:: IJEEIT ::
(International Journal of Electrical Engineering and Information Technology)
Volume 06 Number 02 September 2023

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

Valve Setting Using Fuzzy Logic Method To Control Main Engine Coolant Flow

Didik Dwi Suharso
Semarang Maritime Polytechnic, Indonesia
didik@pip-semarang.ac.id

Imam Syafi'i
Semarang Maritime Polytechnic, Indonesia
imam.safii@pip-semarang.ac.id

Anicitus agung nugroho
Semarang Maritime Polytechnic, Indonesia
anicitusagung@gmail.com

ABSTRACT

The main objective of this research is to improve the operating efficiency and performance of ship main engines by optimizing coolant flow. Improper coolant flow can cause excessive temperature rise in the engine, which in turn can reduce engine life and efficiency. In this research, the coolant flow control system is implemented using the fuzzy logic method. Fuzzy logic allows modeling that is more adaptive to parameter variations and uncertainties in the operational environment. Fuzzy rules are developed based on practical knowledge of experts and operational data related to the host machine. The test was carried out through a simulation using a water flow control circuit that supplies the coolant to the ship's main engine with the main components being an ESP32 microcontroller chip and three DS18B20 temperature sensors. The results of this research show that the use of fuzzy logic in regulating coolant flow is able to provide a faster and more accurate response to changes in engine operational conditions. This has the potential to increase cooling efficiency and prevent engine over-temperatures, which can ultimately increase the overall service life and performance of the host engine. The practical implications of the results of this research can be applied in the development of more intelligent and adaptive control systems for various types of host machines, with the potential to increase operating efficiency and reduce the risk of damage due to excessive temperatures.

Keywords: Ship Engine Cooling, Fuzzy Logic, ESP32, Water Flow Control, Temperature Control

1. Introduction

The main engine is the main driving force on the ship which functions as a converter of mechanicalpower into a driving force so that the ship's propeller can move and then run the ship. In the operation of the main engine that lives for a long time, heat will arise in the engine parts, it will cause an increase in temperature due to heat from combustion, especially in the parts that are in direct contact with the combustion chamber. Diesel engines can move because combustion occurs in the cylinder and from combustion it will produce high heat, if the engine is not cooled overheating can occur and accelerate wear, a cooling system is needed to prevent overheating.

Cooling the main engine is one of the important activities in a ship when the main engine is working or functioning, because the smooth operation of the ship depends on the performance of the main engine, so that parts of the main engine are maintained from stress caused by heat, heat emergence must be controlled. This condition can be overcome by circulating the cooling media using a constant and stable pressure to all parts of the main engine such as the cylinder head, cylinder jacket cooling, and injectors. This activity is an obligation for supervision by the engine crew so that the flow of cooling media remains smooth. Cooling media in diesel engines can be used such as oil, air and water. The three cooling media, water is a cooling medium with a very good level of heat absorption.

In this study, the selenoid valve setting does not need supervision by the crew anymore but has been regulated by the system automatically using a microcontroller with the fuzzy logic method in controlling the flow of coolant to the main engine. From there the author gives the title of his research is "Valve settings with the fuzzy logic method to control the flow of main engine coolant".

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

1.1 Main Engine Cooling

Ships that run for days require an engine that can work optimally so it needs a good engine design. Poor machine design results in the machine not being able to work properly so that the energy produced is wasted in the form of heat energy. To optimize the energy produced, it is necessary to reduce the heat generated, namely using a cooling medium (Cooller) so that the engine functions can work optimally and avoid faster engine damage. Installation of coolers on the main engine on the ship which functions to reduce the heat generated by the main engine. In order for the main engine building in the form of a diesel motor to maintain the voltage output caused by the heat generated by the work of the engine, the engine temperature must be controlled. The way to overcome this situation is to circulate the cooling medium to all components of the main engine with constant pressure such as injectors, cylinder heads and cylinder jacket cooling. The crew must always supervise this main engine so that the cooling media flows smoothly in this system. So that the main engine temperature is in the desired condition, it is necessary to always maintain the working of the cooling system. To produce motion and energy from the results of the combustion process by internal (and external) combustion engines. Imperfection of the engine will reduce the efficiency of energy conversion that occurs from the heat of combustion, some will be wasted through the exhaust channel and some will be absorbed by the material in the combustion chamber. To increase the efficiency of the engine so that the conversion of heat from combustion products into energy and will be converted into mechanical movement, a good circulation of cooling media is needed. Construction of the main engine cooling system can be seen in Figure 1. The working principle of the main engine cooling is to absorb the main engine heat through water as a cooling medium to absorb heat, the components of the system start from the sea water sucked from the sea chest using a sea water pump and before passing through the pump it must first pass through a strainer (filter) then to the cooler to absorb heat and reconnect to the sea. Sea water is directly used in the engine system as a cooling medium for heat absorption. The sea water cooling system only passes to absorb heat and will be wasted back into the sea.

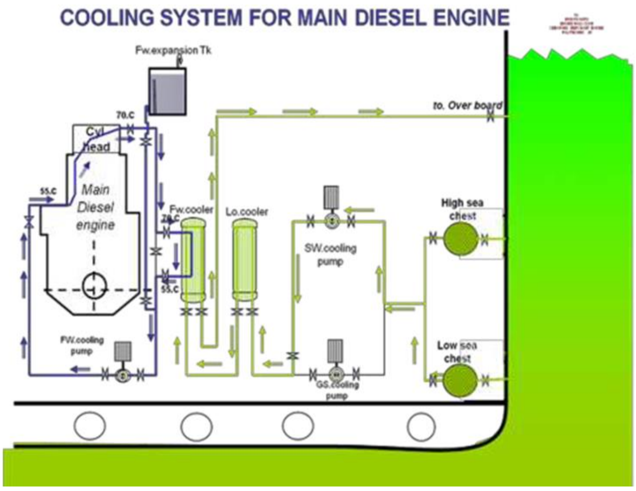


Figure 1. Main Engine Cooling System(*Sistem Pendingin Mesin Induk Kapal | PDF*, n.d.)

2.2 Fuzzy Logic

Fuzzy logic is an approach in computer science and artificial intelligence introduced by Lotfi Zadeh in 1965. The main concept in fuzzy logic is the use of membership degrees to measure the extent to which an object or concept can be included in a set that has defined boundaries. more obscure than sets in classical logic.(Ross, 2010) In other words, fuzzy logic allows modeling situations that are not only right or wrong, but also include varying degrees of truth between the two. This enables more flexible decision making and adapts to uncertainties in many domains.

2.2.1 Fuzzification

Fuzzification is the process of assigning numerical input from a system to a fuzzy set with a certain degree of membership. This membership level can be anywhere in the [0,1] interval.(Jomaa et al., 2019) If 0 then the value is not included in the given fuzzy set, and if 1 then the value is fully included in the fuzzy set. Any value between 0 and 1 represents the degree of uncertainty that the value belongs

:: IJEEIT ::
(International Journal of Electrical Engineering and Information Technology)
Volume 06 Number 02 September 2023

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

to the set. These fuzzy sets are usually described in words, and by assigning system input to the fuzzy sets, we can reason with them in a linguistically natural way.(*Agustianto et al.,2021*)
For example, in the figure 2 below, the meanings of the expressions cold, warm, and hot are represented by the temperature scale mapping function. The point on the scale has three "truth values"—one for each of the three functions. The vertical line in the figure represents a specific temperature as measured by the three arrows (truth value). Since the red arrow points to zero, this temperature can be interpreted as "not hot"; i.e. this temperature has zero membership in the "hot" fuzzy set. The orange arrow (pointing 0.2) might describe it as "slightly warm" and the blue arrow (pointing 0.8) "pretty cool". Therefore, this temperature has 0.2 membership in the "warm" fuzzy set and 0.8 membership in the "cold" fuzzy set. The degree of membership assigned to each fuzzy set is the result of fuzzification.(Passino & Yurkovich, 1998)

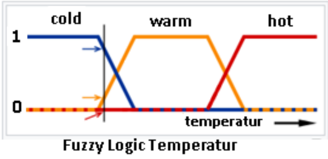


Figure 2. Membership Function

A fuzzy set is often defined as a triangular or trapezoidal curve, because each value will have a slope where the value increases, a peak where the value equals 1 (which can have a length of 0 or greater) and a slope where the value decreases. They can also be defined using the sigmoid function. One common case is the standard logistic function which is defined as(Nguyen & Walker, 2006)

$$S(x) = \frac{1}{1+e^{-x}}$$

which has the following symmetrical properties

$$S(x) + S(-x) = 1$$

From here it follows

$$(S(x) + S(-x)).(S(y) + S(-y)).S(z) + S(-z)) = 1$$

2.2.2 Inference Engine

Inference engine is a key component in a rule-based system or knowledge-based system that uses logic to draw conclusions or make decisions based on existing information in the form of predetermined rules or knowledge. It is part of the inference process in artificial intelligence systems.(Kaswidjanti, 2014) In simple terms, the inference engine functions as a machine that runs the process of reasoning or logical thinking based on predetermined rules. It receives facts or input information, examines the relevant rules, and then generates conclusions or actions based on the logic defined in those rules.

2.2.3 Defuzzification

Defuzzification converts the fuzzy output from the fuzzy inference engine into crisp values so that they can be fed to the controller. The resulting fuzzy results cannot be used in applications where decisions must be taken only on crisp values. The controller can only understand sharp output. So it is necessary to convert the fuzzy output into a crisp value. There is no systematic procedure for choosing a good defuzzification strategy.(Ross, 2010) The choice of defuzzification procedure depends on the nature of the application.
Rule base:
Consider the following two rules in a fuzzy rule base.

R 1 : If x is A then y is C
R 2 : If x is B then y is D

A pictorial representation of the rule base above is shown in the figure 3.
The Center of Largest Area (CoA) method is a simple, computationally effective, and widely used defuzzification that can be seen in figure 4.(Revathi & Sivakumaran, 2016) If the fuzzy set has two sub-regions, then the center of gravity of the sub-region with the largest area can be used to calculate the defuzzification value.

$$x^* = (\int \mu \bar{C}_m(x).x' dx)/(\int \mu\bar{C}_m(x)dx)$$

Here, $\bar{C}_m$ is the largest area, x′ is the center of gravity , $\bar{C}_m$; where , $\bar{C}_m$ =C_3

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

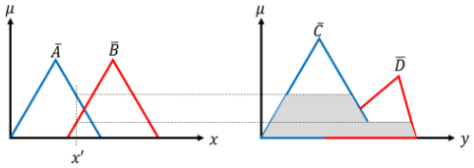


Figure 3. Basic Defuzzification Rules

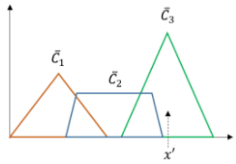


Figure 4. Center of Area (CoA) Defuzzification Method

1.3 ESP32 Microcontroller

Espressif Company (The semiconductor company that manufactures the ESP8266) has released the ESP32 microcontroller which is the successor to the ESP8266. ESP32 not only has WiFi support, but is also equipped with Bluetooth 4.0 (BLE/Bluetooth Smart) making it perfect for almost all IoT projects. One of the strengths of the ESP32 is that it has more GPIOs than the ESP8266. The ESP32 has a total of 48 GPIO pins, only 25 of which connect to the pin headers on both sides of the development board (ESP32 development board). (Cameron, 2023)The construction of the ESP32 pin out can be seen in Figure 5.

1.4 Selenoid Valve

Solenoid valves are electrically controlled valves. The valve has a solenoid, which is an electric coil with a movable ferromagnetic core (plunger) in the center. In the resting position, the plunger closes the pinhole. An electric current through the coil creates a magnetic field. The magnetic field exerts an upward force on the pusher which opens the hole. This is the basic principle used to open and close a solenoid valve. Solenoid valve construction is shown in figure 6. The solenoid valve consists of two main components: the solenoid and the valve body (G). Figure 10. shows the components. The solenoid has an electromagnetic inductive coil (A) around a central iron core called a pusher (E). At rest, it can be normally opened(NO) or normally closed(NC). In the de-energized state, normally open valves open and normally closed valves close. When current flows through the solenoid, the coil is energized and creates a magnetic field. It creates a magnetic attraction with the pusher, sets it in motion and overcomes the spring force (D). If the valve is normally closed, the plunger is lifted so that the seal (F) opens the orifice and allows media flow through the valve. If the valve is normally open, the plunger moves downward so that the seal (F) blocks the orifice and stops the flow of media through the valve. Shading ring (C) prevents vibration and hum in the AC coil.

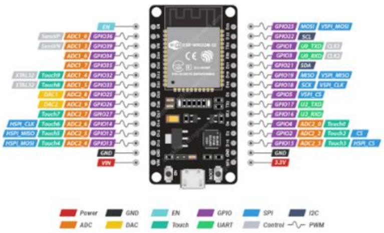


Figure 5. ESP32 Microcontroller Pin Out

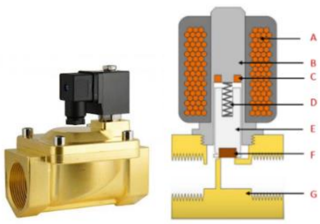


Figure 6. Selenoid Valve

2. Reasearch Method

The method of carrying out this research was carried out in the laboratory of the Semarang Shipping Science Polytechnic with the following stages:

Table 1. Research Process Flow

literature review	Collect literature from various existing sources in the form of reference books, papers, publications and online articles related to research on solenoid valve settings using the fuzzy logic method to control the main engine coolant.
Hardware Design	The hardware design includes the design of electronic circuits that support the implementation of the manufacture of electronic devices for controlling solenoid valves using the fuzzy logic method to control the main engine coolant.
Software Design	Software design which includes program design for the Arduino microcontroller and fuzzy logic for selenoid valve settings
Testing & Evaluation	Carry out system testing and evaluate the results of the system design

:: IJEEIT ::
(International Journal of Electrical Engineering and Information Technology)
Volume 06 Number 02 September 2023

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

2.1 Hardware Design

In this research an ESP32 microcontroller is used which is embedded in the Fuzzy Logic system to regulate the bypass valve which is useful for controlling the main engine coolant. The temperature sensor will read the condition of the liquid temperature (fresh water) which is supplied to the main engine from around 55^{O}C and cooled to around 30^{O}C in a fresh water cooler using sea water. Fresh water flows from the heat exchanger tank through the built-in pump, then exits to the engine parts and then fresh water passes through the cylinder block, heat cylinder and cylinder liner with a temperature of 30°C. After fresh water circulates through the inside of the engine, fresh water returns into the heat exchanger which is then cooled using sea water.

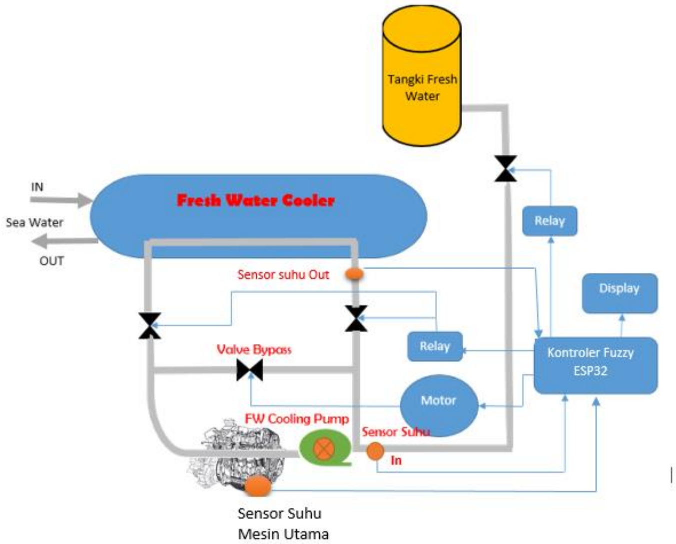


Figure 7. Block Diagram of Bypass Valve Settings Using the Fuzzy Logic Method

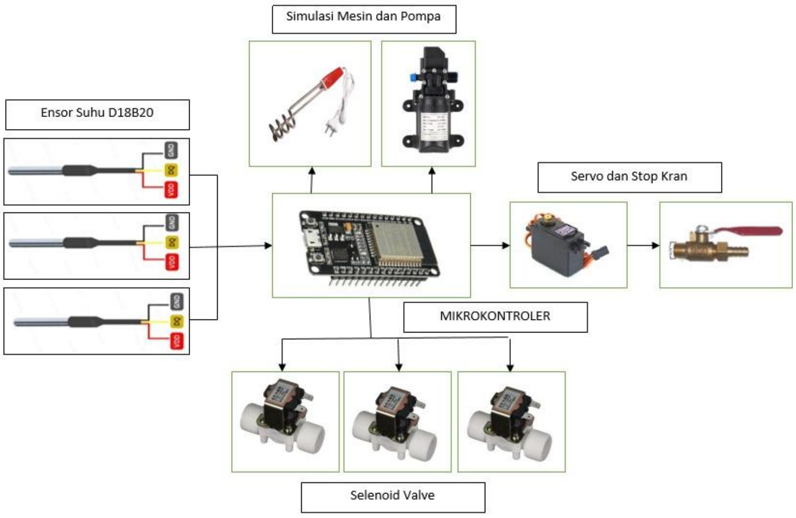


Figure 8. System Hardware Connections

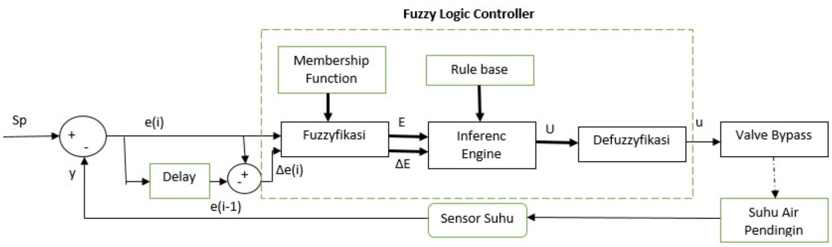


Figure 9. Block Diagram of Temperature Control Using a Fuzzy Logic Controller

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

2.2.1. Temperature Sensor Circuit

The temperature sensor used is a DS18B20 type temperature sensor which has high accuracy and only uses one path to access the resulting temperature. To use this sensor, several software libraries are needed, namely OneWire and DallasTemperature. This sensor will be placed at certain points to measure the temperature of the cooling water at the input of the main engine and the temperature of the water at the output of the main engine, as well as another one to measure the temperature of the engine itself, as shown in Figure 10.

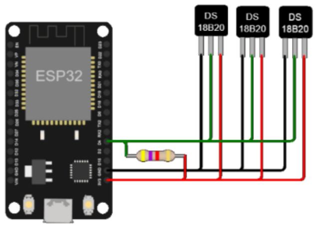


Figure 10. Connection of 3 temperature sensors with ESP32

All sensor outputs are put together to enter GPIO pin 4 and are given a 4k7 resistor. The function of the pull-up resistor is as a pull-up on the DS18B20 temperature sensor which is connected to the 1-Wire communication protocol. In a 1-Wire circuit, a pull-up resistor is used to pull the data leg voltage (GPIO 4) to a high level when there is no data transmission. The 1-Wire protocol uses a data signal with a pull-up architecture which means that the Data leg (GPIO 4) on the sensor is usually "pulled" to a high voltage level by a pull-up resistor. When the sensor wants to transmit data, the Data leg (GPIO 4) is pulled to a low level by the sensor, and data is sent via the voltage variation. The function of the 4.7kΩ pull-up resistor is to ensure that when there is no data transmission, the Data leg (GPIO 4) remains at the high voltage level which is the standard condition for 1-Wire communication. With the pull-up resistor, the data signal can work properly and avoid the ambiguity that can occur if the Data leg (GPIO 4) is not pulled to a high level. It is important to note that this pull-up resistor must be connected between the VCC leg (3.3V on the ESP32) and the Data leg (GPIO 4) on the DS18B20 sensor. Also, ensure that only one pull-up resistor is used in the circuit as the DS18B20 temperature sensor usually only requires one Data connection (GPIO 4). By using a pull-up resistor, 1-Wire communication between the ESP32 and the DS18B20 temperature sensor can work properly and temperature data can be read accurately.

2.2.2. Cooling Tank and Fresh Water Valve Control Circuit

To open the pipe cooling tank, an automatic valve is needed which can be controlled by a microcontroller. This valve will automatically open if the water temperature in the main engine cooling pipe increases and also the bypass valve will be closed slowly using the fuzzy logic method as the water temperature in the cooling pipe increases. By giving a LOW value to pin 27 ESP32, this relay is active and moves the switch between COMM and NO (Normally Open) which is connected to the two valves on the cooling tube. This condition will be carried out if the water temperature in the pipe exceeds 60oC so that it is hot enough and must be reduced so that it is effective in cooling the main engine and the main engine can work properly or normally. If the water temperature in the pipe has fallen below 40oC then this valve will close and the main engine cooling water flow is let through the bypass valve which is regulated by the fuzzy method.

The valve control of the fresh water tank is also carried out using a driver circuit equipped with an automatic relay connected to pin 26 of ESP32. When the cooling pipe is empty or there is no water, it will be filled from the tank by opening the valve with the LOW logic command on pin 26. The circuit will move a relay that connects the solenoid valve to the voltage supply. The circuit diagram for the Fresh Water Tank Driver Valve circuit is as follows:

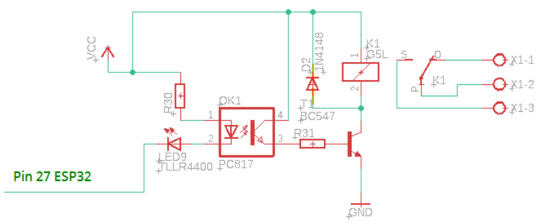


Figure 11. Driver Valve Circuit

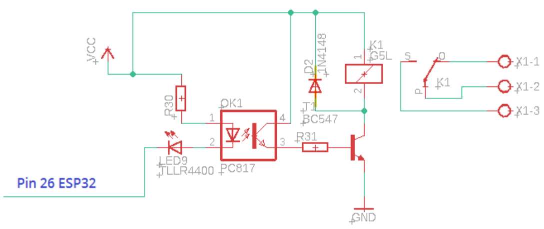


Figure 12. Fresh Water Tank Driver Valve Circuit

:: IJEEIT ::
(International Journal of Electrical Engineering and Information Technology)
Volume 06 Number 02 September 2023

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

2.2.3. Bypass Valve Control Circuit

The bypass valve used to pass cooling water is driven using a stepper motor. This stepper motor rotation will determine how much this valve is opened to drain water in the cooling pipe. A stop valve faucet is coupled to the axle of the stepper motor whose rotation will be regulated by fuzzy logic software.

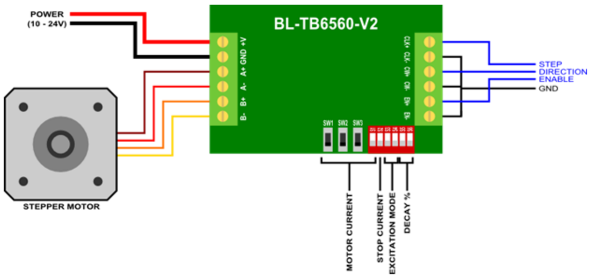


Figure 13. Stepper Motor Driver Circuit

The TB6560 stepper motor driver is a device used to control a stepper motor, which is a type of motor that can rotate in discrete steps. Here is how the TB6560 stepper motor driver operates:
Step 1: Stepper Motor Connection
Connect the stepper motor cables to the appropriate connectors on the TB6560 driver. Be sure to follow the correct connection sequence to prevent damage to the motor or driver.
Step 2: Connect the Power Supply
Connect a suitable (usually DC) power supply to the power supply input terminal on the driver. Check the voltage required by the stepper motor and make sure the power supply is suitable.
Step 3: Set Up Microstepping (Optional)TB6560 drivers generally support multiple microstepping modes. If you want to fine-tune the step motor resolution, you can set the jumper on the driver according to microstepping needs.
Step 4: Set the Motor Rotation Direction
Check the direction markings on the TB6560 driver to determine the desired direction of rotation of the motor. Can adjust the direction by changing the order of the motor wires on the driver connector.
Step 5: Set Pulse and Direction
The TB6560 driver regulates motor movement based on pulses (steps) and direction (direction). Connect the pulse signals from the controller to the PUL+ and PUL- pins of the driver. Additionally, connect the directional signals from the controller to the DIR+ and DIR- pins of the driver.
Step 6: Set Current SettingsAdjust the current setting potentiometer on the driver to adjust the current according to the stepper motor specifications. Make sure the set current does not exceed the maximum current limit of the motor.
Step 7: Set up the Controller (Controller)
Use the ESP32 controller to send the appropriate pulses and directions to the TB6560 driver. Make sure the controller program code is correct and according to the valve opening and closing requirements.
Step 8: Test and Control the Stepper Motor
After all the settings are done, test the stepper motor by running the program code on the controller. Make sure the motor moves in the desired direction and steps.

Table 2. Counting Steps to Drive a Stepper Motor at a Certain Angle

Angle (degree)	Number of Steps of Stepper Motor
1	(Number of steps per lap)/360
5	(Number of steps per lap)/ 72
10	(Number of steps per lap)/ 36
20	(Number of steps per lap) / 24
30	(Number of steps per lap)/ 12
45	(Number of steps per lap)/ 8
60	(Number of steps per lap)/ 6
90	(Number of steps per lap)/ 4

:: IJEEIT ::
(International Journal of Electrical Engineering and Information Technology)
Volume 06 Number 02 September 2023

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

2.3 Software Design

To run the hardware that has been designed, a software design is needed. Software design uses Arduino IDE software with the controller used is ESP32. The first is to initialize, namely the use of the hardware library used, then setting the point in the software to the desired temperature conditions in the cooling process of fresh water flowing on the main engine. Then the blades that have been read are carried out by the Fuzzification process to group the membership functions of the measured temperature, then the reasoning process of the system is carried out which has been carried out by the fuzzification process. Furthermore, the output of the reasoning process will produce a set of control signals that will be used to drive the motor which is coupled to the valve previously defuzzified to produce a control signal from the angle deviation of the valve opening using a motor driver.

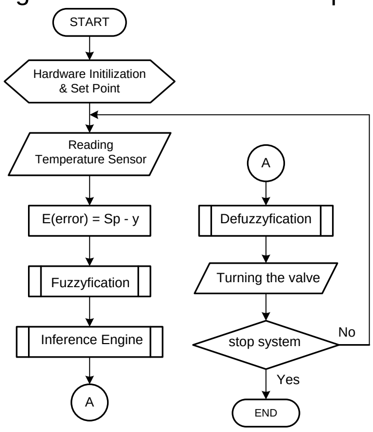


Figure 14. System Flowchart

2.3.1. Reading Temperature Sensor

To find out the address of the DS18B20 sensor connected to the ESP32, you can use the OneWire library and search for sensor addresses automatically. Each sensor has a unique address which will be identified by the program code. Here are the steps: Make sure you have installed the OneWire library for the Arduino IDE (the IDE used for ESP32 programming). then identify 3 ds18b20 temperature sensor addresses by creating c++ code which is uploaded to the ESP32 microcontroller.
After running it will produce as many temperature sensors as installed, in this case a total of 3 sensors are installed, the serial monitor will show the addresses of the 3 sensors in hexadecimal form.

```
sensor1[8] = {0x28,0xEE,0x09,0x81,0xE3,0xAB,0x3C,0x72};
sensor2[8] = {0x28,0x7D,0x7E,0x81,0xE3,0x73,0x3C,0x88};
sensor3[8] = {0x28,0xE3,0xD9,0x81,0xE3,0x33,0x3C,0xD3};
```

The 8 digit hexadecimal address on the DS18B20 sensor is a unique representation of the identity of each sensor in hexadecimal numeral form. In general, this address consists of 8 hexadecimal digits (0-9 and A-F), each digit can have a value from 0 to 15. Each DS18B20 sensor has an address that is different from the others, allowing the system to differentiate each sensor individually. unique. Each digit is represented by a hexadecimal number (0 to 9) or a letter (A to F). The address consists of 8 hexadecimal digits, each separated by a space for easy reading. This address is often used to identify a particular sensor when accessing and reading data from a sensor in a series that has multiple DS18B20 sensors. When the ESP32 or other microcontroller communicates with multiple DS18B20 sensors on a 1-Wire line, these unique addresses allow the device to properly organize and differentiate data from each sensor.

2.3.2. Code For Stepper Motors

Table 2. is based on a stepper motor with a resolution of 200 steps per rotation. To get the number of stepper motor steps for a certain angle, you need to divide the number of steps per stepper motor rotation by the amount of that angle. For example, if a stepper motor has 200 steps per revolution, to go 45 degrees, you would need 200 / 8 = 25 steps.
Make sure you replace the "number of steps per revolution" in the table above with the appropriate value for the stepper motor you are using. Usually, information about the number of steps per rotation can be found in the technical specifications of the stepper motor that you have.

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

2.4 Fuzzy PD Controller Design

The design starts from the fuzzyfication process then proceeds to the inference process with the rules to be determined using the max-min rules, after that the defuzzification process is carried out to obtain a motor control signal that will move the bypass valve open and close in fresh water. The set function that will be used in the fuzzification process is using the triangle rule for errors and delta errors.

$$\mu_A[x] = \begin{cases} 0; & x \le a \text{ atau } x \ge c \\ \frac{x-a}{b-a} & a<x<b \\ \frac{b-x}{b-a} & b<x<c \\ 1; & x=b \end{cases}$$

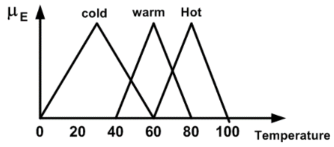


Figure 15. Error Membership Function

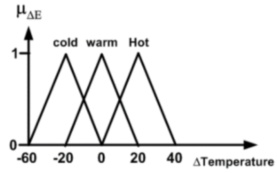


Figure 16. Delta Error Membership Function

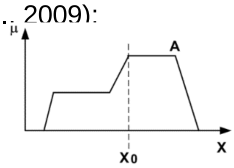
This method is the simplest and most frequently used method for research compared to other methods. Input and output on the mamdani method are in the form of fuzzy sets. The Mamdani method uses the min implication and max aggregation functions so that the Mamdani method is also called the MIN-MAX (min-max inferencing) method.(Abbas et al., 2011) Output n for the rules of the Mamdani method is defined as follows:

$$\mu_B k(y) = max\left[\left[\mu_{A_1^k}(x_i), \mu_{A_2^k}(x_j)\right]\right]_k$$

Tabel 3. Database Inferensi

Δ \ E	Cold	Warm	Hot
Δcold	U_{cold}	U_{cold}	U_{warm}
Δwarm	U_{cold}	U_{warm}	U_{hot}
Δ hot	U_{warm}	U_{hot}	U_{hot}

The defuzzification process is the last process in the fuzzy system. This process is the process of changing the input data that has been included in the fuzzy set to get back its shape (Crisp). In this fuzzy, the defuzzification method used is the centroid or center of area (COA) where the crisp value output is obtained based on the suppression of the decision-making process result curve using the following equation(Wang et al., 2009):



$$x^* = \frac{\sum_{i=1}^{n} x_i \mu(x_i)}{\sum_{i=1}^{n} \mu_B(x_i)}$$

Figure 17. Graph of COA Method Control Signal Value Determination

The Centroid method is also called the Center of Grafity method or the Center of Area (CoA) method. The process of defuzzification in the Centroid method is to take the value of the center point ∗ ( ) of the area in the membership function B.

3. Result And Discussion

After designing the system both hardware and software, the next process is to test each circuit that is made and test the entire system that is built. Starting from testing the address of the temperature sensor used, namely the DS18B20, which is 3 pieces, then proceed with measuring the temperature of each temperature sensor, the results of the temperature sensor are entered into fuzzy logic to test the data set from the fuzzy set through the process of fuzzification, then defuzzification, and finally the process defuzzification. The control signal results are used to drive the stepper motor to rotate the bypass valve. The following results of the simulation design of the ship's main engine cooling system can be seen in Figure 22.

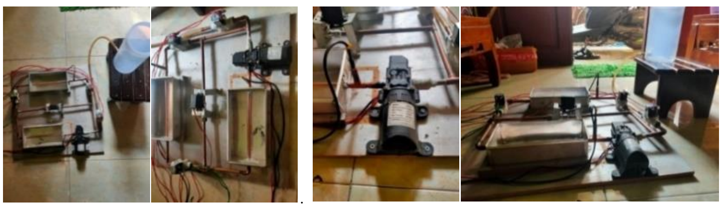


Figure 18. Ship main engine cooling system simulation design

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

3.1. Temperature Sensor Testing

After knowing the respective addresses of the DS18B20 sensors, we then tried to measure the temperature using the addresses of each of these sensors. By requesting a temperature via pin 4 of ESP32, a temperature value will be obtained with the getTempC(sensor) command for each address of the sensor. After the program was run, the three temperature sensors read the air temperature around them, during the experiment it showed that the air temperature around the sensor was ±29 degrees Celsius.

Figure 19. Photo of DS18B20 Sensor Circuit

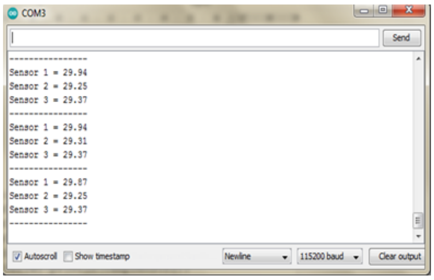


Figure 20. Reading Results of 3 DS18B20 Temperature Sensors

3.1.1. Fuzzy Logic Software Testing

After designing the fuzzy logic algorithm in chapter 2.3.1, next we test the algorithm, which we match between the formula and the experiments carried out. This testing process includes fuzzification process, inference process, and defuzzification process.

3.1.2. Fuzzification Process

This fuzzification process consists of two processes, namely errors and delta errors, in Figure 21. describes the membership set for errors and Figure 28.(Nasution et al., 2011) for the delta error membership set. For example, if the current temperature is 45^{O}C, we will calculate the value of the membership function for the Cold set, the Warm set, and the Hot set.
Calculations for the membership function of the Cold set are as follows:

$$\mu\text{cold} = \frac{60-45}{60-40} = \frac{3}{4} = 0{,}75; \quad \mu\text{warm} = \frac{45-40}{60-40} = \frac{1}{4} = 0{,}25$$

$$\mu\text{hot} = 0 \ \text{ because } \ 45 < 60 \text{ (the boundary of the Hot Set)}$$

The experimental results on the ESP32 microcontroller, which can be seen on the serial monitor, are as follows:

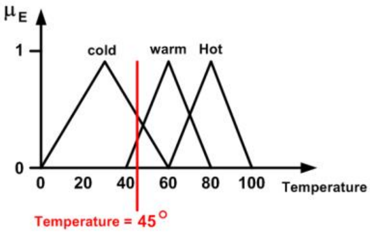


Figure 21. Fuzzy Logic Input Temperature Test of 45^{O}C

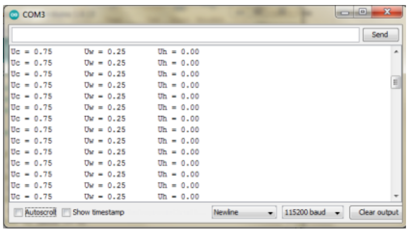


Figure 22. Error Membership Function Test Results

The delta error of a measured temperature is the difference between the current temperature and the previous temperature measured by the sensor and stored in a variable.

Δerror temperature = temperatur(i) – temperature(i-1)

If the previous temperature reads 47^{O}C then there is a difference with the current reading temperature of 45^{O}C of -2^{O}C. The temperature delta graph is as shown in figure 23.
From the graph above if the calculation is carried out with the triangular membership function, the result will be as follows

$$\mu\Delta_{cold} = \frac{0-(-2)}{0-(-20)} = \frac{1}{10} = 0{,}1; \ \mu\Delta_{warm} = \frac{-2-(-20)}{0-(-20)} = \frac{9}{10} = 0{,}9; \ \ \mu\Delta_{hot} = 0 \ \text{ karena } \ -2 < 0 \text{ (batas himpunan } \Delta_{Hot})$$

The experimental results of the Fuzzy logic software for the temperature error delta are shown in figure 23.

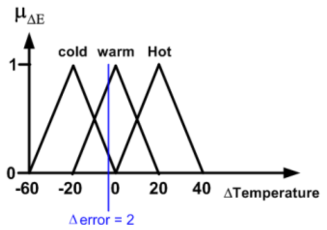


Figure 23. Temperature Error Delta Graph

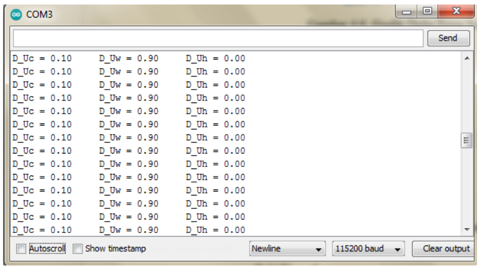


Figure 24. Program Results for Calculating the Delta Error Temperature

Data from the results of the fuzzification process are entered into the inference process according to table 4. and then analyzed with the max-min rule to obtain control signals in fuzzy before deuzyification, namely U_{cold}, U_{warm}, and U_{hot}. If the calculation process is carried out according to the program algorithm, the result will be as follows:Pertama-tama nilai $U_{cold} = 0$; $U_{warm} = 0$; $U_{hot} = 0$;

Rule 1 : $U_{cold} = \max(U_{cold}, \min(E_{cold}, \Delta E_{cold}))$
$= \max(0, \min(0.75, 0.1))$
$= \max(0, 0.1) = 0.1$

Rule 2 : $U_{cold} = \max(U_{cold}, \min(E_{warm}, \Delta E_{cold}))$
$= \max(0.1, \min(0.25, 0.1))$
$= \max(0.1, 0.1) = 0.1$

Rule 3 : $U_{cold} = \max(U_{cold}, \min(E_{cold}, \Delta E_{warm}))$
$= \max(0.1, \min(0.75, 0.9))$
$= \max(0.1, 0.75) = 0.75$

Rule 4 : $U_{warm} = \max(U_{warm}, \min(E_{cold}, \Delta E_{hot}))$
$= \max(0, \min(0.75, 0))$
$= \max(0, 0) = 0$

Rule 5 : $U_{warm} = \max(U_{warm}, \min(E_{warm}, \Delta E_{warm}))$
$= \max(0, \min(0.25, 0.9))$
$= \max(0, 0.25) = 0.25$

Rule 6 : $U_{warm} = \max(U_{warm}, \min(E_{hot}, \Delta E_{cold}))$
$= \max(0.25, \min(0, 0.1))$
$= \max(0.25, 0)= 0.25$

Rule 7 : $U_{hot} = \max(U_{hot}, \min(E_{hot}, \Delta E_{warm}))$
$= \max(0, \min(0, 0.9))$
$= \max(0, 0) = 0$

Rule 8 : $U_{hot} = \max(U_{hot}, \min(E_{warm}, \Delta E_{hot}))$
$= \max(0, \min(0{,}25, 0))$
$= \max(0, 0) = 0$

Rule 9 : $U_{hot} = \max(U_{hot}, \min(E_{hot}, \Delta E_{hot}))$
$= \max(0, \min(0, 0))$
$= \max(0, 0) = 0$

From the calculation process above is generated

$U_{cold} = 0{,}75$; $U_{warm} = 0{,}25$; $U_{hot} = 0$

Tested using a fuzzy logic algorithm, the results can be seen in Figure 25. With three fuzzy control signals that will be mapped in the bypass valve opening process using COA (Center Of Area) rules. If the temperature is too hot then the valve will be closed or the stepper motor is moved to close, if the temperature is cold then the bypass valve will be opened by moving the stepper motor to open, and if the temperature is warm then the valve will be half opened. By making the center variable close equal to 360, open half equal to 8, and open variable equal to 4.

$$U_{kontrol} = \frac{Cmenutup*Uhot+Cseparuh*Uwarm+Cbuka*Ucold}{Uhot+Uarm+Ucold} = \frac{0*360+8*0.25+4*0.75}{0+0.25+0.75} = \frac{5}{1} = 5$$

The value 5 is the number for the step divider in the stepper motor. So the stepper motor is driven by the number of steps per rotation divided by the divisor or in degrees 200/5 = 40^{O}.

The test results with fuzzy logic software produce control signals as can be seen in Figure 26.

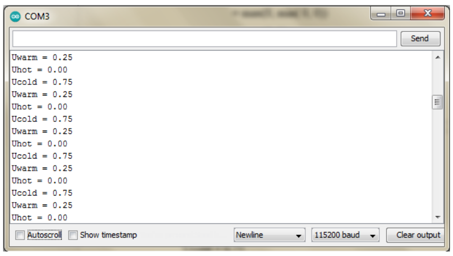


Figure 25. Reasoning Engine Process Results Generate Control Signals

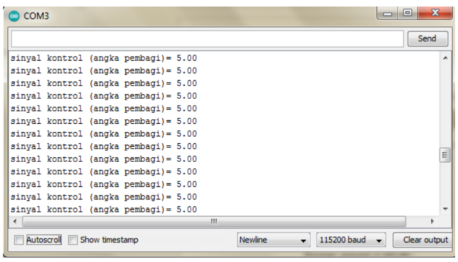


Figure 31. Results of Control Signal Values from the Defuzzification Process

ISSN : 2615-2096 (ONLINE)
ISSN : 2615-2088 (PRINTED)

4. Conclusion

From the results of system design and testing using fuzzy logic, the following conclusions are drawn:

1. The defuzzification process generates a dividing number to determine the angle of rotation of the stepper motor when it opens the bypass valve as a flow of water to cool the main engine in the ship.
2. In this study, the rule that was built for the fuzzy logic process only had 3 membership functions, so that the resulting rotation of the bypass valve opening and closing was still not smooth, to smooth the opening and closing movement of the bypass valve, more membership functions were needed.
3. The tool used is still limited to a prototype by replacing the main engine with a simulation of heating water instead of engine temperature, which is controlled for a maximum temperature of 100o using a water heater.
4. The membership function used for the fuzzification process uses a triangular model because the triangular membership function has a simple and easy-to-understand form. They consist of three parameters: start point, peak point, and end point. This makes it easier to interpret and implement in fuzzy systems.
5. The inference engine process uses the max-min method because it is simple and easy to understand: The "max-min" method is one of the simplest and easiest to understand methods in fuzzy inference. It involves only two stages, namely taking the maximum value from all the rules and then taking the minimum value from that result.
6. The COA method provides results that are easily interpreted physically. The resulting defuzzification value is the centroid value of the large area under the output set curve generated by fuzzy inference.

Reference

A First Course in Fuzzy Logic.pdf. (n.d.).

Abbas, M., Khan, M. S., & Zafar, F. (2011). *Autonomous Room Air Cooler Using Fuzzy Logic Control System*. *2*(5).

Agustianto et al. - 2021—Development of automatic temperature and humidity .pdf. (n.d.).

Cameron, N. (2023). *ESP32 Formats and Communication: Application of Communication Protocols with ESP32 Microcontroller*. Apress. https://doi.org/10.1007/978-1-4842-9376-8

Jomaa, M., Abbes, M., Tadeo, F., & Mami, A. (2019). Greenhouse Modeling, Validation and Climate Control based on Fuzzy Logic. *Engineering, Technology & Applied Science Research*, *9*(4), 4405–4410. https://doi.org/10.48084/etasr.2871

Kaswidjanti, W. (2014). Implementasi Fuzzy Inference System Metode Tsukamoto Pada Pengambilan Keputusan Pemberian Kredit Pemilikan Rumah. *Telematika*, *10*(2). https://doi.org/10.31315/telematika.v10i2.281

Nasution, H., Jamaluddin, H., & Mohd. Syeriff, J. (2011). Energy Analysis for Air Conditioning System Using Fuzzy Logic Controller. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, *9*(1), 139. https://doi.org/10.12928/telkomnika.v9i1.680

Passino, K. M., & Yurkovich, S. (1998). *Fuzzy control*. Addison-Wesley.

Revathi, S., & Sivakumaran, N. (2016). Fuzzy Based Temperature Control of Greenhouse. *IFAC-PapersOnLine*, *49*(1), 549–554. https://doi.org/10.1016/j.ifacol.2016.03.112

Ross, T. J. (2010). *Fuzzy logic with engineering applications* (3rd ed). John Wiley.

Sistem Pendingin Mesin Induk Kapal | PDF. (n.d.). Retrieved 13 September 2023, from https://www.scribd.com/document/372605664/sIstem-Pendingin-Mesin-Induk-Kapal

Wang, T.-M., Liao, I.-J., Liao, J.-C., Suen, T.-W., & Lee, W.-T. (2009). An Intelligent Fuzzy Controller for Air-Condition with Zigbee Sensors. *International Journal on Smart Sensing and Intelligent Systems*, *2*(4), 636–652. https://doi.org/10.21307/ijssis-2017-372