

**SISTEM *MONITORING* SUHU AIR PENDINGIN DAN KAPASITAS
BAHAN BAKAR SOLAR DALAM TANGKI PEMBANGKIT LISTRIK
TENAGA DIESEL (PLTD) BERBASIS ESP-NOW DI PULAU PANGKIL**



SKRIPSI

Diajukan Untuk Memenuhi Persyaratan Mencapai Derajat
Sarjana Teknik (S.T)

Oleh

Afdal Hadi

180120201001

**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK DAN TEKNOLOGI KEMARITIMAN
UNIVERSITAS MARITIM RAJA ALI HAJI
TANJUNGPINANG**

2023

**SISTEM *MONITORING* SUHU AIR PENDINGIN DAN KAPASITAS
BAHAN BAKAR SOLAR DALAM TANGKI PEMBANGKIT LISTRIK
TENAGA DIESEL (PLTD) BERBASIS ESP-NOW DI PULAU PANGKIL**



SKRIPSI

Diajukan Untuk Memenuhi Persyaratan Mencapai Derajat
Sarjana Teknik (S.T)

Oleh

Afdal Hadi

180120201001

Telah Mengetahui dan Disetujui Oleh :

Pembimbing I,

Ir. Sapt Nugraha, S.T., M.Eng
NIP. 198904132015041005

Pembimbing II,

Ahmad Syafiq S.T., M.Si
NIDK. 8938730022

HALAMAN PENGESAHAN

Judul : Sistem Monitoring Suhu Air Pendingin dan Kapasitas Bahan Bakar Dalam Tangki Pembangkit Listrik Tenaga Diesel (PLTD) Berbasis ESP-NOW di Pulau Pangkil

Nama : Afdal Hadi



Nim : 180120201001

Program Studi : Teknik Elektro




Telah dipertahankan di depan Dosen Penguji dan dinyatakan lulus

Pada tanggal 06 Juli 2023

Susunan Tim Pembimbing

Pembimbing : 1. Ir. Sapta Nugraha, S.T., M.Eng ()
2. Ahmad Syafiq S.T., M.Si ()

Susunan Tim Penguji

Penguji : 1. Hollanda Arief Kusuma, S.IK., M.Si ()
2. Muhd Ridho Baihaque, S.T., M.Sc ()
3. Muhamad Mujahidin, S.T., M.T ()

Tanjungpinang, 02 Agustus 2023

Universitas Maritim Raja Ali Haji

Fakultas Teknik dan Teknologi Kemaritiman

Dekan,



Ir. Sapta Nugraha, S.T., M. Eng
NIP. 198904132015041005

PERNYATAAN ORISINALITAS

Saya mahasiswa yang bertanda tangan di bawah ini:

Nama : Afdal Hadi

Nim : 180120201001

Tempat /Tanggal Lahir : Lhokseumawe/ 02 Juni 2000

Menyatakan dengan sebenarnya bahwa skripsi yang berjudul “Sistem Monitoring Suhu Air Pendingin dan Kapasitas Bahan Bakar Dalam Tangki Pembangkit Listrik Tenaga Diesel (PLTD) Berbasis ESP-NOW di Pulau Pangkil “ ini merupakan hasil karya sendiri dan merupakan bukan duplikasi, serta tidak mengutip sebagian atau seluruhnya karya orang lain, kecuali saya nukilkan dan pada setiap penukilan saya sebutkan sumber-sumber nya sesuai dengan batasan dan tata cara pengutipan.

Apabila dikemudian hari terbukti pernyataan saya tidak benar dan melanggar peraturan yang berlaku dalam karya tulis dan hak intelektual, maka saya bersedia ijazah yang telah saya terima untuk dikembalikan ke Universitas Maritim Raja Ali Haji dan menerima sanksi lainnya dengan peraturan yang berlaku.

Tanjungpinang, 08 Juni 2023

Yang membuat pernyataan



fdal Hadi

180120201001

MOTO

“Hidup yang tidak dipertaruhkan
Tidak pernah dimenangkan”

“Dan carilah pada apa yang telah dianugerahkan kepadamu
(kebahagian) negeri akhirat dan janganlah kamu melupakan bagianmu
Dari (kenikmatan) duniawi.”(Q.S Al-Qashas: 77)



HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

“Dengan menyebut nama Allah yang Maha Pengasih lagi Maha Penyayang”

Alhamdulillahirobbil’alamin atas segala kemudahan, limpahan rahmat dan ridho

Nya yang telah engkau limpahkan selama ini.

Saya persembahkan karya sederhana ini kepada orang yang sangat saya sayangi:

1. Kedua orang tua saya tercinta, Karimuddin dan Hafni

Terima kasih atas apa yang ayah dan emak berikan kepada saya. Jujur, itu semua tidak dapat anak mu balas mak dan ayah, terimakasih mak dan ayah berkat doa dan dukungan dari mak dan ayah saya sudah menyelesaikan apa yang emak dan ayah mintak. Terimakasih untuk semuanya.

2. Teruntuk Tunangan saya, Novita Agusyuliana

Terimakasih ya sayang atas support nya, walaupun sukak di marahin tapi ngak pernah berhenti mengingat saya untuk cepat menyelesaikan tanggung jawab untuk orang tua terimakasih ya sayang.

3. Kelima adik saya, Khairul Walidin, Waliyudin, M. Alfatani, M. Alfaizin dan Aisyah Humairah

Terima kasih telah memberikan dukungan dan motivasi sehingga skripsi ini bisa selesai.

4. Persatuan Masyarakat Aceh TPI-Bintan

Terima kasih untuk Bapak/Ibuk, kalian sudah menjadi keluarga kecil dalam perantauan ini, terimakasih juga atas support nya baik finansial atau materialnya. Terimakasih untuk kalian semuanya.

5. Keluarga saya dalam segala hal, Tuah Jantan

Terimakasih buat kalian semuanya berkat bantuan kalian saya selesai terimakasih atas semuanya

6. Teman-teman Teknik Elektro 2018

Terima kasih untuk kalian semuanya teman-teman Elektro 18. Kalian luar biasa , sukses untuk kalian semuanya

7. Seluruh Mahasiswa Elektro

Terimakasih untuk kalian semuanya, saya mohon maaf karna tidak dapat menyebut satu per-satu nama kalian intinya saya sangat berterima kasih kepada kalian semuanya, suksen selalu. Elektro..Jaya..Jaya..Jaya.

KATA PENGANTAR

Assalamualaikum warahmatullahi wabarakatuh

Alhamdulillah, puji dan syukur penulis haturkan kehadiran Allah SWT, karena berkat rahmat dan hidayah-Nya penulis dapat menyelesaikan skripsi dengan judul “Sistem Monitoring Suhu Air Pendingin dan Kapasitas Bahan Bakar Dalam Tangki Pembangkit Listrik Tenaga Diesel (PLTD) Berbasis ESP-NOW di Pulau Pangkil”. Skripsi ini disusun untuk memenuhi salah satu syarat penulis dalam memperoleh gelar Sarjana Teknik (S.T.) pada program S1 Teknik Elektro Fakultas Teknik Universitas Maritim Raja Ali Haji.

Penulis menyadari sekali bahwa penulisan skripsi ini tidak akan terselesaikan tanpa adanya dorongan, sokongan, bantuan dan bimbingan dari berbagai pihak. Oleh karena itu, penulis pada kesempatan ini ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada yang terhormat :

1. Bapak Ir. Sapta Nugraha, S.T., M.Eng. selaku Dekan Fakultas Teknik dan Teknologi Kemaritiman sekaligus Dosen Pembimbing I yang telah memberikan ilmunya, menyediakan waktu, tenaga, pikiran dan bantuan secara langsung dan tidak langsung selama penyusunan skripsi ini.
2. Bapak Ahmad Syafiq S.T., M.Si selaku pembimbing II yang telah memberikan ilmunya, menyediakan waktu, tenaga, pikiran dan bantuan secara langsung dan tidak langsung selama penyusunan skripsi ini.
3. Bapak Tonny Suhendra, S.T., M.Cs. selaku Ketua Jurusan Teknik Elektro sekaligus dosen pembimbing akademik yang juga telah memberikan ilmunya,

menyediakan waktu, tenaga, pikiran dan bantuan secara langsung dalam menyelesaikan skripsi ini.

4. Seluruh Dosen Fakultas Teknik dan Teknologi Kemaritiman Universitas Maritim Raja Ali Haji khususnya Bapak Rozeff Pramana, S.T., M.T., Bapak Ibnu Kahfi Bachtiar, S.T., M.Sc., Bapak Anton Hekso Yuniyanto, S.T., M.Si., Bapak Hollanda Arief Kusuma, S.IK., M.Si., Bapak Muhammad Mujahidin, S.T., M.T., Ibu Rusfa, S.T., M.T., dan Ibu Septia Refly, S.Pd., MSi yang telah banyak memberikan ilmu melalui perkuliahan dan turut serta membangun jurusan Teknik Elektro.
4. Semua keluarga, sahabat, teman dan berbagai pihak yang telah memberikan sokongan dan mendoakan yang terbaik untuk penulis yang tidak dapat disebutkan satu-persatu.

Penulis sangat mengharapkan dan menghargai kritikan maupun bimbingan yang bermanfaat bagi kesempurnaan skripsi ini. Akhirnya, penulis berharap semoga skripsi ini dapat bermanfaat bagi semua pihak pada umumnya. Akhir kata penulis ucapkan terima kasih.

Wassalamualaikum warahmatullahi wabarakatuh.

Tanjungpinang, 21 Juni 2023



Afdal Hadi

DAFTAR ISI

HALAMAN PERSETUJUAN.....	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PERSEMBAHAN.....	v
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xiii
DAFTAR LAMPIRAN.....	xiv
I. PENDAHULUAN.....	1
A. Latar Belakang	1
B. Rumusan Masalah.....	2
C. Batasan Masalah.....	2
D. Tujuan Penelitian	3
E. Manfaat	3
II. TINJAUAN PUSTAKA.....	4
A. Kajian Terdahulu.....	4
B. Landasan Teori.....	5
1. Pembangkit Listrik Tenaga Diesel (PLTD).....	5
2. Radiator	6
3. Suhu.....	6
4. Tangki Bahan Bakar.....	6
5. Sensor Suhu	6
6. Sensor Ultrasonik	7
7. <i>Liquid Crystal Display</i> (LCD).....	8
8. ESP 32	9
9. Modul SD Card	10
10. RTC (<i>Real time clock</i>)	10
11. Arduino IDE	11

12. <i>Error</i> dan Akurasi.....	11
III. METODOLOGI PENELITIAN.....	13
A. Waktu dan Lokasi Penelitian	13
B. Alat dan Bahan Penelitian.....	14
C. Prosedur Penelitian.....	15
D. Perancangan Sistem	16
E. Pengolahan Data dan Analisis.....	19
IV. HASIL DAN PEMBAHASAN.....	23
A. Pengujian Sistem.....	23
1. Pengujian ESP-NOW	23
2. Pengujian LCD 20x4	27
3. Pengujian Micro SD Card	28
4. Pengujian RTC DS3231	29
5. Pengujian Sensor	30
B. Pengembangan <i>Firmware</i>	45
C. Analisis dan Pembahasan.....	53
V. PENUTUP.....	59
DAFTAR PUSTAKA	60
LAMPIRAN.....	63
BIODATA.....	112

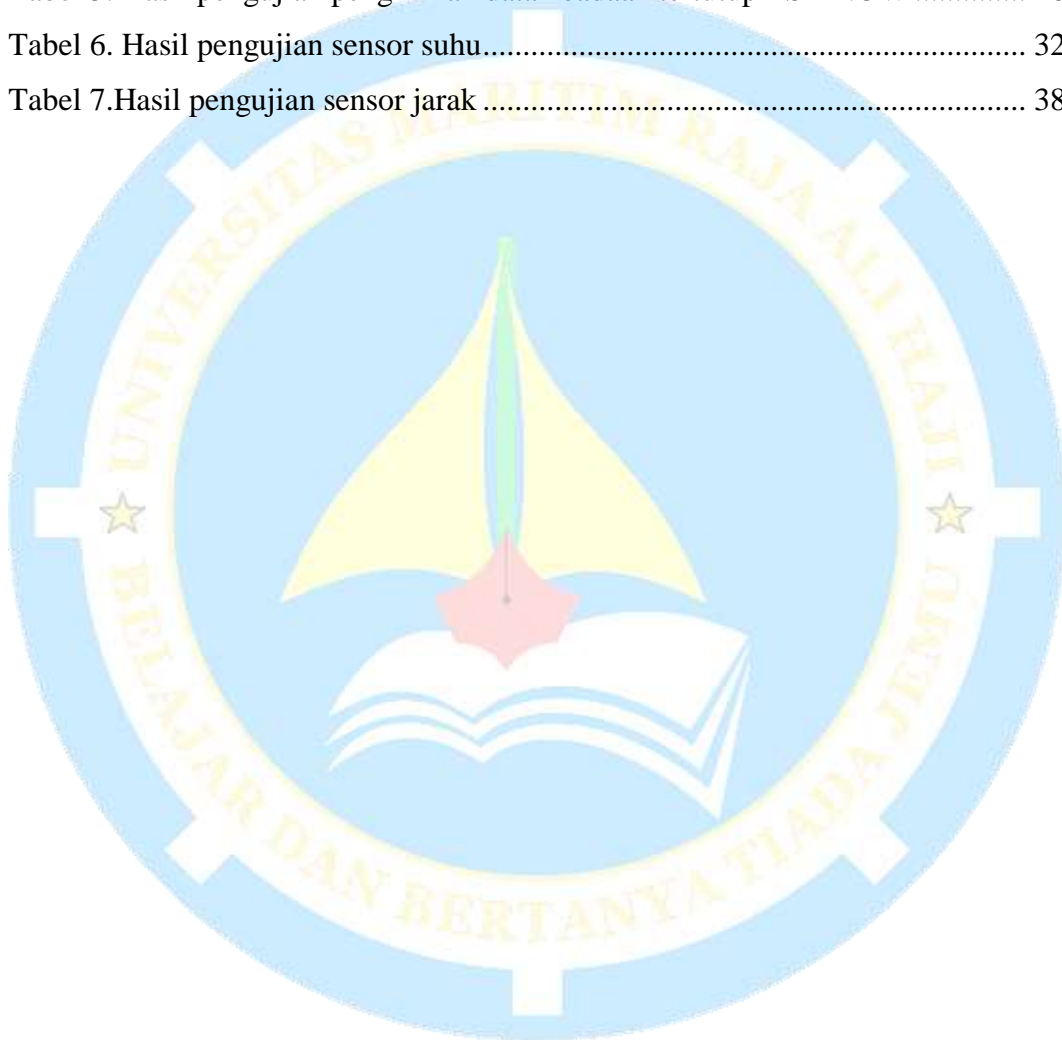
DAFTAR GAMBAR

Gambar 1. Tampilan ilustrasi Pembangkit Listrik Tenaga Listrik.....	5
Gambar 2. Thermocopel tipe K.....	7
Gambar 3. Sensor Ultrasonic A02YYUW	8
Gambar 4. <i>Liquid Crystal Display (LCD)</i>	8
Gambar 5. Mikrokontroler ESP32	9
Gambar 6. Modul SD Card.....	10
Gambar 7. RTC (<i>Real time clock</i>).....	10
Gambar 8. Arduino IDE	11
Gambar 9. Lokasi pengujian sistem Pulau Pangkil.....	13
Gambar 10. Lokasi pengujian sensor, BLKPP Kepulauan Riau.....	13
Gambar 11. Diagram alir penelitian.....	16
Gambar 12. Diagram perancangan perangkat.....	17
Gambar 13. Susunan sistem pada perangkat penelitian	18
Gambar 14. Cara kerja perangkat (a) <i>Master 1</i> , (b) <i>Master 2</i> dan (c) <i>Slave</i>	19
Gambar 15. Thermometer Digital.....	20
Gambar 16. Ilustrasi Pengukuran Bahan Bakar dalam Tangki	22
Gambar 17. Ilustrasi Penempatan Sensor Pada Radiator PLTD	22
Gambar 18. Program <i>Slave</i> ESP-NOW.....	23
Gambar 19. Program <i>Master</i> ESP-NOW	24
Gambar 20. Program RSSI pada perangkat <i>slave</i>	24
Gambar 21. Lokasi Pengujian ESP-NOW pada area terbuka (a) dan area tertutup (b).....	25
Gambar 22. Program pengujian LCD 20x4	28
Gambar 23. Hasil uji LCD 20x4	28
Gambar 24. Program Pengujian SD card	29
Gambar 25. Hasil pengujian micro SD	29
Gambar 26. Pengujian RTC dengan board ESP32.....	30
Gambar 27. Hasil tampilan jam dan tanggal pada serial.....	30
Gambar 28. Rangkaian skematik sensor suhu.....	31
Gambar 29. Persamaan dan garis regresi kalibrasi sensor suhu.....	33
Gambar 30. Tingkat kesalahan dan akurasi suhu	36
Gambar 31. Skematik rangkaian sensor jarak.....	37
Gambar 32. Pengujian sensor ultrasonik A02YYUW terhadap objek.....	37
Gambar 33. Tingkat <i>error</i> dan akurasi jarak.....	41
Gambar 34. Lokasi pengujian keseluruhan sistem.....	42
Gambar 35. Perangkat <i>Master 1</i>	43
Gambar 36. Rangkaian Perangkat <i>Master 1</i>	43
Gambar 37. Perangkat <i>master 2</i>	44

Gambar 38. Rangkaian Perangkat <i>Master 2</i>	44
Gambar 39. Perangkat <i>slave</i> di ruangan CS PLTD Pulau Pangkil	44
Gambar 40. Rangkaian Perangkat Pada <i>Slave</i>	45
Gambar 41. Hasil pengujian perangkat secara keseluruhan.....	45
Gambar 42. Program <i>Library</i> dari (a) <i>Master 1</i> , (b) <i>Master 2</i> dan (c) <i>Slave</i>	46
Gambar 43. Inisialisasi Variabel Perangkat (a) <i>Master 1</i> dan (b) <i>Master 2</i>	46
Gambar 44. Inisialisasi Variabel pada Perangkat <i>Slave</i>	47
Gambar 45. Deklarasi Variabel Perangkat <i>Slave</i>	47
Gambar 46. Fungsi Mengirim Data pada <i>Master 1</i> dan <i>Master 2</i>	48
Gambar 47. Fungsi <i>Void Setup</i> Pada <i>Master 1</i>	48
Gambar 48. Fungsi <i>Void Setup</i> Pada <i>Master 2</i>	49
Gambar 49. Fungsi <i>Void Setup</i> Pada <i>Slave</i>	49
Gambar 50. Program <i>Void Loop</i> Pada <i>Master 1</i>	50
Gambar 51. Program <i>Void Loop</i> Pada <i>Master 2</i>	51
Gambar 52. Program <i>Void Loop Slave</i> (a) LCD dan RTC, (b) <i>Master 1</i> dan <i>Master 2</i>	52
Gambar 53. Diagram Alir Pada <i>Master 1</i> dan <i>Master 2</i>	52
Gambar 54. Diagram Alir <i>Firmware</i> Pengukuran Menggunakan ESP-NOW Pada <i>Slave</i>	53
Gambar 55. Grafik data suhu air pendingin tanggal 16 Juni 2023.....	54
Gambar 56. Grafik data suhu air pendingin tanggal 17 Juni 2023.....	55
Gambar 57. Grafik data suhu air pendingin tanggal 18 Juni 2023.....	55
Gambar 58. Grafik data nilai kapasitas solar tanggal 16 Juni 2023	56
Gambar 59. Grafik data nilai kapasitas solar tanggal 17 Juni 2023.....	57
Gambar 60. Grafik data nilai kapasitas solar tanggal 18 Juni 2023.....	57

DAFTAR TABEL

Tabel 1. Spesifikasi ESP 32	9
Tabel 2. Daftar Alat Penelitian.....	14
Tabel 3. Daftar Bahan Penelitian	14
Tabel 4. Hasil pengujian pengiriman data keadaan terbuka ESP NOW	26
Tabel 5. Hasil pengujian pengiriman data keadaan tertutup ESP-NOW	26
Tabel 6. Hasil pengujian sensor suhu.....	32
Tabel 7. Hasil pengujian sensor jarak	38



DAFTAR LAMPIRAN

Lampiran 1. <i>Firmware</i> pada Perangkat <i>Slave</i>	63
Lampiran 2. Data Pengujian Suhu Pada Keseluruhan Sistem 16 Juni 2023	85
Lampiran 3. Data Pengujian Suhu Pada Keseluruhan Sistem 17 Juni 2023	88
Lampiran 4. Data Pengujian Suhu Pada Keseluruhan Sistem 18 juni 2023	91
Lampiran 5. Data Pengujian Kapasitas Pada Keseluruhan Sistem 16 juni 2023 ..	94
Lampiran 6. Data Pengujian Kapasitas Pada Keseluruhan Sistem 17 juni 2023 .	96
Lampiran 7. Data Pengujian Kapasitas Pada Keseluruhan Sistem 18 juni 2023 .	98
Lampiran 8. Program Kalibrasi Sensor Thermocopel Type-K	100
Lampiran 9. Hasil dari perhitungan nilai regresi	101
Lampiran 10. <i>Firmware</i> pada perangkat <i>master</i> 1	102
Lampiran 11. <i>Firmware</i> pada perangkat Master 2	105
Lampiran 12. Program Kalibrasi Sensor Ultrasonik A002YYUW	108
Lampiran 13. Dokumentasi Penelitian	109

ABSTRAK

Hadi, Afdal.2023. Sistem Monitoring Suhu Air Pendingin dan Kapasitas Bahan Bakar Dalam Tangki Pembangkit Listrik Tenaga Diesel (PLTD) Berbasis ESP-NOW di Pulau Pangki. Skripsi. Tanjungpinang. Jurusan Teknik Elektro Fakultas Teknik dan Teknologi Kemaritiman. Universitas Maritim Raja Ali Haji. Pembimbing I : Ir. Sapta Nugraha, S.T.,M.Eng., Pembimbing II : Ahmad Syafiq S.T.,M.Si.

Pulau Pangkil di Kepulauan Riau merupakan salah satu daerah yang menggunakan sumber listrik dari Pembangkit Listrik Tenaga Diesel (PLTD). Pembangkit Listrik Tenaga Diesel (PLTD) yang berada di Pulau Pangkil beroperasi selama 12 jam. Mesin yang beroperasi perlu dilakukan pengecekan parameter setiap 30 menit untuk mengantisipasi terjadinya *overheat* pada mesin diesel. Namun, parameter yang sangat penting terhadap kinerja mesin PLTD yaitu suhu air pendingin dan kapasitas bahan bakar solar dalam tangki. Penentuan suhu air pendingin dan kapasitas bahan bakar dalam tangki tentunya membutuhkan suatu teknologi. Teknologi tersebut merupakan teknologi yang mampu melakukan pengukuran pada indikator kinerja mesin PLTD diantaranya suhu air pendingin dan kapasitas bahan bakar solar dalam tangki sesuai dengan yang dibutuhkan. Perangkat sistem yang digunakan pada penelitian ini adalah sensor termokopel type-K sebagai sensor suhu dan ultrasonik A02YYUW sebagai sensor jarak. Sensor suhu memiliki tingkat akurasi rata-rata sebesar 98,50 %, sedangkan sensor jarak memiliki rata-rata akurasi sebesar 99,15 %. Tujuan dari penelitian ini adalah merancang sebuah alat sederhana yang mampu mengukur suhu air pendingin dan kapasitas bahan bakar solar dalam tangki yang terdapat di mesin PLTD Pulau Pangkil dalam satu perancangan yang dapat dipantau melalui *Liquid Cristal Display* (LCD). Setelah dilakukan penelitian maka di dapatkan nilai suhu air pendingin tertinggi sebesar 85 °C dan nilai suhu air pendingin terendah sebesar 39 °C. Adapun untuk nilai kapasitas solar tertinggi sebesar 75,67 % dan untuk nilai kapasitas solar terendah sebesar 67,4 %.

Kata Kunci : PLTD, Suhu, Jarak

ABSTRACT

Hadi, Afdal. 2023. ESP-NOW Based Cooling Water Temperature and Fuel Tank Capacity Monitoring System for Diesel Power Plant (PLTD) on Pangki Island. Thesis. Tanjung Pinang. Department of Electrical Engineering, Faculty of Maritime Engineering and Technology. King Ali Hajj Maritime University. Advisor I : Ir. Sapta Nugraha, S.T.,M.Eng., Advisor II : Ahmad Syafiq S.T.,M.Si.

Pangkil Island in the Riau Archipelago is one of the areas that uses a diesel power plant (PLTD) as a source of electricity. The Diesel Power Plant (PLTD) on Pangkil Island operates for 12 hours. The operating engine needs to check parameters every 30 minutes to anticipate overheating in diesel engines. However, the parameters that are very important to the performance of diesel engines are the temperature of the cooling water and the capacity of the diesel fuel tank. Determination of cooling water temperature and diesel fuel tank capacity certainly requires a technology. This technology is a technology that is capable of measuring the performance indicators of diesel engines, including the temperature of the cooling water and the capacity of the diesel fuel tank according to what is needed. The system device used in this study is a K-type thermocouple sensor as a temperature sensor and A02YYUW ultrasonic as a distance sensor. The temperature sensor has an average accuracy rate of 98,50 %, while the proximity sensor has an average accuracy of 99,15 %. The purpose of this research is to design a simple tool that is able to measure the temperature of the cooling water and the capacity of the diesel fuel tank in the Pangkil Island PLTD engine in one design that can be monitored through the Liquid Crystal Display (LCD). After doing the research, the highest cooling water temperature value is 85 °C and the lowest cooling water temperature value is 39 °C. As for the highest solar capacity value of 75.67% and the lowest solar capacity value of 67.4%.

Keyword : PLTD, *Temperatur, Distance*

I. PENDAHULUAN

A. Latar Belakang

Pembangkit listrik yang ada di Indonesia terdiri dari beberapa jenis diantaranya Pembangkit Listrik Tenaga Surya (PLTS), Pembangkit Listrik Tenaga Uap (PLTU), Pembangkit Listrik Tenaga Mesin Gas (PLTMG), Pembangkit Listrik Tenaga Air (PLTA), Pembangkit Listrik Tenaga Bayu (PLTB), Pembangkit Listrik Tenaga Gas dan Uap (PLTGU) dan Pembangkit Listrik Tenaga Diesel (PLTD) (Bachtiar & Syafiq, 2016). Pulau Pangkil di Kepulauan Riau merupakan salah satu daerah yang menggunakan sumber listrik dari Pembangkit Listrik Tenaga Diesel (PLTD). Pembangkit Listrik Tenaga Diesel (PLTD) yang berada di Pulau Pangkil beroperasi selama 12 jam.

Mesin yang beroperasi perlu dilakukan pengecekan parameter setiap 30 menit untuk mengantisipasi terjadinya *overheat* pada mesin diesel. Parameter yang harus dijaga dalam pengoperasian diantaranya tangki bahan bakar, suhu air pendingin, dan suhu pelumas. Namun, parameter yang sangat berpengaruh terhadap kinerja mesin PLTD yaitu suhu air pendingin dan tangki bahan bakar solar. Suhu yang paling rendah pada air pendingin 56,34 °C dan suhu tertinggi 80,85 °C (Fuadi *et al.*, 2020). Kapasitas tangki bahan bakar perlu dilakukan pengukuran untuk mengantisipasi kekurangan bahan bakar saat beroperasi (Rony & Falconi, 2013). Pengukuran yang dilakukan saat ini masih secara manual dengan mencelupkan besi ke dalam tangki untuk mengetahui berapa banyak bahan bakar dalam tangki. Adapun penelitian yang dilakukan oleh Mokhammad *et al.*, (2022) melakukan *monitoring* dan *controlling* HSD tank PLTGU

menggunakan pengontrolan dengan cara melihat dari atas tangki bahan bakar solar untuk mengetahui banyaknya bahan bakar dalam tangki . Hasil pemantauan dan pengukuran dilakukan dengan mencatat pada *form* secara berkala. Sistem yang dilakukan tidak *efisien* sehingga memakan waktu yang cukup lama.

Oleh karena itu diperlukan adanya sebuah inovasi yang mampu memantau dan mengukur sistem pengoperasian PLTD secara *real time*. Berdasarkan permasalahan tersebut maka dirancang sebuah alat yang dapat memantau suhu air pendingin dan mengukur kapasitas bahan bakar dalam tangki pada PLTD Pulau Pangkil berbasis ESP-NOW. Penelitian ini menggunakan sistem berbasis ESP-NOW untuk mengirimkan nilai suhu air pendingin dan kapasitas bahan bakar dalam tangki.

B. Rumusan Masalah

Berdasarkan permasalahan yang terdapat pada latar belakang diatas, rumusan masalah pada penelitian ini adalah sebagai berikut :

1. Bagaimana cara memantau suhu air pendingin mesin diesel pada PLTD Pulau Pangkil ?
2. Bagaimana cara mengukur kapasitas bahan bakar dalam tangki pada PLTD Pulau Pangkil ?

C. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut :

1. Perancangan sistem *monitoring* suhu air pendingin menggunakan sensor termokopel tipe K pada PLTD Pulau Pangkil.

2. Pengukuran kapasitas bahan bakar dalam tangki menggunakan sensor ultrasonik A02YYUW pada PLTD Pulau Pangkil.
3. Hasil pemantauan dan pengukuran di tampilkan di *Liquid Crystal Display* (LCD).
4. Objek penelitian adalah mesin diesel dan tangki bahan bakar solar PLTD Pulau Pangkil.

D. Tujuan Penelitian

Tujuan penelitian ini adalah merancang sistem *monitoring* suhu air pendingin dan kapasitas bahan bakar dalam tangki berbasis ESP-NOW di Pulau Pangkil.

E. Manfaat

Berdasarkan rumusan dan tujuan yang diperoleh manfaat dari penelitian ini yaitu membantu *user* (PLN, petugas PLTD, dosen dan mahasiswa) untuk *monitoring* suhu air pendingin dan kapasita bahan bakar dalam tangki PLTD Pulau Pangkil menggunakan sistem berbasis ESP-NOW.

II. TINJAUAN PUSTAKA

A. Kajian Terdahulu

Fuadi *et al.*, (2020) telah melakukan penelitian suhu air pendingin dengan kontrol dan *monitoring* suhu pada mesin diesel. RTD PT 100 berfungsi mengukur suhu air pendingin, sehingga pengguna dapat memantau suhu mesin. Sensor suhu memiliki kesalahan pembacaan suhu rata-rata 0,031004% adapun seharusnya suhu yang ideal dapat dilihat melalui sistem buka tutup solenoid *valve*, suhu paling rendah 56,34 °C dan suhu tertinggi hanya 80,85 °C.

Selain itu juga dilakukan penelitian Yulianti *et al.*, (2021) pada suhu air pendingin yaitu *prototipe monitoring* suhu air pada genset 1250 kVA berbasis mikrokontroler Arduino Uno 328 via SMS. Sensor suhu DS18B20 digunakan untuk mengukur suhu air pada genset 1250 kVA pada sistem *monitoring* suhu air dapat dipantau melalui notifikasi SMS dengan menggunakan *provider* Simpati dan Axis. *Provider* Simpati dibandingkan dengan *provider* Axis, *provider* simpati dalam kenaikan suhu 1 °C pada heater satu memerlukan waktu 7,5 detik dan saat penurunan memerlukan waktu 7,9 detik.

Pengukuran suhu air pendingin yang dilakukan pada penelitian Randis & Sarminto, (2018) dapat *monitoring* suhu *engine* untuk mencegah terjadinya *overheat*. Sensor DS18B20 berfungsi untuk mengukur suhu air pendingin adapun sistem kontrol yang digunakan yaitu mikrokontroler ESP 32. Hasil percobaan menunjukkan bahwa selisih nilai pembacaan sensor yang alat ukur kalibrasi 0,14 °C sedangkan *margin error* diperoleh 0,14 °C, sesuai dengan pengujian yang dilakukan dari rentang waktu 1 jam suhu yang dihasilkan mencapai 69 °C.

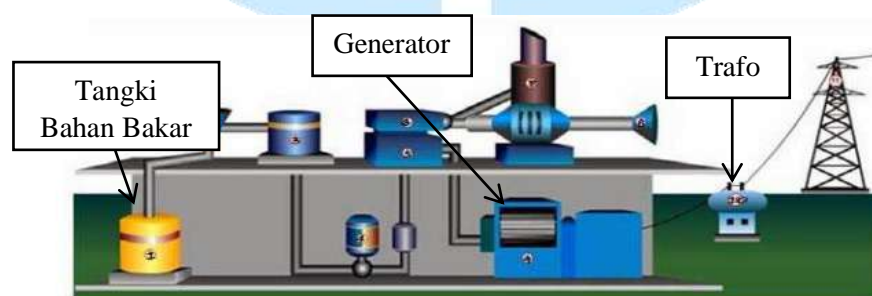
Adapun penelitian pada kapasitas tangki bahan bakar sesuai dengan yang sudah dilakukan yaitu sistem *monitoring* tangki solar menggunakan sensor ultrasonik. Sensor Ultrasonik berfungsi untuk mengukur bahan bakar dalam tangki. Adapun sistem pemberitahuan kepada petugas menggunakan *client server* dimana sistem tersebut berfungsi jika *server* terhubung dengan jaringan LAN untuk mengetahui kapasitas tangki sudah berada pada level tertentu (Kurniawan *et al.*, 2015).

Yusuf *et al.*, (2022) juga melakukan penelitian prototipe sistem *monitoring* dan *controlling* HSD tank PLTGU. Sensor Ultrasonik HC-SR04 berfungsi untuk mengukur level tangki bahan bakar di ketahui bahwa sensor ultrasonik tidak terdapat perbedaan hasil yang signifikan disaat pengukuran.

B. Landasan Teori

1. Pembangkit Listrik Tenaga Diesel (PLTD)

Pembangkit Listrik Tenaga Diesel (PLTD) ialah pembangkit listrik yang menggunakan mesin diesel sebagai penggerak mula (*prime mover*). *Prime* merupakan peralatan yang memiliki fungsi menghasilkan energi mekanis yang digunakan untuk memutar rotor generator (Rahman, 2018). Ilustrasi Pembangkit Listrik Tenaga Diesel dapat dilihat pada Gambar 1.



Gambar 1. Tampilan ilustrasi Pembangkit Listrik Tenaga Listrik (Sumber: bmj.co.id)

2. Radiator

Radiator merupakan alat yang berfungsi sebagai sistem pendingin pada mesin kendaraan. Komponen ini salah satu alat yang berguna untuk memindahkan perpindahan panas dari satu *medium* ke *medium* lainnya dengan tujuan untuk mendinginkan mesin (Hyundai, 2022).

Sistem pendingin terdiri dari tiga macam yaitu sistem pendingin oli, sistem pendingin dengan udara, sistem pendingin dengan air. Secara umum sistem pendingin air sangat banyak digunakan saat ini dikarenakan sistem pendingin salah satu pengembalian suhu normal pada mesin (Fitrianingsih, 2022).

3. Suhu

Suhu merupakan ukuran kuantitas terhadap *temperature* panas dan dingin. Suhu tidak dapat dilihat tetapi dapat dirasakan semakin tinggi suatu benda maka makin tinggi panas yang dimiliki. Suhu hanya dapat diketahui dari akibat yang ditimbulkan pada benda lain (Pangesti, 2021).

4. Tangki Bahan Bakar

Tangki bahan bakar merupakan salah satu hal terpenting untuk menampung BBM baik yang akan digunakan baik dalam keadaan hidup maupun saat mesin tidak digunakan. Tangki memiliki peran yang sangat penting dalam penggunaan sehari-hari (MyPertamina, 2021).

5. Sensor Suhu

Termokopel tipe K merupakan jenis termokopel yang paling umum digunakan dalam berbagai industri misalnya industri minyak dan gas karena rentang ukur pembacaannya yang lebar serta harga yang relatif terjangkau.

Termokopel ini biasanya disebut dengan termokopel chromel-alumel. Adapun prinsip kerja thermocopel tipe-K yaitu terdapat perbedaan suhu pada kedua ujung kawat termokopel yang membuat elektron pada kawat berpindah dari daerah yang bersuhu tinggi ke daerah yang bersuhu rendah. Sementara pada ujung kabel yang dihubungkan dengan nyala api dapat terbaca tegangannya (sentrakalibrasiindustri, 2022). Thermocopel tipe K dapat dilihat pada Gambar 2.



Gambar 2. Thermocopel tipe K
(Sumber : Sentrakalibrasiindustri.com)

6. Sensor Ultrasonik

Sensor ultrasonik adalah sensor yang bekerja berdasarkan prinsip pantulan gelombang suara dan digunakan untuk mendeteksi keberadaan suatu objek atau benda tertentu di depan frekuensi kerja pada daerah di atas gelombang suara dari 40 kHz hingga 400 KHz. Pengukuran yang dilakukan akan menghasilkan nilai digital dimana pin yang digunakan untuk mendapatkan nilai jarak yaitu pin RX sebagai pemberi sinyal dan TX sebagai pemancar sinyal. Adapun rentang jarak pengukuran sensor dari 3 sampai 450 cm (Hudati *et al.*, 2021). Sensor Ultrasonik A02YYUW dapat dilihat pada Gambar 3.



Gambar 3. Sensor Ultrasonic A02YYUW

(Sumber: Hudati *et al.*, 2021)

7. *Liquid Crystal Display (LCD)*

LCD adalah perangkat yang berfungsi sebagai media penampilan dengan pemanfaatan kristal cair sebagai objek penampil utama. LCD juga sangat banyak digunakan seperti media elektronik, televisi, kalkulator. LCD yang digunakan adalah LCD berukuran 20X4 karakter dengan tambahan chip *module* I2C untuk mempermudah program dalam mengakses LCD. Pin LCD I2C yang digunakan memiliki 4 buah pin yaitu, pin SCL, pin SDA, pin VCC dan pin GND (Saepudin & Hafidz, 2020). Bentuk fisik LCD dapat dilihat pada Gambar 4.



Gambar 4. *Liquid Crystal Display (LCD)*

(Sumber : Franux Eletronics)

8. ESP 32

ESP 32 adalah mikrokontroler yang dikenalkan oleh Espressif System merupakan penerus dari mikrokontroler ESP8266. Salah satu kelebihan yang dimiliki oleh ESP32 adalah sudah terdapat Wifi dan Bluetooth di dalamnya, yang akan sangat mempermudah pembuatan sistem IoT yang memerlukan koneksi wireless. Fitur-fitur tersebut tidak ada di dalam ESP 8266, sehingga ESP 32 merupakan sebuah *upgrade* dari ESP 8266 (Muliadi *et al.*, 2020). Bentuk ESP32 dapat dilihat pada Gambar 5. Spesifikasi ESP 32 dapat dilihat pada Tabel 1.



Gambar 5. Mikrokontroler ESP32
(Sumber : BINUS University)

Tabel 1. Spesifikasi ESP 32

No.	Keterangan	Spesifikasi
1.	Tegangan	3.3 Volt
2.	SRAM	512 kB
3.	Flash Memory	16 MB
4.	Bluetooth/Wifi	Ada
5.	CPU	Xtensa dual core LX6 -160MHz
6.	GPIO pin (ADC/DAC)	36 (18/2)
7.	SPI/I2C/UART	4/2/2
8.	Arsitektur	32 bit

Sumber : (Muliadi et al., 2020)

9. Modul SD Card

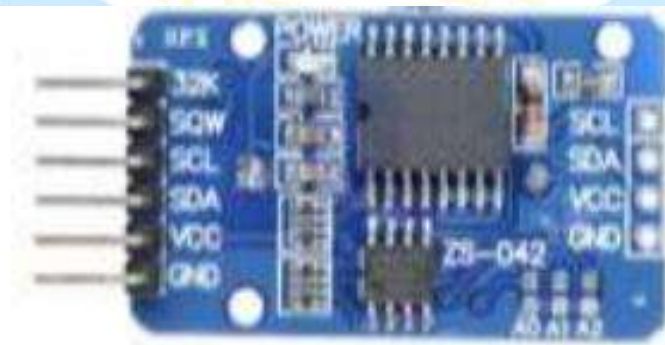
Modul penyimpanan data digunakan untuk menyimpan seluruh data sensor yang dihasilkan dari alat *monitoring*. Modul ini memiliki spesifikasi mendukung *interfacing* SD Card dengan tegangan kerja 3.3 sampai 5 V (Utomo & Wirawan, 2018). Modul SD Card dapat dilihat pada Gambar 6.



Gambar 6. Modul SD Card

10. RTC (Real time clock)

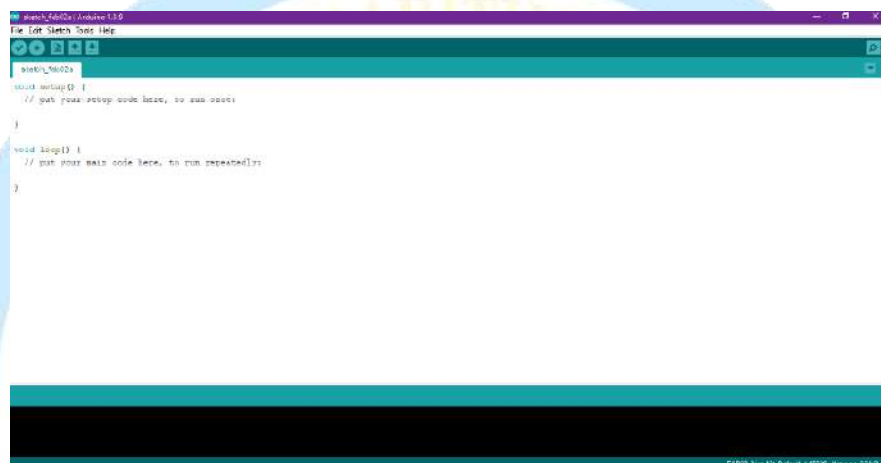
RTC (*Real time clock*) adalah sebuah perangkat yang dapat menerima dan menyimpan data *realtime* berupa deskripsi waktu, seperti hari, tanggal, bulan, tahun. RTC juga mampu menyimpan data perbedaan bulan yang memiliki 30 hari ataupun 31 hari (Rahardjo, 2022). RTC dapat dilihat pada Gambar 7.



Gambar 7. RTC (*Real time clock*)

11. Arduino IDE

Arduino IDE (*Integrate Development Environment*) merupakan sebuah platform perangkat lunak yang ditulis dalam bahasa pemrograman C/C++. Arduino IDE digunakan untuk mengembangkan aplikasi mikrokontroler mulai dari menuliskan *source* program, kompilasi, *upload* hasil dan uji coba secara terminal serial (Arduino.cc, 2023). Tampilan Arduino IDE dapat dilihat pada Gambar 8.



Gambar 8. Arduino IDE
(Sumber : Arduino.cc)

12. Error dan Akurasi

Error (kesalahan) merupakan nilai yang menunjukkan penyimpangan dalam pengukuran tertentu (Advanced Instructional Systems Inc, 2011). Nilai *error* didapatkan dari persamaan berikut :

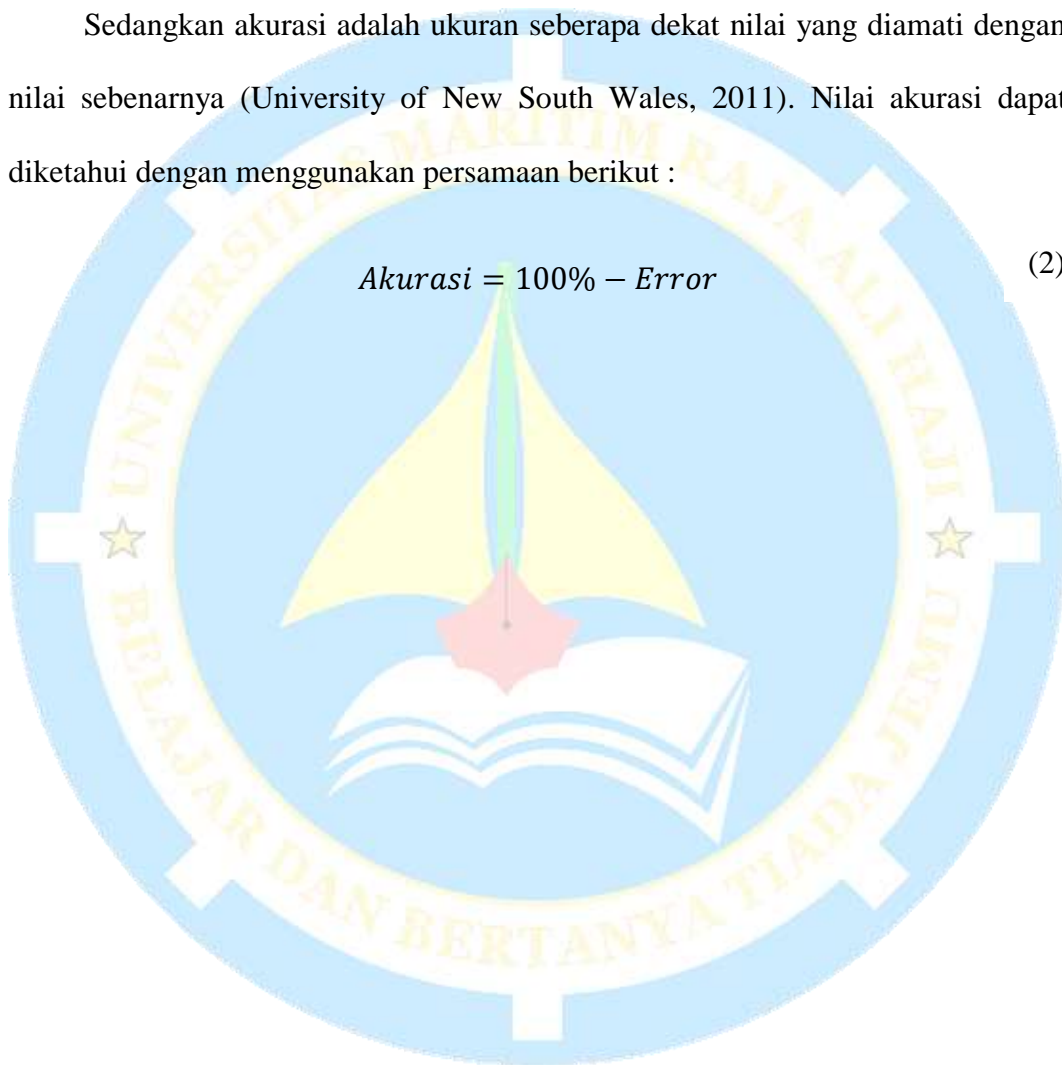
$$Error = \left| \left(\frac{Data\ Kalibrator - Data\ Sensor}{Data\ kalibrator} \right) \right| \times 100\% \quad (1)$$

Error dapat terjadi karena 2 kesalahan. Kesalahan yang pertama adalah *random error*. *Random error* merupakan kesalahan yang bersifat sementara dan tidak dapat diprediksi dan diperkirakan. Sedangkan yang kedua adalah

systemathic error. Kesalahan ini adalah kesalahan yang bermula dari pengaruh-pengaruh yang dapat diduga dengan pasti atau ditimbulkan oleh adanya faktor tetap yang mengakibatkan hasil dari pengujian tersebut lebih tinggi atau lebih rendah dari nilai sebenarnya (Munir, 2003).

Sedangkan akurasi adalah ukuran seberapa dekat nilai yang diamati dengan nilai sebenarnya (University of New South Wales, 2011). Nilai akurasi dapat diketahui dengan menggunakan persamaan berikut :

$$Akurasi = 100\% - Error \quad (2)$$



III. METODOLOGI PENELITIAN

A. Waktu dan Lokasi Penelitian

Penelitian ini dilakukan pada bulan Maret - Juni 2023. Perancangan dan kalibrasi sensor dilakukan di Balai Latihan Kerja dan Pengembangan Profesi (BLKPP) Kepulauan Riau dapat dilihat pada Gambar 9. Lokasi pengujian dilakukan di Pulau Pangkil yang terletak di Kec. Teluk Bintan, Kabupaten Bintan, Kepulauan Riau yang dapat dilihat pada Gambar 10.



Gambar 9. Lokasi pengujian sistem Pulau Pangkil
(Sumber : Google Map)



Gambar 10. Lokasi pengujian sensor, BLKPP Kepulauan Riau
(Sumber : Google Maps)

B. Alat dan Bahan Penelitian

Pada penelitian ini diperlukan beberapa alat dan bahan untuk mendukung penelitian. Alat dan bahan yang digunakan dalam penelitian dapat dilihat pada Tabel 2 dan Tabel 3.

Tabel 2. Daftar Alat Penelitian

No.	Nama Alat	Jumlah	Satuan
1.	Laptop	1	Pcs
2.	Solder	1	Pcs
3.	Gunting	1	Pcs
4.	Obeng	1	Pcs

Tabel 3. Daftar Bahan Penelitian

No.	Nama Alat	Jumlah	Satuan
1.	ESP32	3	Pcs
2.	Sensor Thermocopel Tipe K	1	Pcs
3.	Sensor Ultrasonic A02YYUW	1	Pcs
4.	<i>Box</i>	1	Pcs
5.	<i>Powerbank</i>	1	Pcs
6.	LCD 2004	1	Pcs
7.	<i>Module Micro SD Card</i>	1	Pcs
8.	PCB	1	Pcs
9.	JST Pin	1	Pcs
10.	Kabel <i>Jumper</i>	10	Pcs
11.	RTC DS3231	1	Pcs

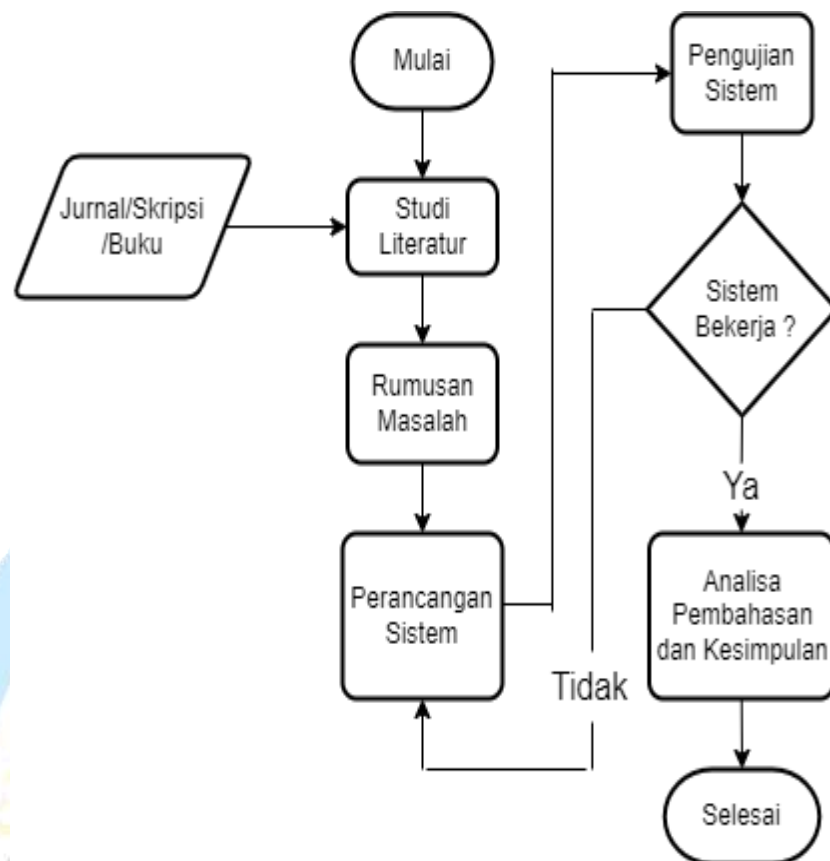
C. Prosedur Penelitian

Penelitian ini dimulai dari pengumpulan studi literatur yang didapatkan dari jurnal, skripsi atau buku-buku yang berkaitan dengan penelitian. Kajian terdahulu dan landasan teori berisi referensi hal-hal yang diperlukan dalam penelitian ini.

Rumusan masalah merupakan dasar pokok permasalahan yang didapatkan berdasarkan pengamatan yang dilakukan dilapangan. Rumusan masalah diangkat dari latar belakang permasalahan yang terjadi. Berdasarkan studi kajian terdahulu, landasan teori dan perumusan masalah, suatu perancangan perangkat diperlukan guna memecahkan permasalahan pada penelitian yang dilakukan.

Perancangan alat *monitoring* suhu air pendingin dan kapasitas bahan bakar di dalam tangki dilakukan dengan 2 tahapan yaitu perancangan *hardware* dan *firmware*. Perancangan *hardware* dilakukan dengan menguji fungsional dari setiap komponen yang digunakan. Setelah diuji, beberapa komponen tersebut digabungkan menjadi sebuah sistem *hardware*. Sedangkan perancangan *firmware* dilakukan dengan menguji perangkat bekerja sesuai dengan *firmware* yang telah di-*upload*.

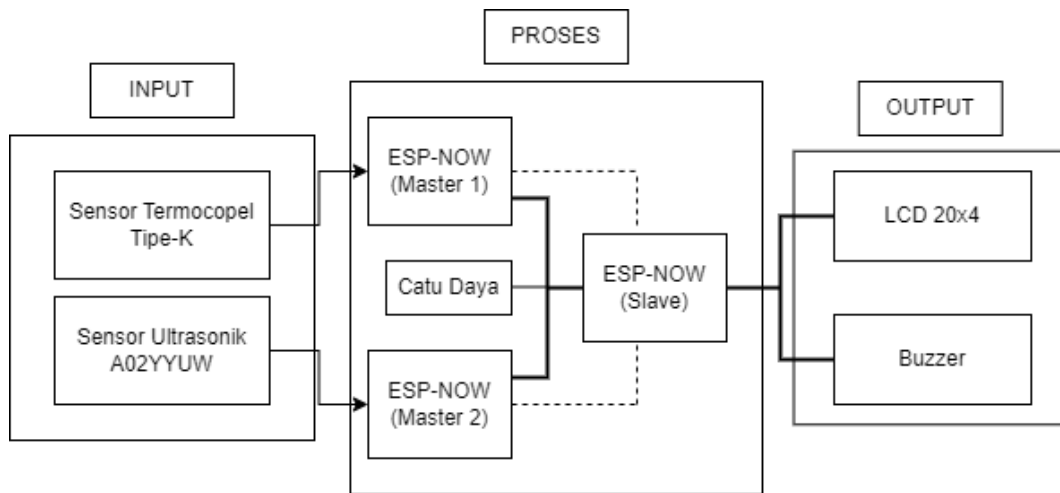
Tahap pengujian adalah tahap dimana seluruh sistem diuji untuk mengetahui alat bekerja atau tidak. Apabila alat tidak berhasil bekerja maka akan dilakukan perancangan ulang dan jika berhasil maka dilanjutkan dengan menganalisis hasil yang didapat dari hasil uji yang dilakukan untuk mendapatkan kesimpulan yang diinginkan. Diagram alir penelitian ditampilkan pada Gambar 11.



Gambar 11. Diagram alir penelitian

D. Perancangan Sistem

Perancangan perangkat *monitoring* suhu air pendingin dan kapasitas bahan bakar pada tangki PLTD Pulau Pangkil ini terdiri dari beberapa komponen utama yaitu sensor thermocopel tipe K, sensor ultrasonik A02YYUW, ESP32. Perangkat terdiri dari 3 ESP-NOW, yaitu *master 1* yang ditempatkan pada ruangan mesin PLTD, *master 2* ditempatkan di tangki bahan bakar solar dan *slave* yang ditempatkan ruangan CS PLTD Pulau Pangkil. Perangkat menggunakan *powerbank* sebagai sumber daya utama. Diagram perancangan perangkat dapat dilihat pada Gambar 12.

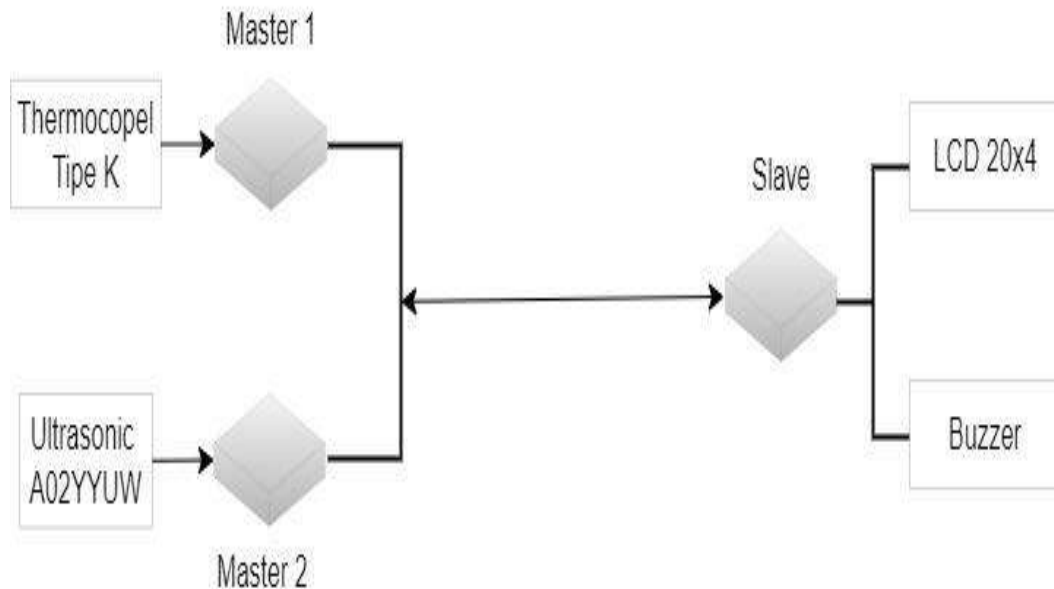


Gambar 12. Diagram perancangan perangkat

Susunan perangkat *monitoring* suhu air pendingin dan kapasitas bahan bakar dalam tangki pada PLTD menggunakan sensor thermocopel tipe K sebagai sensor suhu. Pada penelitian ini juga digunakan sensor ultrasonik untuk mengukur kapasitas bahan bakar dalam tangki pada PLTD. Setelah sensor-sensor tersebut berhasil membaca data yang diinginkan, maka akan dikirimkan dan diproses ESP-NOW *master*. Data tersebut akan dikirimkan ke ESP-NOW *slave* yang berada di ruangan CS PLTD Pulau Pangkil. Setelah data diterima oleh ESP-NOW *slave*, maka akan dilanjutkan pengiriman data nilai pada LCD dan *buzzer*.

Jangkauan jarak antara *master* dan perangkat *slave* adalah ± 220 M dengan keadaan *Line of Sight* (LOS) untuk dapat mengirimkan data tanpa adanya penghalang di antara kedua perangkat tersebut (Prastyo, 2022). Seluruh data suhu atau kapasitas yang diambil akan dikirimkan dari perangkat *master* ke perangkat *slave* dengan menggunakan komunikasi ESP-NOW. Setelah data dikirim ke *Slave*, data akan dikirim ke LCD untuk ditampilkan. Adapun sistem kerja buzzer yaitu apabila nilai suhu 90°C dan nilai kapasitas 5 % maka akan bekerja untuk

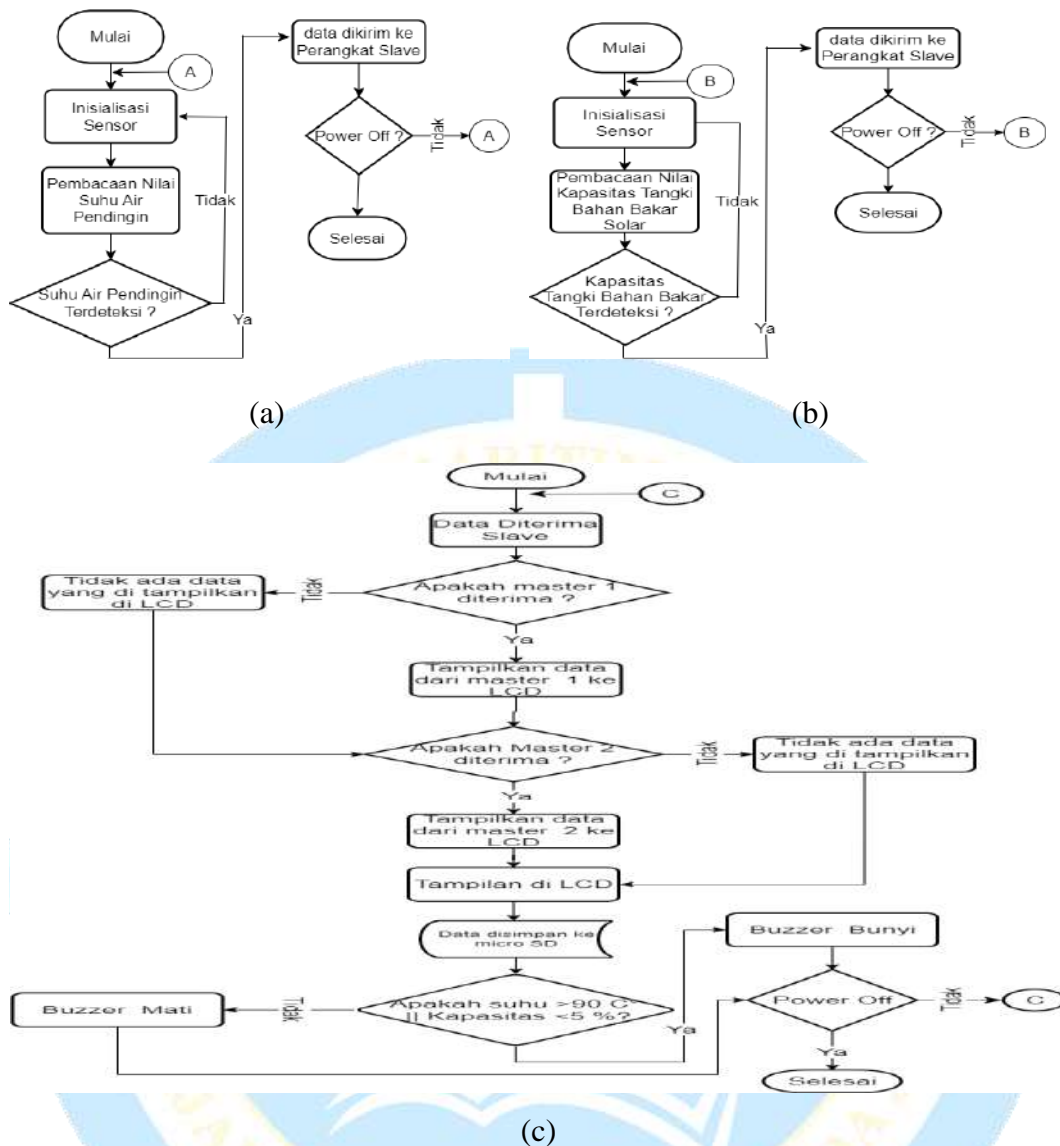
memberikan notifikasi berbentuk suara buzzer secara berulang-ulang. Susunan sistem pada perangkat penelitian dapat dilihat pada Gambar 13.



Gambar 13. Susunan sistem pada perangkat penelitian

Cara kerja perangkat *Master* dimulai dengan menghubungkan perangkat ke *power supply*. Setelah *power supply* terhubung, sensor-sensor yang digunakan pada penelitian akan diinisialisasi. Sensor-sensor tersebut akan membaca suhu air pendingin dan kapasitas bahan bakar dalam tangki. Hasil pembacaan tersebut akan diolah oleh mikrokontroler ESP-NOW *Master*. Setelah itu akan dikirimkan ke ESP-NOW yang berfungsi sebagai *Slave* di PLTD Pulau Pangkil.

Perangkat *slave* berfungsi untuk menerima data yang dikirim oleh perangkat *master*. Setelah data diterima, perangkat LCD dan buzzer ke ruangan *customer service* PLTD Pulau Pangkil. Cara kerja perangkat *master* dan *slave* dapat dilihat pada Gambar 14.



Gambar 14. Cara kerja perangkat (a) *Master 1*, (b) *Master 2* dan (c) *Slave*

E. Pengolahan Data dan Analisis

Nilai dari setiap sensor akan dikalibrasi menggunakan standar masing masing sensor. Nilai yang didapatkan dari kalibrator akan dibandingkan dengan nilai yang didapatkan oleh sensor.

Kalibrasi sensor suhu air pendingin dilakukan dengan melihat pengukuran suhu air pendingin pada tampilan thermometer digital. Sistem pengambilan data suhu setiap 10 menit sekali untuk mempermudah kalibrasi. Kemudian data suhu

yang sudah didapatkan pada tampilan digital tersebut diuji menggunakan sensor thermocopel tipe K untuk mendapatkan nilai suhu dari masing-masing pengujian. Nilai yang didapat dari sensor Thermocopel tipe K dan Thermometer digital. Kemudian data dibandingkan dengan metode regresi sehingga didapatkan persamaan yang digunakan sebagai persamaan kalibrasi. Adapun kalibrator yang digunakan yaitu termometer digital dapat dilihat pada Gambar 15.



Gambar 15. Thermometer Digital

Regresi yang digunakan untuk mengkalibrasi sensor suhu adalah regresi linear. Regresi linear merupakan salah satu jenis regresi yang persamaannya menggambarkan hubungan karakteristik kualitas maupun kuantitas. Persamaan regresi linear digambarkan dengan garis lurus (Harsiti *et al.*,2022). Persamaan umum regresi linear adalah sebagai berikut:

$$y = a + bx \quad (3)$$

Dimana :

a : Konstanta

b : Koefisien Regresi

y : Variabel dependen (variabel tak bebas)

x : Variabel independen (variabel bebas)

Perbandingan sensor jarak juga dilakukan dengan melihat nilai yang didapatkan dari sensor Ultrasonik A02YYUW dan Meteran. Adapun nilai yang dikirim oleh sensor Ultrasonik A02YYUW yaitu berbentuk digital. Sistem pengambilan data sensor jarak setiap 10 menit sekali untuk mempermudah

kalibrasi. Kemudian data jarak yang sudah didapatkan pada tampilan meteran tersebut diuji menggunakan sensor Ultrasonik A02YYUW untuk mendapatkan nilai jarak dari masing-masing pengujian.

Setelah kalibrasi dilakukan, akan diukur nilai *error* dan akurasi dengan menggunakan Persamaan (1) dan Persamaan (2). Hal ini dilakukan untuk mengetahui seberapa jauh tingkat kesalahan sensor terhadap alat ukur standar dan tingkat kedekatan pengukuran sensor terhadap alat ukur standar. Adapun regresi di gunakan untuk mendapatkan persamaan yang menggambarkan hubungan karakteristik kualitas maupun kuantitas menggunakan Persamaan (3).

Pengujian selanjutnya dilakukan untuk mengetahui jarak terjauh ESP-NOW dapat mengirimkan data. Pengujian ini dilakukan dengan mengirimkan beberapa data di beberapa titik yang berbeda. Pengujian dimulai dari jarak yang terdekat hingga jarak terjauh. Uji ini dilakukan untuk mengetahui jarak terjauh ESP-NOW mengirimkan data tanpa adanya *packet loss*. Adapun cara pengujian dilakukan dengan jarak terdekat sampai jarak terjauh sehingga kondisi data pada pengujian tidak dapat terkirim.

Kemudian akan dilakukan uji lapangan untuk mendapatkan pengukuran suhu air pendingin dan kapasitas bahan bakar dalam tangki pada PLTD Pulau Pangkil. Keseluruhan data yang telah didapat akan dikirim ke LCD agar bisa dipantau oleh petugas. Satuan nilai yang dikirimkan ke LCD adalah *Celsius* (C°) untuk sensor suhu dan Kapasitas (%) untuk sensor jarak. Adapun sistem untuk mengetahui kapasitas bahan bakar dalam tangki solar yaitu dengan mengukur ketinggian bahan bakar yang tersedia pada tangki. Rumus untuk menghitung kapasitas bahan bakar dapat dilihat pada Persamaan (4).

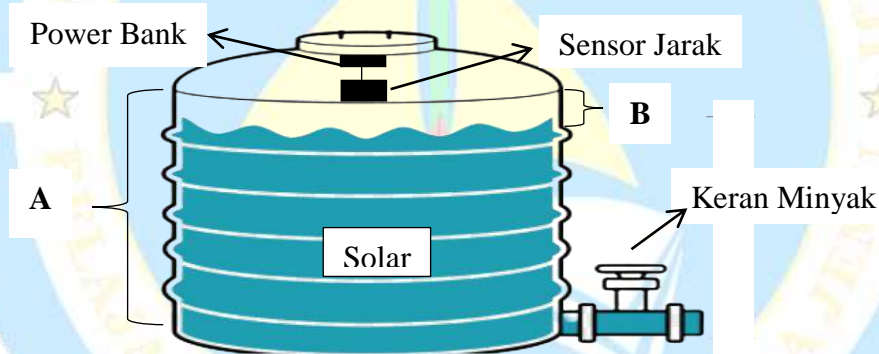
$$Kapasitas = \frac{a-b}{a} \times 100\% \quad (4)$$

Dimana:

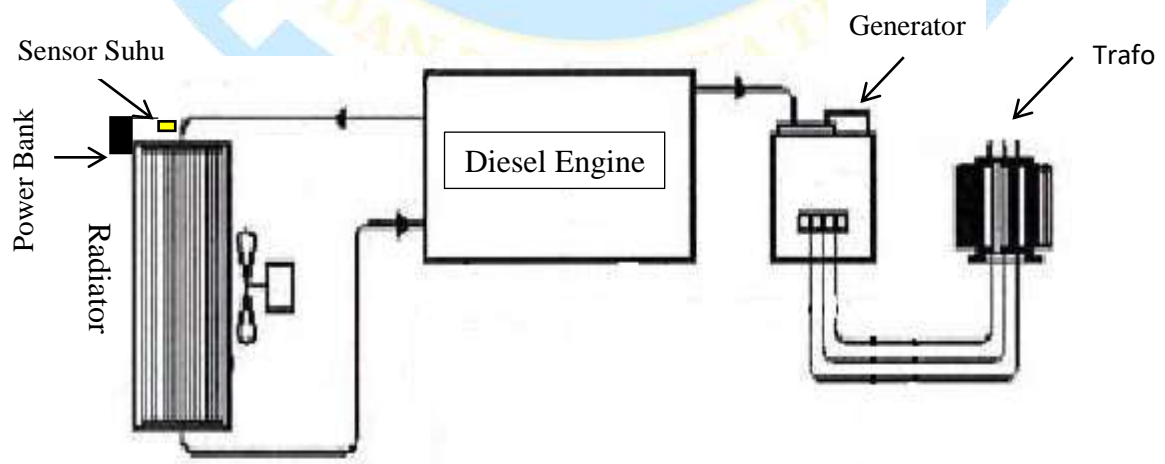
a : Tinggi Tangki

b : Pembacaan Sensor

Ilustrasi pengukuran bahan bakar dalam tangki dapat dilihat pada Gambar 16. Nilai suhu dan kapasitas akan menjadi dasar untuk mengetahui pengoperasian berjalan normal. Adapun ilustrasi pengukuran suhu air pendingin dapat dilihat pada Gambar 17. Setelah dilakukan pengambilan data dari suhu air pendingin dan bahan bakar dalam tangki PLTD Pulau Pangkil lalu dilakukan pengiriman data dari *master* ke *slave*. Keberhasilan pengiriman data ke LCD dan *buzzer* harus dipantau untuk mengetahui data berhasil dikirim atau gagal dikirim.



Gambar 16. Ilustrasi Pengukuran Bahan Bakar dalam Tangki



Gambar 17. Ilustrasi Penempatan Sensor Pada Radiator PLTD

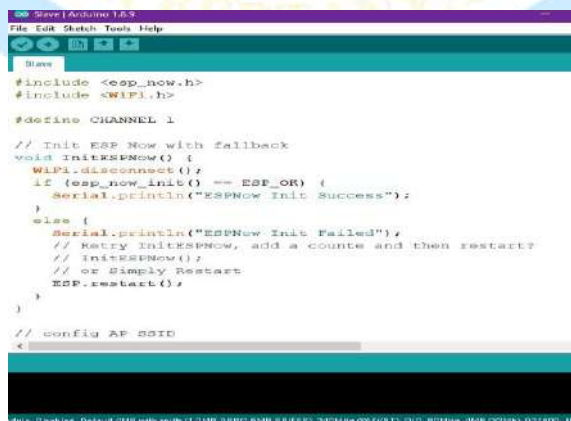
IV. HASIL DAN PEMBAHASAN

A. Pengujian Sistem

Proses pengujian sistem pada penelitian ini dilakukan dengan beberapa tahapan. Tahapan tersebut dimulai dari pengujian ESP-NOW, pengujian setiap sensor dalam pengambilan data dan pengujian sistem secara keseluruhan. Tahapan pengujian sistem dilakukan sebagai berikut.

1. Pengujian ESP-NOW

Pengujian ESP-NOW dilakukan untuk mengetahui dua perangkat dapat terhubung secara baik atau tidak. Pengujian pengiriman data pada perangkat *slave* akan dilakukan pada beberapa titik mulai titik terdekat hingga titik terjauh. Sistem pengujian ESP-NOW juga dilakukan dengan memastikan bahwasanya jalur dari setiap perangkat sudah sesuai dengan jalur yang digunakan *slave* atau tidak, dikarenakan apabila jalur dari *master* berbeda dengan *slave* maka akan mengakibatkan kegagalan pengiriman data sehingga perlu adanya penyesuaian jalur antara *master* dan *slave*. Program sederhana pengujian pengiriman data ESP-NOW dapat dilihat pada Gambar 18 dan Gambar 19.



```
Serial Arduino 1.6.3
File Edit Sketch Tools Help

Slave

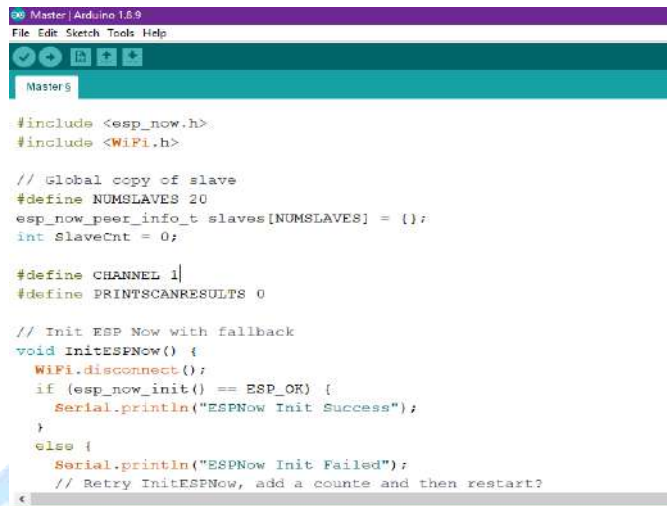
#include <esp_now.h>
#include <WiFi.h>

#define CHANNEL 1

// Init ESP Now with fallback
void InitESPNow() {
  WiFi.disconnect();
  if (esp_now_init() == ESP_OK) {
    Serial.println("ESPNow Init Success");
  }
  else {
    Serial.println("ESPNow Init Failed");
    // Retry InitESPNow, add a count and then restart?
    // InitESPNow();
    // or Simply Restart
    ESP.restart();
  }
}

// config AP SSID
<
```

Gambar 18. Program *Slave* ESP-NOW



```

Master | Arduino 1.8.9
File Edit Sketch Tools Help

Master$

#include <esp_now.h>
#include <WiFi.h>

// Global copy of slave
#define NUMSLAVES 20
esp_now_peer_info_t slaves[NUMSLAVES] = {};
int SlaveCnt = 0;

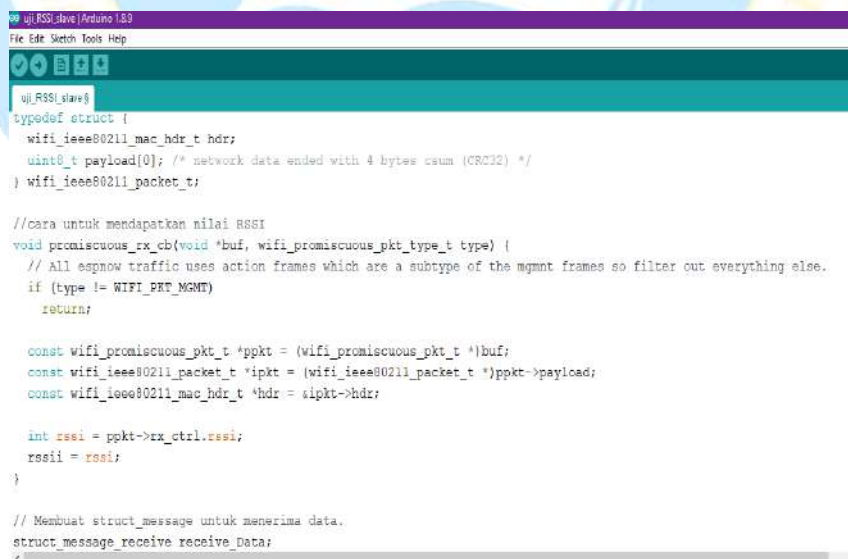
#define CHANNEL 1
#define PRINTSCANRESULTS 0

// Init ESP Now with fallback
void initESPNow() {
  WiFi.disconnect();
  if (esp_now_init() == ESP_OK) {
    Serial.println("ESPNow Init Success");
  }
  else {
    Serial.println("ESPNow Init Failed");
    // Retry InitESPNow, add a count and then restart?
  }
}

```

Gambar 19. Program *Master* ESP-NOW

Adapun cara mengetahui kekuatan sinyal saat pengujian jarak ESP-NOW yaitu dengan menggunakan *Receive Signal Strength Indicator* (RSSI). RSSI merupakan teknik yang digunakan untuk mengukur indikator kekuatan sinyal yang diterima oleh sebuah perangkat (Puspitasari, 2011). Program pengujian kekuatan sinyal pada perangkat *slave* dapat dilihat pada Gambar 20. Adapun program pengujian RSSI dapat dilihat pada Lampiran 1.



```

uji_RSSI_slave | Arduino 1.8.9
File Edit Sketch Tools Help

uji_RSSI_slave$

typedef struct {
  wifi_ieee80211_mac_hdr_t hdr;
  uint8_t payload[0]; /* network data ended with 4 bytes csam (CRC32) */
} wifi_ieee80211_packet_t;

//cara untuk mendapatkan nilai RSSI
void promiscuous_rx_cb(void *buf, wifi_promiscuous_pkt_type_t type) {
  // All espnow traffic uses action frames which are a subtype of the mgmt frames so filter out everything else.
  if (type != WIFI_PKT_MGMT)
    return;

  const wifi_promiscuous_pkt_t *ppkt = (wifi_promiscuous_pkt_t *)buf;
  const wifi_ieee80211_packet_t *ipkt = (wifi_ieee80211_packet_t *)ppkt->payload;
  const wifi_ieee80211_mac_hdr_t *hdr = &ipkt->hdr;

  int rssi = ppkt->rx_ctrl.rssi;
  rssi = rssi;
}

// Membuat struct_message untuk menerima data.
struct_message_receive receive_Data;

```

Gambar 20. Program RSSI pada perangkat *slave*

Pengujian dilakukan dengan menempatkan *slave* di kantor BLKPP Tanjungpinang Kepulauan Riau sedangkan *master* akan diuji beberapa titik untuk melihat keberhasilan pengiriman data *master* ke *slave*. Lokasi pengujian pengiriman data terbuka ESP-NOW dari jarak 20 m – 230 m dan pengujian pengiriman data tertutup antara ESP-NOW dari 5 m – 35 m yang dapat dilihat pada Gambar 21. Adapun hasil pengujian pengiriman data keadaan terbuka dapat dilihat pada Tabel 4 dan keadaan tertutup dapat dilihat pada Tabel 5.



(a)



(b)

Gambar 21. Lokasi Pengujian ESP-NOW pada area terbuka (a) dan area tertutup (b)

Tabel 4. Hasil pengujian pengiriman data keadaan terbuka ESP NOW

No	Jarak (m)	RSSI (dBm)	Status
1.	20	-80	(Ter kirim)
2.	40	-82	(Ter kirim)
3.	60	-84	(Ter kirim)
4.	80	-82	(Ter kirim)
5.	100	-85	(Ter kirim)
6.	120	-86	(Ter kirim)
7.	140	-83	(Ter kirim)
8.	160	-84	(Ter kirim)
9.	180	-87	(Ter kirim)
10	190	-88	(Ter kirim)
11	200	-91	(Ter kirim)
12	210	-92	(Ter kirim)
13	220	-94	(Tidak Ter kirim)
14	230	-94	(Tidak Ter kirim)

Tabel 5. Hasil pengujian pengiriman data keadaan tertutup ESP-NOW

No	Jarak (m)	RSSI (dBm)	Status
1.	5	-60	(Ter kirim)
2.	15	-80	(Ter kirim)
3.	25	-88	(Ter kirim)
4.	30	-91	(Ter kirim)
5.	35	-94	(Tidak Ter kirim)

Hasil pengujian pengiriman data antar ESP-NOW menunjukkan bahwa ESP-NOW dapat mengirim data antara perangkat *slave* dan *master*. Jarak pengiriman data mempengaruhi keberhasilan pengiriman data. Pengiriman data di lapangan terbuka yang dilakukan di bawah jarak 210 m dengan nilai RSSI -92 dBm mengalami keberhasilan yang baik. Dari seluruh data yang dikirimkan, semuanya berhasil diterima oleh *slave*. Adapun pengiriman data diatas jarak 220 m dengan nilai RSSI mencapai -94 dBm tidak dapat diterima sama sekali.

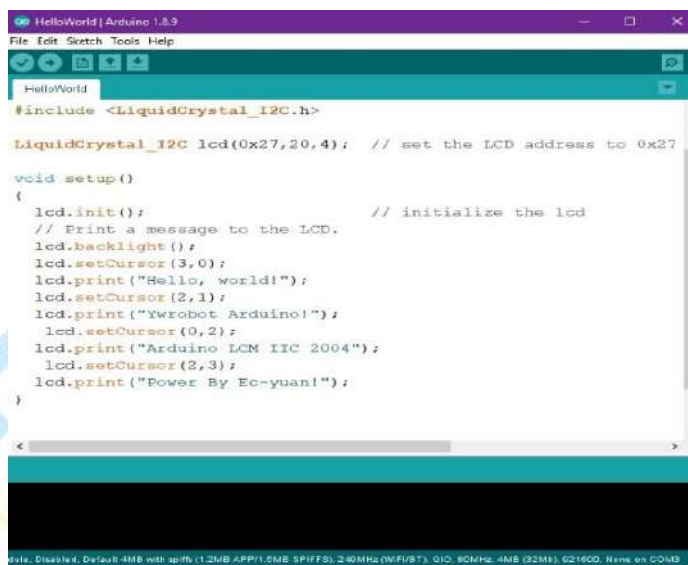
Adapun pengujian pengiriman data dengan kondisi tertutup seperti ruangan, bangunan dan pohon-pohonan juga sangat mempengaruhi pengiriman data, dimana pengujian dengan jarak 30 m dengan nilai RSSI -91 dBm mengalami keberhasilan yang baik. Sedangkan untuk pengiriman diatas jarak 35 m dengan nilai RSSI mencapai -94 dBm mengalami kegagalan pengiriman data.

Berdasarkan pengujian jarak dari ESP-NOW didapatkan pula nilai RSSI dari perangkat *slave*. Sesuai dengan hasil pengujian, didapatkan apabila jarak pengiriman dari perangkat *master* ke *slave* tidak jauh maka nilai RSSI yang didapatkan besar, namun sebaliknya apabila jarak pengirimannya jauh maka nilai RSSI yang didapatkan kecil. Pada pengujian ESP-NOW didapatkan apabila nilai RSSI di bawah -94 dBm maka status data terkirim. Namun apabila nilai RSSI di atas -94 dBm maka status data tidak terkirim.

2. Pengujian LCD 20x4

Pengujian LCD 20x4 terdiri dari pin GND, VCC, SDA dan SCL lalu semua pin dihubungkan ke ESP 32. Pengecekan dilakukan dengan menggunakan *library*

example *Liquid Cristal I2C* pada Arduino IDE berikut. Program LCD dapat dilihat pada Gambar 22. Adapun hasil uji LCD dapat dilihat pada Gambar 23.



```

HelloWorld | Arduino 1.8.9
File Edit Sketch Tools Help

HelloWorld
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,20,4); // set the LCD address to 0x27

void setup()
{
  lcd.init(); // initialize the lcd
  // Print a message to the LCD.
  lcd.backlight();
  lcd.setCursor(3,0);
  lcd.print("Hello, world!");
  lcd.setCursor(2,1);
  lcd.print("Ywrobot Arduino!");
  lcd.setCursor(0,2);
  lcd.print("Arduino LCM IIC 2004");
  lcd.setCursor(2,3);
  lcd.print("Power By Ec-yuan!");
}

```

Gambar 22. Program pengujian LCD 20x4

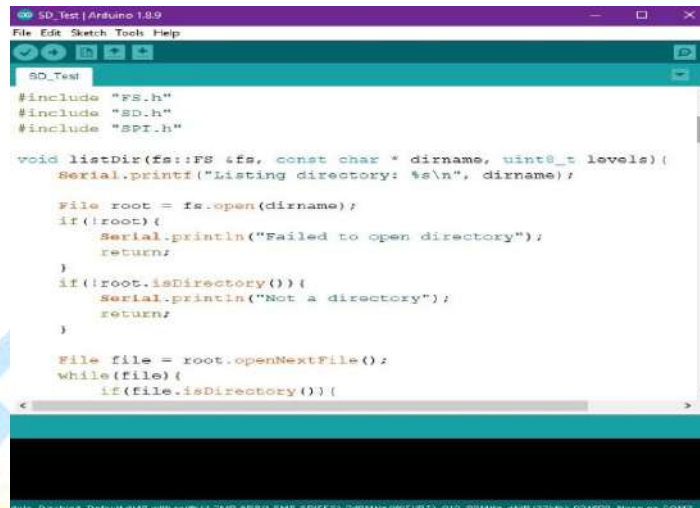


Gambar 23. Hasil uji LCD 20x4

3. Pengujian Micro SD Card

Pengujian modul *micro SD* dilakukan untuk mengetahui modul berfungsi dan dapat membaca *micro SD* beserta isinya. Pengujian dilakukan dengan menghubungkan pin modul *micro SD* ke *board ESP 32*. Pengujian ini menggunakan *firmware SD test* pada *example library SD* yang ada di *software*

arduino IDE dapat dilihat pada Gambar 24. Hasil pengujian micro SD dapat dilihat pada Gambar 25.



```

SD_Test | Arduino 1.9.9
File Edit Sketch Tools Help
SD_Test
#include "FS.h"
#include "SD.h"
#include "SPI.h"

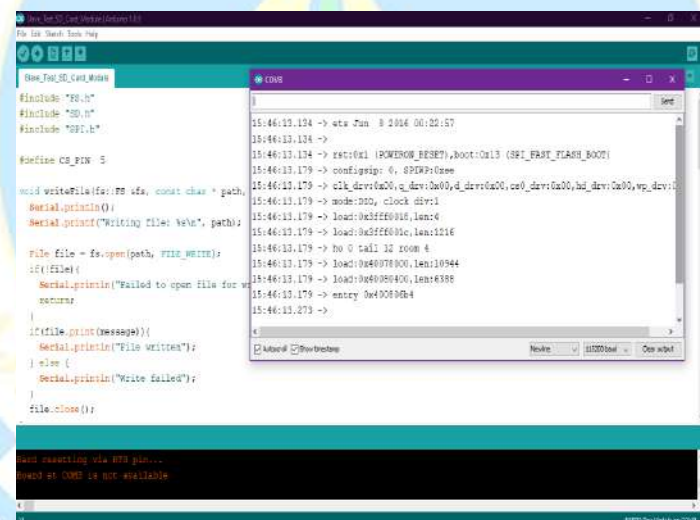
void listDir(fs::FS &fs, const char * dirname, uint8_t levels){
  Serial.printf("Listing directory: %s\n", dirname);

  File root = fs.open(dirname);
  if(!root){
    Serial.println("Failed to open directory");
    return;
  }
  if(!root.isDirectory()){
    Serial.println("Not a directory");
    return;
  }

  File file = root.openNextFile();
  while(file){
    if(file.isDirectory()){

```

Gambar 24. Program Pengujian SD card



```

SD_Test | Arduino 1.9.9
File Edit Sketch Tools Help
SD_Test | Serial Monitor
#include "FS.h"
#include "SD.h"
#include "SPI.h"

#define CS_PIN 5

void writeFile(fs::FS &fs, const char * path,
  Serial.println();
  Serial.print("Writing file: %s\n", path);

  File file = fs.open(path, FILE_WRITE);
  if(!file){
    Serial.println("Failed to open file for writing");
    return;
  }
  if(!file.print(message)){
    Serial.println("File written");
  } else {
    Serial.println("Write failed");
  }
  file.close();

  Serial.println("File written");
}

void setup() {
  Serial.begin(9600);
  while (!Serial) continue;

  Serial.println("Initializing the SD card...");
  if (!SD.begin(4)) {
    Serial.println("SD card initialization failed!");
    return;
  }
  Serial.println("SD card initialized successfully.");
}

void loop() {
  listDir(SD, "/", 0);
  delay(5000);
}

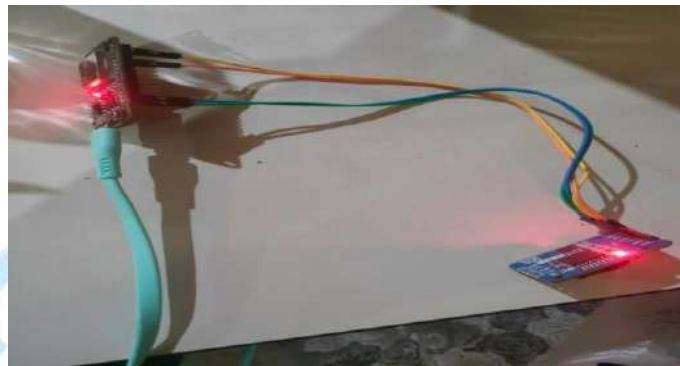
```

Gambar 25. Hasil pengujian micro SD

4. Pengujian RTC DS3231

RTC DS3231 terdiri dari pin GND, VCC, SCL dan SDA. Pada RTC DS3231 dilakukan pengecekan alamat I2C dari RTC DS3231 dengan menggunakan I2C Scanner. Pengujian RTC DS3231 dapat dilakukan dengan

menghubungkan pin RTC dengan *board* ESP 32 yang dapat dilihat pada Gambar 26. Selanjutnya adalah tampilan jam dan tanggal yang dapat dilihat pada Gambar 27.



Gambar 26. Pengujian RTC dengan board ESP32

```

// Date and time functions using a DS1307 RTC module
#include "RTClib.h"
#include <liquidcrystal_i2c.h>
#include <Wire.h>

#define Buzzer_PIN 4
LiquidCrystal_I2C lcd(0x27, 20, 4);
RTC_DS1307 rtc;

void setup() {
  Serial.begin(115200);
  Serial.println();
  delay(3000); // wait for module open
  pinMode(Buzzer_PIN, OUTPUT);
  // initialize LCD
}

void loop() {
  // Get the current time
  RTCDateTime now = rtc.now();
  Serial.print(now.toString());
  Serial.println();
  digitalWrite(Buzzer_PIN, HIGH);
  delay(100);
  digitalWrite(Buzzer_PIN, LOW);
  delay(100);
}

```

Gambar 27. Hasil tampilan jam dan tanggal pada serial

5. Pengujian Sensor

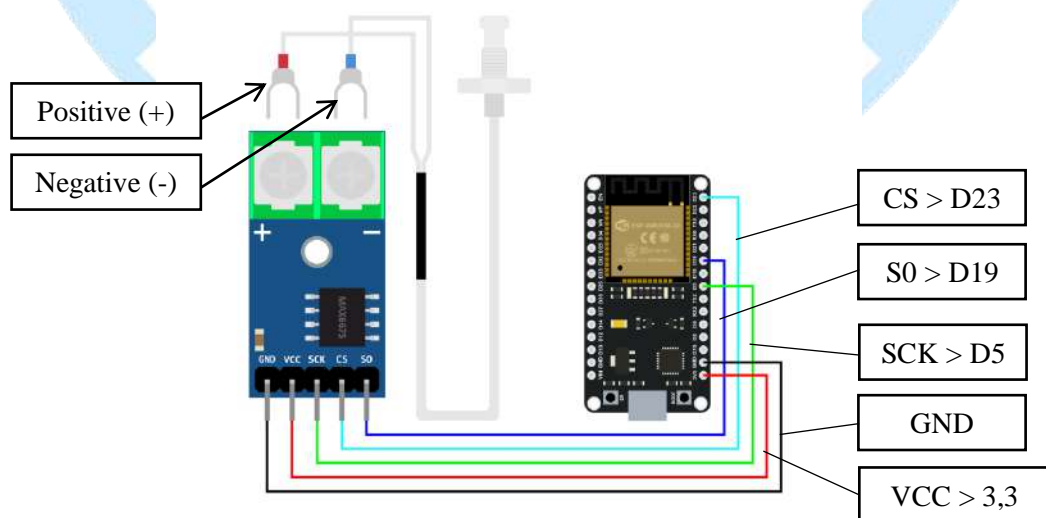
Pengujian sensor merupakan pengujian pengambilan data setiap sensor yang digunakan menggunakan mikrokontroler ESP32 dengan cara menjalankan perintah program yang telah dirancang untuk masing-masing sensor guna mengetahui nilai yang didapatkan dan hasil perbandingan nilai yang didapatkan

dari alat ukur *standard*. Besaran perbandingan tersebut nantinya akan dinotasikan ke besaran *persentase* perselisihan yang dihasilkan dan digunakan sebagai persamaan kalibrasi sensor. Pengujian sensor terbagi dua yaitu:

a. Pengujian Sensor Suhu

Pengujian sensor suhu dilakukan guna mengetahui selisih antara sensor suhu dan alat pengukur suhu standar *thermometer* digital yang digunakan sebagai kalibrator. Kalibrator digunakan sebagai pembandingan alat ukur tersebut akan menjadi acuan pada penelitian ini.

Pengujian sensor suhu dilakukan dengan menguji setiap sampel terhadap suhu yaitu sebanyak 5 jenis sampel suhu yaitu suhu air panas, suhu lilin, suhu kompor, suhu matahari dan suhu mesin motor. Proses pengujian sampel tersebut untuk mendapatkan nilai suhu dari masing-masing sampel suhu yang dihasilkan oleh Thermocopel tipe K, data yang dikirimkan yaitu data analog, data tersebut diolah modul Max 6675 menjadi data digital, lalu dikirim ke ESP 32. Berikut adalah skematik rangkaian dapat dilihat pada Gambar 28.



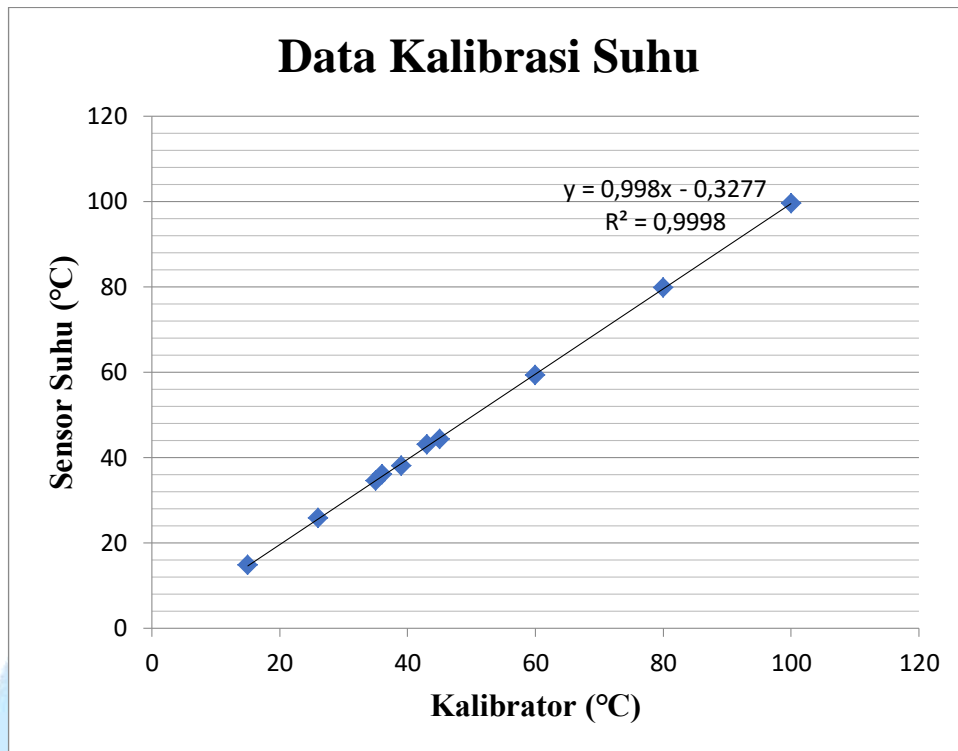
Gambar 28. Rangkaian skematik sensor suhu

Dari pola *scatter* masing-masing nilai suhu dari inputan digital sampel suhu diperoleh pola metode regresi. Kemudian, uji sampel dilakukan sebanyak 10 kali pengujian yang mana 1 kali pengujian berlangsung selama ± 2 menit untuk mendapatkan nilai uji sampel yang mendekati nilai kalibrator dan terukur stabil. Adapun program kalibrasi yang digunakan dapat dilihat pada Lampiran 8.

Hasil perbandingan antara sensor suhu dan hasil yang diperoleh kalibrator ketika diuji dengan sampel suhu panas dan dingin dapat dilihat Tabel 6. Persamaan dan garis regresi yang didapatkan dari hasil pengujian ditampilkan pada Gambar 29. Adapun perhitungan nilai regresi dapat dilihat pada Lampiran 9.

Tabel 6. Hasil pengujian sensor suhu

No	Kalibrator (°C)	Sensor (°C)	Error (%)	Akurasi (%)
1.	15	14,75	1,67	98,33
2.	26	25,75	0,83	99,85
3.	35	34,50	1,43	98,57
4.	36	36	0	100
5.	39	38	2,56	97,59
6.	43	43	0	100
7.	45	44,25	1,66	98,33
8.	60	59,25	1,25	98,00
9.	80	79,75	0,31	99,75
10	100	99,50	0,5	99,50



Gambar 29. Persamaan dan garis regresi kalibrasi sensor suhu

Persamaan dan garis regresi tersebut menunjukkan hubungan nilai sensor suhu dan kalibrator dengan hasil koefisien determinasi (R^2) adalah 0,9998. Hal ini menunjukkan bahwa hubungan tersebut mempunyai garis regresi yang baik dikarenakan koefisien determinasi mendekati 1 (Basuki, 2017).

Lalu untuk mengetahui tingkat kesalahan dan tingkat akurasi sensor suhu digunakan persamaan 1 dan persamaan 2 sebagai berikut :

Untuk sampel suhu = 15 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{15,00 - 14,75}{15,00} \right) \right| \times 100\% \\ &= 1.67 \% \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 1.67 \% \\ &= 98.33 \% \end{aligned}$$

Untuk sampel suhu = 26 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{26,00 - 25,75}{26,00} \right) \right| \times 100\% \\ &= 0,96\% \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 0,96\% \\ &= 99,85\% \end{aligned}$$

Untuk sampel suhu = 35 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{35,00 - 34,50}{35,00} \right) \right| \times 100\% \\ &= 1,43\% \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 1,43\% \\ &= 98,57\% \end{aligned}$$

Untuk sampel suhu = 36 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{36,00 - 36,00}{36,00} \right) \right| \times 100\% \\ &= 0 \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 0\% \\ &= 100\% \end{aligned}$$

Untuk sampel suhu = 39 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{39,00 - 38,00}{39,00} \right) \right| \times 100\% \\ &= 2,56\% \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 2,56\% \\ &= 97,43\% \end{aligned}$$

Untuk sampel suhu = 43 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{43,00 - 43,00}{43,00} \right) \right| \times 100\% \\ &= 0 \end{aligned}$$

$$\text{Akurasi} = 100\% - 0\%$$

$$= 100\%$$

Untuk sampel suhu = 45 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{45,00 - 44,25}{45,00} \right) \right| \times 100\% \\ &= 1,67\% \end{aligned}$$

$$\text{Akurasi} = 100\% - 1,67\%$$

$$= 98,33\%$$

Untuk sampel suhu = 60 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{60,00 - 59,25}{60,00} \right) \right| \times 100\% \\ &= 1,25\% \end{aligned}$$

$$\text{Akurasi} = 100\% - 1,25\%$$

$$= 98,00\%$$

Untuk sampel suhu = 80 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{80,00 - 79,75}{80,00} \right) \right| \times 100\% \\ &= 0,31\% \end{aligned}$$

$$\text{Akurasi} = 100\% - 0,31\%$$

$$= 99,75\%$$

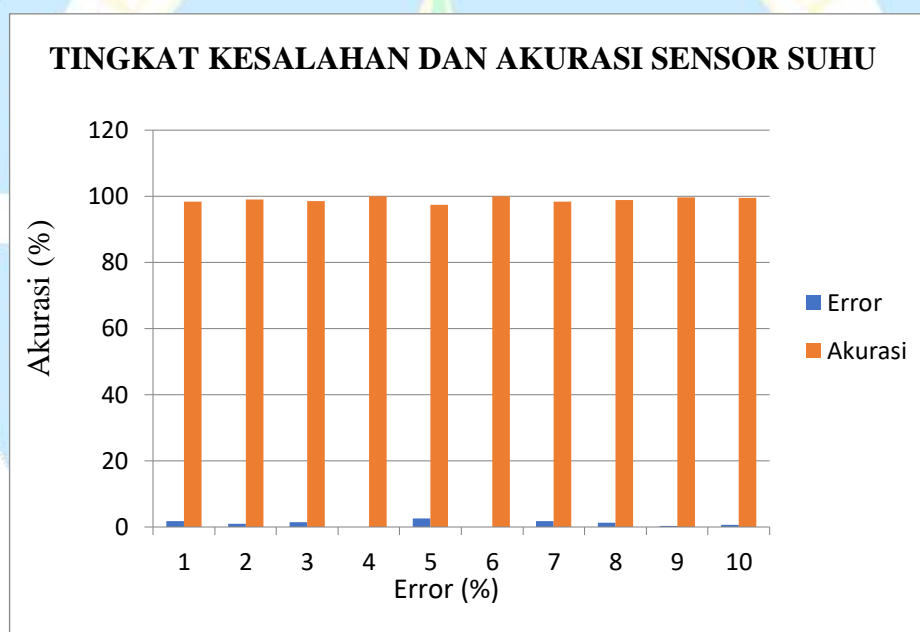
Untuk sampel suhu = 100 °C

$$\begin{aligned} \text{Error} &= \left| \left(\frac{100,00 - 99,50}{100,00} \right) \right| \times 100\% \\ &= 0,5\% \end{aligned}$$

$$\text{Akurasi} = 100\% - 0,5\%$$

$$= 99,00\%$$

Berdasarkan perhitungan tersebut, didapatkan data tingkat kesalahan sensor suhu memiliki nilai kesalahan tertinggi pada sampel suhu 39 °C sebesar 2,56 % dan tingkat kesalahan terendah pada sampel suhu 36 °C dan suhu 43 °C sebesar 0 %. Data tingkat akurasi tertinggi memiliki nilai akurasi pada suhu 36 °C dan suhu 43 °C sebesar 100% dan tingkat akurasi terendah pada suhu 39 °C sebesar 97,59 %. Berdasarkan perhitungan tersebut juga diketahui bahwa rata-rata *error* adalah sebesar 1 %. Hal itu menunjukkan bahwa sensor dapat bekerja dengan baik karena rata-rata tingkat kesalahan yang didapatkan mendekati 0 (Munir, 2003). Data tingkat kesalahan dan tingkat akurasi sensor suhu dapat dilihat pada Gambar 30.



Gambar 30. Tingkat kesalahan dan akurasi suhu

b. Pengujian Sensor Jarak

Pengujian sensor jarak dilakukan untuk mengetahui selisih antara sensor jarak dan alat pengukur meteran yang digunakan sebagai kalibrator. Kalibrator digunakan sebagai pembanding pengukuran sensor jarak. Sehingga, hasil dari kedua perbandingan alat ukur tersebut akan menjadi acuan pada penelitian ini.

Adapun data yang dikirimkan oleh sensor jarak berupa data digital. Skematik rangkaian sensor jarak dapat dilihat pada Gambar 31.

Proses pengujian jarak atau tinggi yang dapat diukur oleh sensor ultrasonik A02YYUW ini berkisar 3 cm sampai 450 cm dapat dilihat pada Gambar 32. Keluaran sensor ultrasonik ini sebagai masukan bagi mikrokontroler berupa data digital yang akan diproses menjadi nilai jarak atau tinggi. Adapun program kalibrasi sensor jarak dapat dilihat pada Lampiran 12.



Gambar 31. Skematik rangkaian sensor jarak



Gambar 32. Pengujian sensor ultrasonik A02YYUW terhadap objek

Perbandingan yang dilakukan dalam pengukuran yang dihasilkan oleh sensor Ultrasonik dan Meteran dari jarak 3 Cm sampai 450 Cm untuk mendapatkan kesalahan dan akurasi dari sensor dapat dilihat pada Tabel 7.

Tabel 7.Hasil pengujian sensor jarak

No	Kalibrator (Cm)	Sensor (Cm)	Error (%)	Akurasi (%)
1.	10	9,80	2	98
2.	50	50,20	0,4	99,6
3.	80	79,00	1,25	98,75
4.	110	109,25	0,68	99,32
5.	140	139,80	0,14	99,86
6.	200	199,30	0,35	99,65
7.	240	239,50	0,21	99,79
8.	280	279,60	0,14	99,86
9.	400	386,80	3,3	96,7
10	450	449,80	0,04	99,96

Adapun cara menghitung *error* dan akurasi yang didapatkan dari perbandingan pengukuran antara meteran dengan sensor ultrasonik A02YYUW dapat dihitung berdasarkan rumus berikut :

Untuk sampel jarak = 10 Cm

$$Error = \left| \left(\frac{10 - 9,80}{10} \right) \right| \times 100\%$$

$$= 2 \%$$

$$Akurasi = 100\% - 2 \%$$

$$= 98 \%$$

Untuk sampel jarak = 50 Cm

$$\begin{aligned} \text{Error} &= \left| \left(\frac{50 - 49,80}{50} \right) \right| \times 100\% \\ &= -0,4 \% \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 0,4 \% \\ &= 100 \% \end{aligned}$$

Untuk sampel jarak = 80 Cm

$$\begin{aligned} \text{Error} &= \left| \left(\frac{80 - 79}{80} \right) \right| \times 100\% \\ &= 1,25 \% \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 1.25 \% \\ &= 98.75 \% \end{aligned}$$

Untuk sampel jarak = 110 Cm

$$\begin{aligned} \text{Error} &= \left| \left(\frac{110 - 109,25}{110,00} \right) \right| \times 100\% \\ &= 0,68 \% \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 0,68\% \\ &= 99,32 \% \end{aligned}$$

Untuk sampel jarak = 140 Cm

$$\begin{aligned} \text{Error} &= \left| \left(\frac{140 - 139,80}{140} \right) \right| \times 100\% \\ &= 0,14 \% \end{aligned}$$

$$\begin{aligned} \text{Akurasi} &= 100\% - 0,14\% \\ &= 99,86 \% \end{aligned}$$

Untuk sampel jarak = 200 Cm

$$\begin{aligned} \text{Error} &= \left| \left(\frac{200 - 199,30}{200} \right) \right| \times 100\% \\ &= 0,35 \% \end{aligned}$$

$$\text{Akurasi} = 100\% - 0,35 \%$$

$$= 99,65 \%$$

Untuk sampel jarak = 240 Cm

$$\begin{aligned} \text{Error} &= \left| \left(\frac{240 - 239,50}{240} \right) \right| \times 100\% \\ &= 0,21 \% \end{aligned}$$

$$\text{Akurasi} = 100\% - 0,21\%$$

$$= 99,79\%$$

Untuk sampel jarak = 280 Cm

$$\begin{aligned} \text{Error} &= \left| \left(\frac{280 - 279,60}{280} \right) \right| \times 100\% \\ &= 0,14 \% \end{aligned}$$

$$\text{Akurasi} = 100\% - 0,14\%$$

$$= 99,86 \%$$

Untuk sampel jarak = 400 Cm

$$\begin{aligned} \text{Error} &= \left| \left(\frac{400 - 399,80}{400} \right) \right| \times 100\% \\ &= 3,3 \% \end{aligned}$$

$$\text{Akurasi} = 100\% - 3,3\%$$

$$= 96,7 \%$$

Untuk sampel jarak = 450 Cm

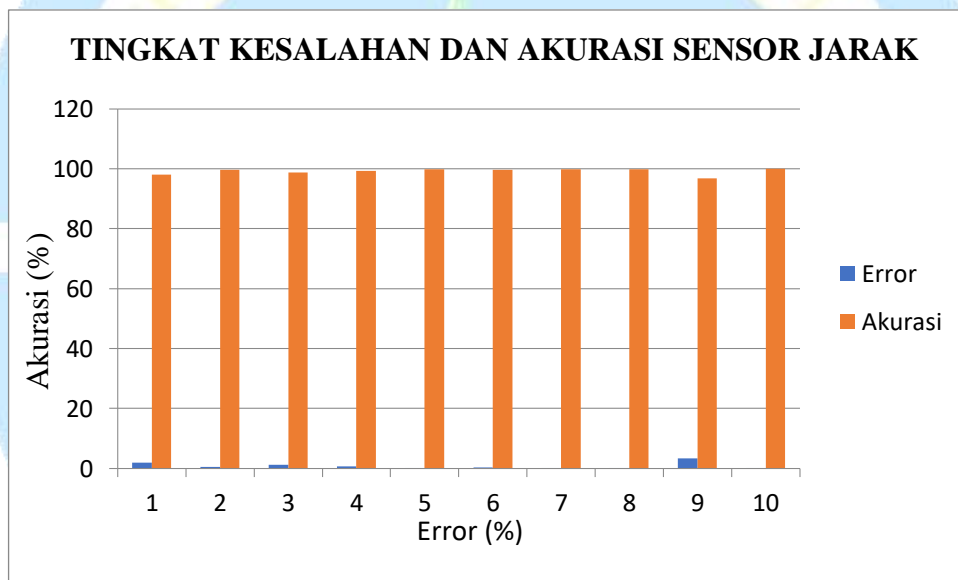
$$Error = \left| \left(\frac{450 - 449,80}{450} \right) \right| \times 100\%$$

$$= 0,04 \%$$

$$Akurasi = 100\% - 0,04\%$$

$$= 99,95\%$$

Melalui persamaan tersebut, perbandingan nilai jarak yang dihasilkan diolah dan diperoleh tingkat kesalahan dan tingkat akurasi sensor ultrasonik A02YYUW dengan meteran dapat dilihat pada Gambar 33.



Gambar 33. Tingkat *error* dan akurasi jarak

Berdasarkan karakteristik sensor ultrasonik A02YYUW dapat menghitung dengan rentang jarak 3 cm – 450 cm, sedangkan dari data hasil pengukuran didapat bahwa untuk jarak 3 cm menghasilkan persentase pembacaan *error* dan selebihnya hanya terjadi persentase kesalahan yang kecil. Hal ini menandakan bahwa sensor ultrasonik bekerja dengan baik. Artinya sensor hanya dapat bekerja dengan jarak minimal 3 cm dan maksimal 450 cm. Umumnya semakin jauh jarak

yang diukur semakin kecil kesalahan. Perbedaan jarak hasil pengujian dengan jarak sesungguhnya dapat disebabkan oleh adanya *kebisingan*. Hal itu sesuai dengan penelitian yang dilakukan Supriyadi (2011), bahwa kebisingan dapat mempengaruhi hasil data yang dikirimkan.

Berdasarkan perhitungan tersebut, didapatkan data tingkat kesalahan sensor jarak memiliki nilai kesalahan tertinggi pada sampel jarak 400 cm sebesar 3,3 % dan tingkat kesalahan terendah pada sampel jarak 450 cm sebesar 0,04 %. Data tingkat akurasi memiliki nilai akurasi tertinggi pada jarak 450 cm sebesar 99,95 % dan tingkat akurasi terendah pada jarak 400 cm sebesar 96,7%.

1. Pengujian Keseluruhan Sistem

Pengujian perangkat secara keseluruhan dilakukan guna menampilkan nilai suhu air pendingin dan kapasitas bahan bakar solar tangki, nilai suhu dan kapasitas bahan bakar solar dalam tangki pada Pembangkit Listrik Tenaga Diesel (PLTD) Pulau Pangkil nantinya dapat dipantau pada LCD agar dapat diakses oleh pengguna. Pengujian dilakukan di Pembangkit Listrik Tenaga Diesel (PLTD) Pulau Pangkil untuk menjadi landasan dalam pengujian keseluruhan sistem nantinya. Lokasi pengujian perangkat keseluruhan dapat dilihat pada Gambar 34.



Gambar 34. Lokasi pengujian keseluruhan sistem

Pada pengujian sistem secara keseluruhan, perangkat *master* 1 diletakkan di mesin PLTD Pulau Pangkil untuk mengukur suhu sedangkan untuk *master* 2 diletakkan di tangki solar PLTD Pulau Pangkil untuk mengukur kapasitas bahan bakar dalam tangki kedua *master* tersebut berfungsi mengirim data ke *slave*. Perangkat *slave* yang berfungsi untuk menerima data dan menampilkan pada LCD diletakkan di kantor CS PLTD Pulau Pangkil. Perangkat *master* 1 dapat dilihat pada Gambar 35. Rangkaian perangkat *master* 1 dapat dilihat pada Gambar 36 dan perangkat *master* 2 dapat dilihat pada Gambar 37. Rangkaian perangkat *master* 2 dapat dilihat pada Gambar 38 Sedangkan perangkat *slave* dapat dilihat pada Gambar 39 dan rangkaian perangkat *slave* dapat dilihat pada Gambar 40.



Gambar 35. Perangkat *Master* 1



Gambar 36. Rangkaian Perangkat *Master* 1



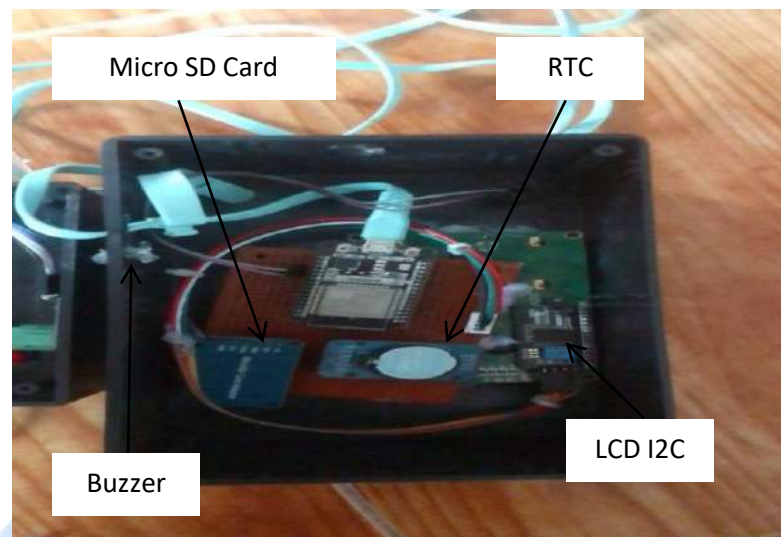
Gambar 37. Perangkat *master* 2



Gambar 38. Rangkaian Perangkat *Master* 2



Gambar 39. Perangkat *slave* di ruangan CS PLTD Pulau Pangkil



Gambar 40. Rangkaian Perangkat Pada Slave

Pengujian dilakukan dalam kurun waktu 1x12 jam dengan hasil yang akan ditampilkan di LCD. Hasil pengujian perangkat secara keseluruhan dapat dilihat pada Gambar 41.



Gambar 41. Hasil pengujian perangkat secara keseluruhan

B. Pengembangan *Firmware*

Firmware (perangkat lunak) yang digunakan pada penelitian ini menggunakan aplikasi Arduino IDE. *Firmware* pada Arduino IDE berisi *library*, *void setup*, *void loop* dan variabel-variabel yang digunakan. *Firmware* dimulai dengan melakukan inisialisasi *library* yang digunakan dalam *instrument* ini. Daftar *library* yang digunakan pada *master 1*, *master 2* dan *slave* dapat dilihat pada Gambar 42.

```

2 #include <esp_now.h>
3 #include <esp_wifi.h>
4 #include <WiFi.h>
5
6 #include "max6675.h"

```

(a)

```

2 #include <esp_now.h>
3 #include <esp_wifi.h>
4 #include <WiFi.h>

```

(b)

```

2 #include <esp_now.h>
3 #include <esp_wifi.h>
4 #include <WiFi.h>
5
6 #include "RTCLib.h"
7 #include <LiquidCrystal_I2C.h>
8 #include <Wire.h>
9
10 #include "FS.h"
11 #include "SD.h"
12 #include "SPI.h"

```

(c)

Gambar 42. Program *Library* dari (a) *Master 1*, (b) *Master 2* dan (c) *Slave*

Selanjutnya menginisialisasi pin komponen dan deklarasi nama variabel. *Master 1* sensor Thermocopel tipe-K menggunakan pin SCK terhubung ke pin D5, CS terhubung ke pin D23 dan SO terhubung ke pin D19 dan *master 2* sensor Ultrasonik A02YYUW menggunakan pin RX terhubung ke pin RX 2 dan TX terhubung ke pin TX 2. Adapun program inialisasi variabel untuk sensor dapat dilihat pada Gambar 43. Sedangkan *slave* terdiri beberapa deklarasi variabel diantaranya deklarasi variabel LCD, deklarasi variabel RTC, deklarasi variabel *micro* SD dan deklarasi variabel buzzer. Program inialisasi variabel *slave* dapat dilihat pada Gambar 44 dan deklarasi variabel slave dapat dilihat pada Gambar 45

```

34// Deklarasi variabel untuk alamat pin koneksi I2C01 dengan sensor K-Type Thermocouple.
35 int thermoD0 = 15;
36 int thermoCS = 23;
37 int thermoCLK = 5;

```

(a)

```

28// Deklarasi variabel untuk menerima data hasil pembacaan sensor Ultrasonik A02YYUW.
29 unsigned char data[4]={};
30 float distance_cm;
31 float kapasitas;
32 String kapasitas_Send = "000";

```

(b)

Gambar 43. Inialisasi Variabel Perangkat (a) *Master 1* dan (b) *Master 2*

```

86 //inisialisasi pin micro sd
87 #define CS_PIN      5
88 //inisialisasi pin buzzer
89 #define Buzzer_PIN  4
90
91 // Inisialisasi dan pengaturan LCD.
92 // 0x27 adalah alamat i2c di LCD.
93 // 20  adalah jumlah kolom LCD.
94 // 4   adalah jumlah naris LCD.
95 LiquidCrystal_I2C lcd(0x27,20,4);
96
97 // Inisialisasi RTC_DS3231 sebagai rtc.
98 RTC_DS3231 rtc;
99
100 char month_names[12][4] = {"Jan", "Feb", "Mar", "Apr", "Mei", "Jun", "Jul", "Ags", "Sep", "Okt", "Nov", "Des"};
101

```

Gambar 44. Inisialisasi Variabel pada Perangkat *Slave*

```

25 // Deklarasi variabel untuk millis (millis untuk mengambil data dari RTC dan menampilkan data yang diterima dari master).
26 unsigned long previousMillis_UpdateRTC = 0;
27 const long interval_UpdateRTC = 1000;
28
29 String receive_Temp_Val = "";
30 String receive_Dist_Val = "";
31
32 byte last_Temp_Val_length;
33 byte last_Dist_Val_length;
34
35 String time_now = "";
36 String date_now = "";
37
38 String fileName = "";
39
40 bool saveDataFromMaster_1 = false;
41 bool saveDataFromMaster_2 = false;
42
43 bool Buzzer_ON_Temp = false;
44 bool Buzzer_ON_Dist = false;
45 byte Buzzer_Dur_Temp = 0;
46 byte Buzzer_Dur_Dist = 0;
47
48 bool writeFile_Success = false;
49 int rssi;
..

```

Gambar 45. Deklarasi Variabel Perangkat *Slave*

Adapun fungsi untuk mengirim data pada ESP-NOW *master 1* dan *master 2* dapat dilihat pada Gambar 46. Setelah itu masuk ke fungsi *void setup*, pada fungsi *void setup* komunikasi serial menggunakan baudrate 115200 untuk melihat keluaran data pada serial monitor, fungsi yang digunakan yaitu “*Serial.begin*

(115200)". Adapun fungsi *void setup* pada *master 1* dapat dilihat pada Gambar 47, *master 2* dapat dilihat pada Gambar 48 dan *slave* dapat dilihat pada Gambar 49.

```

56 // _____ Subrutin untuk mengirimkan data.
57 void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
58   Serial.print("\r\nStatus pengiriman data terakhir : \t");
59   Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Delivery Success" : "Delivery Fail");
60   Serial.println("-----");
61 }

```

Gambar 46. Fungsi Mengirim Data pada *Master 1* dan *Master 2*

```

84 Serial.begin(115200);
85 delay(1000);
86
87 // Tetapkan perangkat pada mode "Wi-Fi Station".
88 Serial.println();
89 Serial.println("Atur wifi ke mode Wi-Fi Station.");
90 WiFi.mode(WIFI_STA);
91
92 //-----
93 int cur_WiFiChannel = WiFi.channel();
94 if (cur_WiFiChannel != CHANNEL) {
95   esp_wifi_set_promiscuous(true);
96   esp_wifi_set_channel(CHANNEL, WIFI_SECOND_CHAN_NONE);
97   esp_wifi_set_promiscuous(false);
98 }
99 //-----
100 // Mendaftarkan fungsi "callback function OnDataSent" yang akan dipanggil saat pengiriman data.
101 esp_now_register_send_cb(OnDataSent);
102
103 // Register Peer.
104 esp_now_peer_info_t peerInfo;
105 memcpy(peerInfo.peer_addr, broadcastAddress, 6);
106 peerInfo.encrypt = false;
107
108 Serial.println();
109 Serial.print("WiFi Channel : ");
110 Serial.println(WiFi.channel());
111
112 Serial.println();
113 Serial.print("Mac Address : ");
114 Serial.println(WiFi.macAddress());
115
116 //-----Init ESP-NOW.
117 Serial.println();

```

Gambar 47. Fungsi *Void Setup* Pada *Master 1*

```

92 Serial.begin(115200);
93 // Pengaturan Hardware Serial_2 ESP32. Hardware Serial_2 digunakan untuk berkomunikasi dengan sensor Ultrasonic A02YYUW.
94 Serial2.begin(9600, SERIAL_BN1, RMD2, TXD2);
95 delay(1000);
96
97 // Tetapkan perangkat pada mode "Wi-Fi Station".
98 Serial.println();
99 Serial.println("Atur wifi ke mode Wi-Fi Station.");
100 WiFi.mode(WIFI_STA);
101 int cur_WiFiChannel = WiFi.channel();
102 if (cur_WiFiChannel != CHANNEL) {
103     esp_wifi_set_promiscuous(true);
104     esp_wifi_set_channel(CHANNEL, WIFI_SECOND_CHAN_NONE);
105     esp_wifi_set_promiscuous(false);
106 }
107 //-----
108 // Register Peer.
109 esp_now_peer_info_t peerInfo;
110 memcpy(peerInfo.peer_addr, broadcastAddress, 6);
111 peerInfo.encrypt = false;
112 Serial.println();
113 Serial.print("Wi-Fi Channel : ");
114 Serial.println(WiFi.channel());
115 Serial.println();
116 Serial.print("Mac Address : ");
117 Serial.println(WiFi.macAddress());
118 //-----Init ESP-NOW.
119 Serial.println();
120 Serial.println("-----");
121 Serial.println("Mulai menginisialisasi ESP-NOW...");
122 if (esp_now_init() != ESP_OK) {
123     Serial.println("Error initializing ESP-NOW");
124     Serial.println("Restart ESP32...");
125     Serial.println("-----");

```

Gambar 48. Fungsi *Void Setup* Pada *Master 2*

```

468 Serial.begin(115200);
469 Serial.println();
470 delay(2000);
471
472 pinMode(Busser_PIN, OUTPUT);
473 digitalWrite(Busser_PIN, LOW);
474 for (byte i = 0; i < 4; i++) {
475     digitalWrite(Busser_PIN, !digitalRead(Busser_PIN));
476     delay(1000);
477 }
478 // Menjalankan inisialisasi dan menerapkan pengaturan LCD.
479 lcd.init();
480 // menyalakan LCD backlight.
481 lcd.backlight();
482 delay(1000);
483 //-----Memulai dan memeriksa ketersediaan rtc.
484 lcd.clear();
485 lcd.setCursor(0,0);
486 lcd.print("Memeriksa RTC...");
487 delay(2000);
488
489 //Serial.println();
490 //Serial.println("Memeriksa RTC...");
491 if (!rtc.begin()) {
492     delay(500);
493     lcd.clear();
494     lcd.setCursor(0,0);
495     lcd.print("RTC tidak ditemukan!");
496
497     //Serial.println("RTC tidak ditemukan");
498     //Serial.println("Mohon periksa pemasangan RTC dengan ESP32.");
499     while(1) {
500         delay(100);
501     }

```

Gambar 49. Fungsi *Void Setup* Pada *Slave*

Setelah semua program yang ada di *void setup* berhasil, maka selanjutnya masuk ke fungsi *void loop*. Pada program *void loop* terdapat beberapa fungsi berupa perulangan pengambilan data pada *master 1* yaitu sensor thermocopel tipe-*k* dan *master 2* sensor Ultrasonik A02YYUW. Fungsi *void loop* pada *master 1* dapat dilihat pada Gambar 50, *master 2* dapat dilihat pada Gambar 51. *Slave* terdiri dari RTC DS3231 berfungsi mengambil data waktu pengiriman dari *master 1* dan *master 2*, LCD berfungsi sebagai menampilkan data waktu, tanggal dan data pengukuran dari *master*, sedangkan *micro SD* sebagai penyimpan data tanggal, waktu, *master 1* dan *master 2*. Adapun fungsi *void loop slave* dapat dilihat pada Gambar 52.

```

151 //-----Timer/Millis untuk pembacaan sensor K-Type Thermocouple.
152 // Mengambil nilai suhu dari sensor K-Type Thermocouple MAX6675 setiap 1 detik (1000 milidetik). Lihat variabel interval_K_Type_Thermocouple.
153 unsigned long currentMillis_K_Type_Thermocouple = millis();
154 if (currentMillis_K_Type_Thermocouple - previousMillis_K_Type_Thermocouple >= interval_K_Type_Thermocouple) {
155     previousMillis_K_Type_Thermocouple = currentMillis_K_Type_Thermocouple;
156
157     read_K_Type_Thermocouple();
158 }
159 //-----Timer/Millis untuk mengirimkan data ke penerima/slave.
160 // Mengirimkan data ke penerima/slave setiap 1 menit (60000 milidetik). Lihat variabel interval_Send.
161 unsigned long currentMillis_Send = millis();
162 if (currentMillis_Send - previousMillis_Send >= interval_Send) {
163     previousMillis_Send = currentMillis_Send;
164
165     Serial.println();
166     Serial.println("-----");
167     Serial.print("Thermocouple : ");
168     Serial.print(thermocouple_celcius_Send);
169     Serial.println(" °C");
170     Serial.println("-----");
171
172     Serial.println();
173     Serial.println("----Send data");
174
175     //:::::::::::Atur data/pesan yang akan dikirim.
176     send_Data.ID_Sender = Master_ID;
177     send_Data.Data_Send = thermocouple_celcius_Send;
178     //:::::::::::
179
180     //:::::::::::Kirim data/pesan via ESP-NOW.
181     esp_err_t result = esp_now_send(broadcastAddress, (uint8_t *) &send_Data, sizeof(send_Data));
182
183     if (result == ESP_OK) {
184         Serial.println("Sent with success");

```

Gambar 50. Program *Void Loop* Pada *Master 1*

```

155 //-----Timer/Millis untuk pembacaan sensor Ultrasonic A02YUW.
156 // Mengambil nilai jarak dari sensor Ultrasonic A02YUW setiap 0.1 detik (100 milidetik). Lihat variabel interval_Ultrasonic.
157 unsigned long currentMillis_Ultrasonic = millis();
158 if (currentMillis_Ultrasonic - previousMillis_Ultrasonic >= interval_Ultrasonic) {
159     previousMillis_Ultrasonic = currentMillis_Ultrasonic;
160
161     read_Ultrasonic_A02YUW();
162 }
163 //-----Timer/Millis untuk mengirimkan data ke penerima/slave.
164 // Mengirimkan data ke penerima/slave setiap 10 menit (600000 milidetik). Lihat variabel interval_Send.
165 unsigned long currentMillis_Send = millis();
166 if (currentMillis_Send - previousMillis_Send >= interval_Send) {
167     previousMillis_Send = currentMillis_Send;
168
169     Serial.println();
170     Serial.println("-----");
171     Serial.print("Kapasitas : ");
172     Serial.print(kapasitas_Send);
173     Serial.println(" #");
174     Serial.println("-----");
175     Serial.println();
176     Serial.println("-----Send data");
177     //:-----Atur data/pesan yang akan dikirim.
178     send_Data.ID_Sender = Master_ID;
179     send_Data.Data_Send = kapasitas_Send;
180     //:-----
181     //:-----Kirim data/pesan via ESP-NOW.
182     esp_err_t result = esp_now_send(broadcastAddress, (uint8_t *) &send_Data, sizeof(send_Data));
183     if (result == ESP_OK) {
184         Serial.println("Sent with success");
185     }
186     else {
187         Serial.println("Error sending the data");
188     }
}

```

Gambar 51. Program *Void Loop* Pada *Master 2*

```

597 void loop () {
598     // put your main code here, to run repeatedly:
599     //-----Timer/Millis untuk memperbaharui Jam,Tanggal dan lainnya.
600     unsigned long currentMillis_UpdateRTC = millis();
601     if (currentMillis_UpdateRTC - previousMillis_UpdateRTC >= interval_UpdateRTC) {
602         previousMillis_UpdateRTC = currentMillis_UpdateRTC;
603         // Mengupdate Jam dan Tanggal (Time & date).
604         getTimeDate();
605         //:-----Menampilkan Jam dan Tanggal ke LCD.
606         lcd.setCursor(0, 0);
607         lcd.print("Jam   : ");
608         lcd.print(time_now);
609         lcd.setCursor(0, 1);
610         lcd.print("Tgl   : ");
611         lcd.print(date_now);
612         lcd.setCursor(0, 2);
613         lcd.print("Suhu  :");
614         if (last_Temp_Val_length != receive_Temp_Val.length()) {
615             lcd.setCursor(8, 2);
616             lcd.print("      "); }
617         lcd.setCursor(8, 2);
618         lcd.print(receive_Temp_Val);
619         lcd.print(" ");
620         lcd.print(char(223));
621         lcd.print("C");
622         lcd.setCursor(0, 3);
623         lcd.print("Kapasitas :");
624         if (last_Dist_Val_length != receive_Dist_Val.length()) {
625             lcd.setCursor(12, 3);
626             lcd.print("      "); }
627         lcd.setCursor(12, 3);
628         lcd.print(receive_Dist_Val);
629         lcd.print(" #");
630         Serial.println("rssi\t= " + String(rssi) + " dBm");
}

```

(a)

```

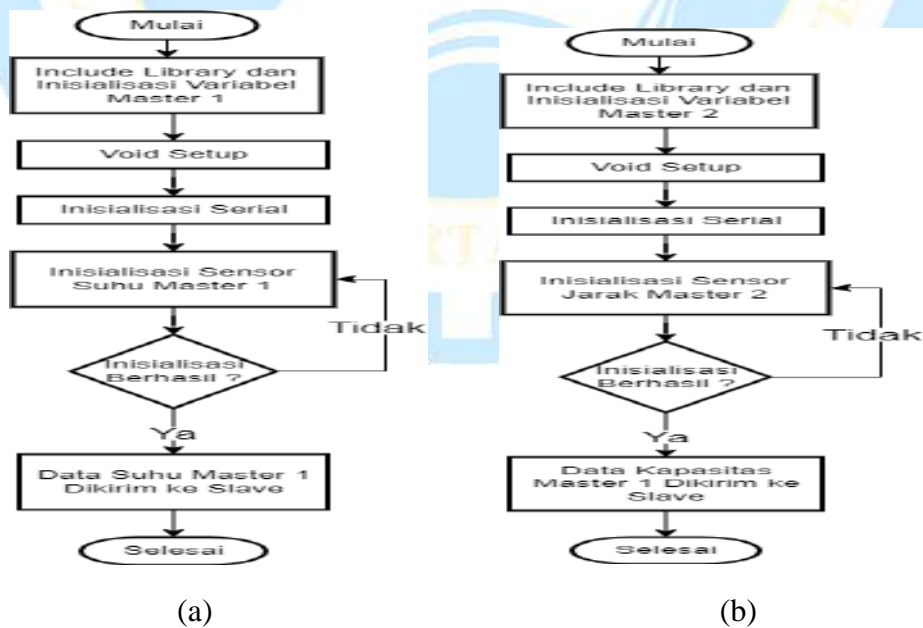
640 // Create a file on the SD card and write the data labels
641 File file = SD.open(fileName);
642
643 if(!file) {
644 //Serial.println();
645 //Serial.println("File doesn't exist");
646 //Serial.println("Creating file...");
647 writeFile(SD, fileName.c_str(), "TANGGAL, JAM, SUHU, KAPASITAS");
648 }
649 else {
650 //Serial.println();
651 //Serial.println("File already exists");
652 }
653 file.close();
654 }
655
656 if (saveDataFromMaster_1 == true) {
657 String savedData = "\r\n" + date_now + "," + time_now + ",Master_1," + receive_Temp_Val + ",C," + rssi + "dBm";
658
659 // Append the data to file
660 appendFile(SD, fileName.c_str(), savedData.c_str());
661
662 saveDataFromMaster_1 = false;
663 }
664
665 if (saveDataFromMaster_2 == true) {
666 String savedData = "\r\n" + date_now + "," + time_now + ",Master_2," + receive_Dist_Val + ",dB," + rssi + "dBm";
667
668 // Append the data to file
669 appendFile(SD, fileName.c_str(), savedData.c_str());
670
671 saveDataFromMaster_2 = false;
672 }
673 }

```

(b)

Gambar 52. Program *Void Loop Slave* (a) LCD dan RTC, (b) *Master 1* dan *Master 2*

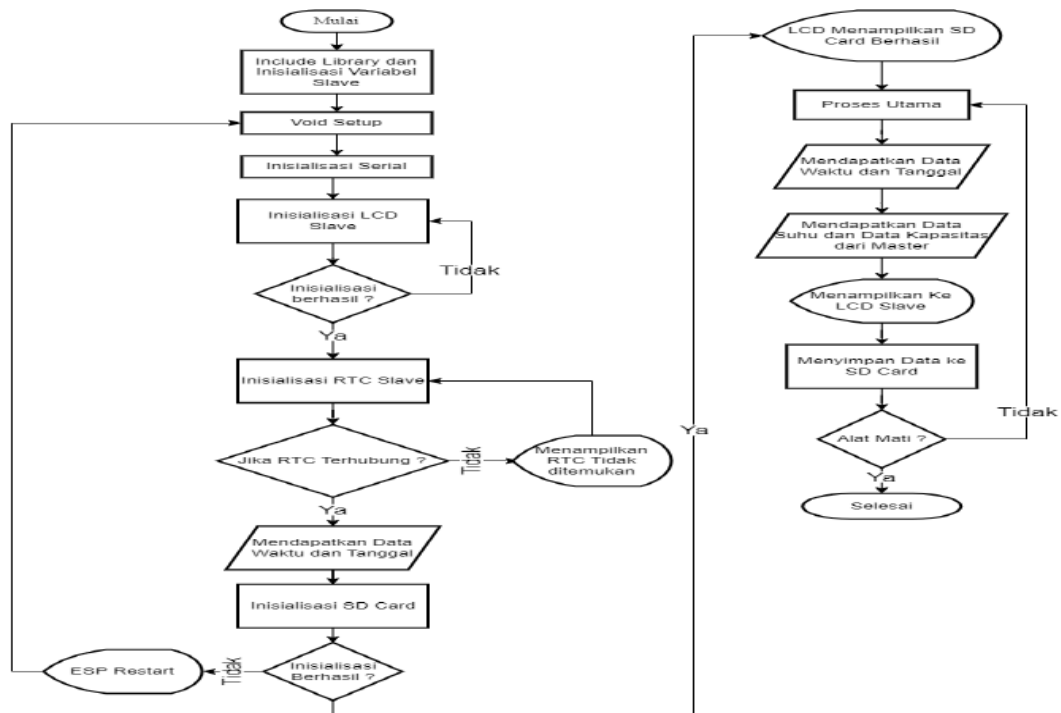
Setelah pengambilan data selesai, maka ESP-NOW *master* dan *slave* mati maka pengambilan data selesai. Diagram alir *firmware* pengukuran suhu air pendingin dan kapasitas bahan bakar dalam tangki, *master 1* dan *master 2* dapat dilihat pada Gambar 53 dan *slave* dapat dilihat pada Gambar 54.



(a)

(b)

Gambar 53. Diagram Alir Pada *Master 1* dan *Master 2*



Gambar 54. Diagram Alir *Firmware* Pengukuran Menggunakan ESP-NOW Pada *Slave*

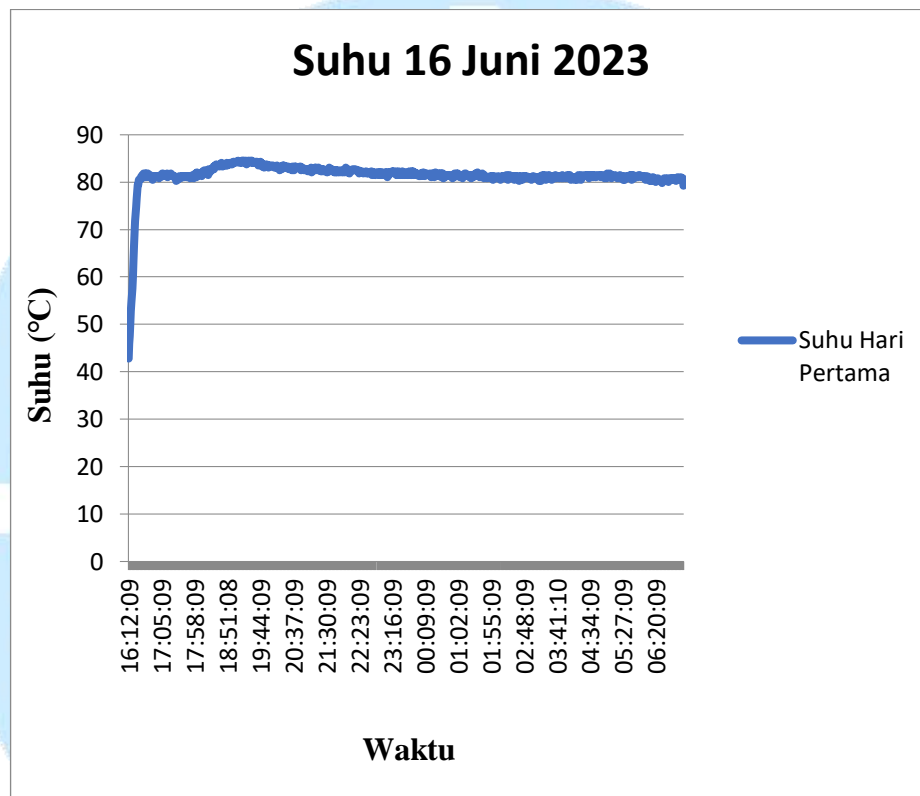
C. Analisis dan Pembahasan

Pengujian sistem secara keseluruhan di PLTD Pulau Pangkil dilakukan dalam setengah hari (1x12 jam) dengan menggunakan *powerbank* 10.000 mAH sebagai *power supply* dari perangkat tersebut. Hal ini dikarenakan pengujian sistem secara keseluruhan dilakukan untuk mengetahui suhu air pendingin dan kapasitas bahan bakar pada tangki PLTD Pulau Pangkil.

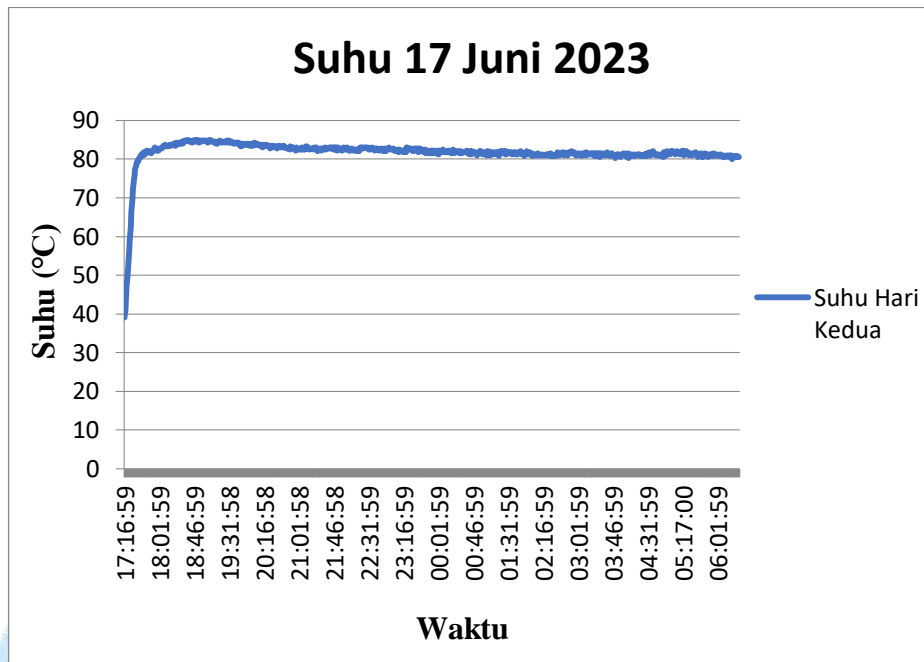
Hasil pengujian sistem secara keseluruhan menunjukkan hasil yang baik. Sebagian besar data dapat dikirim dari perangkat *master* 1 dan *master* 2 selanjutnya diteruskan ke LCD yang ada pada perangkat *slave* agar dapat dipantau oleh pengguna.

Pengujian sistem dilakukan dengan cara mengirimkan data per 1 menit sekali dengan total data suhu sebanyak 1.540 data selama 12 Jam. Hasil dari

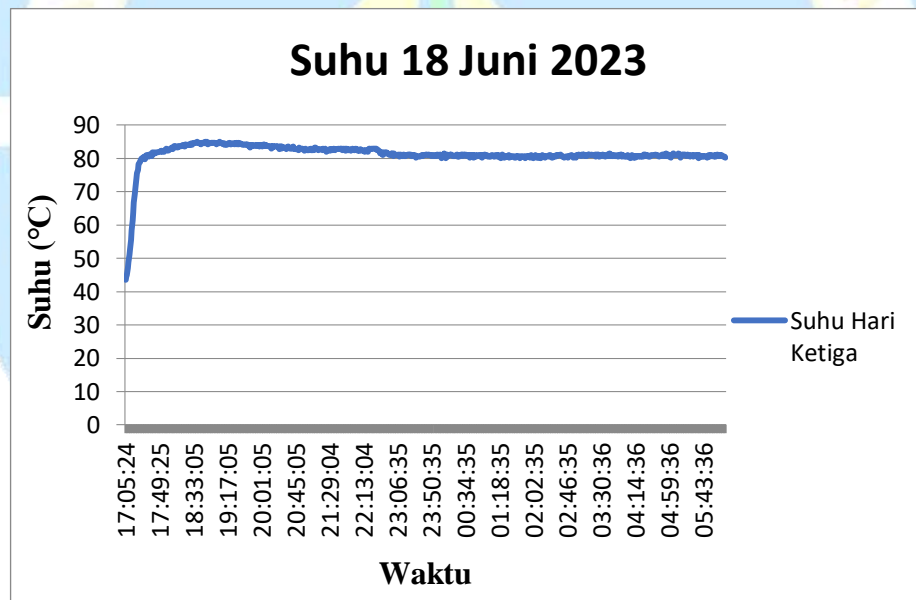
pengujian suhu air pendingin pada mesin PLTD Pulau Pangkil menggunakan sistem keseluruhan pengujian hari pertama dapat dilihat pada Gambar 55, untuk hari kedua dapat dilihat pada Gambar 56 dan untuk hari ketiga dapat dilihat pada Gambar 57. Data suhu dapat dilihat pada Lampiran 2, Lampiran 3 dan Lampiran 4.



Gambar 55. Grafik data suhu air pendingin tanggal 16 Juni 2023



Gambar 56. Grafik data suhu air pendingin tanggal 17 Juni 2023

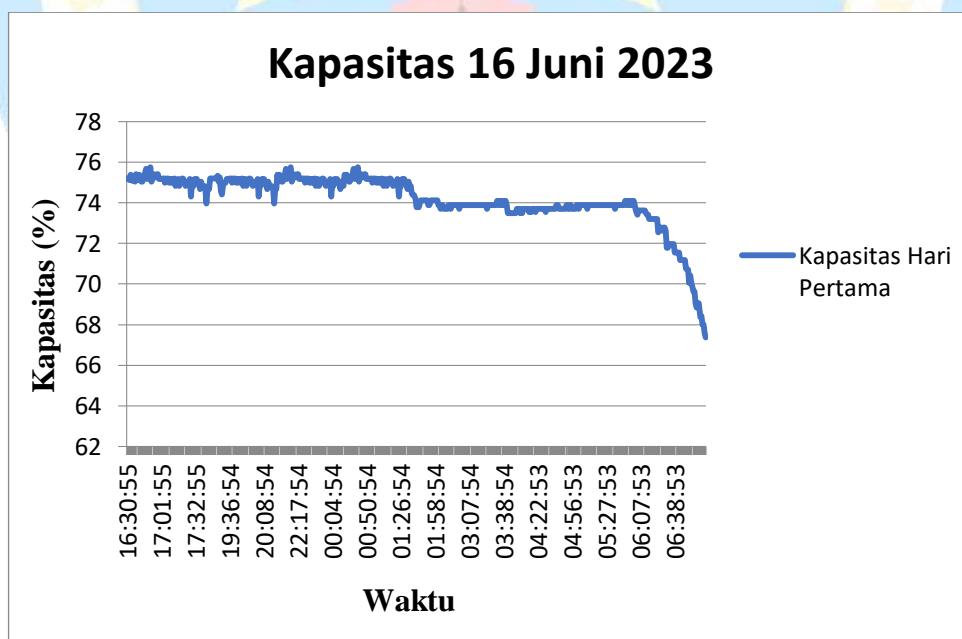


Gambar 57. Grafik data suhu air pendingin tanggal 18 Juni 2023

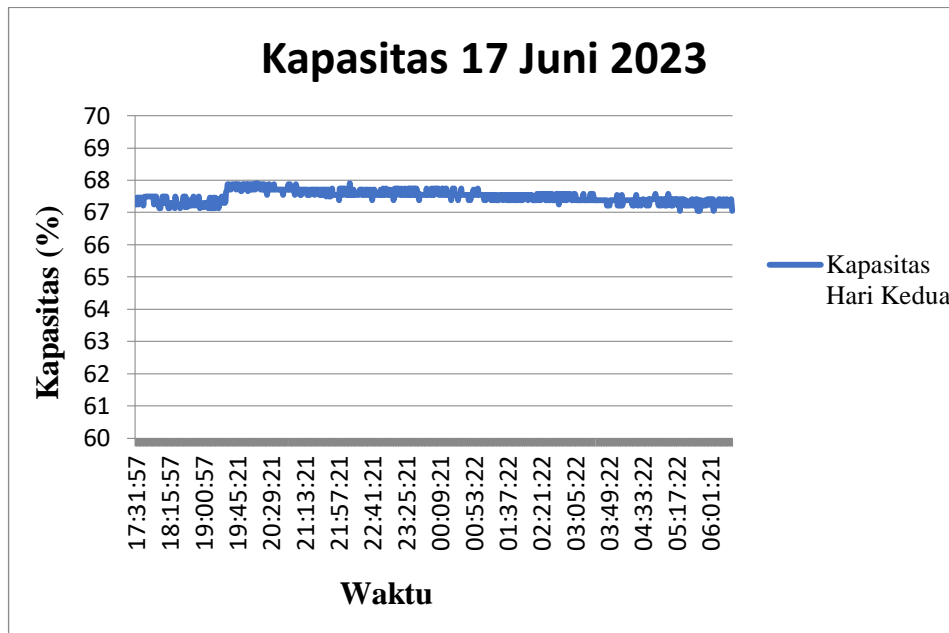
Dari grafik tersebut menghasilkan suhu tertinggi terdapat sebesar 85,00 °C. Hal itu dikarenakan beban pemakaian listrik meningkat. Sedangkan suhu terendah sebesar 39 °C dikarenakan kondisi awal mesin beroperasi. Dari pengujian

tersebut juga diperoleh data rata-rata suhu hari pertama sebesar $81,42\text{ }^{\circ}\text{C}$, suhu hari kedua sebesar $81,76\text{ }^{\circ}\text{C}$ dan suhu hari ketiga sebesar $81,30\text{ }^{\circ}\text{C}$. Hal ini menunjukkan bahwa suhu air pendingin mencapai kinerja ideal pada suhu $50\text{ }^{\circ}\text{C}$ - $83\text{ }^{\circ}\text{C}$ jika suhu melewati batas tersebut maka tenaga mesin dapat berkurang (Hardi *et al.*, 2022). Berdasarkan hal tersebut pengoperasian Pembangkit Listrik Tenaga Diesel (PLTD) Pulau Pangkil masih dalam keadaan normal dalam pengoperasian.

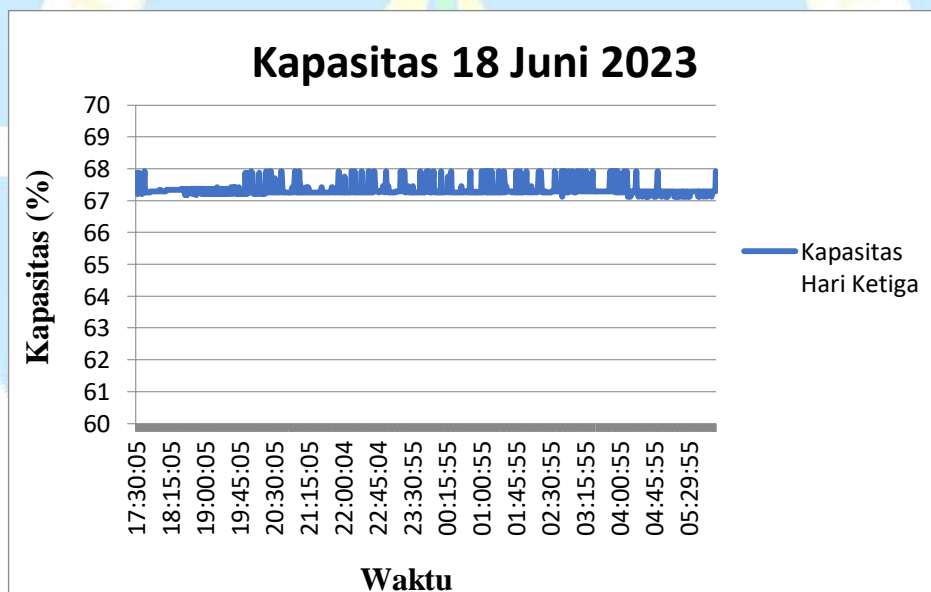
Selain suhu air pendingin, nilai kapasitas solar pada tangki Pembangkit Listrik Tenaga Diesel (PLTD) Pulau pangkil juga berhasil didapatkan dari pengujian. Hasil pengujian data kapasitas solar hari pertama dapat dilihat pada Gambar 58, data kapasitas solar hari kedua dapat dilihat pada Gambar 59 dan data kapasitas solar hari ketiga dapat dilihat pada Gambar 60. Data kapasitas dapat dilihat pada Lampiran 5, Lampiran 6 dan Lampiran 7.



Gambar 58. Garafik data nilai kapasitas solar tanggal 16 Juni 2023



Gambar 59. Grafik data nilai kapasitas solar tanggal 17 Juni 2023

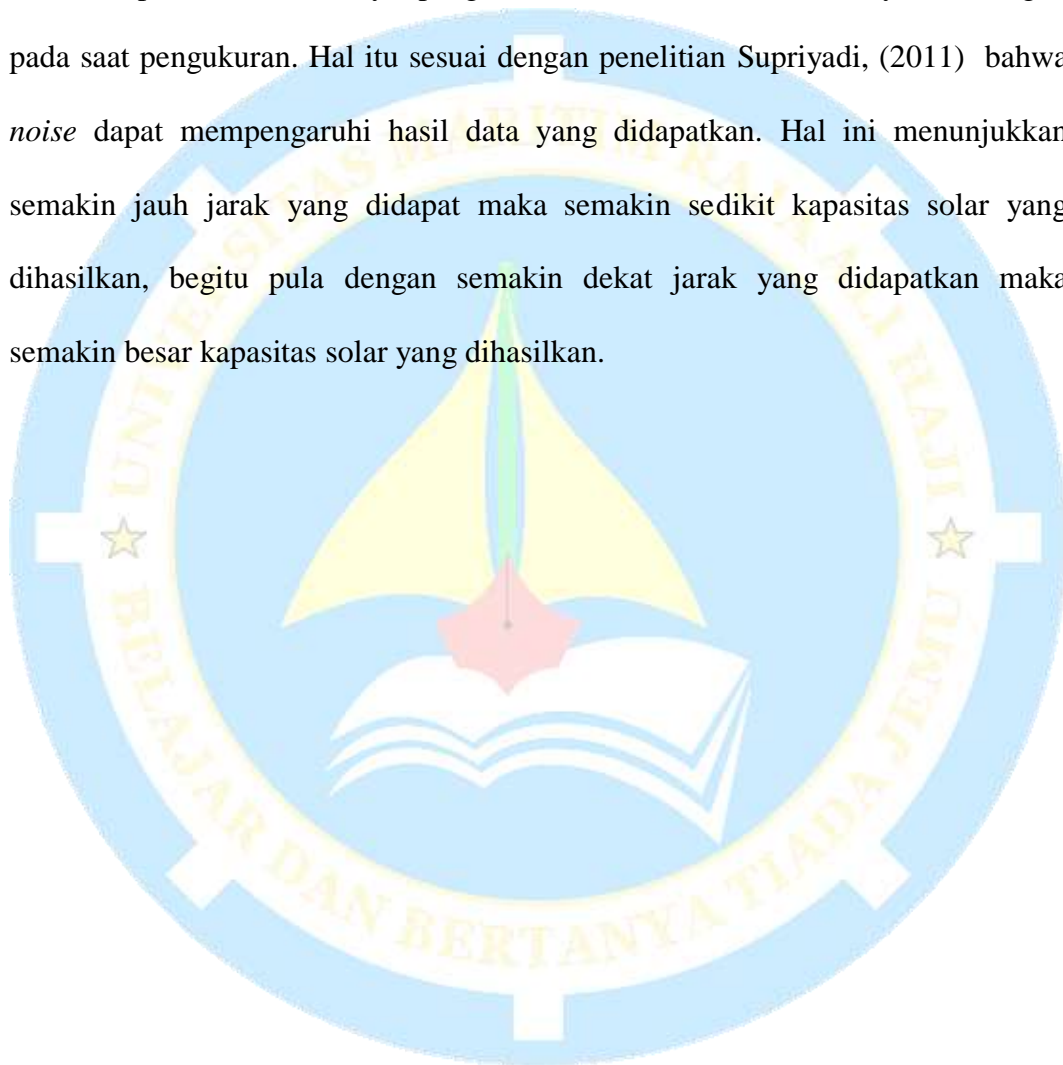


Gambar 60. Grafik data nilai kapasitas solar tanggal 18 Juni 2023

Berdasarkan grafik hasil pengujian, diketahui bahwa nilai kapasitas solar tertinggi yang diperoleh sebesar 75,67 % sedangkan nilai kapasitas solar terendah sebesar 67,04 %. Berdasarkan data tersebut diperoleh nilai kapasitas bahan bakar solar PLTD Pulau Pangkil hari pertama sebesar 74,47 % namun adanya

penurunan pada hari kedua dan hari ketiga yaitu dikarenakan adanya pengisian solar pada hari pertama sehingga kapasitas bahan bakar solar hari kedua sebesar 67,50 % dan kapasitas bahan bakar hari ketiga sebesar 67,34 % dikarenakan tidak adanya pengisian bahan bakar dalam tangki pemakaian.

Adapun tidak stabilnya pengukuran disebabkan oleh adanya kebisingan pada saat pengukuran. Hal itu sesuai dengan penelitian Supriyadi, (2011) bahwa *noise* dapat mempengaruhi hasil data yang didapatkan. Hal ini menunjukkan semakin jauh jarak yang didapat maka semakin sedikit kapasitas solar yang dihasilkan, begitu pula dengan semakin dekat jarak yang didapatkan maka semakin besar kapasitas solar yang dihasilkan.



V. PENUTUP

A. Kesimpulan

Sistem *monitoring* suhu air pendingin dan kapasitas bahan bakar solar dalam tangki di PLTD Pulau Pangkil berhasil dirancang. Cara kerja perangkat bermula dari data suhu air pendingin dan kapasitas bahan bakar solar dalam tangki yang terbaca lalu diolah ESP 32 *master* 1 yang diletakkan di mesin PLTD Pulau Pangkil dan *master* 2 yang diletakkan di tangki bahan bakar solar PLTD Pulau Pangkil. Kemudian data dikirimkan ke ESP32 *slave* yang berada di ruang *customer service* PLTD Pulau Pangkil. Sistem ini menggunakan sensor suhu dan sensor jarak. Sensor suhu memiliki tingkat akurasi rata-rata 98,50 % dan sensor jarak memiliki rata-rata akurasi sebesar 99,15 %. Setelah dilakukan pengujian dengan mendapatkan data suhu air pendingin dan kapasitas bahan bakar solar dalam tangki, didapatkan hasil bahwa suhu air pendingin memiliki nilai rata-rata 81,49 °C dan kapasitas bahan bakar solar dalam tangki rata-rata 69,70 %. Nilai tersebut membuktikan bahwa PLTD Pulau Pangkil masih di bawah batas minimal.

B. Saran

Berdasarkan hasil penelitian yang diperoleh dari data-data lapangan, pada dasarnya penelitian ini berjalan dengan baik. Namun ada beberapa saran untuk melanjutkan penelitian ini yaitu dengan menambahkan beberapa indikator pengukuran diantaranya pengukuran tegangan yang dikeluarkan, pengukuran beban pemakaian dan pengukuran oli mesin.

DAFTAR PUSTAKA

- Advanced Instructional Systems Inc. (2011). *Measurements and Error Analysis*. 1–20. https://www.webassign.net/question_assets/unccolphysem1/measurements/manual.pdf
- Arduino.cc. (2023). *Arduino Integrated Development Environment (IDE) Web Editor*. Arduino.Cc. <https://www.arduino.cc/en/software>
- Bachtiar, I. K., & Syafiq, M. (2016). Rancangan Implementasi Pembangkit Listrik Tenaga Surya (PLTS) Skala Rumah Tangga menggunakan Software HOMER: *Jurnal Sustainable: Jurnal Hasil Penelitian Dan Industri Terapan*, 5(2), 17–25. <https://doi.org/10.31629/sustainable.v5i2.368>
- Basuki, A. T. (2017). Pengantar Ekonometrika (Dilengkapi Penggunaan Eviews). *Danisa Media*, 135. <https://ekonometrikblog.files.wordpress.com/2018/02/buku-ekonometrika-1.pdf>
- Febriana, F., Hardi, S., & Fauzi. (2022). Analisis Sistem Kontrol Dan Instrumentasi Preheater Pada Engine Di Pt. Sumberdaya Sewatama Sumbagut 2 Peaker 250 Mw. 224. *Jurnal Tektro*, 06(02), 224–231. <https://ejurnal.pnl.ac.id/TEKTRO/article/download/3736/2926>
- Fitrianingsih. (2022). *Radiator Mobil: Fungsi, Manfaat dan Jenis-jenisnya*. Qoala. <https://www.qoala.app/id/blog/gaya-hidup/otomotif/fungsi-radiator-mobil/>
- Fuadi, M. H., Purwandi, A. W., Studi, P., & Telekomunikasi, J. (2020). Rancang Bangun Kontrol Dan Monitoring Suhu Pada Mesin Diesel Menggunakan Web Mobile. 31–37. <https://doi.org/10.33795/jartel.v10i1.183>
- Harsiti, Muttaqin, Z., & Srihartini, E. (2022). Penerapan Metode Regresi Linier Sederhana Untuk Prediksi Persediaan Obat Jenis Tablet. *JSiI (Jurnal Sistem Informasi)*, 9(1), 12–16. <https://doi.org/10.30656/jsii.v9i1.4426>
- Hudati, I., Kusuma, D. Y., Permatasari, N. B., & Pebriani, R. R. (2021). Sensor Ultrasonik Waterproof A02YYUW Berbasis Arduino Uno pada Sistem Pengukuran Jarak. *Jurnal Listrik, Instrumentasi Dan Elektronika Terapan (JuLIET)*, 2(2), 14–19. <https://doi.org/10.22146/juliet.v2i2.71146>
- Hyundai. (2022). *Radiator Mobil: Kenali Fungsi, Cara Kerja Hingga Perawatannya*. Hyundai. <https://www.hyundai.com/id/en/hyundai-story/articles/radiator-mobil-kenali-fungsi,-cara-kerja-hingga-perawatannya-0000000122>

- Kurniawan, I., Priyatman, H., & Elbani, A. (2015). Rancang Bangun Sistem Monitoring Penggunaan Bahan Bakar Minyak (Bbm) Dan Temperatur Pada Generator Menggunakan Sms Berbasis Pengendali Mikro. 2(1), 1–10. <https://jurnal.untan.ac.id/index.php/jteuntan/article/view/11136/10581>
- Mokhammad, Y., Priyandoko, G., & Setiawidayat, S. (2022). Prototipe Sistem Monitoring dan Controlling HSD Tank PLTGU Grati Berbasis IoT. *Jambura Journal of Electrical and Electronics Engineering*, 4(2), 159–168. <https://doi.org/10.37905/jjee.v4i2.14396>
- Muliadi, Imran, A., & Rasul, M. (2020). Pengembangan Tempat Sampah Pintar Menggunakan Esp32. *Jurnal Media Elektrik*, 17(2), 2721–9100. <https://doi.org/10.59562/metrik.v17i2.14193>
- Munir. (2003). *Metode Numerik Informatika*. <https://anyflip.com/xqzik/disb/basic/351-400>
- MyPertamina. (2021). Bagian-bagian Terpenting pada Fuel Tank serta Cara Merawatnya agar Tetap Optimal. MyPertamina. <https://mypertamina.id/bagianbagian-terpenting-pada-fuel-tank-serta-cara-merawatnya-agar-tetap-optimal>
- Pangesti, R. (2021). Suhu: Pengertian, Alat Ukur dan Skala. Detikedu. <https://www.detik.com/edu/detikpedia/d-5866521/suhu-pengertian-alat-ukur-dan-skala>
- Prastyo, E. A. (2022). *ESP-NOW Komunikasi Dua Arah Antara Papan ESP32*. <https://www.arduino.biz.id/2022/07/esp-now-komunikasi-dua-arah-antara.html>
- Puspitasari, N. F. (2011). Analisis RSSI (Receive Signal Strength Indicator) Terhadap Ketinggian Perangkat Wi-Fi Di Lingkungan Indoor. *Jurnal Ilmiah Dasi*, 15(04), 32–38. <https://media.neliti.com/media/publications/90018-ID-none.pdf>
- Rahardjo, P. (2022). Sistem Penyiraman Otomatis Menggunakan Sensor Kelembaban Tanah Berbasis Mikrokontroler Arduino Mega 2560 Pada Tanaman Mangga Harum Manis Buleleng Bali. *Majalah Ilmiah Teknologi Elektro*, 21(1), 31-35. <https://doi.org/10.24843/mite.2022.v21i01.p05>
- Rahman, E. S. (2018). Studi Tentang Proses Pembangkitan Listrik Tenaga Diesel PT . PLN (Persero) Wilayah Sulselrabar Sektor Tello Makassar. *Jurnal Teknik Elektro*, 2(1), 1-6. http://eprints.unm.ac.id/19435/1/JURNAL_MASRIANTO_132404_0019.pdf
- Randis, & Sarminto. (2018). Aplikasi Internet of Things Monitoring Suhu Engine. *Jurnal Teknik Mesin Univ. Muhammadiyah Metro*, 7(2), 153–158.

<http://dx.doi.org/10.24127/trb.v7i2.791>

Rony, M. A., & Falconi. (2013). Sistem Monitoring Volume Tangki Solar Menggunakan Sensor Ultrasonic. *IO(1)*, 50–58. <http://dx.doi.org/10.36080/bit.v10i1.510>

Saepudin, N., & Hafidz, M. (2020). Rancang Bangun Aplikasi Untuk Prototipe Sistem Monitoring Konsumsi Energi Listrik Pada Peralatan Elektronik Berbasis Pzem-004t. *Jurnal Teknik Informatika*, *12(2)*, 22–28. https://d1wqtxts1xzle7.cloudfront.net/86937132/871-Article_Text-1481-1-10-20200824-libre.pdf

Sapardi, & Yulianti, B. (2018). *Prototip Monitoring Suhu air Pada Genset 1250Kva Berbasis Microcontroller Arduino Uno 328Pvia SMS*.11–17. <https://doi.org/10.35968/jti.v7i0.304>

sentrakalibrasiindustri. (2022). *Termokopel Tipe K Ditinjau Dari Warna Kabel dan Voltasenya*. Sentra Kalibrasi Industri. <https://www.sentrakalibrasiindustri.com>

Supriyadi, T. (2011). Penggunaan Sensor Ultrasonik Sebagai Pendeteksi Ketinggian Air Sungai Pada Sistem Peringatan Dini. *Industrial Research Workshop and National Seminar*, 143–147. <https://doi.org/10.35313/irwns.v2i0.377>

University of New South Wales. (2011). First Year Physics Laboratory Manual. In *First Year Physics Laboratory Manual*. http://www.animations.physics.unsw.edu.au/sf/toolkits/Errors_and_Error_Estimation.pdf

Utomo, A. P., & Wirawan, N. A. (2018). Perancangan Alat Monitoring Air Conditioner Menggunakan Mikrokontroler Wemos. *Jurnal Telematika Edisi Industrial Engineering Seminar and Call for Paper (IESC)*, 44–53. <https://journal.ithb.ac.id/telematika/article/view/229/iesc2018>

LAMPIRAN

Lampiran 1. *Firmware* pada Perangkat *Slave*

```
//=====Memasukkan/menyertakan library.
#include <esp_now.h>
#include <esp_wifi.h>
#include <WiFi.h>
#include "RTCLib.h"
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include "FS.h"
#include "SD.h"
#include "SPI.h"
//=====

#define CS_PIN 5
#define Buzzer_PIN 4
// Mengatur channel Wi-Fi.
// Rentang channel yang dikonfigurasi dari 1~13 saluran.
// Pengirim/Master dan Penerima/Slave harus menggunakan channel Wi-Fi yang sama.
#define CHANNEL 1
#define temperature_Threshold_Value 90.00
#define kapasitas_Threshold_Value_1 10.00
#define kapasitas_Threshold_Value_2 230.00
// Deklarasi variabel untuk millis (millis untuk mengambil data dari RTC dan menampilkan data yang diterima dari master).
unsigned long previousMillis_UpdateRTC = 0;
const long interval_UpdateRTC = 1000;
String receive_Temp_Val = ""
String receive_Dist_Val = "";
```



```

byte last_Temp_Val_length;
byte last_Dist_Val_length;
String time_now = "";
String date_now = "";
String fileName = "";
bool saveDataFromMaster_1 = false;
bool saveDataFromMaster_2 = false;
bool Buzzer_ON_Temp = false;
bool Buzzer_ON_Dist = false;
byte Buzzer_Dur_Temp = 0;
byte Buzzer_Dur_Dist = 0;
bool writeFile_Success = false;
int rssii;
//=====typedef struct untuk
menerima data.
// Harus sesuai dengan "typedef struct" pada pengirim/master.
typedef struct struct_message_receive {
    String ID_Sender;
    String Data_Receive;
} struct_message_receive;
typedef struct {
    unsigned frame_ctrl: 16;
    unsigned duration_id: 16;
    uint8_t addr1[6]; /* receiver address */
    uint8_t addr2[6]; /* sender address */
    uint8_t addr3[6]; /* filtering address */
    unsigned sequence_ctrl: 16;
    uint8_t addr4[6]; /* optional */
} wifi_ieee80211_mac_hdr_t;

```

```

typedef struct {
    wifi_ieee80211_mac_hdr_t hdr;
    uint8_t payload[0]; /* network data ended with 4 bytes csum (CRC32) */
} wifi_ieee80211_packet_t;

//mengidentifikasi/memanggil RSSI
void promiscuous_rx_cb(void *buf, wifi_promiscuous_pkt_type_t type) {
    // All espnow traffic uses action frames which are a subtype of the mgmnt
    frames so filter out everything else.
    if (type != WIFI_PKT_MGMT)
        return;
    const wifi_promiscuous_pkt_t *ppkt = (wifi_promiscuous_pkt_t *)buf;
    const wifi_ieee80211_packet_t *ipkt = (wifi_ieee80211_packet_t *)ppkt-
    >payload;
    const wifi_ieee80211_mac_hdr_t *hdr = &ipkt->hdr;
    int rssi = ppkt->rx_ctrl.rssi;
    rssi = rssi;
}

// Membuat struct_message untuk menerima data.
struct_message_receive receive_Data;

//=====

// Inisialisasi dan pengaturan LCD.
// 0x27 adalah alamat i2c di LCD.
// 20 adalah jumlah kolom LCD.
// 4 adalah jumlah naris LCD.
LiquidCrystal_I2C lcd(0x27,20,4);

// Inisialisasi RTC_DS3231 sebagai rtc.
RTC_DS3231 rtc;

char month_names[12][4] = {"Jan", "Feb", "Mar", "Apr", "Mei", "Jun", "Jul",
"Ag", "Sep", "Okt", "Nov", "Des"};

//

```

```

_____Subrutin untuk menerima data dari pengirim/Master.

void OnDataRecv(const uint8_t * mac_addr, const uint8_t *incomingData, int
len) {
    //-----Mendapatkan MAC ADDRESS
    pengirim/Master.
    char macStr[18];
    sprintf(macStr, sizeof(macStr), "%02x:%02x:%02x:%02x:%02x:%02x",
mac_addr[0], mac_addr[1], mac_addr[2], mac_addr[3], mac_addr[4],
mac_addr[5]);
    //-----
    memcpy(&receive_Data, incomingData, sizeof(receive_Data)); //--> Menyalin
informasi pada variabel "incomingData" ke dalam variabel struktur
"receive_Data".
    //-----Mencetak MAC ADDRESS pengirim/Master
    dan Byte yang diterima.
    Serial.println();
    Serial.println("-----Menerima Data/Pesan");
    Serial.print("Pesan diterima dari : ");
    Serial.println(macStr);
    Serial.print("Byte yang diterima : ");
    Serial.println(len);
    //-----
    //-----Print semua data yang diterima dari
    pengirim.
    Serial.println("Data/Pesan yang diterima : ");
    Serial.println(receive_Data.ID_Sender);
    Serial.println(receive_Data.Data_Receive);
    Serial.println("-----");
    //-----
    //-----Mengambil semua data dari variabel struktur
    "terima Data" jika ID Pengirim/Master sesuai.
    // #1 = ID Master_1

```

```

if (receive_Data.ID_Sender == "#1") {
    if (checkIsNumeric(receive_Data.Data_Receive) == true) {
        receive_Temp_Val = receive_Data.Data_Receive;
        if (receive_Temp_Val.toFloat() > temperature_Threshold_Value) {
            Buzzer_ON_Temp = true;
        }
        if (receive_Temp_Val.toFloat() <= temperature_Threshold_Value) {
            Buzzer_ON_Temp = false;
            Buzzer_Dur_Temp = 0;
            if (digitalRead(Buzzer_PIN) == HIGH) digitalWrite(Buzzer_PIN, LOW);
        }
        saveDataFromMaster_1 = true;
    } else {
        receive_Temp_Val = "Error";
    }
}
// #2 = ID Master_2
if (receive_Data.ID_Sender == "#2") {
    if (checkIsNumeric(receive_Data.Data_Receive) == true) {
        receive_Dist_Val = receive_Data.Data_Receive;
        if (receive_Dist_Val.toFloat() < kapasitas_Threshold_Value_1) {
            Buzzer_ON_Dist = true;
        }
        if (receive_Dist_Val.toFloat() >= kapasitas_Threshold_Value_1) {
            Buzzer_ON_Dist = false;
            Buzzer_Dur_Dist = 0;
            if (digitalRead(Buzzer_PIN) == HIGH) digitalWrite(Buzzer_PIN, LOW);
        }
        if (receive_Dist_Val.toFloat() < kapasitas_Threshold_Value_2) {

```

```

    Buzzer_ON_Dist = true;
}
if (receive_Dist_Val.toFloat() >= kapasitas_Threshold_Value_2) {
    Buzzer_ON_Dist = false;
    Buzzer_Dur_Dist = 0;
    if (digitalRead(Buzzer_PIN) == HIGH) digitalWrite(Buzzer_PIN, LOW);
}
saveDataFromMaster_2 = true;
} else {
    receive_Dist_Val = "Error";
}
}
}
//-----
}
getTimeDate()
void getTimeDate() {
    // Mendapatkan waktu terakhir/sekarang/terbaru dari RTC.
    DateTime now = rtc.now();
    //-----Memasukkan jam,menit dan detik ke variable
    string time_now.
    time_now = "";
    if (now.hour() < 10) {
        time_now += "0" + String(now.hour());
    } else {
        time_now += String(now.hour());
    }
    time_now += ":";
    if (now.minute() < 10) {
        time_now += "0" + String(now.minute());

```

```
} else {  
    time_now += String(now.minute());  
}  
time_now += ":";  
if (now.second() < 10) {  
    time_now += "0" + String(now.second());  
} else {  
    time_now += String(now.second());  
}  
//-----  
  
//----- Memasukkan tanggal, bulan dan tahun ke  
variable string date_now.  
date_now = "";  
if (now.day() < 10) {  
    date_now += "0" + String(now.day());  
} else {  
    date_now += String(now.day());  
}  
  
date_now += "-";  
  
date_now += String(month_names[now.month()-1]);  
  
date_now += "-";  
  
date_now += String(now.year());  
//-----  
}
```

```

checkIsNumeric()
// Function untuk memeriksa isi data string berupa numerik/angka.
bool checkIsNumeric (String str_Check) {
  int str_Check_length = str_Check.length();
  for (int i = 0; i < str_Check_length; i++) {
    if (str_Check[i] != '-' && str_Check[i] != '.') {
      if (!isDigit(str_Check[i])) {
        return false;
      }
    }
  }
  return true;
}

initSDCard()
void initSDCard() {
  bool initSDCard_Success = false;
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Mengakses modul");
  lcd.setCursor(0,1);
  lcd.print("SD Card...");
  delay(2000);
  //Serial.println();
  //Serial.println("Mengakses modul SD Card...");
  for (byte i = 0; i < 4; i++) {
    if(!SD.begin(CS_PIN)){
      initSDCard_Success = false;
    } else {
      initSDCard_Success = true;
    }
  }
}

```

```
    break;
  }
  delay(1000);
}
if (initSDCard_Success == true) {
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Mengakses modul");
  lcd.setCursor(0,1);
  lcd.print("SD Card berhasil.");
  delay(2000);
} else {
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Gagal membaca kartu");
  lcd.setCursor(0,1);
  lcd.print("SD Card.");
  lcd.setCursor(0,2);
  lcd.print ("Restart ESP32...");
  lcd.setCursor(0,3);
  lcd.print ("Silahkan Tunggu...");
  //Serial.println();
  //Serial.println("Card Mount Failed");
  //Serial.println("Restart ESP32 2 seconds from now.");
  //Serial.println();
  delay(2000);
  ESP.restart();
}
uint8_t cardType = SD.cardType();
```



```

if(cardType == CARD_NONE){
  while(1) {
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Kartu SD Card");
    lcd.setCursor(0,1);
    lcd.print("tidak terpasang !");
    delay(2000);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Silahkan pasang");
    lcd.setCursor(0,1);
    lcd.print("Kartu SD Card.");
    lcd.setCursor(0,2);
    lcd.print ("Kemudian");
    lcd.setCursor(0,3);
    lcd.print ("Restart ESP32.");
    delay(2000);
    //Serial.println("No SD card attached");
    //delay(500);
  }
}
// Serial.print("SD Card Type : ");
// if(cardType == CARD_MMC){
//   Serial.println("MMC");
// } else if(cardType == CARD_SD){
//   Serial.println("SDSC");
// } else if(cardType == CARD_SDHC){
//   Serial.println("SDHC");

```

```

// } else {
    // Serial.println("UNKNOWN");
// }
// uint64_t cardSize = SD.cardSize() / (1024 * 1024);
// Serial.printf("SD Card Size : %lluMB\n", cardSize);
// Serial.printf("Total space : %lluMB\n", SD.totalBytes() / (1024 * 1024));
// Serial.printf("Used space : %lluMB\n", SD.usedBytes() / (1024 * 1024));
}
writeFile()
// Subrutin untuk membuat dan menulis file baru ke SD card.
void writeFile(fs::FS &fs, const char * path, const char * message) {
    Serial.println();
    Serial.printf("Writing file: %s\n", path);
    File file = fs.open(path, FILE_WRITE);
    if(!file) {
        Serial.println("Failed to open file for writing");
        writeFile_Success = false;
        return;
    }
    if(file.print(message)) {
        Serial.println("File written");
        writeFile_Success = true;
    } else {
        Serial.println("Write failed");
        writeFile_Success = false;
    }
    file.close();
}
appendFile()

```

```
// Subrutin untuk menuliskan data baru ke file yang sudah tersedia di SD card.
void appendFile(fs::FS &fs, const char * path, const char * message) {
  Serial.println();
  Serial.printf("Appending to file: %s\n", path);

  File file = fs.open(path, FILE_APPEND);

  if(!file) {
    Serial.println("Failed to open file for appending");
    return;
  }
  if(file.print(message)) {
    Serial.println("Message appended");
  } else {
    Serial.println("Append failed");
  }

  file.close();
}

check_File()
// Subrutin untuk memeriksa file penyimpanan data.
void check_File() {
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Memeriksa file");
  lcd.setCursor(0,1);
  lcd.print("Penyimpanan...");
  delay(2000);
}
```

```
getTimeDate();

fileName = "/" + date_now + ".txt";

// If the data.txt file doesn't exist
// Create a file on the SD card and write the data labels
File file = SD.open(fileName);

if(!file) {
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("File penyimpanan");
  lcd.setCursor(0,1);
  lcd.print("tidak ditemukan !");
  lcd.setCursor(0,2);
  lcd.print ("Mulai membuat file");
  lcd.setCursor(0,3);
  lcd.print ("penyimpanan...");
  delay(2000);

  //Serial.println();
  //Serial.println("File doesn't exist");
  //Serial.println("Creating file...");

  writeFile(SD, fileName.c_str(), "TANGGAL, JAM, MASTER,
SUHU/KAPASITAS");

  if (writeFile_Success == true) {
    lcd.clear();
```

```
lcd.setCursor(0,0);  
lcd.print("Berhasil membuat");  
lcd.setCursor(0,1);  
lcd.print("file penyimpanan.");  
delay(2000);  
} else {  
  while(1) {  
    lcd.clear();  
    lcd.setCursor(0,0);  
    lcd.print("Gagal membuat");  
    lcd.setCursor(0,1);  
    lcd.print("file penyimpanan.");  
    lcd.setCursor(0,2);  
    lcd.print("Silahkan periksa");  
    lcd.setCursor(0,3);  
    lcd.print("kartu SD Card.");  
    delay(3000);  
    lcd.clear();  
    delay(1000);  
  }  
}  
}  
else {  
  lcd.clear();  
  lcd.setCursor(0,0);  
  lcd.print("File penyimpanan");  
  lcd.setCursor(0,1);  
  lcd.print("ditemukan.");  
  delay(2000);
```

```

//Serial.println();
//Serial.println("File already exists");
}
file.close();
}
VOID SETUP()
void setup () {
// put your setup code here, to run once:
Serial.begin(115200);
Serial.println();
delay(2000);
pinMode(Buzzer_PIN, OUTPUT);
digitalWrite(Buzzer_PIN, LOW);
for (byte i = 0; i < 4; i++) {
digitalWrite(Buzzer_PIN, !digitalRead(Buzzer_PIN));
delay(1000);
}
// Menjalankan inisialiasasi dan menerapkan pengaturan LCD.
lcd.init();
// menyalakan LCD backlight.
lcd.backlight();
delay(1000);
//-----Memulai dan memeriksa ketersediaan rtc.
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Memeriksa RTC...");
delay(2000);
//Serial.println();

```

```

//Serial.println("Memeriksa RTC...");
if (!rtc.begin()) {
  delay(500);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("RTC tidak ditemukan!");

  //Serial.println("RTC tidak ditemukan");
  //Serial.println("Mohon periksa pemasangan RTC dengan ESP32.");
  while(1) {
    delay(100);
  }
}
delay(500);
lcd.clear();
lcd.setCursor(0,0);
lcd.print("RTC Ok.");
delay(2000);

//Serial.println("RTC ditemukan");
//-----
//-----Pengaturan Jam dan Tanggal (Time dan
Date).

// rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0)); //--> rtc.adjust(DateTime(Tahun,
Bulan, Hari/Tanggal, Jam, Menit, Detik));

//-----

delay(1000);
initSDCard();
delay(1000);
check_File();

```

```

delay(1000);
// Tetapkan perangkat pada mode "Wi-Fi Station".
Serial.println();
Serial.println("Atur wifi ke mode Wi-Fi Station.");
WiFi.mode(WIFI_STA);
//-----
int cur_WIFIchannel = WiFi.channel();
if (cur_WIFIchannel != CHANNEL) {
    esp_wifi_set_promiscuous(true);
    esp_wifi_set_channel(CHANNEL, WIFI_SECOND_CHAN_NONE);
    esp_wifi_set_promiscuous(false);
}
//-----
esp_wifi_set_promiscuous(true);
esp_wifi_set_promiscuous_rx_cb(&promiscuous_rx_cb);
//Serial.println();
//Serial.print("WiFi Channel : ");
//Serial.println(WiFi.channel());
//Serial.println();
//Serial.print("Mac Address : ");
//Serial.println(WiFi.macAddress());
//-----Init ESP-NOW.
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Inisialisasi");
lcd.setCursor(0,1);
lcd.print("ESP-NOW...");
delay(2000);
//Serial.println();

```



```

//Serial.println("Mulai menginisialisasi ESP-NOW...");
if (esp_now_init() != ESP_OK) {
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Inisialisasi");
  lcd.setCursor(0,1);
  lcd.print("ESP-NOW gagal !");
  lcd.setCursor(0,2);
  lcd.print ("Restart ESP32...");
  lcd.setCursor(0,3);
  lcd.print ("Silahkan tunggu...");
  delay(2000);
  //Serial.println("Error initializing ESP-NOW");
  //Serial.println("Restart ESP32...");
  //delay(1000);
  ESP.restart();
}
lcd.clear();
lcd.setCursor(0,0);
lcd.print("Inisialisasi ESP-NOW");
lcd.setCursor(0,1);
lcd.print("berhasil.");
delay(2000);
//Serial.println("Inisialisasi ESP-NOW berhasil.");
//-----

// Mendaftarkan fungsi "callback function OnDataRecv" yang akan dipanggil
saat data diterima.

esp_now_register_recv_cb(OnDataRecv);

lcd.clear();

```

```

}
VOID LOOP()
void loop () {
  // put your main code here, to run repeatedly:
  unsigned long currentMillis_UpdateRTC = millis();
  if (currentMillis_UpdateRTC - previousMillis_UpdateRTC >=
interval_UpdateRTC) {
    previousMillis_UpdateRTC = currentMillis_UpdateRTC;

    // Mengupdate Jam dan Tanggal (Time & date).
    getTimeDate();
    //:.....
    //Serial.println();
    //Serial.println("----Waktu dan Tanggal");
    //Serial.print("Jam : ");
    //Serial.println(time_now);
    //Serial.print("Tgl : ");
    //Serial.println(date_now);
    //Serial.println("----");
    //:.....
    //:.....:Menampilkan Jam dan Tanggal ke LCD.
    lcd.setCursor(0, 0);
    lcd.print("Jam  :");
    lcd.print(time_now);
    lcd.setCursor(0, 1);
    lcd.print("Tgl  :");
    lcd.print(date_now);
    lcd.setCursor(0, 2);
    lcd.print("Suhu  :");

```

```

if (last_Temp_Val_length != receive_Temp_Val.length()) {
    lcd.setCursor(8, 2);
    lcd.print("    ");
}
lcd.setCursor(8, 2);
lcd.print(receive_Temp_Val);
lcd.print(" ");
lcd.print(char(223));
lcd.print("C");
lcd.setCursor(0, 3);
lcd.print("Kapasitas :");
if (last_Dist_Val_length != receive_Dist_Val.length()) {
    lcd.setCursor(12, 3);
    lcd.print("    ");
}
★
lcd.setCursor(12, 3);
lcd.print(receive_Dist_Val);
lcd.print(" %");
Serial.println("rssi\t= " + String(rssi) + " dBm");
last_Temp_Val_length = receive_Temp_Val.length();
last_Dist_Val_length = receive_Dist_Val.length();

//:Kondisi untuk menyimpan data yang diterima dari Master_1 dan
Master_2.

if (saveDataFromMaster_1 == true || saveDataFromMaster_2 == true) {
    fileName = "/" + date_now + ".txt";
    // If the file .txt file doesn't exist
    // Create a file on the SD card and write the data labels
    File file = SD.open(fileName);
    if(!file) {

```

```

//Serial.println();
//Serial.println("File doesn't exist");
//Serial.println("Creating file...");
writeFile(SD, fileName.c_str(), "TANGGAL, JAM, SUHU,
KAPASITAS");
}
else {
//Serial.println();
//Serial.println("File already exists");
}
file.close();
}
if (saveDataFromMaster_1 == true) {
String saveData = "\r\n" + date_now + "," + time_now + ",Master_1," +
receive_Temp_Val + ",°C," + rssii + "dBm";//Menyimpan data
// Append the data to file
appendFile(SD, fileName.c_str(), saveData.c_str());
saveDataFromMaster_1 = false;
}
if (saveDataFromMaster_2 == true) {
String saveData = "\r\n" + date_now + "," + time_now + ",Master_2," +
receive_Dist_Val + ",%," + rssii + "dBm";//Menyimpan data
// Append the data to file
appendFile(SD, fileName.c_str(), saveData.c_str());
saveDataFromMaster_2 = false;
}
//:.....:
//:.....:Kondisi untuk menyalakan dan mematikan Buzzer.
if (Buzzer_ON_Temp == true) {
digitalWrite(Buzzer_PIN, HIGH);

```

```
Buzzer_Dur_Temp++;  
if (Buzzer_Dur_Temp > 10) {  
    Buzzer_Dur_Temp = 10;  
    digitalWrite(Buzzer_PIN, LOW);  
}  
}  
if (Buzzer_ON_Dist == true) {  
    digitalWrite(Buzzer_PIN, HIGH);  
    Buzzer_Dur_Dist++;  
    if (Buzzer_Dur_Dist > 10) {  
        Buzzer_Dur_Dist = 10;  
        digitalWrite(Buzzer_PIN, LOW);  
    }  
}  
if (Buzzer_ON_Dist == true) {  
    digitalWrite(Buzzer_PIN, HIGH);  
    Buzzer_Dur_Dist++;  
    if (Buzzer_Dur_Dist > 230) {  
        Buzzer_Dur_Dist = 10;  
        digitalWrite(Buzzer_PIN, LOW);  
    }  
}  
}  
}
```

Lampiran 2. Data Pengujian Suhu Pada Keseluruhan Sistem 16 Juni 2023

16 Juni 2023				
No	Date	Time	°C	RSSI
1	16 Juni 2023	16:12:09	52,72	-80 dBm
2	16 Juni 2023	16:20:09	75,675	-87 dBm
3	16 Juni 2023	16:30:09	81,325	-86 dBm
4	16 Juni 2023	16:40:09	81,375	-68 dBm
5	16 Juni 2023	16:50:09	81,05	-81 dBm
6	16 Juni 2023	17:00:09	81,3	-82 dBm
7	16 Juni 2023	17:10:09	81,425	-81 dBm
8	16 Juni 2023	17:20:09	81,125	-82 dBm
9	16 Juni 2023	17:30:09	80,925	-81 dBm
10	16 Juni 2023	17:40:09	81,175	-81 dBm
11	16 Juni 2023	17:50:09	81,25	-82 dBm
12	16 Juni 2023	18:00:09	81,575	-80 dBm
13	16 Juni 2023	18:10:09	82,1	-82 dBm
14	16 Juni 2023	18:20:08	82,625	-80 dBm
15	16 Juni 2023	18:30:08	83,55	-87 dBm
16	16 Juni 2023	18:40:08	83,6	-86 dBm
17	16 Juni 2023	18:50:08	83,8	-68 dBm
18	16 Juni 2023	19:00:08	84,2	-81 dBm
19	16 Juni 2023	19:10:08	84,275	-82 dBm
20	16 Juni 2023	19:20:08	84,25	-81 dBm
21	16 Juni 2023	19:30:08	84,15	-82 dBm
22	16 Juni 2023	19:40:09	83,725	-81 dBm
23	16 Juni 2023	19:50:09	83,475	-81 dBm
24	16 Juni 2023	20:00:09	83,3	-82 dBm
25	16 Juni 2023	20:10:09	83,025	-80 dBm
26	16 Juni 2023	20:20:09	83,175	-82 dBm
27	16 Juni 2023	20:30:09	82,925	-80 dBm
28	16 Juni 2023	20:40:09	83,05	-87 dBm
29	16 Juni 2023	20:50:09	82,7	-86 dBm
30	16 Juni 2023	21:00:09	82,575	-68 dBm
31	16 Juni 2023	21:10:09	82,75	-81 dBm
32	16 Juni 2023	21:20:09	82,475	-82 dBm
33	16 Juni 2023	21:30:09	82,5	-81 dBm
34	16 Juni 2023	21:40:09	82,25	-82 dBm
35	16 Juni 2023	21:50:09	82,275	-81 dBm

36	16 Juni 2023	22:00:09	82,4	-81 dBm
37	16 Juni 2023	22:10:09	82,6	-82 dBm
38	16 Juni 2023	22:20:09	82,125	-80 dBm
39	16 Juni 2023	22:30:09	81,975	-82 dBm
40	16 Juni 2023	22:40:09	81,825	-80 dBm
41	16 Juni 2023	22:50:09	81,825	-87 dBm
42	16 Juni 2023	23:00:09	81,75	-86 dBm
43	16 Juni 2023	23:10:09	81,95	-68 dBm
44	16 Juni 2023	23:20:09	81,85	-81 dBm
45	16 Juni 2023	23:30:09	81,8	-82 dBm
46	16 Juni 2023	23:40:09	81,875	-81 dBm
47	16 Juni 2023	23:50:09	81,6	-82 dBm
48	16 Juni 2023	00:00:09	81,65	-81 dBm
49	16 Juni 2023	00:10:09	81,525	-81 dBm
50	16 Juni 2023	00:20:09	81,55	-82 dBm
51	16 Juni 2023	00:30:09	81,475	-80 dBm
52	16 Juni 2023	00:40:09	81,375	-82 dBm
53	16 Juni 2023	00:50:09	81,35	-80 dBm
54	16 Juni 2023	01:00:09	81,325	-87 dBm
55	16 Juni 2023	01:10:09	81,525	-86 dBm
56	16 Juni 2023	01:20:09	81,275	-68 dBm
57	16 Juni 2023	01:30:09	81,475	-81 dBm
58	16 Juni 2023	01:40:09	81,125	-82 dBm
59	16 Juni 2023	01:50:09	81	-81 dBm
60	16 Juni 2023	02:00:09	81,025	-82 dBm
61	16 Juni 2023	02:10:09	80,975	-81 dBm
62	16 Juni 2023	02:20:09	81,1	-81 dBm
63	16 Juni 2023	02:30:09	81,05	-82 dBm
64	16 Juni 2023	02:40:09	80,925	-80 dBm
65	16 Juni 2023	02:50:09	80,925	-82 dBm
66	16 Juni 2023	03:00:10	80,95	-80 dBm
67	16 Juni 2023	03:10:10	80,8	-87 dBm
68	16 Juni 2023	03:20:10	81,1	-86 dBm
69	16 Juni 2023	03:30:10	80,975	-68 dBm
70	16 Juni 2023	03:40:10	81,15	-81 dBm
71	16 Juni 2023	03:50:10	81,1	-82 dBm
72	16 Juni 2023	04:00:10	80,975	-81 dBm
73	16 Juni 2023	04:10:10	80,95	-82 dBm
74	16 Juni 2023	04:20:10	81,35	-81 dBm

75	16 Juni 2023	04:30:09	81,3	-81 dBm
76	16 Juni 2023	04:40:09	81,35	-82 dBm
77	16 Juni 2023	04:50:09	81,225	-80 dBm
78	16 Juni 2023	05:00:09	81,35	-82 dBm
79	16 Juni 2023	05:10:09	81,15	-80 dBm
80	16 Juni 2023	05:20:09	81,05	-87 dBm
81	16 Juni 2023	05:30:09	81,025	-86 dBm
82	16 Juni 2023	05:40:09	81,1	-68 dBm
83	16 Juni 2023	05:50:09	81,25	-81 dBm
84	16 Juni 2023	06:00:09	80,95	-82 dBm
85	16 Juni 2023	06:10:09	80,55	-81 dBm
86	16 Juni 2023	06:20:09	80,45	-82 dBm
87	16 Juni 2023	06:30:09	80,45	-81 dBm
88	16 Juni 2023	06:40:09	80,55	-82 dBm



Lampiran 3. Data Pengujian Suhu Pada Keseluruhan Sistem 17 Juni 2023

17 Juni 2023				
No	Date	Time	°C	RSSI
1	17 juni 2023	17:30:09	51,8	-80 dBm
2	17 juni 2023	17:40:09	76,65	-87 dBm
3	17 juni 2023	17:50:09	81,325	-86 dBm
4	17 juni 2023	18:00:09	82,05	-68 dBm
5	17 juni 2023	18:10:09	82,575	-81 dBm
6	17 juni 2023	18:20:08	83,475	-82 dBm
7	17 juni 2023	18:30:08	83,825	-81 dBm
8	17 juni 2023	18:40:08	84,325	-82 dBm
9	17 juni 2023	18:50:08	84,65	-81 dBm
10	17 juni 2023	19:00:08	84,65	-81 dBm
11	17 juni 2023	19:10:08	84,675	-82 dBm
12	17 juni 2023	19:20:08	84,45	-80 dBm
13	17 juni 2023	19:30:08	84,4	-82 dBm
14	17 juni 2023	19:40:09	84,45	-80 dBm
15	17 juni 2023	19:50:09	84,1	-87 dBm
16	17 juni 2023	20:00:09	83,775	-86 dBm
17	17 juni 2023	20:10:09	83,8	-68 dBm
18	17 juni 2023	20:20:09	83,625	-81 dBm
19	17 juni 2023	20:30:09	83,475	-82 dBm
20	17 juni 2023	20:40:09	83,225	-81 dBm
21	17 juni 2023	20:50:09	83,125	-82 dBm
22	17 juni 2023	21:00:09	82,825	-81 dBm
23	17 juni 2023	21:10:09	82,625	-81 dBm
24	17 juni 2023	21:20:09	82,775	-82 dBm
25	17 juni 2023	21:30:09	82,7	-80 dBm
26	17 juni 2023	21:40:09	82,625	-82 dBm
27	17 juni 2023	21:50:09	82,825	-80 dBm
28	17 juni 2023	22:00:09	82,675	-87 dBm
29	17 juni 2023	22:10:09	82,675	-86 dBm
30	17 juni 2023	22:20:09	82,4	-68 dBm
31	17 juni 2023	22:30:09	82,7	-81 dBm
32	17 juni 2023	22:40:09	82,825	-82 dBm
33	17 juni 2023	22:50:09	82,6	-81 dBm
34	17 juni 2023	23:00:09	82,45	-82 dBm
35	17 juni 2023	23:10:09	82,6	-81 dBm

36	17 juni 2023	23:20:09	82,125	-81 dBm
37	17 juni 2023	23:30:09	82,5	-82 dBm
38	17 juni 2023	23:40:09	82,475	-80 dBm
39	17 juni 2023	23:50:09	82,25	-82 dBm
40	17 juni 2023	00:00:09	81,95	-80 dBm
41	17 juni 2023	00:10:09	81,9	-87 dBm
42	17 juni 2023	00:20:09	82	-86 dBm
43	17 juni 2023	00:30:09	81,9	-68 dBm
44	17 juni 2023	00:40:09	81,775	-81 dBm
45	17 juni 2023	00:50:09	81,725	-82 dBm
46	17 juni 2023	01:00:09	81,6	-81 dBm
47	17 juni 2023	01:10:09	81,575	-82 dBm
48	17 juni 2023	01:20:09	81,5	-81 dBm
49	17 juni 2023	01:30:09	81,6	-81 dBm
50	17 juni 2023	01:40:09	81,65	-82 dBm
51	17 juni 2023	01:50:09	81,475	-80 dBm
52	17 juni 2023	02:00:09	81,525	-82 dBm
53	17 juni 2023	02:10:09	81,3	-80 dBm
54	17 juni 2023	02:20:09	81,125	-87 dBm
55	17 juni 2023	02:30:09	81,075	-86 dBm
56	17 juni 2023	02:40:09	81,1	-68 dBm
57	17 juni 2023	02:50:09	81,325	-81 dBm
58	17 juni 2023	03:00:10	81,575	-82 dBm
59	17 juni 2023	03:10:10	81,3	-81 dBm
60	17 juni 2023	03:20:10	81,325	-82 dBm
61	17 juni 2023	03:30:10	81,25	-81 dBm
62	17 juni 2023	03:40:10	81,2	-81 dBm
63	17 juni 2023	03:50:10	81,25	-82 dBm
64	17 juni 2023	04:00:10	80,975	-80 dBm
65	17 juni 2023	04:10:10	81,05	-82 dBm
66	17 juni 2023	04:20:10	80,925	-80 dBm
67	17 juni 2023	04:30:09	81,075	-87 dBm
68	17 juni 2023	04:40:09	81,175	-86 dBm
69	17 juni 2023	04:50:09	81,35	-68 dBm
70	17 juni 2023	05:00:09	81,05	-81 dBm
71	17 juni 2023	05:10:09	81,725	-82 dBm
72	17 juni 2023	05:20:09	81,55	-81 dBm
73	17 juni 2023	05:30:09	81,625	-82 dBm
74	17 juni 2023	05:40:09	81,35	-81 dBm

75	17 juni 2023	05:50:09	80,975	-81 dBm
76	17 juni 2023	06:00:09	80,975	-82 dBm
77	17 juni 2023	06:10:09	81,075	-81 dBm
78	17 juni 2023	06:20:09	80,75	-82 dBm
79	17 juni 2023	06:30:09	80,625	-81 dBm



Lampiran 4. Data Pengujian Suhu Pada Keseluruhan Sistem 18 juni 2023

18 juni 2023				
No	Date	Time	°C	RSSI
1	18 Juni 2023	17:30:09	50,775	-80 dBm
2	18 Juni 2023	17:40:09	73,275	-87 dBm
3	18 Juni 2023	17:50:09	80,35	-86 dBm
4	18 Juni 2023	18:00:09	81,325	-68 dBm
5	18 Juni 2023	18:10:09	81,975	-81 dBm
6	18 Juni 2023	18:20:08	82,55	-82 dBm
7	18 Juni 2023	18:30:08	83,375	-81 dBm
8	18 Juni 2023	18:40:08	83,775	-82 dBm
9	18 Juni 2023	18:50:08	84,25	-81 dBm
10	18 Juni 2023	19:00:08	84,65	-81 dBm
11	18 Juni 2023	19:10:08	84,675	-82 dBm
12	18 Juni 2023	19:20:08	84,65	-80 dBm
13	18 Juni 2023	19:30:08	84,525	-82 dBm
14	18 Juni 2023	19:40:09	84,35	-80 dBm
15	18 Juni 2023	19:50:09	84,475	-87 dBm
16	18 Juni 2023	20:00:09	84,125	-86 dBm
17	18 Juni 2023	20:10:09	83,8	-68 dBm
18	18 Juni 2023	20:20:09	83,8	-81 dBm
19	18 Juni 2023	20:30:09	83,675	-82 dBm
20	18 Juni 2023	20:40:09	83,45	-81 dBm
21	18 Juni 2023	20:50:09	83,225	-82 dBm
22	18 Juni 2023	21:00:09	83,2	-81 dBm
23	18 Juni 2023	21:10:09	82,825	-81 dBm
24	18 Juni 2023	21:20:09	82,6	-82 dBm
25	18 Juni 2023	21:30:09	82,825	-80 dBm
26	18 Juni 2023	21:40:09	82,75	-82 dBm
27	18 Juni 2023	21:50:09	82,525	-80 dBm
28	18 Juni 2023	22:00:09	82,825	-87 dBm
29	18 Juni 2023	22:10:09	82,7	-86 dBm
30	18 Juni 2023	22:20:09	82,675	-68 dBm
31	18 Juni 2023	22:30:09	82,475	-81 dBm
32	18 Juni 2023	22:40:09	82,6	-82 dBm
33	18 Juni 2023	22:50:09	82,75	-81 dBm
34	18 Juni 2023	23:00:09	81,575	-82 dBm
35	18 Juni 2023	23:10:09	81,225	-81 dBm

36	18 Juni 2023	23:20:09	80,95	-81 dBm
37	18 Juni 2023	23:30:09	81,025	-82 dBm
38	18 Juni 2023	23:40:09	80,775	-80 dBm
39	18 Juni 2023	23:50:09	80,85	-82 dBm
40	18 Juni 2023	00:00:09	81,025	-80 dBm
41	18 Juni 2023	00:10:09	80,9	-87 dBm
42	18 Juni 2023	00:20:09	80,9	-86 dBm
43	18 Juni 2023	00:30:09	80,975	-68 dBm
44	18 Juni 2023	00:40:09	80,925	-81 dBm
45	18 Juni 2023	00:50:09	80,875	-82 dBm
46	18 Juni 2023	01:00:09	80,8	-81 dBm
47	18 Juni 2023	01:10:09	80,825	-82 dBm
48	18 Juni 2023	01:20:09	80,7	-81 dBm
49	18 Juni 2023	01:30:09	80,725	-81 dBm
50	18 Juni 2023	01:40:09	80,55	-82 dBm
51	18 Juni 2023	01:50:09	80,55	-80 dBm
52	18 Juni 2023	02:00:09	80,425	-82 dBm
53	18 Juni 2023	02:10:09	80,6	-80 dBm
54	18 Juni 2023	02:20:09	80,5	-87 dBm
55	18 Juni 2023	02:30:09	80,725	-86 dBm
56	18 Juni 2023	02:40:09	80,65	-68 dBm
57	18 Juni 2023	02:50:09	80,775	-81 dBm
58	18 Juni 2023	03:00:10	80,675	-82 dBm
59	18 Juni 2023	03:10:10	80,8	-81 dBm
60	18 Juni 2023	03:20:10	81,025	-82 dBm
61	18 Juni 2023	03:30:10	81,025	-81 dBm
62	18 Juni 2023	03:40:10	80,95	-81 dBm
63	18 Juni 2023	03:50:10	80,975	-82 dBm
64	18 Juni 2023	04:00:10	80,8	-80 dBm
65	18 Juni 2023	04:10:10	80,8	-82 dBm
66	18 Juni 2023	04:20:10	80,575	-80 dBm
67	18 Juni 2023	04:30:09	80,6	-87 dBm
68	18 Juni 2023	04:40:09	80,85	-86 dBm
69	18 Juni 2023	04:50:09	81	-68 dBm
70	18 Juni 2023	05:00:09	80,9	-81 dBm
71	18 Juni 2023	05:10:09	80,925	-82 dBm
72	18 Juni 2023	05:20:09	81,075	-81 dBm
73	18 Juni 2023	05:30:09	81,1	-82 dBm
74	18 Juni 2023	05:40:09	80,85	-81 dBm

75	18 Juni 2023	05:50:09	80,625	-81 dBm
76	18 Juni 2023	06:00:09	80,75	-82 dBm
77	18 Juni 2023	06:10:09	80,84091	-81 dBm



Lampiran 5. Data Pengujian Kapasitas Pada Keseluruhan Sistem 16 juni 2023

16 juni 2023				
No	Date	Time	Kapasitas %	RSSI
1	16 juni 2023	18:00:09	75,21667	-76 dBm
2	16 juni 2023	18:10:09	75,27083	-90 dBm
3	16 juni 2023	18:20:08	75,35833	-90 dBm
4	16 juni 2023	18:30:08	75,15	-90 dBm
5	16 juni 2023	18:40:08	75,0375	-88 dBm
6	16 juni 2023	18:50:08	74,96667	-89 dBm
7	16 juni 2023	19:00:08	74,95417	-89 dBm
8	16 juni 2023	19:10:08	74,88333	-90 dBm
9	16 juni 2023	19:20:08	74,9875	-87 dBm
10	16 juni 2023	19:30:08	75,15	-88 dBm
11	16 juni 2023	19:40:09	75,0375	-88 dBm
12	16 juni 2023	19:50:09	74,98333	-90 dBm
13	16 juni 2023	20:00:09	74,97083	-90 dBm
14	16 juni 2023	20:10:09	74,87917	-90 dBm
15	16 juni 2023	20:20:09	75,33333	-88 dBm
16	16 juni 2023	20:30:09	75,27083	-89 dBm
17	16 juni 2023	20:40:09	75,1125	-89 dBm
18	16 juni 2023	20:50:09	75,05833	-90 dBm
19	16 juni 2023	21:00:09	74,92917	-87 dBm
20	16 juni 2023	21:10:09	75,04167	-88 dBm
21	16 juni 2023	21:20:09	75,35833	-88 dBm
22	16 juni 2023	21:30:09	75,25417	-90 dBm
23	16 juni 2023	21:40:09	75,13333	-90 dBm
24	16 juni 2023	21:50:09	75,05833	-90 dBm
25	16 juni 2023	22:00:09	74,92917	-88 dBm
26	16 juni 2023	22:10:09	74,8375	-89 dBm
27	16 juni 2023	22:20:09	74,06667	-89 dBm
28	16 juni 2023	22:30:09	74,075	-90 dBm
29	16 juni 2023	22:40:09	73,85833	-87 dBm
30	16 juni 2023	22:50:09	73,84167	-88 dBm
31	16 juni 2023	23:00:09	73,85833	-88 dBm
32	16 juni 2023	23:10:09	73,875	-90 dBm
33	16 juni 2023	23:20:09	73,85833	-90 dBm
34	16 juni 2023	23:30:09	73,9375	-90 dBm
35	16 juni 2023	23:40:09	73,75	-88 dBm

36	16 juni 2023	23:50:09	73,6	-89 dBm
37	16 juni 2023	00:00:09	73,66667	-89 dBm
38	16 juni 2023	00:10:09	73,675	-90 dBm
39	16 juni 2023	00:20:09	73,725	-87 dBm
40	16 juni 2023	00:30:09	73,75833	-88 dBm
41	16 juni 2023	00:40:09	73,775	-88 dBm
42	16 juni 2023	00:50:09	73,85833	-90 dBm
43	16 juni 2023	01:00:09	73,875	-90 dBm
44	16 juni 2023	01:10:09	73,875	-90 dBm
45	16 juni 2023	01:20:09	73,85833	-88 dBm
46	16 juni 2023	01:30:09	74	-89 dBm
47	16 juni 2023	01:40:09	73,62917	-89 dBm
48	16 juni 2023	01:50:09	73,27917	-90 dBm
49	16 juni 2023	02:00:09	72,65833	-87 dBm
50	16 juni 2023	02:10:09	71,775	-88 dBm
51	16 juni 2023	02:20:09	70,95833	-88 dBm
52	16 juni 2023	02:30:09	69,39167	-90 dBm
53	16 juni 2023	02:40:09	74,92917	-90 dBm
54	16 juni 2023	02:50:09	74,90417	-90 dBm
55	16 juni 2023	03:00:10	74,975	-88 dBm
56	16 juni 2023	03:10:10	74,97917	-89 dBm
57	16 juni 2023	03:20:10	75,13333	-89 dBm
58	16 juni 2023	03:30:10	75,05833	-90 dBm
59	16 juni 2023	03:40:10	74,92917	-87 dBm
60	16 juni 2023	03:50:10	74,95417	-88 dBm
61	16 juni 2023	04:00:10	74,92083	-88 dBm
62	16 juni 2023	04:10:10	75,39167	-90 dBm
63	16 juni 2023	04:20:10	75,225	-90 dBm
64	16 juni 2023	04:30:09	75,09583	-90 dBm
65	16 juni 2023	04:40:09	75,0375	-88 dBm
66	16 juni 2023	04:50:09	74,9625	-89 dBm
67	16 juni 2023	05:00:09	75,05833	-89 dBm
68	16 juni 2023	05:10:09	75,35417	-90 dBm
69	16 juni 2023	05:20:09	75,26667	-87 dBm
70	16 juni 2023	05:30:09	75,1125	-88 dBm
71	16 juni 2023	05:40:09	75,02083	-88 dBm
72	16 juni 2023	05:50:09	74,9625	-88 dBm
73	16 juni 2023	06:00:09	74,67917	

Lampiran 6. Data Pengujian Kapasitas Pada Keseluruhan Sistem 17 juni 2023

17 juni 2023				
No	Date	Time	Kapasitas (%)	RSSI
1	17 Juni 2023	18:00:09	67,39583	-75 dBm
2	17 Juni 2023	18:10:09	67,4625	-90 dBm
3	17 Juni 2023	18:20:08	67,425	-90 dBm
4	17 Juni 2023	18:30:08	67,31667	-90 dBm
5	17 Juni 2023	18:40:08	67,2625	-88 dBm
6	17 Juni 2023	18:50:08	67,24583	-89 dBm
7	17 Juni 2023	19:00:08	67,31667	-89 dBm
8	17 Juni 2023	19:10:08	67,2875	-90 dBm
9	17 Juni 2023	19:20:08	67,25	-87 dBm
10	17 Juni 2023	19:30:08	67,2875	-88 dBm
11	17 Juni 2023	19:40:09	67,32917	-88 dBm
12	17 Juni 2023	19:50:09	67,4125	-90 dBm
13	17 Juni 2023	20:00:09	67,825	-90 dBm
14	17 Juni 2023	20:10:09	67,8	-90 dBm
15	17 Juni 2023	20:20:09	67,8375	-88 dBm
16	17 Juni 2023	20:30:09	67,81667	-89 dBm
17	17 Juni 2023	20:40:09	67,79167	-89 dBm
18	17 Juni 2023	20:50:09	67,725	-90 dBm
19	17 Juni 2023	21:00:09	67,725	-87 dBm
20	17 Juni 2023	21:10:09	67,70833	-88 dBm
21	17 Juni 2023	21:20:09	67,725	-88 dBm
22	17 Juni 2023	21:30:09	67,675	-90 dBm
23	17 Juni 2023	21:40:09	67,64167	-90 dBm
24	17 Juni 2023	21:50:09	67,675	-90 dBm
25	17 Juni 2023	22:00:09	67,62083	-88 dBm
26	17 Juni 2023	22:10:09	67,60417	-89 dBm
27	17 Juni 2023	22:20:09	67,60833	-89 dBm
28	17 Juni 2023	22:30:09	67,65417	-90 dBm
29	17 Juni 2023	22:40:09	67,59167	-87 dBm
30	17 Juni 2023	22:50:09	67,60833	-88 dBm
31	17 Juni 2023	23:00:09	67,59167	-88 dBm
32	17 Juni 2023	23:10:09	67,59167	-90 dBm
33	17 Juni 2023	23:20:09	67,6	-90 dBm
34	17 Juni 2023	23:30:09	67,6	-90 dBm
35	17 Juni 2023	23:40:09	67,62083	-88 dBm
36	17 Juni 2023	23:50:09	67,625	-76 dBm
37	17 Juni 2023	00:00:09	67,60833	-90 dBm

38	17 Juni 2023	00:10:09	67,56667	-90 dBm
39	17 Juni 2023	00:20:09	67,58333	-90 dBm
40	17 Juni 2023	00:30:09	67,64583	-88 dBm
41	17 Juni 2023	00:40:09	67,60417	-89 dBm
42	17 Juni 2023	00:50:09	67,5875	-89 dBm
43	17 Juni 2023	01:00:09	67,54583	-90 dBm
44	17 Juni 2023	01:10:09	67,50833	-87 dBm
45	17 Juni 2023	01:20:09	67,57917	-88 dBm
46	17 Juni 2023	01:30:09	67,50833	-88 dBm
47	17 Juni 2023	01:40:09	67,49167	-90 dBm
48	17 Juni 2023	01:50:09	67,475	-90 dBm
49	17 Juni 2023	02:00:09	67,475	-90 dBm
50	17 Juni 2023	02:10:09	67,475	-88 dBm
51	17 Juni 2023	02:20:09	67,49167	-89 dBm
52	17 Juni 2023	02:30:09	67,475	-89 dBm
53	17 Juni 2023	02:40:09	67,44167	-90 dBm
54	17 Juni 2023	02:50:09	67,5	-87 dBm
55	17 Juni 2023	03:00:10	67,45833	-88 dBm
56	17 Juni 2023	03:10:10	67,47083	-88 dBm
57	17 Juni 2023	03:20:10	67,43333	-90 dBm
58	17 Juni 2023	03:30:10	67,4	-90 dBm
59	17 Juni 2023	03:40:10	67,4375	-90 dBm
60	17 Juni 2023	03:50:10	67,375	-88 dBm
61	17 Juni 2023	04:00:10	67,30833	-89 dBm
62	17 Juni 2023	04:10:10	67,35833	-89 dBm
63	17 Juni 2023	04:20:10	67,34167	-90 dBm
64	17 Juni 2023	04:30:09	67,34167	-87 dBm
65	17 Juni 2023	04:40:09	67,34167	-88 dBm
66	17 Juni 2023	04:50:09	67,33333	-88 dBm
67	17 Juni 2023	05:00:09	67,43333	-90 dBm
68	17 Juni 2023	05:10:09	67,33333	-90 dBm
69	17 Juni 2023	05:20:09	67,35	-90 dBm
70	17 Juni 2023	05:30:09	67,3125	-88 dBm
71	17 Juni 2023	05:40:09	67,32917	-88 dBm
72	17 Juni 2023	05:50:09	67,2625	-88 dBm
73	17 Juni 2023	06:00:09	67,25	-90 dBm
74	17 Juni 2023	06:10:09	67,34583	-90 dBm
75	17 Juni 2023	06:20:09	67,2125	-90 dBm
76	17 Juni 2023	06:30:09	67,27083	-75 dBm

Lampiran 7. Data Pengujian Kapasitas Pada Keseluruhan Sistem 18 juni 2023

18 juni 2023				
No	Date	Time	Kapasitas (%)	RSSI
1	18 Juni 2023	18:00:09	67,49167	-76 dBm
2	18 Juni 2023	18:10:09	67,32083	-90 dBm
3	18 Juni 2023	18:20:08	67,29167	-90 dBm
4	18 Juni 2023	18:30:08	67,3	-90 dBm
5	18 Juni 2023	18:40:08	67,33333	-88 dBm
6	18 Juni 2023	18:50:08	67,3375	-89 dBm
7	18 Juni 2023	19:00:08	67,3125	-89 dBm
8	18 Juni 2023	19:10:08	67,33333	-90 dBm
9	18 Juni 2023	19:20:08	67,325	-87 dBm
10	18 Juni 2023	19:30:08	67,275	-88 dBm
11	18 Juni 2023	19:40:09	67,2625	-88 dBm
12	18 Juni 2023	19:50:09	67,275	-90 dBm
13	18 Juni 2023	20:00:09	67,3	-90 dBm
14	18 Juni 2023	20:10:09	67,25	-90 dBm
15	18 Juni 2023	20:20:09	67,37083	-88 dBm
16	18 Juni 2023	20:30:09	67,36667	-89 dBm
17	18 Juni 2023	20:40:09	67,30417	-89 dBm
18	18 Juni 2023	20:50:09	67,46667	-90 dBm
19	18 Juni 2023	21:00:09	67,37917	-87 dBm
20	18 Juni 2023	21:10:09	67,31667	-88 dBm
21	18 Juni 2023	21:20:09	67,35833	-88 dBm
22	18 Juni 2023	21:30:09	67,3375	-90 dBm
23	18 Juni 2023	21:40:09	67,28333	-90 dBm
24	18 Juni 2023	21:50:09	67,25	-90 dBm
25	18 Juni 2023	22:00:09	67,26667	-88 dBm
26	18 Juni 2023	22:10:09	67,28333	-89 dBm
27	18 Juni 2023	22:20:09	67,32083	-89 dBm
28	18 Juni 2023	22:30:09	67,3	-90 dBm
29	18 Juni 2023	22:40:09	67,47083	-87 dBm
30	18 Juni 2023	22:50:09	67,3375	-88 dBm
31	18 Juni 2023	23:00:09	67,3875	-88 dBm
32	18 Juni 2023	23:10:09	67,4125	-90 dBm
33	18 Juni 2023	23:20:09	67,38333	-90 dBm
34	18 Juni 2023	23:30:09	67,25833	-90 dBm
35	18 Juni 2023	23:40:09	67,4625	-88 dBm
36	18 Juni 2023	23:50:09	67,35417	-76 dBm
37	18 Juni 2023	00:00:09	67,27083	-90 dBm

38	18 Juni 2023	00:10:09	67,39583	-90 dBm
39	18 Juni 2023	00:20:09	67,41667	-90 dBm
40	18 Juni 2023	00:30:09	67,325	-88 dBm
41	18 Juni 2023	00:40:09	67,325	-89 dBm
42	18 Juni 2023	00:50:09	67,32083	-89 dBm
43	18 Juni 2023	01:00:09	67,2875	-90 dBm
44	18 Juni 2023	01:10:09	67,325	-87 dBm
45	18 Juni 2023	01:20:09	67,25417	-88 dBm
46	18 Juni 2023	01:30:09	67,5375	-88 dBm
47	18 Juni 2023	01:40:09	67,38333	-90 dBm
48	18 Juni 2023	01:50:09	67,38333	-90 dBm
49	18 Juni 2023	02:00:09	67,36667	-90 dBm
50	18 Juni 2023	02:10:09	67,32917	-88 dBm
51	18 Juni 2023	02:20:09	67,4125	-89 dBm
52	18 Juni 2023	02:30:09	67,2875	-89 dBm
53	18 Juni 2023	02:40:09	67,4125	-90 dBm
54	18 Juni 2023	02:50:09	67,25833	-87 dBm
55	18 Juni 2023	03:00:10	67,3625	-88 dBm
56	18 Juni 2023	03:10:10	67,45833	-88 dBm
57	18 Juni 2023	03:20:10	67,425	-90 dBm
58	18 Juni 2023	03:30:10	67,425	-90 dBm
59	18 Juni 2023	03:40:10	67,41667	-90 dBm
60	18 Juni 2023	03:50:10	67,35417	-88 dBm
61	18 Juni 2023	04:00:10	67,29167	-89 dBm
62	18 Juni 2023	04:10:10	67,35417	-89 dBm
63	18 Juni 2023	04:20:10	67,59583	-90 dBm
64	18 Juni 2023	04:30:09	67,38333	-87 dBm
65	18 Juni 2023	04:40:09	67,30417	-88 dBm
66	18 Juni 2023	04:50:09	67,32917	-88 dBm
67	18 Juni 2023	05:00:09	67,225	-90 dBm
68	18 Juni 2023	05:10:09	67,225	-90 dBm
69	18 Juni 2023	05:20:09	67,33333	-90 dBm
70	18 Juni 2023	05:30:09	67,24167	-88 dBm
71	18 Juni 2023	05:40:09	67,19167	-88 dBm
72	18 Juni 2023	05:50:09	67,225	-88 dBm
73	18 Juni 2023	06:00:09	67,25833	-90 dBm
74	18 Juni 2023	06:10:09	67,25833	-90 dBm
75	18 Juni 2023	06:20:09	67,25833	-90 dBm

Lampiran 8. Program Kalibrasi Sensor Thermocopel Type-K

```
#include "max6675.h"

int thermoDO = 19;
int thermoCS = 23;
int thermoCLK = 5;

MAX6675 thermocouple(thermoCLK, thermoCS, thermoDO);

void setup() {
  // put your setup code here, to run once:

  Serial.begin(115200);
  delay(1000);

  Serial.println();
  Serial.println("MAX6675 test");
  // wait for MAX chip to stabilize
  delay(1000);
}

void loop() {
  // put your main code here, to run repeatedly:

  Serial.println();
  Serial.println("-----");
  Serial.print("C = ");
  Serial.println(thermocouple.readCelsius());
  Serial.print("F = ");
  Serial.println(thermocouple.readFahrenheit());
  Serial.println("-----");

  // For the MAX6675 to update, you must delay AT LEAST 250ms between
  reads!
  delay(1000);
}
```

Lampiran 9. Hasil dari perhitungan nilai regresi

$$y = a + bx$$

$$a = \frac{(\sum x)(\sum y) - (\sum x)(\sum xy)}{n(\sum x^2) - (\sum x)^2}$$

$$a = \frac{1.368.0871 - 1.370.0119}{288.170 - 229.441}$$

$$a = \frac{19.247,75}{58.729}$$

$$a = 0,327738$$

Maka nilai konstantanya adalah $a = 0,327738$

$$\frac{n(\sum xy) - (\sum x)(\sum y)}{n(\sum x^2) - (\sum x)^2}$$

$$\frac{10(28.601,5) - (479)(474)}{10(28817) - (229441)}$$

$$b = \frac{589629,75}{58729}$$

$$b = 0,997969$$

Maka nilai koefisiennya adalah $b = 0,997969$

Lampiran 10. Firmware pada perangkat master 1

```

#include <esp_now.h>
#include <esp_wifi.h>
#include <WiFi.h>
#include "max6675.h"
#define CHANNEL 1
uint8_t broadcastAddress[] = {0xB0, 0xB2, 0x1C, 0x97, 0x9A, 0xF0}; //address
yang digunakan address dari slave
String Master_ID = "#1";
unsigned long previousMillis_K_Type_Thermocouple = 0;
const long interval_K_Type_Thermocouple = 1000;
unsigned long previousMillis_Send = 0;
const long interval_Send = 10000; //waktu pengiriman
int thermoDO = 19;
int thermoCS = 2023;
int thermoCLK = 5;
String thermocouple_celcius_Send = "UN";
typedef struct struct_message_send {
  String ID_Sender;
  String Data_Send;
} struct_message_send;
struct_message_send send_Data;
MAX6675 thermocouple(thermoCLK, thermoCS, thermoDO);
void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
  Serial.print("\r\nStatus pengiriman data terakhir : \t");
  Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Delivery Success" :
"Delivery Fail");
  Serial.println("  ");
}
void read_K_Type_Thermocouple() {
  thermocouple_celcius_Send = String(thermocouple.readCelsius());
}
}

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  delay(1000);
  Serial.println();

```

```

Serial.println("Atur wifi ke mode Wi Fi Station.");
WiFi.mode(WIFI_STA);
int cur_WIFIchannel = WiFi.channel();
if (cur_WIFIchannel != CHANNEL) {
  esp_wifi_set_promiscuous(true);
  esp_wifi_set_channel(CHANNEL, WIFI_SECOND_CHAN_NONE);
  esp_wifi_set_promiscuous(false);
}
Serial.println();
Serial.print("WiFi Channel : ");
Serial.println(WiFi.channel());
Serial.println();
Serial.print("Mac Address : ");
Serial.println(WiFi.macAddress());
Serial.println();
Serial.println(" ");
Serial.println("Mulai menginisialisasi ESP NOW...");
if (esp_now_init() != ESP_OK) {
  Serial.println("Error initializing ESP NOW");
  Serial.println("Restart ESP32...");
  Serial.println(" ");
  delay(1000);
  ESP.restart();
}
Serial.println("Inisialisasi ESP NOW berhasil.");
Serial.println(" ");
}
esp_now_register_send_cb(OnDataSent);
esp_now_peer_info_t peerInfo;
memcpy(peerInfo.peer_addr, broadcastAddress, 6);
peerInfo.encrypt = false;
Serial.println();
Serial.println(" ");
Serial.println("Mulai add Peers...");
if (esp_now_add_peer(&peerInfo) != ESP_OK){
  Serial.println("Add peer gagal !");
  Serial.println(" ");
  return;
}
Serial.println("Add Peer berhasil.");

```



```

Serial.println("      ");
}
void loop() {
  // put your main code here, to run repeatedly:
  unsigned long currentMillis_K_Type_Thermocouple = millis();
  if (currentMillis_K_Type_Thermocouple
previousMillis_K_Type_Thermocouple >= interval_K_Type_Thermocouple) {
  previousMillis_K_Type_Thermocouple =
currentMillis_K_Type_Thermocouple;

  read_K_Type_Thermocouple(); //pembacaan suhu
}
  unsigned long currentMillis_Send = millis();
  if (currentMillis_Send - previousMillis_Send >= interval_Send) {
    previousMillis_Send = currentMillis_Send;
    Serial.println();
    Serial.println("      ");
    Serial.print("Thermocouple : ");
    Serial.print(thermocouple_celcius_Send);
    Serial.println(" °C");
    Serial.println("      ");
    Serial.println();
    Serial.println("  Send data");
    send_Data.ID_Sender = Master_ID;
    send_Data.Data_Send = thermocouple_celcius_Send;
    esp_err_t result = esp_now_send(broadcastAddress, (uint8_t *) &send_Data,
sizeof(send_Data));

    if (result == ESP_OK) {
      Serial.println("Sent with success");
    }
    else {
      Serial.println("Error sending the data");
    }
  }
}
}

```

Lampiran 11. Firmware pada perangkat Master 2

```

#include <esp_now.h>
#include <esp_wifi.h>
#include <WiFi.h>
#define CHANNEL 1
#define RXD2 16
#define TXD2 17
uint8_t broadcastAddress[] = {0xB0, 0xB2, 0x1C, 0x97, 0x9A, 0xF0}; // #
String Master_ID = "#2";
unsigned char data[4]={};
float distance_cm;
float kapasitas;
String kapasitas_Send = "UN";
unsigned long previousMillis_Ultrasonic = 0;
const long interval_Ultrasonic = 100;
unsigned long previousMillis_Send = 0;
const long interval_Send = 10000; //waktu pengiriman
typedef struct struct_message_send {
  String ID_Sender;
  String Data_Send;
} struct_message_send;

// Membuat struct_message untuk menerima data.
struct_message_send send_Data;
void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
  Serial.print("\r\nStatus pengiriman data terakhir : \t");
  Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Delivery Success" :
"Delivery Fail");
  Serial.println(" ");
}
void read_Ultrasonic_A02YYUW() {
  do{
    for(int i=0;i<4;i++) {
      data[i]=Serial2.read();
    }
  } while(Serial2.read()!==0xff);

  Serial2.flush();

  if(data[0]==0xff) {
    int sum;
    sum=(data[0]+data[1]+data[2])&0x00FF;
    if(sum==data[3]) {
      distance_cm=(data[1]<<8)+data[2];
      kapasitas = (240.0 distance_cm/10) / 240.0 * 100.0;

```

```

if(distance_cm>30) {
    kapasitas_Send = String(kapasitas);
} else {
    kapasitas_Send = "BLL";
}
}
}
}

void setup() {
    // put your setup code here, to run once:
    Serial.begin(115200);
    Serial2.begin(9600, SERIAL_8N1, RXD2, TXD2);
    delay(1000);
    Serial.println();
    Serial.println("Atur wifi ke mode Wi Fi Station.");
    WiFi.mode(WIFI_STA);
    int cur_WiFiChannel = WiFi.channel();
    if (cur_WiFiChannel != CHANNEL) {
        esp_wifi_set_promiscuous(true);
        esp_wifi_set_channel(CHANNEL, WIFI_SECOND_CHAN_NONE);
        esp_wifi_set_promiscuous(false);
    }
    Serial.println();
    Serial.print("WiFi Channel : ");
    Serial.println(WiFi.channel());
    Serial.println();
    Serial.print("Mac Address : ");
    Serial.println(WiFi.macAddress());
    Serial.println();
    Serial.println(" ");
    Serial.println("Mulai menginisialisasi ESP NOW...");
    if (esp_now_init() != ESP_OK) {
        Serial.println("Error initializing ESP NOW");
        Serial.println("Restart ESP32...");
        Serial.println(" ");
        delay(1000);
        ESP.restart();
    }
    Serial.println("Inisialisasi ESP NOW berhasil.");
    Serial.println(" ");
    esp_now_register_send_cb(OnDataSent);
    esp_now_peer_info_t peerInfo;
    memcpy(peerInfo.peer_addr, broadcastAddress, 6);
    peerInfo.encrypt = false;
    Serial.println();

```

```

Serial.println(" ");
Serial.println("Mulai add Peers...");
if (esp_now_add_peer(&peerInfo) != ESP_OK){
  Serial.println("Add peer gagal !");
  Serial.println(" ");
  return;
}
Serial.println("Add Peer berhasil.");
Serial.println(" ");
}

void loop() {
  // put your main code here, to run repeatedly:
  unsigned long currentMillis_Ultrasonic = millis();
  if (currentMillis_Ultrasonic - previousMillis_Ultrasonic >= interval_Ultrasonic)
  {
    previousMillis_Ultrasonic = currentMillis_Ultrasonic;

    read_Ultrasonic_A02YYUW();
  }
  unsigned long currentMillis_Send = millis();
  if (currentMillis_Send - previousMillis_Send >= interval_Send) {
    previousMillis_Send = currentMillis_Send;

    Serial.println();
    Serial.println(" ");
    Serial.print("Kapasitas : ");
    Serial.print(kapasitas_Send);
    Serial.println(" %");
    Serial.println(" ");

    Serial.println();
    Serial.println("  Send data");
    send_Data.ID_Sender = Master_ID;
    send_Data.Data_Send = kapasitas_Send;
    esp_err_t result = esp_now_send(broadcastAddress, (uint8_t *) &send_Data,
    sizeof(send_Data));

    if (result == ESP_OK) {
      Serial.println("Sent with success");
    }
    else {
      Serial.println("Error sending the data");
    }
  }
}
}

```

Lampiran 12. Program Kalibrasi Sensor Ultrasonik A002YYUW

```

#define RXD2 16
#define TXD2 17

unsigned char data[4]={};
float distance;

void setup() {
  // put your setup code here, to run once:

  Serial.begin(115200);
  Serial2.begin(9600, SERIAL_8N1, RXD2, TXD2);
  delay(1000);
}

void loop() {
  // put your main code here, to run repeatedly:

  do{
    for(int i=0;i<4;i++) {
      data[i]=Serial2.read();
    }
  } while(Serial2.read()!==0xff);

  Serial2.flush();

  if(data[0]==0xff) {
    int sum;
    sum=(data[0]+data[1]+data[2])&0x00FF;
    if(sum==data[3]) {
      distance=(data[1]<<8)+data[2];
      if(distance>30) {
        Serial.print("distance=");
        Serial.print(distance/10);
        Serial.println("cm");
      } else {
        Serial.println("Below the lower limit");
      }
    } else Serial.println("ERROR");
  }

  delay(100);
}

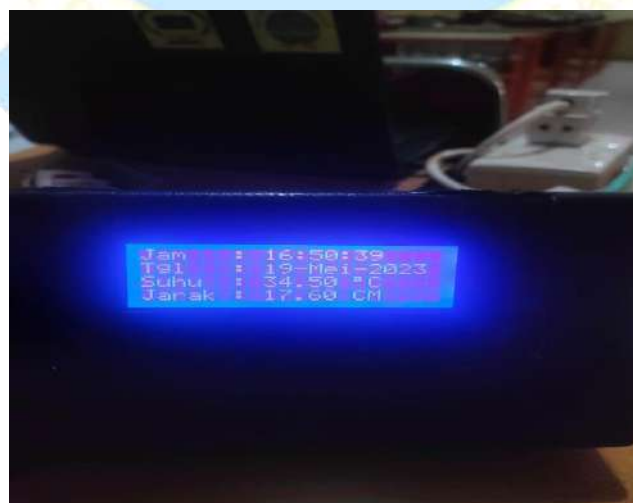
```

Lampiran 13. Dokumentasi Penelitian

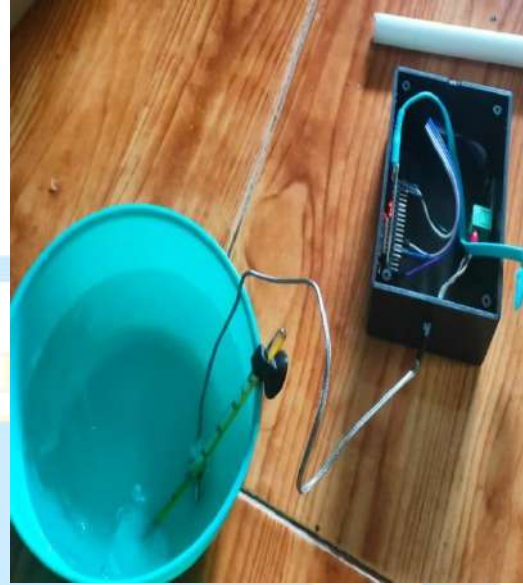
a. Lokasi Penelitian



b. Pengujian jarak ESP NOW



c. Proses Pengkalibrasian Sensor



d. Pengujian Sensor Secara Keseluruhan





BIODATA



Afdal Hadi, dilahirkan di Kota Lhokseumawe pada tanggal 06 juni 2000. Anak pertama dari lima bersaudara dari pasangan Karimudin Pala dan Hafni.

Pada tahun 2012, penulis menamatkan sekolah SD Negeri 10 Banda Sakti Lhokseumawe dan melanjutkan pendidikan MTSS Ulumuddin Lhokseumawe Penulis menyelesaikan pendidikan Sekolah Madrasah Tasnawiyah Swasta Ulummuddin pada tahun 2015. Pada tahun yang sama pula penulis melanjutkan ke SMK Swasta Ulumuddin Lhokseumawe dan tamat pada tahun 2018. Pada tahun 2018, penulis diterima sebagai mahasiswa Universitas Maritim Raja Ali Haji di Jurusan Teknik Elektro Fakultas Teknik dan Teknologi Kemaritiman.

Selama menjadi mahasiswa, penulis bergabung kedalam kepengurusan Lembaga Dakwah Fakultas (LDF) pada tahun 2018, Lalu penulis melanjutkan kepengurusan di Himpunan Mahasiswa Teknik Elektro pada tahun 2019. Pada tahun 2019 penulis mencoba bergabung dalam keorganisasian external kampus yaitu Himpunan Mahasiswa Islam (HMI), penulis di angkat menjadi Kepala Bidang P3A. Ditahun 2020 penulis terpilih sebagai Ketua Himpunan Mahasiswa Teknik Elektro. Pada tahun 2022 penulis di angkat sebagai sekretaris 2 di PERMASA. Penulis juga pernah mengikuti beberapa perlombaan baik LKTI,TTG, Syarhil Quran dan lain diantaranya.

Penulis menyelesaikan studi di Fakultas Teknik dan Teknologi Kemaritiman Universitas Maritim Raja Ali Haji dengan melakukan penelitian yang berjudul “Sistem Monitoring Suhu Air Pendingin dan Kapasitas Bahan Bakar Dalam Tangki Pembangkit Listrik Tenaga Diesel (PLTD) Berbasis ESP NOW di Pulau Pangkil”.