Indonesian Journal of Maritime Technology (ISMATECH)
Vol. 04, Issue 02, December 2026
eISSN: 3025-518X
DOI: https://doi.org/10.35718/ismatech.v4i2.8482359

Design and Development of an IoT-Based High-Temperature Monitoring System for Early Fire Detection in Ship Engines Using Thermocouple

Suardi[1*], Syahrul Gunawan Sitanggang[1], Faisal Mahmuddin[2], Abdul Basiq Zainal Abidin[2], Aung Ye Kyaw[3], A. Ahmad Fauzan Ihsan[4], Novita Vadina Marta Ria[5]

[1]Department of Naval Architecture, Kalimantan Institute of Technology, Balikpapan, 76127, Indonesia
[2]Department of Marine Engineering, Engineering Faculty, Hasanuddin University, Gowa 92171, Indonesia
[3]Department of Marine Administration, Ministry of Transport and Communications, Myanmar
[4]Department of Informatics Engineering, Faculty of Engineering, Hasanuddin University, Gowa 92171, Indonesia
[5]Power Plant Engineering Technology, Faculty of Vocational Studies, State University of Malang, Malang, 65145, Indonesia

KEYWORDS
Internet of Things;
Ship Engine;
Temperature Monitoring;
K-Type Thermocouple;
Early Warning System.

ABSTRACT – The operational safety of ship engines is a vital aspect of the maritime industry. A major risk threatening vessel safety is fire caused by engine overheating. To mitigate this risk, a real-time temperature monitoring system capable of providing early warnings when temperatures reach dangerous thresholds is required. This research aims to design an Internet of Things (IoT)-based temperature monitoring system that tracks ship engine temperatures in real-time and transmits data to users via the Blynk IoT application. The system utilizes a K-type thermocouple temperature sensor connected to a MAX6675 module for accurate temperature readings. The sensor is interfaced with an Arduino Uno microcontroller, which transmits temperature data via serial communication to a NodeMCU ESP8266. The NodeMCU acts as the transmitter, sending data to the Blynk Cloud platform over a Wi-Fi connection. Additionally, the system features an I2C LCD to display real-time temperature readings and a buzzer to provide an audible alert if the temperature exceeds the set threshold. Test results demonstrate that the system effectively monitors ship engine temperatures in real-time, triggers a buzzer alert during overheating events, and consistently transmits temperature data to the Blynk application. Consequently, this system enhances vessel operational safety by effectively providing temperature information and early warnings to the crew.

*Corresponding Author | Suardi | ✉ (suardi@lecturer.itk.ac.id)

INTRODUCTION

The engine room is one of the most important areas in ship operations because it contains the main engine, power generation equipment, and various mechanical systems that support ship operations [1]. The ship engine functions as a power-generating unit, either as a main engine or an auxiliary engine, and operates at high intensity during ship operation [1][2]. The operation of various equipment in the engine room generates heat, which may cause an increase in temperature within the engine room [3]. Uncontrolled temperature, humidity, and air pressure conditions can affect the performance and service life of equipment installed in the engine room [4]. Therefore, monitoring the environmental conditions of the engine room is essential to maintain the safety and reliability of ship operations [1][4].

Temperature is one of the important parameters that must be considered during ship engine operation because an increase in temperature may indicate a malfunction in the engine system [5]. Continuous operation at excessively high temperatures can reduce engine performance and shorten the service life of engine components [5][6]. Monitoring temperature, humidity, and air circulation in the engine room is necessary to help maintain proper engine operating conditions. In addition, based on the provisions stated in the report, the engine room environmental temperature needs to be maintained below the specified limit to support crew comfort and occupational safety [7]. Therefore, temperature monitoring is an important aspect of preventing overheating conditions in ship engines [5][6].

The operating temperature of the main engine also has a specific range that needs to be considered during ship operation. The normal operating temperature of the main engine generally ranges from 50°C to 60°C and can rise to approximately 80°C under overheating conditions [8]. Based on the information provided in the

This is an open access article under the CC BY-SA license.
Copyright © 2026 by Author

manual book used in the report, the cooling-water temperature during sailing is in the range of 60–70°C, indicating that the cooling system is operating under optimal conditions [9]. An extreme increase in temperature may indicate a malfunction in the cooling system that requires immediate attention. Therefore, a system capable of continuously measuring temperature is required so that changes in engine conditions can be detected at an early stage [10].

Ship accidents are one of the risks that can have significant consequences for crew safety, material losses, and shipping operations[11]. One of these risks is overheating in the engine area due to uncontrolled temperature increases or malfunctions in the cooling system [11]. On May 29, 2021, a fire occurred on the passenger ship Karya Indah while sailing from Ternate to Sanana, North Maluku, which, according to the report, was associated with a delay in detecting the fire in the engine room[12]. Furthermore, on September 24, 2024, a fire occurred on the tugboat Dwi Randa in the waters of Lubuk Ruso, which, according to the report, was associated with engine overheating [13]. These incidents demonstrate that delays in detecting temperature increases may increase the risk of engine damage and fire [11][13].

Conventional monitoring systems generally rely on direct inspections by ship crews, meaning that information regarding temperature changes cannot always be obtained continuously [14]. Periodic inspections also have limitations because temperature increases may occur rapidly when a malfunction occurs in the engine [15]. The application of Internet of Things (IoT) technology can provide a solution by enabling automatic and real-time temperature monitoring. In this study, a Thermocouple sensor is used to detect engine temperature, while an Arduino Uno processes the data and transmits it to a NodeMCU ESP8266 through serial communication [14]. The temperature data are subsequently transmitted through an Internet connection to the Blynk platform, allowing users to monitor the engine temperature through a smartphone.

In addition to functioning as a monitoring system, the developed system is equipped with a warning mechanism using a buzzer when the temperature reaches the predetermined threshold. The integration of a Thermocouple, Arduino Uno, NodeMCU ESP8266, LCD, buzzer, and Blynk enables temperature data to be displayed locally and transmitted remotely. Through this system, changes in temperature can be detected more rapidly, allowing users to take appropriate action before the engine reaches a more hazardous condition. The implementation of an IoT-based monitoring system is expected to improve the operational safety of ship engines through continuous temperature monitoring and early warning [15]. Based on these problems, this study aims to design an IoT-based hot-temperature monitoring system as an early fire warning system for ship engines that can be accessed in real time.

MATERIAL AND METHODS

This study employed an experimental design to develop and evaluate an Internet of Things (IoT)-based temperature monitoring system for ship engines.

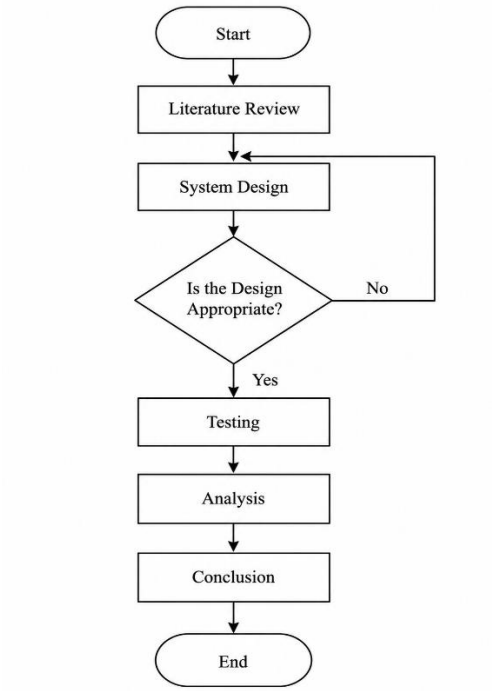


Figure 1. Research Flowchart

The research stages consisted of literature review, system design, hardware assembly, software development, component testing, sensor calibration, installation of the temperature sensor on the ship engine, data acquisition, and data analysis [16]. The experimental approach was used to evaluate the capability of the developed system to measure engine temperature, display measurement results locally, transmit data wirelessly, and provide an early warning when the measured temperature exceeds the programmed threshold. The research workflow was designed to ensure that each stage was completed sequentially before the system was tested under actual engine operating conditions.

System Configuration

The developed monitoring system was designed as an integrated hardware and software system consisting of a K-type Thermocouple, MAX6675 thermocouple interface module, Arduino Uno, NodeMCU ESP8266, LCD 16×2, buzzer, power supply, and Blynk IoT platform. The Thermocouple detects the temperature of the ship engine, while the MAX6675 converts the thermocouple signal into temperature data that can be processed by the Arduino Uno [17]. The Arduino Uno functions as the primary microcontroller for acquiring sensor data and controlling the local display and alarm. The measured temperature is subsequently transmitted through serial communication to the NodeMCU ESP8266, which forwards the data to the Blynk Cloud through a Wi-Fi connection for remote monitoring [18].

The system also incorporates two output mechanisms. The LCD 16×2 provides local temperature information, whereas the buzzer functions as an audible warning when the measured temperature reaches or exceeds the programmed threshold. Thus, the system provides both local and remote monitoring capabilities. The system architecture used in this study is shown in Figure 2.

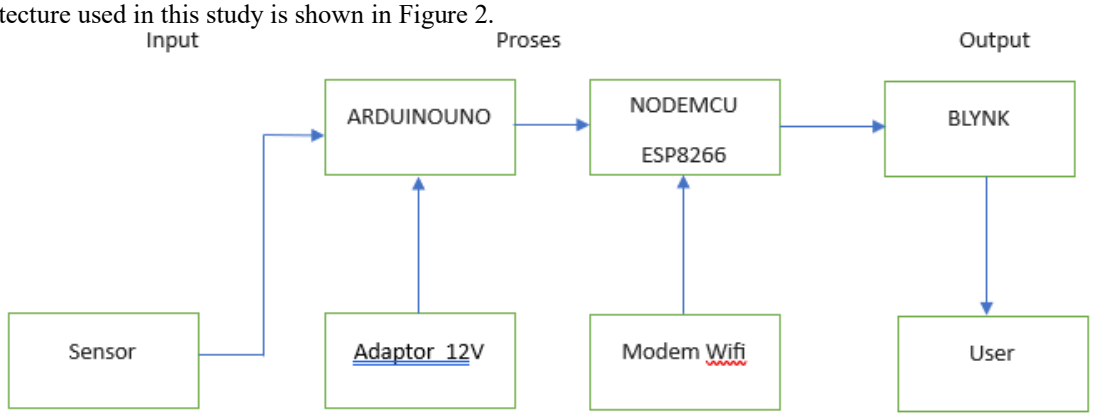


Figure 2. Configuration of the IoT-based Ship Engine Temperature Monitoring System.

Figure 2 shows the data flow within the proposed system. The thermocouple functions as the temperature sensing element, while the Arduino Uno processes the sensor output and provides the temperature information to the LCD and NodeMCU. The NodeMCU ESP8266 serves as the communication interface between the local monitoring system and the Blynk cloud platform [19]. This architecture enables the temperature to be observed locally through the LCD and remotely through a smartphone connected to the Blynk application. The warning mechanism is implemented by comparing the measured temperature with a predetermined threshold. When the measured temperature reaches or exceeds the programmed threshold, the Arduino activates the buzzer as an audible warning. This configuration provides two forms of information, namely local visual information through the LCD and remote information through the IoT platform.

Hardware Configuration

The prototype was assembled using a K-type Thermocouple connected to a MAX6675 module as the temperature sensing unit. An Arduino Uno was used as the main processing unit, while the NodeMCU ESP8266 served as the wireless communication module. A 16×2 I2C LCD was used to display the measured temperature locally, and a buzzer was incorporated as an audible warning indicator [20]. The prototype was assembled using a breadboard, jumper wires, and a 12 V adapter as the power source. The main hardware components used in the study are summarized in Table 1.

Table 1. Main hardware components and their functions

Component	Function
K-type Thermocouple	Measures the temperature of the ship engine
MAX6675	Interfaces the Thermocouple with the Arduino Uno
Arduino Uno	Processes temperature data and controls the local outputs
NodeMCU ESP8266	Transmits temperature data through Wi-Fi
LCD 16×2 I2C	Displays temperature locally
Buzzer	Provides an audible warning
Breadboard	Supports temporary circuit assembly
Jumper wires	Provides electrical connections between components
12 V adapter	Provides electrical power to the prototype
Blynk IoT	Provides remote temperature monitoring

Table 1 indicates that the system is divided into five main functional groups: sensing, processing, communication, visualization, and warning. The thermocouple and MAX6675 module perform the sensing function, while the Arduino Uno performs the primary data-processing function. The NodeMCU ESP8266 provides wireless connectivity, whereas the LCD and buzzer provide local visual and audible outputs, respectively.

The physical arrangement of the components is shown in Figure 3.



Figure 3. Prototype components of the ship engine temperature monitoring system.

Figure 3 illustrates the hardware components used to construct the prototype. For the journal article, the original photograph may be retained, but it is recommended that unnecessary background areas be cropped and the image resolution be improved so that each component can be identified clearly.

Hardware Interconnection

The hardware interconnection was designed to establish communication between the temperature sensor, Arduino Uno, LCD, NodeMCU ESP8266, and buzzer. The K-type thermocouple is connected to the MAX6675 interface, which communicates with the Arduino Uno through an SPI-based connection. The LCD communicates with the Arduino Uno through the I2C interface, while the NodeMCU receives temperature data from the Arduino Uno through serial communication. The buzzer is connected to a digital output of the Arduino Uno and functions as the local warning device.

The complete wiring configuration is presented in Figure 4.

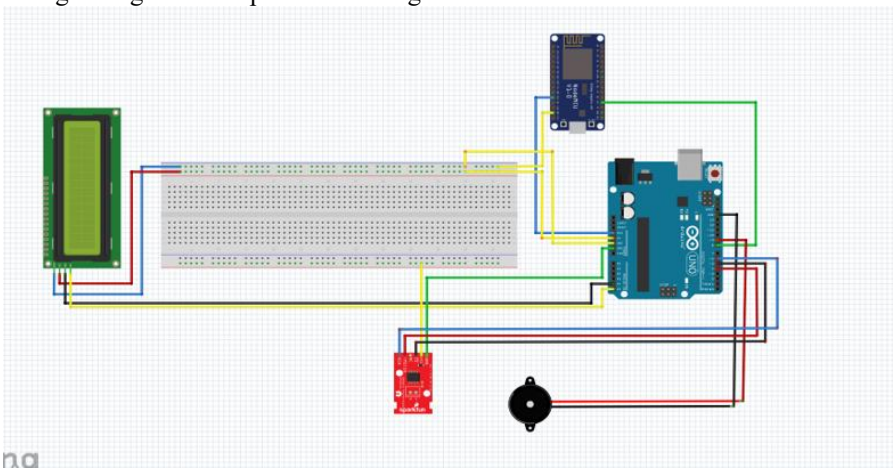


Figure 4. Wiring Configuration of the Temperature Monitoring System.

The main connections used in the prototype are summarized in Table 2.

Table 2. Main Electrical Connections between the System Components

Component	Pin	Arduino Uno
MAX6675	GND	GND
MAX6675	VCC	5 V
MAX6675	SCK	D6
MAX6675	CS	D5
MAX6675	SO/DO	D4
LCD I2C	VCC	5 V
LCD I2C	GND	GND
LCD I2C	SDA	A4
LCD I2C	SCL	A5
NodeMCU	RX	D8
NodeMCU	GND	GND
NodeMCU	3.3 V	3.3 V
Buzzer	Positive	D9
Buzzer	Negative	GND

The thermocouple interface uses digital pins D4, D5, and D6 of the Arduino Uno for temperature data communication, while the LCD uses the I2C interface through pins A4 and A5. The NodeMCU receives serial data through its RX connection, and the buzzer is controlled through digital pin D9. These connections correspond to the wiring configuration documented in the experimental setup.

Software and IoT Integration

The software component was developed using the Arduino IDE for programming the Arduino Uno and NodeMCU ESP8266. The Arduino Uno program is responsible for reading the thermocouple measurement, processing the temperature value, displaying the result on the LCD, transmitting the data to the NodeMCU, and controlling the buzzer. The NodeMCU program receives the temperature data through serial communication and transmits the data to the Blynk IoT platform through a Wi-Fi connection. The Blynk interface was developed to provide remote access to the temperature data through a smartphone. The final monitoring interface is shown in Figure 5.

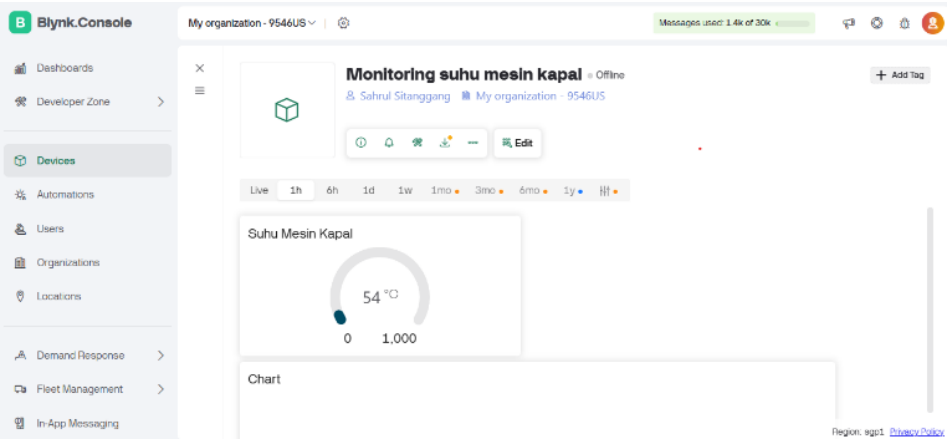


Figure 5. Blynk interface for remote monitoring of ship engine temperature.

Figure 5 demonstrates the remote monitoring function of the developed system. The temperature value measured by the thermocouple is transferred from the Arduino Uno to the NodeMCU and subsequently transmitted to the Blynk platform. This configuration enables the temperature condition of the engine to be monitored remotely without requiring direct access to the prototype.

Sensor Installation and Measurement Procedure

The temperature sensor was installed on the ship engine at a location considered representative of the monitored thermal condition. The installation position was selected by considering accessibility, proximity to the engine heat source, equipment safety, sensor positioning, and the communication connection between the Arduino Uno and NodeMCU. The objective was to obtain temperature measurements from the engine while maintaining the operational condition of the monitoring equipment.
The installation of the monitoring device on the ship engine is presented in Figure 6.



Figure 6. Installation of the temperature monitoring system on the ship engine.

Figure 6 shows the placement of the monitoring system in the engine area. The sensor was positioned at a location where temperature changes could be detected while allowing the electronic components and communication module to remain appropriately positioned. The installation arrangement was intended to facilitate temperature observation while maintaining the operational accessibility of the system. The measurement procedure was subsequently performed under actual engine operating conditions. Temperature data were collected and displayed locally on the LCD, while the NodeMCU transmitted the data to the Blynk platform for remote observation.

Sensor Calibration

Sensor calibration was performed to evaluate the measurement agreement between the developed thermocouple system and a reference temperature-measuring instrument. A Thermogun thermometer was used as the comparison instrument. The calibration was conducted for approximately 10 min with measurements taken at

30-s intervals, resulting in 20 paired temperature observations. The calibration procedure and comparison instrument are documented in the original report.
The measurement error for each observation was calculated using:

$$E_i = \left| \frac{T_{TC,i} - T_{ref,i}}{T_{ref,i}} \right| \times 100\% \quad (1)$$

where E_i is the percentage error at observation i ,$T_{TC,i}$ is the temperature measured by the thermocouple system, and $T_{ref,i}$is the temperature measured by the reference thermometer.
The mean absolute percentage error (MAPE) was calculated as:

$$MAPE = \frac{1}{n} \sum_{i=1}^{n} E_i \quad (2)$$

where n represents the total number of calibration observations.

Table 3. Calibration results of the thermocouple temperature sensor

No.	Time	Thermocouple (°C)	Thermogun (°C)	Error (%)
1	14:00:30	38	40	5.0
2	14:01:00	40	41	2.4
3	14:01:30	42	43	2.3
4	14:02:00	44	45	2.2
5	14:02:30	46	48	4.2
…	…	…	…	…
18	14:09:00	70	71	1.4
19	14:09:30	71	72	1.4
20	14:10:00	72	72	0.0

The calibration results show that the thermocouple measurements followed the reference thermometer measurements with relatively small deviations. The reported errors ranged from 0% to 5.0%, indicating that the sensor was capable of providing temperature measurements with reasonable agreement with the reference instrument over the tested temperature range. Based on the 20 observations, the average absolute percentage error was approximately 2.54%. The calibration procedure therefore provides an experimental basis for using the thermocouple as the temperature sensing element in the subsequent ship-engine monitoring test.

System Performance Testing

Following calibration, the complete monitoring system was tested to evaluate the integration of the sensing, processing, display, communication, and warning functions. The testing consisted of component-level verification and integrated system testing. At the component level, the Arduino Uno, NodeMCU ESP8266, LCD, and thermocouple were checked to ensure that each component could operate according to its intended function. The original test results indicated that these components operated properly. The integrated system was subsequently operated to measure the temperature of the ship engine. The measured temperature was displayed on the LCD and transmitted through the NodeMCU to the Blynk platform. The buzzer was configured to operate when the measured temperature reached or exceeded the programmed threshold.The component-level testing results are summarized in Table 4.

Table 4. Component-level functional testing

Component	Test result	Functional status
Arduino Uno	Program executed properly	Good
NodeMCU ESP8266	Program executed properly	Good
LCD	Display operated properly	Good
Thermocouple	Temperature data detected	Good

Open Access: https://journal.itk.ac.id/index.php/ismatech

Table 4 indicates that the main electronic components were operational before the complete system was deployed. This preliminary verification was important to minimize the possibility that an individual component failure would affect the interpretation of the integrated monitoring results.

Field Data Acquisition

Field measurements were conducted on the ship engine to observe temperature variation during operation. Temperature data were collected continuously and recorded at one-minute intervals during the monitoring period. The measured values were displayed locally through the LCD and transmitted to the Blynk application through the NodeMCU ESP8266. The field measurement data were subsequently analyzed descriptively by determining the observed temperature range and identifying changes in temperature during different engine operating conditions. The data were also used to evaluate whether the developed system was capable of continuously monitoring engine temperature and transmitting the measurement results to the remote monitoring platform.

Data Analysis

The measurement data obtained from the calibration and field testing stages were analyzed quantitatively to evaluate the measurement performance and monitoring capability of the developed system. The calibration data were analyzed by comparing the temperature measured by the K-type thermocouple with the temperature obtained from the Thermogun reference instrument. The measurement error for each observation was calculated as the absolute percentage difference between the thermocouple and reference temperature values, while the Mean Absolute Percentage Error (MAPE) was used to determine the overall measurement error across all calibration observations. This analysis was used to assess the degree of agreement between the developed temperature measurement system and the reference instrument.

The field measurement data were analyzed using descriptive analysis to characterize the temperature variation of the ship engine during operation. The minimum, maximum, and observed temperature trends were determined from the recorded measurements, while changes in temperature were examined in relation to the operating condition of the engine. In addition, the functionality of the IoT communication system was evaluated by observing the transmission of temperature data from the Arduino Uno through the NodeMCU ESP8266 to the Blynk platform. The analysis was therefore focused on two main performance aspects, measurement accuracy of the thermocouple system and functional capability of the IoT-based monitoring system. The results of these analyses were subsequently used to determine the feasibility of the developed system for real-time ship engine temperature monitoring and early warning applications.

RESULTS AND DISCUSSION

Prototype Implementation and System Functionality

The developed monitoring system integrates a K-type thermocouple with a MAX6675 module, Arduino Uno, NodeMCU ESP8266, LCD 16×2, and buzzer as the main hardware components. The Arduino Uno functions as the primary controller for acquiring temperature data from the thermocouple, while the NodeMCU ESP8266 transfers the measured temperature data to the Blynk IoT platform through a wireless network. The LCD provides local temperature information, whereas the buzzer is configured as an audible warning when the measured temperature reaches the predefined threshold. The assembled prototype and its components are shown in Figure 7.



Figure 7. Prototype of the IoT-based ship engine temperature monitoring system.

The prototype demonstrates the integration of the sensing, processing, local display, communication, and warning functions within a single monitoring system. The component test reported in the thesis showed that the Arduino Uno, NodeMCU ESP8266, LCD, and thermocouple operated properly during preliminary testing. This result indicates that the individual hardware components were able to perform their intended functions before the system was installed for field measurement. The system architecture also provides two monitoring paths. Temperature information can be observed directly through the LCD near the engine, while the same information is transmitted through the NodeMCU ESP8266 to the Blynk platform for remote observation. This dual-monitoring approach is useful because local monitoring remains available when remote communication is interrupted, while the IoT interface allows temperature information to be accessed without continuous physical inspection of the engine area. The role of the LCD as a local interface and the NodeMCU as a wireless data transmitter is consistent with the system configuration described in the thesis.

Sensor Calibration Results

The calibration results are shown in Table 5.

Table 5. Thermocouple Calibration Results

No.	Time	Thermocouple (°C)	Thermogun (°C)	Error (%)
1	14:00:30	38	40	5.0
2	14:01:00	40	41	2.4
3	14:01:30	42	43	2.3
4	14:02:00	44	45	2.2
5	14:02:30	46	48	4.2
6	14:03:00	48	49	2.0
7	14:03:30	50	52	3.8
8	14:04:00	51	53	3.8
9	14:04:30	53	55	3.6
10	14:05:00	55	56	1.8
11	14:05:30	57	58	1.7
12	14:06:00	59	60	1.7
13	14:06:30	61	62	1.6
14	14:07:00	63	65	3.1
15	14:07:30	65	67	3.0
16	14:08:00	67	69	2.9
17	14:08:30	68	70	2.9
18	14:09:00	70	71	1.4
19	14:09:30	71	72	1.4
20	14:10:00	72	72	0.0

The calibration results show that the thermocouple measurement followed the reference thermometer consistently. The lowest measurement error was 0%, while the highest error was 5.0%. Based on the 20 measurement points, the average measurement error was approximately 2.54%. The relatively small difference between the thermocouple and reference measurements indicates that the developed sensor provided consistent temperature readings within the tested range. The largest difference occurred at the beginning of the measurement, where the thermocouple recorded 38 °C while the reference thermometer recorded 40 °C. At the final measurement point, both instruments recorded 72 °C, resulting in an error of 0%. The calibration result supports the use of the thermocouple as the temperature sensing element in the proposed monitoring system. However, the result represents the performance of the prototype under the tested conditions and should not be interpreted as a general accuracy specification for all thermocouple installations.

Temperature Monitoring During Engine Operation

The monitoring test was conducted for approximately one hour on 22 May 2025. The recorded temperature changed according to the operating condition of the engine.

Table 6. Ship Engine Temperature Monitoring Results

No.	Time	Temperature (°C)
1	14:00	64
2	14:05	61
3	14:10	58
4	14:15	62
5	14:20	66
6	14:25	56
7	14:30	53
8	14:35	48
9	14:40	53
10	14:45	55
11	14:50	46
12	14:55	52
13	14:59	56

The complete measurement dataset contained 60 one-minute observations. The temperature initially reached 64 °C and increased to a maximum of approximately 67 °C around 14:18–14:19. After that period, the temperature gradually decreased to approximately 52–53 °C before increasing again toward the end of the observation period. The complete dataset indicates that the temperature was dynamic during engine operation rather than remaining constant. This variation demonstrates the importance of continuous monitoring because the temperature changed according to the operating condition of the engine. The highest value observed in the one-hour dataset was 67 °C, while the report also recorded a maximum temperature of 72 °C during the broader measurement/test sequence.

Temperature Characteristics Under Different Engine Conditions

The temperature measurements demonstrated a clear variation in response to different ship-engine operating conditions, with the thermocouple recording approximately 32–38 °C when the engine was not operating and 45–57 °C under light to moderate operation. Under the optimal operating condition observed in the field test, the temperature increased to approximately 67 °C, indicating that the sensor was able to detect thermal changes associated with engine operation. The broader measurement sequence used during the experimental evaluation recorded a maximum temperature of 72 °C, which was obtained during the calibration test and therefore represents a different measurement condition from the one-hour field dataset. The increase in temperature with engine operation indicates that the developed sensing system was responsive to changes in the thermal condition of the engine and can be used to identify temperature trends during operation.

IoT-Based Temperature Monitoring Performance

The IoT monitoring test demonstrated that the temperature data measured by the thermocouple could be transmitted from the Arduino Uno to the NodeMCU ESP8266 and subsequently displayed through the Blynk platform. During operation, the measured temperature was processed by the Arduino Uno and transferred through serial communication before being forwarded by the NodeMCU to the Blynk Cloud using a Wi-Fi connection. The Blynk interface enabled the measured temperature to be observed remotely, providing an additional monitoring capability without requiring the operator to remain directly beside the engine. The successful transmission and visualization of the temperature data indicate that the developed IoT architecture was able to integrate local temperature measurement with remote monitoring functionality. Therefore, the system can support early identification of temperature changes in the ship-engine environment by providing accessible temperature information to the operator during operation.

Early Warning Function

The buzzer was programmed to activate when the measured temperature reached the predetermined threshold of 100 °C. This function was designed to provide an immediate local warning when an extreme temperature condition occurred.

The alarm logic can be expressed as:

$$T < 100°C \rightarrow \text{Buzzer OFF}$$

$$T \geq 100°C \rightarrow \text{Buzzer ON}$$

During the recorded operational test, the maximum observed engine temperature was below the programmed 100 °C threshold. Therefore, the temperature monitoring data primarily demonstrated the sensing and transmission functions rather than a real overheating event at the alarm threshold. This distinction is important. The prototype successfully implements the alarm mechanism, but the field measurements reported in the study did not represent an actual fire or extreme overheating condition. Consequently, the system should be regarded as an early-warning prototype rather than a complete certified marine fire-detection system.

CONCLUSION

The IoT-based ship-engine temperature monitoring system was successfully developed using a K-type thermocouple integrated with a MAX6675 module, Arduino Uno, NodeMCU ESP8266, LCD, buzzer, and Blynk platform. The calibration results showed an average absolute percentage error of 2.54%, with the thermocouple measurement reaching a maximum recorded temperature of 72 °C, indicating satisfactory agreement with the Thermogun reference measurements. During the one-hour field measurement, the system successfully recorded temperature variations from 48 to 67 °C, with the highest temperature occurring during the higher engine operating condition. The developed system also successfully transmitted temperature data from the Arduino Uno through the NodeMCU ESP8266 to the Blynk platform, while the LCD provided local temperature information for on-site monitoring. Overall, the developed system demonstrates the feasibility of combining thermocouple-based temperature measurement with IoT communication to support continuous ship-engine temperature monitoring and provide an early warning function for potentially excessive thermal conditions.

REFERENCES

[1] Y. Wardi, D. Bak, and F. Al Aziz, "Sistem Monitoring Kondisi Mesin Kapal Berbasis Internet of Things (IoT) dan Predictive Maintenance Menggunakan Mikrokontroler dan Machine Learning di Edge," *ESDJ Electron. Sea Def. J.*, vol. 1, pp. 67–74, 2025, [Online]. Available: https://journal.akademitnial.ac.id/index.php/ESDJ/article/view/161/130

[2] Suardi S, Pratama BY, Pawara MU, Yosefan A, Anugerah AN, Abdurrahman L, Tasrief M, Klara S. Optimizing generator power usage through LED lighting distribution on tugboats: A case study of a 26.80-meter vessel. Indonesian Journal of Maritime Technology. 2025 Apr 25;3(1).

[3] D. A. N. Analisis and K. Sistem, "Perancangan Model Sistem Monitoring Mesin Kapal Berbasis Internet Of Things (Iot): Arsitektur 4-Tier Dengan Deteksi Anomali Berbasis Machine Learning," vol. 6, no. 03, pp. 203–213, 2024.

[4] N. N. Saphira, M. R. Pangestu, and M. K. Naf'an, "Pengembangan Sistem Smart Ship Safety Melalui Kolaborasi IoT Dan Condition-Based Maintenance Mesin Kapal," *Bluepr. J.*, vol. 1, no. 3, pp. 104–109, 2025.

[5] M. R. Mauk, S. I. Pella, and H. F. J. Lami, "Sistem Internet of Things (Iot) Deteksi Kebakaran Lahan Pada Daerah Tanpa Infrastruktur Internet," *J. Ilm. Flash*, vol. 10, no. 1, pp. 31–35, 2024, [Online]. Available: https://www.researchgate.net/publication/387516660

[6] A. Riyadi, E. Nursanti, and J. H. Galuh, "Perancangan Sistem Monitoring Deteksi Kebakaran Berbasis Internet of Things (IoT) Untuk Menurunkan Risiko Kebakaran," *J. Valtech*, vol. 7, no. 1, pp. 123–128, 2024, doi: 10.36040/valtech.v7i1.9272.

[7] M. V. Gultom and I. S. Putro, "Sistem Deteksi Kebakaran Berbasis Iot Dengan Mikrokontroler Esp32," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 2, 2025, doi: 10.23960/jitet.v13i2.6236.

[8] D. H. Pranata and L. Latipah, "Prototype Pendeteksi Kebakaran Dini Berbasis Internet of Things (IoT) dengan Notifikasi Telegram," *J. Ilmu Komput. dan Bisnis*, vol. 15, no. 1, pp. 116–124, 2024, doi: 10.47927/jikb.v15i1.709.

[9] F. Salam and O. Alexander, "Perancangan Monitoring Suhu Berbasis Internet of Things Dengan Node Mcu Esp8266, Dht 11 Dan Thingspeak," *J. Ilm. Inform.*, vol. 11, no. 01, pp. 22–26, 2023, doi: 10.33884/jif.v11i01.6546.

[10] Suardi AI, Pawara TM, Alamsyah TH. Patrol ship design to guard the natuna seas. Int. J. Mar. Eng. Innov. Res. 2022 Sep 22;7(3):171-9. doi: https://doi.org/10.12962/j25481479.v7i3

[11] P. O. Putra, H. Setiawan, and Z. R. Mair, "Implementasi Alat Monitoring Suhu Ruangan Berbasis Internet Of Things (IoT) Menggunakan Metode MQTT dan HTML Pada Ruangan Server Universitas Indo Global Mandiri Palembang," *J. Softw. Eng. Comput. Intell.*, vol. 1, no. 1, pp. 40–50, 2023, doi: 10.36982/jseci.v1i1.3046.

[12] Setiawan W, Kyaw AY. Plan for the power requirements of the lights in the fishing boat room using LED lights. Indonesian Journal of Maritime Technology. 2023 Jun 27;1(1). doi: https://doi.org/10.35718/ismatech.v1i1.877

[13] Y. A. Yunus, R. Satra, and F. Fattah, "Sistem Monitoring Suhu Pada Inkubator Penetas Telur Berbasis IoT," *LINIER Lit. Inform. dan Komput.*, vol. 1, no. 4, pp. 389–394, 2024, doi: 10.33096/linier.v1i4.2538.

[14] J. R. Material and M. Energi, "FT-UMSU FT-UMSU," vol. 7, no. 2, pp. 335–342, 2024.

[15] H. Firmansyah and W. Indrasari, "Internet of Things-Based Real-Time Thermal Monitoring: Utilizing the Affordable MLX90640 Thermal Camera and the Industrial MAX6675 Thermocouple Module with a Linear Regression Characterization Method," *Positron*, vol. 15, no. 1, pp. 94–103, 2025, doi: 10.26418/positron.v15i1.89878.

[16] Aditya Putra Mahendra, Sri Mulyanto Herlambang, and Prima Yudha Yudianto, "Rancang Bangun Sistem Monitoring Kinerja Main Engine Kapal Berbasis IOT," *Globe Publ. Ilmu Tek. Teknol. Kebumian, Ilmu Perkapalan*, vol. 2, no. 4, pp. 139–159, 2024, doi: 10.61132/globe.v2i4.575.

[17] C. Prastyadi, B. G. Irianto, H. G. Ariswati, D. Titisari, S. Nyatte, and S. Misra, "Analysis of the Accuracy of Thermocouple Sensors at the Incubator Calibrator Laboratory Equipped with Internet of Thing-Based Data Storage," *Indones. J. Electron. Electromed. Eng. Med. Informatics*, vol. 4, no. 4, pp. 160–167, 2025, doi: 10.35882/ijeeemi.v4i4.158.

[18] D. P. Prakosa and S. Wijayanta, "Internet of Things (IoT)-Based Brake Fluid Condition Detection and Vapor Lock Prevention Device," *J. Nas. Pendidik. Tek. Inform.*, vol. 15, no. 1, pp. 24–34, 2026, doi: 10.23887/janapati.v15i1.105070.

[19] Clara S. Perencanaan distribusi penerangan untuk ruangan di atas kapal tb liberty 217 gt menggunakan metode zonal cavity. INOVTEK POLBENG. 2023 Jun 13.

[20] Suardi S, Maulana MK, Ikhwani RJ, Pawara MU, Mahmuddin F, Tasrief M. Design and Implementation of Solar Cells as an Alternative Power Source for Pinisi Ships. Computational And Experimental Research In Materials And Renewable Energy. 2024 Nov 25;7(2):93-104.