Model 6PSHTb-26D Marine Diesel Engine," Osaka, Japan, 1981.
- [24] H. K. Doan, T. P. Mac, A. T. Dinh, and H. B. T. Nguyen, "Development of a Hardware-in-the-Loop Framework for Marine Engine Diagnostics: Performance Evaluation of a Standalone Hybrid ANN Application using PLC-Based Signal Emulation," *Journal Européen des Systèmes Automatisés*, vol. 59, no. 1, pp. 181-193, 2026.
- [25] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861-874, 2006.
- [26] Modbus Organization, "Modbus Application Protocol Specification V1.1b," Dec. 28, 2006.