IoT Based Temperature Control System Using Node MCU ESP 8266

Rifqi Firmansyah[1,*], Muhamad Yusuf[2], Pressa P. Surya Saputra[3],

Muhammad Eko Prasetyo[1], Fahmi Mahardi Mochtar[1], Fandik Agung Kurniawan[1]

[1] *Department of Electrical Engineering Universitas Negeri Surabaya, Surabaya, Indonesia*
[2] *Department of Electrical Engineering Politeknik Negeri Cilacap, Cilacap, Indonesia*
[3] *Department of Electrical Engineering Universitas Muhammadiyah Gresik, Gresik, Indonesia*
[*] *Corresponding author. Email: rifqifirmansyah@unesa.ac.id*

ABSTRACT

Water heaters are currently needed especially in the industrial world. This heater is used to maintain the temperature of the liquid so that the products produced are appropriate with standard specifications. Along with technological developments in recent years, circulating water heaters still use conventional technology. This technology is relatively simple because it only applies the on/off system. This is not effective because the user has not been able to set the desired water temperature automatically and cannot be adjusted remotely. The purpose of this research was to overcome conventional technology systems by automatically updating the system based on IoT systems. The data update was useful for stabilizing responses at the desired setpoint of users and could be monitored remotely. This automated technology assistance was more effective and easier in terms of using water heaters in everyday life. To support this research in the manufacture of modern or automatic water heaters the system employs a PI controller system that uses analytical tuning methods. This analytical method was effective for finding PI control parameters in the form of K_p and K_i. Retrieval of supporting data in this study used the Node MCU ESP8266 microcontroller, tube or plant container, heater, DS18B20 temperature sensor, Blink application as a GUI in a water heater monitor. The results of the system response used analytical methods are delay time= 10.94, Rise time (5% -95%) = 9.97, Rise time (10% -90%) = 8.81, Settling time (2%) = 16.04, Settling time (5 %) = 15.04. The system showed that the comparison of the results of the temperature stability of modern or automatic water heaters was more stable because it had response value equivalent to the setpoint.

Keywords: *IoT, Water Heater, PI Controller, and ESP8266*

1. INTRODUCTION

As known to the bath is one of the most common human needs to do. Aside from cleansing the body of dirt, bathing can also be a relaxation to stretch the muscles of a stiff body. What is needed to be able to relax the body in a bath is to use additional warm water. Warm water is often cooked first in a container or pan. When it's boiling, pour the boiling water into the tub which has been filled with cold water. In the form of this process, it is really ineffective and requires more energy. Besides that, we are not able to adjust the heat level according to our needs.

Now there are many water heater devices as a solution for the problems being sold on the market. The basic principle of the devices uses two containers where one of the containers is used as a water heater. But these devices still have drawbacks, where the device uses the on/off (conventional) method as well as without a monitoring system that is able to increase the efficiency of time for users. Therefore, the researcher has developed an automatic water heater device which is able to control the desired temperature requirements. The device is also equipped with a monitoring-based process control system.

In the control system process, there are often linear systems that are difficult to control and will affect the accuracy of the process results. In controls, PID controllers are often combined in various combinations, such as PI, PD, and PID controls themselves. This combination is more widely used because of the level of simplicity in the linear system. The disadvantage of the PID controller combination is a lack of effectiveness in the value of nonlinear systems. In addition, PID combination controllers can be used effectively in linear systems [1].

Industrial temperature control systems are also needed to provide products that meet standard specifications. Therefore, to maintain the product in accordance with standard specifications, it is necessary to accelerate or increase the desired temperature within a certain time without producing overshoot [2].

In this study, the system employs controller PI (proportional-integral). The PI controller is susceptible to change in system parameters and to obtain zero overshoot. PI control system is very simple but can still be used effectively in a linear system[3][4][5]. There are several supporting methods for the system such as Ziegler-Nichols, Cohen-Coon, CHR, analytics, even trial and error adjustment methods.

In the PI combination controller, the characterization is divided into 3 parameters, namely Kp Gain, time constant T constant and time delay L. This characterization is a rule for PI controllers to get simple response steps. Therefore, these parameters will support the development of simple settings for PI controllers that depend on delay time [6]. The current study was carried out using a combination of PI controllers with analytical methods to control the system that works on water heaters. In preparing the system this research was supported by Arduino IDE which was used to read the value of the response and The DS 18B20 sensor used to support the reading of temperature value of the water heater.

2. METHOD

2.1. Modeling of Temperature Control System

2.1.1. Water Heater

The basic principle of this study used 3 different tanks. The first tank used a container with a hot temperature which would be adjusted by the user, the second tank consisted of water of room temperature and the last tank was for mixed water. The first tank utilized heater as a water heater element using a DS18B20 temperature sensor to measure the temperature output, then adjust the output of water on the second tank using a control valve as an actuator [7][8].

During the process of heating water by adjusting the required temperature, the researcher utilized the ESP module. The module functions as a device for monitoring and controlling the temperature of the tank remotely through the internet. This device would be expected to be able to monitor and to control the performance of process control systems using a laptop or smartphone [9][10][11].

If assumed in temperature, this heating system could be represented by linear differential equations such as in (1)

$$Q_h = Q_c + Q_o + Q_i + Q_l \tag{1}$$

Where,

Q_h = heat flow provided by the heating element.

Q_c = heat flow into the water tank.

Q_o = heat flow is a loss by hot water in the tank.

Q_i = heat flow carried by the incoming cold water to the tank.

Q_l = heat flow through heat insulation.

The heat flow into the water tank *Qc* given in (2),

$$Q_c = C\frac{\partial\tau_t}{\partial t} \tag{2}$$

Where,

C = heat capacity of water in the tank.

τ_t= water temperature in the tank.

Equation (3) is heat flow is a loss by hot water in the tank

$$Q_o = V.H.\tau_t \tag{3}$$

Where,

V = flow of water from the tank

H = Specific heat of water

Equation (4) is heat flow is a loss by hot water in the tank.

$$Q_i = V.H.\tau_i \tag{4}$$

Where

τ_i = Water temperature into the tank

And heat flow through heat insulation is described in (5)

$$Q_l = \frac{\tau_t - \tau_e}{R} \tag{5}$$

Where,

τ_e = temperature of air that surrounds the tank

R = thermal resistance from heat insulation.

From entering the (2), (3), (4), and (5), we get (6)

$$Q_h = C\frac{\partial\tau_t}{\partial t} + V.H.\tau_t + V.H.\tau_t + \frac{\tau_t - \tau_e}{R} \tag{6}$$

2.1.2. *PnID and Diagram Block of Temperature Control System*

In this paper, we used a closed-loop control system. There were three tanks used in the system as displayed in Figure 1. first tank functioned as increasing water temperature, second tank for normal temperature and the last tank functioned as disturbance.

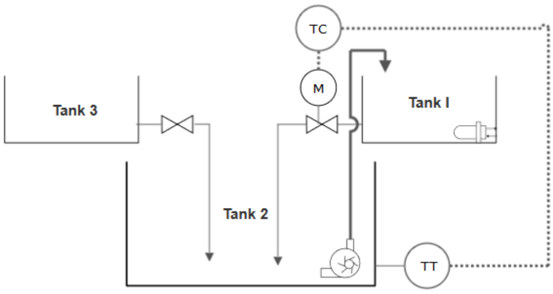

Figure 1 PnID of the system

Closed-loop control systems as represented in Figure 2 had input device, processing device, and output device.

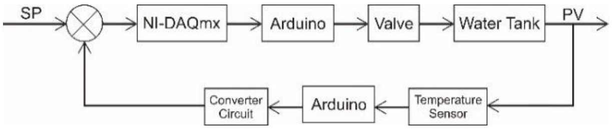

Figure 2 Block Diagram System

From Figure 2, the brains of all systems were in Node MCU ESP8266. besides it was used to monitor the system response directly, Node MCU ESP8266 was also used to control the temperature of the system through the internet so that the user could monitor and control everywhere and every time they wanted.

2.1.3. *PnID and Diagram Block of Temperature Control System*

The PI controller as shown in Figure 3 was a feedback mechanism as control controls used in a system control industry. Control on the PID controller was divided into P control (proportional) and controller I (integral).

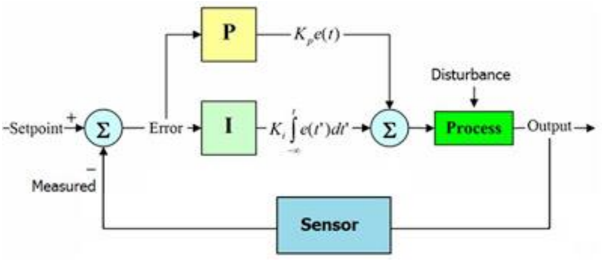

Figure 3 Diagram Block of PI Controller

PI Controller was designed using program C in Arduino IDE software, whereas the center of setting the mixture temperature on the main tank and setting the servo rotation angle on the actuator. Then PI Controller set the parameter values Kp, Ki.

2.1.4. *Temperature Sensors*

As shown in Figure 4 DS18B20 sensor was a temperature sensor that had waterproof properties. This sensor is capable of measuring temperatures up to 125℃, accuracy of 9 to 12-bit, with a value of the level of accuracy of ± 0.5℃. on the DS18B20 sensor has a unique code of 64-Bit embedded in each chip, allowing the use of sensors via just one cable. In reading the DS18B20 sensor output value in the graphical form, use the Matlab GUI which is connected to the Node MCU ESP 8266.

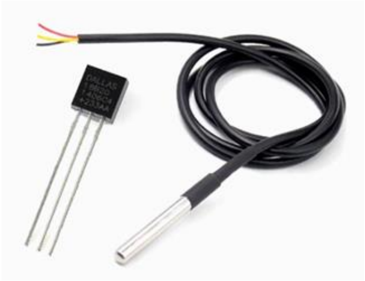

Figure 4 Temperature Sensor DS18B20

For temperature leveling, a DS18B20 temperature sensor is issued with the Node MCU ESP 8266 and is assembled using an additional 4.7 KΩ resistor as shown in Figure 5. The sensor reading system would work to mix mixed water in the main tank. After reading, the output was read inside the GUI Blynk app.

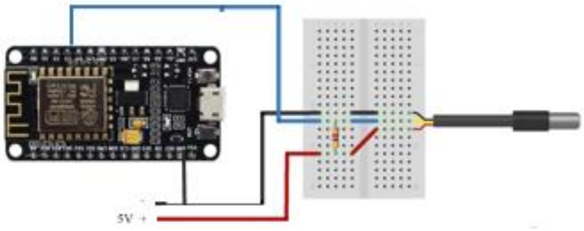

Figure 5 Wiring of Temperature Sensors

2.1.5. *Servo Motor*

A servo motor as pointed in Figure 6 is the main driver or an actuator in this water heater. The servo motor applied a feedback control system to make it easier to adjust the position of the rotation angle at the motor output. For servo motor setting was limited only up to 90˚. With the help of PWM (Pulse Wide Modulation) could determine the rotation angle on the servo motor shaft.

Figure 6 Servo Motor

For designing a servo motor by connecting to Node MCU ESP8266 it was quite simple by connecting the cables to the servo motor. For the yellow cable, connect to D4 as data. Then connect the red cable to the 5V supply and the black cable to ground. Figure 7 is a wiring diagram of a servo motor for the system.

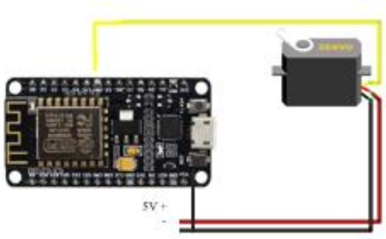

Figure 7 Design of Servo Motor

2.1.6. Blynk App

Blynk app as represented in Figure 8 had been designed for the Internet of Things. It could control hardware remotely, display sensor data, store data, visualize it, and do many other things. It was a digital dashboard where you could build graphic interface for your project by simply dragging and dropping widgets.

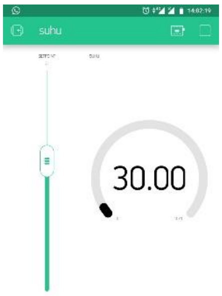

Figure 8 GUI Blynk App

2.1.7. Node MCU ESP 8266

Node MCU is an open-source IoT platform. Consisting of hardware in the form of a System On Chip ESP8266 from ESP8266 made by Espressif System as well as firmware used using the Lua scripting programming language as pointed out in Figure 9. Node MCU itself refers to the firmware used in the hardware development kit Node MCU just like Arduino, but the difference is the MCU Node ESP 8266 module is equipped with WIFI, so it can be used in internet-based or IoT hardware requirements [12].

Figure 9 Design of Servo Motor

Node MCU provides access to GPIO (General Purpose Input / Output) and pins mapping tables are part of the API documentation. The following table explains ESP8266 pins.

Table 1. Number Pin ESP 8266 Module

I/O Index	PIN	I/O Index	PIN
0	GPIO16	6	GPIO12
1	GPIO5	7	GPIO13
2	GPIO4	8	GPIO15
3	GPIO0	9	GPIO3
4	GPIO2	10	GPIO1
5	GPIO14	11	GPIO9

2.2. Control Method

2.2.1. PI Controller

The method used in this research employs the PI Controller with an analytical approach which was the PI Controller had proportional and integral properties to signal errors. In designing this PI Controller we applied the first-order plant. It was expected that proportional Gain Kp and Ti integral time could make the system achieving the desired response [13] [14].

The first order plant equation is obtained (7)

$$\frac{C(S)}{R(S)} = \frac{K}{\tau s+1} \tag{7}$$

For the equation from the PI Controller is acquired (8)

$$\frac{U(S)}{E(S)} = K_p\left(1+\frac{1}{\tau_i s}\right) \tag{8}$$

From the (7) and (8), we get the equation of the transfer function (9)

$$\frac{C(S)}{R(S)} = \frac{K_p\left(1+\frac{1}{\tau_i s}\right)\frac{K}{\tau s+1}}{1+K_p\left(1+\frac{1}{\tau_i s}\right)\frac{K}{\tau s+1}} \tag{9}$$

With the value of Ti = T, first-order system design obtained results with transfer function values

$$\tau^* = \frac{\tau_i}{Kp.K} \tag{10}$$

$$K^* = 1 \tag{11}$$

3. RESULT AND DISCUSSION

This section discussed system modeling, PI controller design, and implementation and result.

3.1. Modeling System

System modeling was carried out using the first-order system characteristic approach with the required parameters k and Ti. This method was intended to get the transfer model or function from the system. The transfer function was used as a reference to design the parameters of the PI controller and simulation to determine the system response. The following is the system response with a 2.3V setpoint shown in Figure 10.

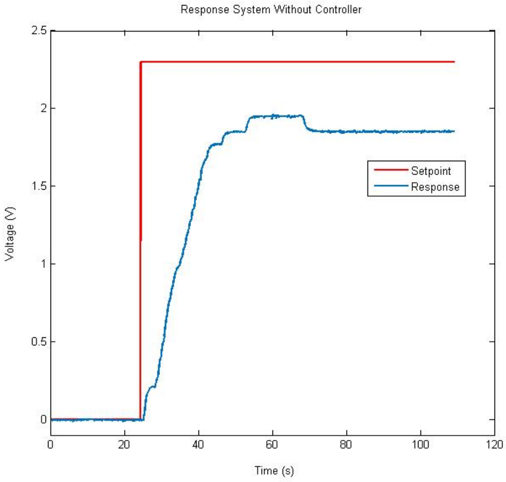

Figure 10 Response System Without a Controller

Based on Figure 10, and using equations (10), (11), the parameters of the order-1 system characteristics are K = 1.7, and τ_i = 0.05.

3.2. PI Controller Design

The PI controller design was done to get the parameters of the PI controller, Kp, and Ki. In the previous discussion, it was determined that the transfer function with setpoint 2.3 V was used as a reference to determine the parameters of the PI controller. Based on the PI controller with a value of Kp = 1.7 and τ_i = 0.05., the results of the system response were not maximal. To overcome this problem, we changed the values of Kp and Ti in the hope that the responses obtained would be better. Changes were made by trial and error, but the changes remain based on the values of Kp and τ_iobtained from the above equation. After conducting experiments and improvements, the PI parameters were changed to Kp = 1.8 and τ_i = 0.01. in our opinion, this parameter when applied to the resulting system is better than before.

3.3. Implementation and Result

After getting the Kp and Ki values, the next step was to implement the PI controller on the heat exchanger. The heat exchanger was tested according to a fixed setpoint, a changed setpoint and with a load disturbance.

3.2.1. Fixed Setpoint

For system testing, the setpoint used was 31°C. The response from the system could be shown in Figure 11.

After adding the PI controller, the system response would change. To find out the changes, the parameters of the system response were calculated based on the image above, which was at setpoint 31°C. The parameters to be calculated were τd (Time Delay), τr (Rise Time), τs (Settling Time) with a value of τ = 12.47 seconds. Real plant response parameters with a setpoint 31°C using a PI controller are shown in Table 2.

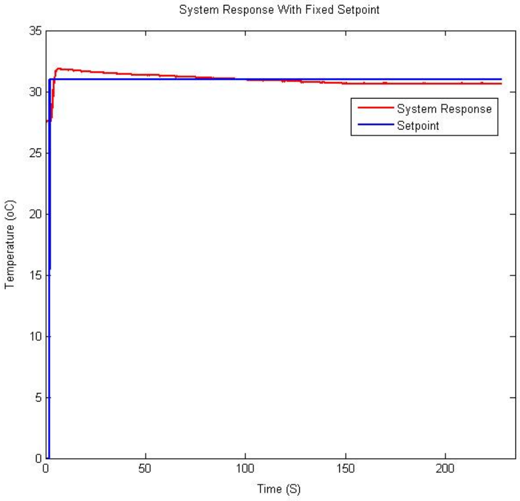

Figure 11 Response System with Fixed Setpoint

Table 2. Response System with Fixed Setpoint

Setpoint 2,3V	T	td	Rise Time		Settling Time	
			Tr (5%-95%)	tr (10%-90%)	ts 5%	Ts 2%
Analitic	12.47	10.94	9.97	8.81	15.44	16.04

3.2.2. Setpoint Changed

Tests were carried out by giving varied setpoints of 33°C and 35°C. Real system response with varied setpoints are shown in Figure 12.

3.2.3. Load Disturbance

To test the robustness of the system was carried out by adding a PI controller. The loading process was carried out by adding water with a low temperature to the mixing tank. Real system responses with load disturbance are shown in Figure 13. Based on Figure 13, the response parameters τd (Time Delay), τr (Rise Time), τs (Settling Time) can be calculated. Real plant response parameters with a 31ºC setpoint using a PI controller are shown in Table 3.

Table 3. Response System with Load Disturbance

Setpoint 2,3V	T	td	Rise Time		Settling Time	
			Tr (5%-95%)	tr (10%-90%)	ts 5%	Ts 2%
Analitic	12.81	11.61	14.8	12.1	19.81	21.61

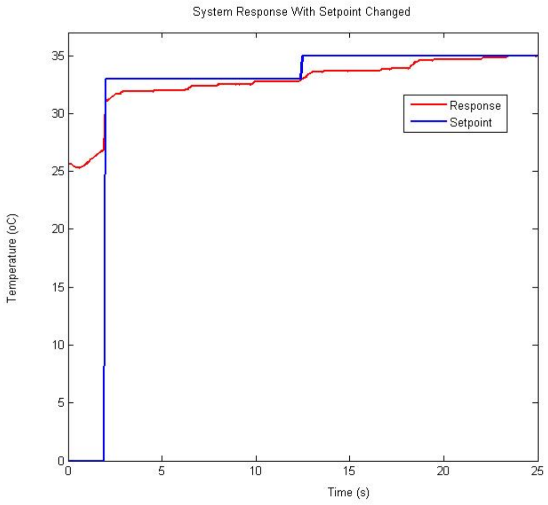

Figure 12 System Response with Setpoint Changed

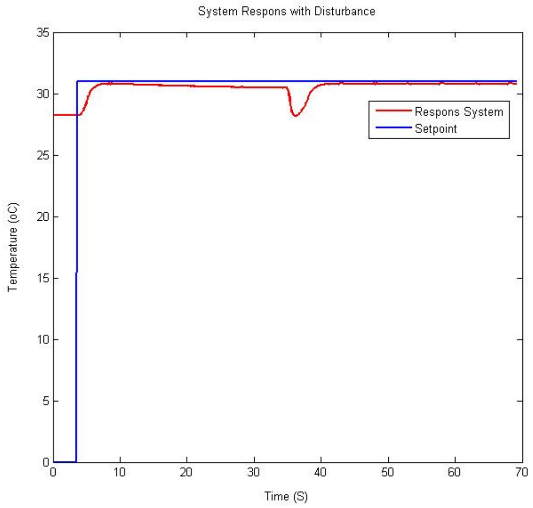

Figure 13 System Response with Load Disturbance

4. CONCLUSION

Based on the experiments that have been carried out, it could be concluded that testing carried out using Arduino IDE software was able to process real system response data with real-time. With the first-order system response characteristic approach, PI controller parameters were obtained, namely Kp = 1.8, and Ki = 0.01. PI controller designed to control the temperature response in the tank managed to improve the system response before adding a PI controller. The response after adding the PI controller was able to reach the setpoint value.

REFERENCES

[1] T.P.Mote, Dr.S.D.Lokhande, “Temperature Control System Using ANFIS”, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Vol.2, Issue-1, March 2012.

[2] Bakhtiar I.Saeed, “Zero Overshoot and fast Transient Response using a Fuzzy Logic Controler,” Procd. Of the Inter. Conf. On Autom. And Computing, pp. 116-120, 2011.

[3] Rubiyah Yusof, Sigeru Omatu, Marzuki Khalid, “Application of SelfTuning PI(PID) Controller to Temperature Control Syste,” Procd. Of the Inter. Conf. On third IEEE Conf., pp. 1181-1186, Vol. 2, 1994.

[4] T.Kavita, B.Naga parvathi, M.Lavanya, M.Arivalagan, “Temperature Control Water bath system using PID Controller”, International Journal of Applied Engineering Research, ISSN 0973-4562 Vol. 10 No.4, 2015.

[5] Om Prakash Verma, Rajesh Singla, Rajesh Kumar, “Intelligent Temperature Controller for WaterBath System”, International Journal of Electrical and Computer Engineering Vol.6, No:9, 2012.

[6] Evaristo Kahoraho and Juan J. Gude, “New Tuning rules for PI and fractional PI controllers”, ESIDE, Vol. 6. 2014

[7] Om Prakash Verma, Himanshu Gupta,” Fuzzy Logic Based Water Bath Temperature Control System”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 4, April 2012 ISSN: 2277 128X.

[8] PS Haranath, “A SIMPLE CONSTRUCTION OF WATER BATHS”, Indian journal of physiology and pharmacology · November 1963.

[9] Rupert Gouws, Estie le Roux, “Efficiency and cost analysis of a designed in-line water heating system compared to a conventional water heating system in

South Africa”, Journal of Energy in Southern Africa ,Vol 23 No 3, August 2012.

[10] H. Thybo, L. Larsen, “Control of a water-based floor heating system,” IEEE Int. Conf. on Control Applications (CCA), Singapore, Oct. 1-3, 2007, pp.: 288-294.

[11] L. Jinhai, F. Lide, C. Suosheng, Kong Xiangjie, “Analysis of dynamic model of heating system after step change of eater supply temperature,” 10th Int. Conf. on Control, Automation, Robotics and Vision, (ICARCV), Hanoi, Dec. 17-20, 2008, pp.: 1924-1928.

[12] Firmansyah R, Widodo A, Romadhon AD, Hudha MS, Saputra PP, Lestari NA. “The prototype of infant incubator monitoring system based on the internet of things using NodeMCU ESP8266”. InJournal of Physics: Conference Series 2019 Feb (Vol. 1171, No. 1, p. 012015). IOP Publishing.

[13] Ogata, Katsuhiko, “Modern Control Engineering,” fifth edition. Canada.

[14] Ogata, Katsuhiko, “Discrete Control System,” university of Minnesota.