

TUGAS AKHIR

**SISTEM PEMANTAUAN DAN PENGENDALIAN
SUHU DAN KADAR OKSIGEN PADA MODEL
TAMBAK UDANG VANAME DENGAN TEKNOLOGI
IOT**

Diajukan untuk memenuhi syarat

Memperoleh gelar Sarjana Teknik pada

Jurusan Teknik Elektro

Fakultas Sains dan Teknologi Universitas Sanata Dharma



Disusun oleh:

PATRA L TOBING

NIM: 205114043

**JURUSAN TEKNIK ELEKTRO
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS SANATA DHARMA
YOGYAKARTA**

2024

FINAL PROJECT

**MONITORING AND CONTROLLING SYSTEM OF
TEMPERATURE AND OXYGEN LEVEL IN
VANAME SHRIMP POND MODEL WITH IOT
TECHNOLOGY**

In partial fulfillment of the requirements

To achieve the degree of Bachelor of Engineering

Electrical Engineering Study Program

Faculty of Science and Technology, Sanata Dharma University



PATRA L TOBING
NIM: 205114043

**DEPARTMENT OF ELECTRICAL ENGINEERING
FACULTY OF SCIENCE AND TECHNOLOGY
SANATA DHARMA UNIVERSITY
YOGYAKARTA
2024**

LEMBAR PERSETUJUAN

TUGAS AKHIR

**SISTEM PEMANTAUAN DAN PENGENDALIAN
SUHU DAN KADAR OKSIGEN PADA MODEL
TAMBAK UDANG VANAME DENGAN TEKNOLOGI
IOT
(MONITORING AND CONTROLLING SYSTEM OF
TEMPERATURE AND OXYGEN LEVEL IN VANAME
SHRIMP POND MODEL WITH IOT TECHNOLOGY)**

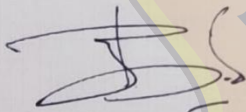
Oleh:

PATRAL TOBING

NIM: 205114043

Telah disetujui oleh:

Pembimbing I



Ir. Damar Widjaja Ph.D.

Tanggal: 25 Juli 2024

LEMBAR PENGESAHAN

TUGAS AKHIR

**SISTEM PEMANTAUAN DAN PENGENDALIAN
SUHU DAN KADAR OKSIGEN PADA MODEL
TAMBAK UDANG VANAME DENGAN TEKNOLOGI
IOT
(MONITORING AND CONTROLLING SYSTEM OF
TEMPERATURE AND OXYGEN LEVEL IN VANAME
SHRIMP POND MODEL WITH IOT TECHNOLOGY)**

Disusun oleh:

PATRA L TOBING

NIM: 205114043

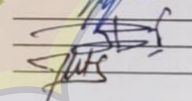
Telah dipertahankan di depan tim penguji
pada tanggal 11 Juli 2024
dan dinyatakan memenuhi syarat

Susunan Tim Penguji:

Nama Lengkap

Ketua : Ir. A. Bayu Primawan, D. Tech. Sc.
Sekretaris : Ir. Damar Widjaja, Ph.D.
Anggota : Ir. Djoko Untoro Suwarno, S.Si., M.T.

Tanda Tangan



Yogyakarta, 26 Juli 2024

Fakultas Sains dan Teknologi
Universitas Sanata Dharma
Dekan,

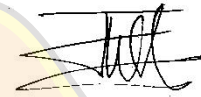


Ir. Drs. Haris Sriwindono, M. Kom., Ph.D.

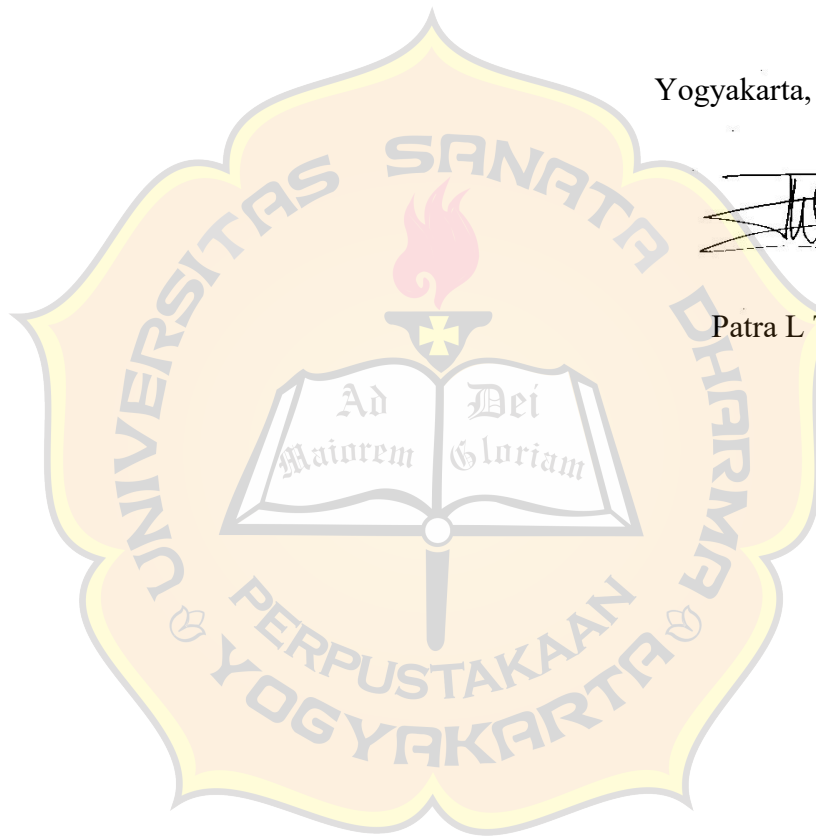
LEMBAR PERNYATAAN KEASLIAN KARYA

Saya menyatakan dengan sesungguhnya bahwa tugas akhir yang berjudul “SISTEM PEMANTAUAN DAN PENGENDALIAN SUHU DAN KADAR OKSIGEN PADA MODEL TAMBAK UDANG VANAME DENGAN TEKNOLOGI IOT” ini tidak memuat karya atau bagian karya orang lain, kecuali yang telah disebutkan dalam kutipan dan daftar pustaka sebagaimana layaknya karya ilmiah.

Yogyakarta, 5 Juli 2024



Patra L Tobing



LEMBAR HALAMAN PERSEMBAHAN DAN MOTTO HIDUP

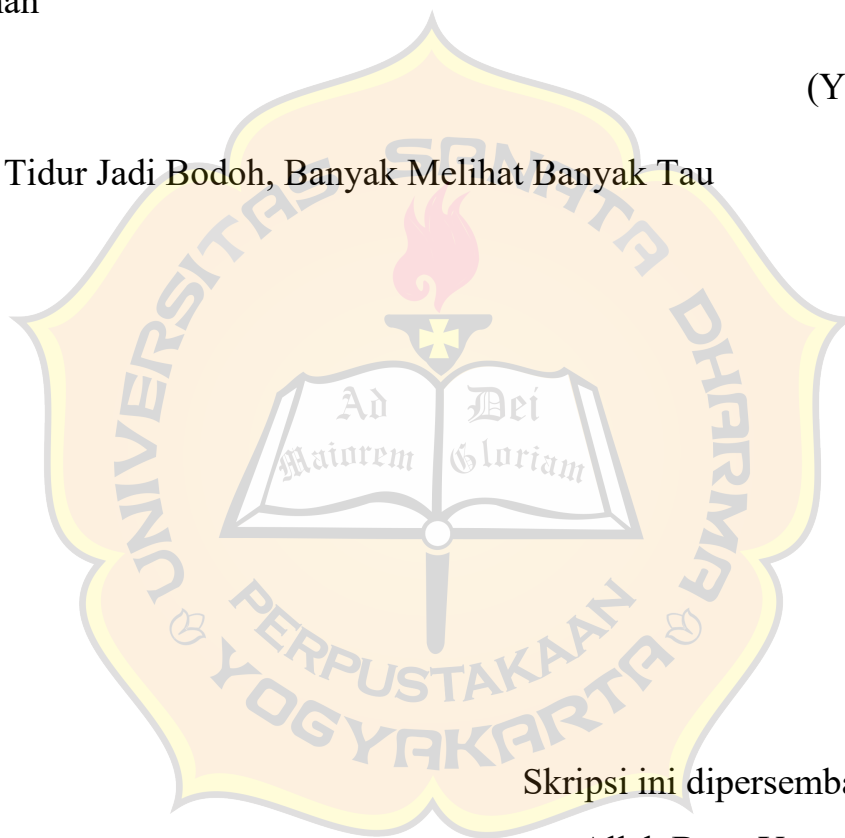
“Takut akan TUHAN adalah permulaan pengetahuan, tetapi orang bodoh menghina hikmat dan didikan”

(amsal 1:7)

"Diberkatilah orang yang mengandalkan Tuhan, yang menaruh harapannya pada Tuhan"

(Yeremia 17:7)

“Banyak Tidur Jadi Bodoh, Banyak Melihat Banyak Tau



Skripsi ini dipersembahkan untuk:

Allah Bapa Yang Maha Kuasa

Kedua Orang Tua yang aku cintai

Kakak dan Abang yang aku sayangi

LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS

Yang bertanda tangan di bawah ini, saya mahasiswa Universitas Sanata Dharma:

Nama : PATRA L TOBING

NIM : 205114043

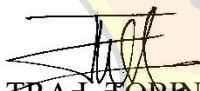
Demi pengembangan ilmu pengetahuan, saya memberikan kepada perpustakaan Universitas Sanata Dharma karya ilmiah saya yang berjudul:

SISTEM PEMANTAUAN DAN PENGENDALIAN SUHU DAN KADAR OKSIGEN PADA MODEL TAMBAK UDANG VANAME DENGAN TEKNOLOGI IOT

Beserta perangkat yang diperlukan (bila ada). Dengan demikian saya memberikan kepada Perpustakaan Universitas Sanata Dharma hak untuk menyimpan, mengalihkan dalam bentuk media lain, mengelolanya dalam bentuk pangkalan data, mendistribusikan secara terbatas, dan mempublikasikannya di Internet atau media lain untuk kepentingan akademis tanpa perlu meminta izin dari saya maupun memberikan royalti kepada saya selama tetap mencantumkan nama saya sebagai penulis.

Demikian pernyataan ini yang saya buat dengan sebenarnya.

Yogyakarta, 5 Juli 2024


PATRA L TOBING

INTISARI

Udang vaname (*Litopenaeus vannamei*) adalah komoditas perikanan dengan nilai ekonomi tinggi yang memiliki peran penting di pasar domestik dan global. Budidaya udang ini di Indonesia menggunakan metode intensif untuk meningkatkan produktivitas tambak dengan cepat. Namun, budidaya ini menghadapi berbagai tantangan seperti kualitas air yang buruk, penyakit, pakan tidak memadai, dan perubahan iklim yang dapat menyebabkan gagal panen. Pengendalian kualitas air, terutama kadar oksigen terlarut dan suhu, adalah faktor kunci dalam keberhasilan budidaya udang vaname.

Penelitian ini bertujuan untuk merancang sistem pemantauan dan pengendalian suhu serta kadar oksigen pada tambak udang dengan memanfaatkan teknologi Internet of Things (IoT). Sistem ini menggunakan Arduino Mega sebagai pengendali utama dan ESP32 sebagai pengirim data ke platform Blynk. Sensor yang digunakan meliputi dua sensor DO SKU SEN 0237 untuk mengukur kadar oksigen terlarut dan dua sensor DS18B20 untuk mengukur suhu air. Data yang diperoleh dari sensor diproses oleh Arduino Mega dan diteruskan ke ESP32 untuk dikirimkan ke platform Blynk, yang memungkinkan pemantauan dan pengendalian jarak jauh secara real-time melalui smartphone.

Keunggulan sistem ini adalah kemampuannya untuk tetap beroperasi meskipun terjadi gangguan koneksi internet, karena pemantauan masih dapat dilakukan melalui Arduino Mega. Selain itu, sistem ini dilengkapi dengan empat motor DC 12V yang akan memompa air umpan ke dalam bak uji jika terdeteksi kondisi abnormal. Sensor oksigen SKU SEN0237 digunakan karena dapat mengukur kadar oksigen terlarut tanpa dipengaruhi oleh oksigen di udara sekitar, sementara motor DC digunakan untuk penanganan kondisi air yang tidak normal.

Hasil penelitian menunjukkan bahwa sistem yang dirancang mampu bekerja dengan baik dalam mendeteksi dan mengontrol kondisi air di tambak udang. Dengan adanya sistem ini, diharapkan keberhasilan budidaya udang vaname dapat meningkat melalui pengendalian kualitas air yang lebih baik dan pemantauan yang lebih efisien.

Kata kunci: Udang vaname, kualitas air, kendali oksigen dan suhu.

ABSTRACT

Vaname shrimp (*Litopenaeus vannamei*) is a fishery commodity with high economic value that plays an important role in domestic and global markets. This shrimp farming in Indonesia uses intensive methods to increase pond productivity quickly. However, this farming faces various challenges such as poor water quality, disease, inadequate feed, and climate change that can lead to crop failure. Water quality control, especially dissolved oxygen levels and temperature, are key factors in the success of vaname shrimp farming.

This research aims to design a system for monitoring and controlling temperature and oxygen levels in shrimp ponds by utilising Internet of Things (IoT) technology. This system uses Arduino Mega as the main controller and ESP32 as the data sender to the Blynk platform. The sensors used include two DO SKU SEN 0237 sensors to measure dissolved oxygen levels and two DS18B20 sensors to measure water temperature. The data obtained from the sensors is processed by Arduino Mega and forwarded to ESP32 for transmission to the Blynk platform, which allows real-time remote monitoring and control via smartphone.

The advantage of this system is its ability to continue operating despite internet connection interruptions, as monitoring can still be done through Arduino Mega. In addition, the system is equipped with four 12V DC motors that will pump feed water into the test basin if abnormal conditions are detected. The oxygen sensor SKU SEN0237 is used because it can measure dissolved oxygen levels without being affected by oxygen in the surrounding air, while the DC motors are used for handling abnormal water conditions.

The results show that the designed system can effectively detect and control water conditions in shrimp ponds. With this system, it is expected that the success of vaname shrimp farming can be increased through better water quality control and more efficient monitoring.

Keywords: Vaname shrimp, water quality, oxygen and temperature control.

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yesus Kristus atas hikmat dan anugerah yang Ia berikan sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul “SISTEM PEMANTAUAN DAN PENGENDALIAN SUHU DAN KADAR OKSIGEN PADA MODEL TAMBAK UDANG VANAME DENGAN TEKNOLOGI IOT”.

Tugas Akhir ini disusun sebagai salah satu syarat kelulusan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Sains dan Teknologi, Universitas Sanata Dharma Yogyakarta. Penulisan Tugas Akhir ini tidak lepas dari campur tangan dan bantuan dari banyak pihak. Penulis menyadari bahwa ada banyak pihak yang turut memberikan dukungan, bimbingan serta arahan sehingga penelitian ini dapat diselesaikan dengan baik. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada:

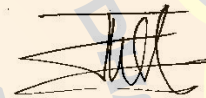
1. Tuhan Yesus Kristus, yang senantiasa menolong, membimbing dan menyertai setiap langkahku.
2. Bapak Ir. Drs. Haris Sriwindono, M.Kom., Ph.D., selaku Dekan Fakultas Sains dan Teknologi Universitas Sanata Dharma.
3. Bapak Ir. Martanto, M.T., selaku Ketua Program Studi Teknik Elektro, Fakultas Sains dan Teknologi, Universitas Sanata Dharma.
4. Bapak Damar Widjaja Ph.D., selaku Dosen Pembimbing Tugas Akhir yang selalu memberikan masukan dan dorongan, kepada penulis untuk berkembang dan berproses, selalu sabar dan meluangkan waktunya untuk bimbingan sehingga tugas akhir dapat diselesaikan dengan baik.
5. Bapak Ir. Augustinus Bayu Primawan D.Tech.Sc. Dan Bapak Ir. Djoko Untoro Suwarno S.Si., M.T. selaku Dosen Penguji tugas akhir yang telah memberi masukan, bimbingan serta saran untuk menyempurnakan penulisan tugas akhir ini.
6. Bapak dan ibu dosen yang mengajarkan banyak hal dan memberikan pengalaman dalam proses pembelajaran selama penulis menempuh pendidikan di Program Studi Teknik Elektro, Fakultas Sains dan Teknologi, Universitas Sanata Dharma.
7. Orang tua dan seluruh keluarga yang senantiasa mendukung dan mendoakan yang terbaik sehingga penulis dapat menyelesaikan tugas akhir ini.
8. Semua teman-teman yang sudah memberikan dukungannya selama penulisan tugas akhir ini.

9. Seluruh staf dan karyawan Universitas Sanata Dharma atas pelayanan yang diberikan untuk mahasiswa.
10. Seluruh pihak yang tidak dapat disebutkan satu persatu yang membantu penyelesaian tugas akhir ini.

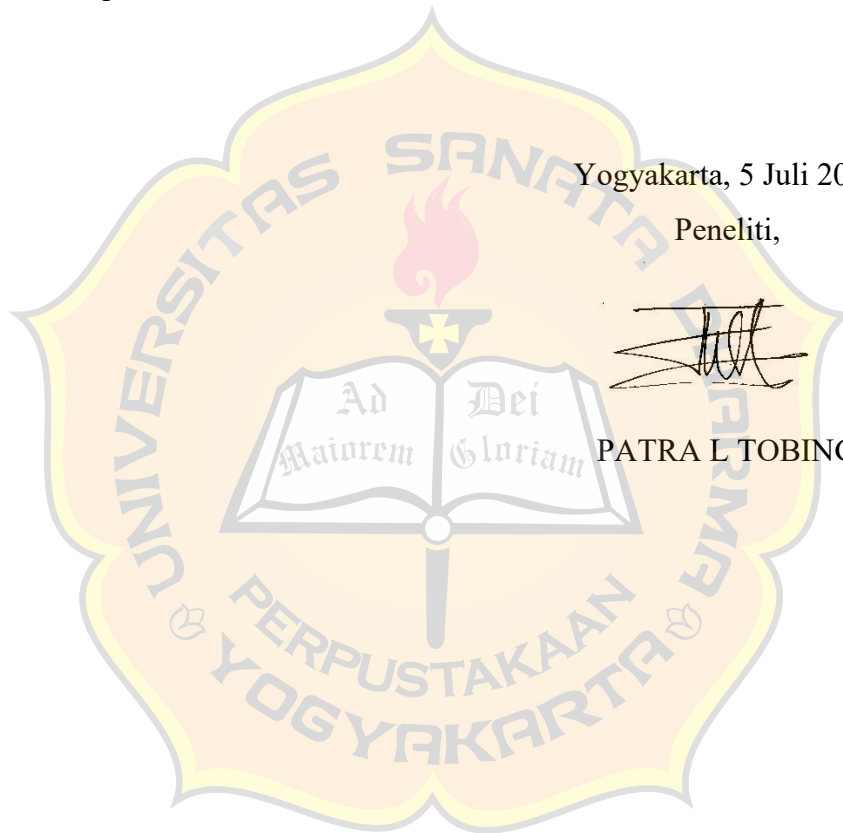
Penulis menyadari bahwa dalam penulisan tugas akhir ini, masih banyak terdapat kekurangan. Oleh karena itu, penulis mengharapkan adanya masukan, kritik dan saran yang bersifat membangun untuk menyempurnakan kekurangan tugas akhir ini. Penulis berharap tugas akhir ini dapat bermanfaat untuk penelitian selanjutnya terkait dengan topik yang diambil oleh penulis. Sekian dan Terimakasih.

Yogyakarta, 5 Juli 2024

Peneliti,



PATRA L TOBING



DAFTAR ISI

Halaman Sampul (Bahasa Indonesia)	ii
Halaman Sampul (Bahasa Inggris)	iii
Halaman Pengesahan	iv
Halaman Pernyataan Keaslian Karya	v
Halaman Persembahan	vi
Lembar Publikasi Karya Ilmiah	vii
INTISARI	viii
ABSTRACT	ix
KATA PENGANTAR	x
DAFTAR ISI	xii
DAFTAR GAMBAR	xiv
DAFTAR TABEL	xv
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Tujuan dan Manfaat	3
1.3. Batasan Masalah	3
1.4. Metodologi Penelitian	4
BAB II DASAR TEORI	6
2.1. Kualitas Air	6
2.2. Dasar Sistem Kendali	7
2.2.1. Sistem Kendali Loop Tertutup	7
2.3. Mikrokontroler	8
2.3.1. Arduino Mega 2560	8
2.3.2. NodeMCU ESP 32	9
2.3.3. Arduino <i>IDE</i>	10
2.4. Sensor	11
2.4.1. Sensor Dissolved Oxygen (DO)	11
2.4.2. Sensor Suhu DS18B20	13
2.5. Aktuator	13
2.5.1. Pompa Air	13
2.5.2. Aerator	14
2.5.3. <i>Relay</i>	15

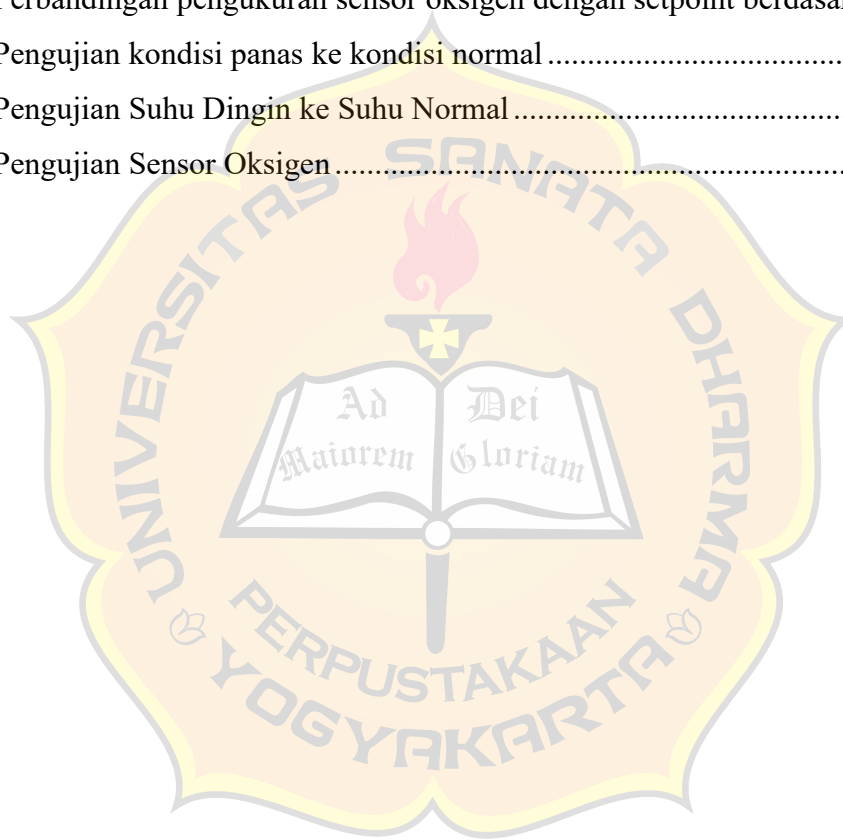
2.6. <i>Internet Of Things</i>	15
2.6.1. <i>ByInk Apps</i>	16
BAB III RANCANGAN PENELITIAN	17
3.1. Diagram Blok Sistem	17
3.2. Perancangan Perangkat Keras Mekanis	18
3.3. Perancangan Perangkat Elektronis	19
3.3.1. Rangkaian ESP 32 dengan Arduino Mega	20
3.3.2. Sensor DO (SKU SEN0237)	22
3.3.3. Sensor Suhu (DS18B20)	23
3.3.4. Perancangan OLED	24
3.4. Perancangan Pompa Air	25
3.5. Perancangan Perangkat Lunak	27
3.5.1. <i>Flowchart</i> Sistem keseluruhan	27
3.5.2. <i>Flowchart</i> Sistem pada Oled	29
3.5.3. Desain GUI pada Aplikasi Biynk	29
BAB IV HASIL DAN PEMBAHASAN	31
4.1 Implementasi Perangkat Keras	31
4.2 Hasil Pengujian Sistem	33
4.2.1 Pengukuran Nilai pada Sensor Oksigen dan Sensor Suhu	36
4.2.2 Pengujian Suhu Panas ke Suhu Normal	37
4.2.3 Pengujian Suhu Dingin ke Suhu Normal	38
4.2.4 Pengujian Sensor Oksigen	40
4.3 Implementasi Perangkat Lunak	42
4.3.1 Program pada Arduino Mega	42
4.3.2 Program pada ESP 32	45
BAB V KESIMPULAN DAN SARAN	47
5.1. Kesimpulan	47
5.2. Saran	47
DAFTAR PUSTAKA	48
LAMPIRAN	50

DAFTAR GAMBAR

Gambar 2. 1 Diagram Sistem Kendali Loop Tertutup	7
Gambar 2. 3 Arduino Mega 2560	8
Gambar 2.4. ESP 32	10
Gambar 2. 5 Arduino IDE	10
Gambar 2. 6 Hubungan temperatur dengan tegangan kalibrasi	11
Gambar 2.7. Sensor Dissolved Oxygen (DO)	12
Gambar 2.8. Sensor Suhu DSI8B20	13
Gambar 2.9. Pompa Air DC 12V	14
Gambar 2.10. Aerator	14
Gambar 2.11. <i>Relay</i> empat Channel	15
Gambar 2.12. IoT	15
Gambar 2. 13. Blynk Server	16
Gambar 3.1. Diagram Blok Sistem Secara Umum.....	17
Gambar 3.2. Perancangan Kolam Dua Tingkat Tampak Depan.....	18
Gambar 3.3. Perancangan Kolam Dua Tingkat Sisi Samping.....	18
Gambar 3.4. Rangkaian Elektronis Keseluruhan Sistem.....	20
Gambar 3.5. Schematic Rangkaian ESP 32 dengan Arduino Mega.....	21
Gambar 3. 6. Pemasangan Pin ESP pada Arduino Mega.....	21
Gambar 3. 7. Schematic DO pada Arduino.....	22
Gambar 3.8. Pemasangan Pin Sensor pada Arduino Mega.....	22
Gambar 3.9. Pemasangan Pin Sensor Suhu pada Arduino Mega.....	23
Gambar 3.10. Schematic Rangkaian Sensor Suhu dan Arduino Mega.....	23
Gambar 3.11. Pemasangan Pin OLED pada Arduino Mega.....	24
Gambar 3.12. Schematic OLED dan Arduino Mega.....	24
Gambar 3.13. Pemasangan Pin pada <i>Relay</i> ke Arduino Mega; <i>Relay</i> ke Motor; <i>Relay</i> ke Power Supply.....	25
Gambar 3.14 diagram perkabelan Arduino Mega, <i>Relay</i> , Pompa DC dan Power Supply...26	
Gambar 3.15. Flowchart Sistem secara umum.....	27
Gambar 3.16 Gambar 3.16. Flowchart Pengendalian Pada Arduino Mega.....	28
Gambar 3.18. Flowchart Sistem pada Oled.....	29
Gambar 3.19. Desain GUI pada Aplikasi ByInk.	30
Gambar 4.1 Keseluruhan Sistem.....	31
Gambar 4.2 Kotak Panel.....	32
Gambar 4.3 Grafik Kenaikan Suhu.....	39
Gambar 4.4 Kenaikan Kadar Oksigen Terlarut.....	42
Gambar 4.5 Program Library Arduino Mega 2560.....	42
Gambar 4.6 Deklarasi pin pada Arduino Mega.....	43
Gambar 4.7 Void Setup.....	43
Gambar 4.8 Void Setup.....	44
Gambar 4.9 Program On Off <i>Relay</i>	44
Gambar 4.10 Program Pembacaan Nilai Sensor.....	45
Gambar 4.11 Tampilan pada Oled.....	45
Gambar 4.12 Program deklarasi pin dan <i>library</i> pada ESP 32.....	46
Gambar 4.13 Program untuk mengirim data ke <i>blynk apps</i>	46

DAFTAR TABEL

Tabel 2.1 Nilai <i>Setpoint</i> Kadar Oksigen Terlarut Berdasarkan suhu.	12
Tabel 3.1. Alamat Pin ESP dengan Arduino Mega	21
Tabel 3.2. Pemasangan Pin Sensor DO pada Arduino Mega.	23
Tabel 3.3. Pemasangan Pin Sensor Suhu pada Arduino Mega.	24
Tabel 3.4. Pemasangan Pin OLED pada Arduino Mega.	25
Tabel 4. 1 Keterangan Bagian Perangkat	32
Tabel 4. 2 Perbandingan pengukuran sensor suhu dengan alat ukur pembanding	36
Tabel 4.3 Perbandingan pengukuran sensor oksigen dengan setpoint berdasarkan suhu	36
Tabel 4.4 Pengujian kondisi panas ke kondisi normal	37
Tabel 4.5 Pengujian Suhu Dingin ke Suhu Normal	39
Tabel 4.6 Pengujian Sensor Oksigen	41



BAB I

PENDAHULUAN

1.1. Latar Belakang

Udang vaname (*Litopenaeus vannamei*) termasuk jenis komoditas perikanan dengan nilai ekonomis tinggi baik di pasar domestik maupun global. Budidaya udang vaname memiliki peranan yang penting untuk memenuhi kebutuhan pasar dalam dan luar negeri. Sistem Budidaya udang secara intensif telah menjadi pola budidaya yang dilakukan sebagian besar petambak Indonesia dalam meningkatkan produktifitas tambak secara cepat. Indonesia tercatat menempati urutan kedua produsen perikanan budidaya dunia. Nilai produksi perikanan budidaya rata-rata naik sebesar 19,2% per tahun. Komoditas unggulan ekspor dari perikanan budidaya adalah udang, rumput laut, tilapia, mutiara, kerapu, dan ikan hias. Produksi perikanan budidaya Indonesia terus menunjukkan tren yang positif. Peningkatan produksi perikanan budidaya secara langsung juga dapat ditunjang dengan meningkatkan pengetahuan dan keterampilan dari masyarakat maupun tenaga kerja[1].

Air merupakan sumber daya alam yang sangat penting bagi kelangsungan hidup ikan agar dapat hidup sehat dan tumbuh secara maksimal. Kadar oksigen dan suhu dalam air budidaya harus diperhatikan dengan baik. Jika kadar oksigen dan suhu tidak diperhatikan maka akan berakibat fatal pada metabolisme tubuh ikan sehingga energi ikan untuk bergerak, berkembang, dan bereproduksi akan terganggu. Oksigen terlarut merupakan parameter kunci kualitas air. Tersedianya oksigen terlarut dalam air sangat menentukan kehidupan udang dan ikan. Kadar oksigen terlarut yang rendah dapat berpengaruh terhadap fungsi dan lambatnya pertumbuhan, bahkan dapat mengakibatkan kematian, Permasalahan yang sering muncul bagi petani adalah cuaca ekstrem, cuaca ekstrim merupakan salah satu penyebab terjadinya gagal panen bagi petani. Perubahan suhu yang tidak normal akan mengakibatkan kemampuan biota air untuk bertahan hidup semakin berkurang[2].

Pada tahun 2022-2023 sektor budidaya udang masih menghadapi berbagai tantangan yang menyebabkan terjadinya gagal panen. Beberapa faktor utama yang menjadi penyebab gagal panen yaitu; kualitas air yang buruk, penyakit akibat kualitas air yang tidak optimal, pakan yang tidak memadai, pengelolaan tambak yang kurang baik, dan perubahan iklim[3][4].

Pada dasarnya budidaya ikan terdapat empat parameter mendasar untuk menjaga kelangsungan hidup ikan antara lain tingkat Ph, salinitas, oksigen terlarut, dan suhu. Oleh karena itu pembudidaya udang perlu memikirkan bagaimana cara mengubah kebiasaan yang selama ini dilakukan dalam budidaya udang dengan cara manual ke cara otomatis.

Pada tahun 2014 Stevanus Hari Wijatmika melakukan penelitian “Alat Ukur Kadar Oksigen Terlarut Dengan Metode Elektrolisis Berbasis ATmega8535” [5]. Alat yang dibuat menggunakan mikrokontroler ATmega8535 dan menggunakan sensor gas oksigen KE-25 dan menggunakan metode elektrolisis untuk pengukuran kadar oksigen. Sistem ini berhasil melakukan pengukuran kadar oksigen terlarut, tetapi proses pengukuran membutuhkan waktu yang sangat lama dan hasil pengukuran dipengaruhi kadar oksigen udara sekitar dan pemantauan masih dilakukan melalui lcd.

Pada tahun 2020, Novia Indriani Putri melakukan penelitian yaitu “Sistem Monitoring dan Pengendalian Jarak Jauh Tingkat Keasaman dan Salinitas Kolam Ikan Koi” [6]. Alat yang dibuat menggunakan Arduino Mega sebagai mikrokontroler. Terdapat dua buah sensor yaitu sensor pH (SEN0161) dan sensor TDS. Kedua sensor tersebut nantinya terhubung dengan air pada kolam ikan dan memberikan keluaran berupa pemberitahuan terhadap kualitas air. Sensor akan mengirimkan data ke Arduino Mega. Setelah itu data diproses oleh arduino untuk menjalankan motor servo. Motor servo membuka katup penampung cairan asam, basa atau garam sesuai dengan data yang diberikan oleh sensor. Hasil nilai pembacaan dari sensor pH dan salinitas akan ditampilkan pada sebuah aplikasi pada ponsel atau *smartphone*.

Penelitian yang akan dilakukan penulis berjudul ”Sistem Pemantauan dan Pengendalian Suhu dan Kadar Oksigen Pada Model Tambak Udang Dengan Teknologi IoT”. Sistem ini menggunakan Arduino Mega Sebagai pengendali sistem dan menggunakan ESP 32 sebagai pengirim data ke platform Blynk. Penggunaan Arduino Mega sebagai pengendali sistem utama bertujuan untuk pemantauan secara langsung agar tidak terganggu apabila terjadi koneksi buruk yang menghambat proses pengiriman data atau data tidak terkirim data ke platform Blynk, sehingga pemantauan masih tetap dilakukan jika terjadi koneksi buruk. Penelitian ini menggunakan dua buah sensor DO SKU SEN 0237 dan dua buah sensor DS18B20 data dari sensor akan diolah oleh arduino mega, data dari arduino mega diteruskan ke ESP32. ESP 32 bertugas untuk mengirimkan data ke platform Blynk.

Berdasarkan penelitian sebelumnya, penelitian yang dilakukan penulis memiliki perbedaan dan pembaruan. Penelitian ini menggunakan sensor dengan tipe berbeda dari sebelumnya yaitu sensor oksigen SKU SEN0237 sensor ini dapat mengukur kadar oksigen terlarut tanpa dipengaruhi oksigen udara di sekitar, menggunakan empat buah motor DC 12V yang akan memompa air umpan kedalam bak uji jika terjadi kondisi abnormal. Oleh karena itu penelitian ini dilakukan untuk merancang sistem pengendalian suhu dan kadar oksigen pada model tambak udang yang ter-monitor secara jarak jauh melalui *smartphone*. Sistem monitoring akan dipantau secara online yang terhubung ke *smartphone* melalui platform Blynk, dan sistem kontrol dan pemantauan secara langsung menggunakan mikrokontroler Arduino Mega.

1.2. Tujuan dan Manfaat

Tujuan dari penelitian ini adalah menghasilkan sistem pengendalian suhu dan kadar oksigen yang ter-monitor secara jarak jauh melalui *smartphone*.

Manfaat dari penelitian sistem ini memberikan pemantauan suhu dan kadar oksigen secara real-time, memungkinkan petani udang untuk dengan cepat merespon perubahan kondisi lingkungan saat kondisi membahayakan udang.

1.3. Batasan Masalah

Batasan masalah pada penelitian ini, yaitu:

1. Pengiriman data menggunakan ESP 32.
2. Mikrokontroler yang digunakan sebagai pengendali adalah Arduino Mega.
3. Menggunakan dua sensor suhu DS18B20 memiliki ketelitian $0,5^{\circ}\text{C}$ pada suhu 10°C - 85°C sehingga mempermudah pembacaan oleh mikrokontroler.
4. menggunakan dua Sensor Oksigen SKU SEN 0237 yang akan memantau kondisi Oksigen terlarut dalam air.
5. Penelitian menggunakan model tambak udang dengan ukuran 60cm x 30cm x 35cm (PxLxT).
6. Pemantauan secara langsung dipantau melalui OLED dan tampilan jarak jauh menggunakan *smartphone*.
7. Thermostat digunakan untuk mengatur dan memantau suhu air bak umpan

8. OLED digunakan untuk menampilkan data keluaran dari sensor berbentuk angka dan huruf.
9. Aerator digunakan untuk meningkatkan nilai oksigen ke dalam permukaan air, sehingga akan lebih banyak oksigen yang masuk kedalam air untuk menyuplai oksigen udang vaname.
10. Pompa air digunakan untuk mengisi air pada kolam sesuai kebutuhan.

1.4. Metodologi Penelitian

Metode yang digunakan pada penelitian ini adalah:

1. Studi pustaka

Tahap awal dalam penelitian ini dimulai dari pengumpulan informasi dengan mencari sumber yang terkait dengan judul tugas akhir berupa referensi buku tentang pemeliharaan Udang, dan referensi terkait berupa jurnal yang berisi *datasheet* sensor Suhu DS18B20 dan Sensor Oksigen Sku SEN0237, Serta pengendalian dan monitoring kualitas air pada tambak udang.

2. Perancangan sistem

Perancangan perangkat lunak dan perangkat keras bertujuan untuk merancang program yang optimal untuk sistem yang akan dibuat, pembuatan flowchart untuk memprogram mikrokontroler, motor, dan *smartphone*

3. Pembuatan alat

tahap ini akan dilakukan pembuatan sistem mekanis, kontrol, dan elektronik pada alat. Pembuatan sistem mekanis dilakukan dengan merakit konstruksi prototipe kolam dua tingkat dan merancang komponen kelistrikan. Pembuatan sistem kontrol dilakukan dengan membuat program untuk yang digunakan untuk mengatur jalannya sistem elektronika. Pembuatan sistem elektronik dilakukan dengan pengkabelan setiap komponen kelistrikan.

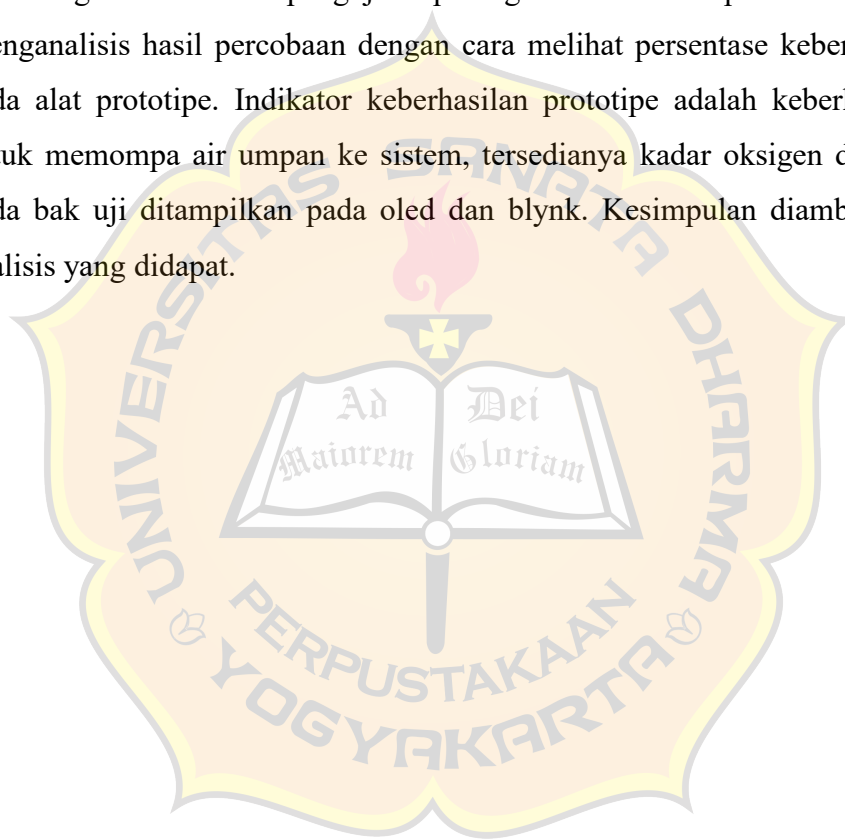
4. Pengujian sistem dan pengambilan data

Tahap pengujian alat dilakukan agar sesuai dengan yang ditetapkan, menguji tingkat keberhasilan komunikasi dengan *smartphone*. Pengujian kondisi air saat dalam kondisi panas, kondisi dingin, dan saat kondisi panas dingin secara bersamaan yang termonitor secara jarak jauh melalui *smartphone*, menguji ketelitian sensor Suhu dan sensor oksigen dalam mendeteksi kualitas air kolam. Data yang diambil ada dua, yaitu:

- a. Data Primer : Kemampuan dalam pengaturan sensor dan aktuator, yaitu data Nilai Suhu dan oksigen sebelum dan sesudah diatur, berjalannya pompa saat kondisi suhu panas, dingin, aerator memompa oksigen saat kadar DO berkurang. tampilan GUI dapat dilihat melalui *smartphone*.
- b. Data sekunder : Data hasil keluaran dari sensor suhu DS18B20 dan sensor oksigen sku sen 0237 dapat ditampilkan di-platform *blynk*.

5. Analisis dan kesimpulan

Menganalisis hasil pengujian perangkat keras berupa waktu yang dibuat Menganalisis hasil percobaan dengan cara melihat persentase keberhasilan sistem pada alat prototipe. Indikator keberhasilan prototipe adalah keberhasilan pompa untuk memompa air umpan ke sistem, tersedianya kadar oksigen dan suhu stabil pada bak uji ditampilkan pada oled dan blynk. Kesimpulan diambil berdasarkan analisis yang didapat.



BAB II DASAR TEORI

2.1. Kualitas Air

Kualitas air merupakan salah satu faktor keberhasilan pada budidaya ikan. Manajemen kualitas air adalah merupakan faktor penting dalam kehidupan organisme akuatik karena berpengaruh terhadap reproduksi, pertumbuhan, dan kelangsungan hidup udang[7].

Salah satu parameter penting kualitas air dalam budidaya udang adalah oksigen terlarut yang dikonsumsi udang untuk proses respirasi. Tingkat konsumsi udang vaname antara lain bergantung pada ukuran (stadia) udang vaname (faktor internal) dan status makan (faktor eksternal). Jika kebutuhan oksigen dalam air tidak terpenuhi maka akan menyebabkan penurunan kondisi kesehatan dan bahkan menyebabkan kematian pada udang.

Kondisi Temperatur atau suhu sangat mempengaruhi pertumbuhan budidaya udang, bila terjadi peningkatan suhu secara ekstrim, dapat mengakibatkan kematian pada udang. Temperatur yang cocok untuk pertumbuhan udang vaname berkisar pada 26-30°C, udang akan mengalami stress pada suhu diatas 35°C [8].

Suhu atau *temperatur* sangat berpengaruh terhadap oksigen terlarut karena suhu juga mempengaruhi terhadap parameter yang mempengaruhi oksigen, apabila suhu meningkat maka oksigen terlarut akan menurun ini dikarenakan metabolisme organisme yang meningkat mengakibatkan konsumsi oksigen tinggi, kadar oksigen menurun dan karbon dioksida meningkat. Pada suhu lingkungan turun mendadak akan terjadi degradasi eritrosit sehingga proses respirasi (pernafasan atau pengambilan oksigen) terganggu. Sebaliknya, pada suhu yang meningkat tinggi akan menyebabkan ikan bergerak aktif, tidak mau berhenti makan, dan metabolisme cepat meningkat sehingga kotoran menjadi lebih banyak.

Kotoran yang banyak akan menyebabkan kualitas air disekitarnya menjadi buruk. Sementara kebutuhan oksigen meningkat, tetapi ketersediaan oksigen air buruk sehingga ikan akan kekurangan oksigen dalam darah. Akibatnya ikan menjadi stres dan terganggu keseimbangannya. Stres akibat peningkatan suhu air pada ikan berdampak terhadap kinerja dan kesehatan ikan berupa gangguan fungsi sel-sel darah, salah satunya yaitu eritrosit.

Tabel 2.1. Parameter Kualitas Air

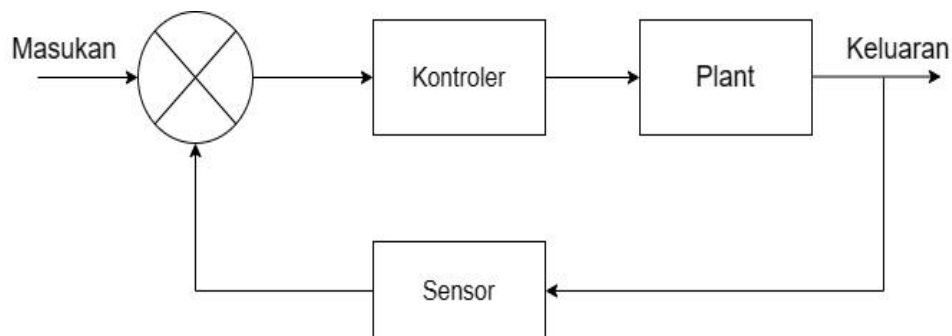
Parameter kualitas air	Satuan	Alat
Temperatur	°C	DSI8B20
Oksigen Terlarut	mg/L	SKU SEN-0237

2.2. Dasar Sistem Kendali

Sistem kendali (*Control System*) adalah proses pengendalian suatu komponen untuk mengatur atau merubah suatu sistem di dalamnya, berdasarkan masukan dan keluaran yang dihasilkan [9]. Sistem kendali memiliki tiga unsur yaitu masukan, proses dan keluaran. Berdasarkan tipe kendalinya, sistem kendali terdiri atas dua sistem kendali yaitu sistem kendali *loop* terbuka dan sistem kendali *loop* tertutup. Penelitian ini berlandaskan sistem kendali *loop* tertutup.

2.2.1. Sistem Kendali Loop Tertutup

Sistem kontrol *loop* tertutup adalah sistem kontrol yang sinyal keluarannya mempunyai pengaruh langsung pada aksi pengontrolan. Sensor sebagai komponen masukan yang mengukur keluaran atau kondisi sistem dan mengubahnya menjadi sinyal listrik atau sinyal lain yang dapat diproses oleh sistem kontrol. Mikrokontroler komponen yang menerima sinyal dari sensor dan membandingkannya dengan nilai target. Berdasarkan perbedaan antara nilai yang diukur dan yang diinginkan, kontroler menghasilkan sinyal kendali yang mengatur plant ke dalam sistem untuk memperbaiki atau mempertahankan keluaran sesuai dengan target. Sistem kontrol *loop* tertutup juga merupakan sistem kontrol berumpan balik. Dengan kata lain, istilah “*loop* tertutup” berarti menggunakan aksi umpan balik untuk memperkecil kesalahan sistem. Gambar 2.2 merupakan diagram sistem kendali *loop* tertutup.



Gambar 2. 1 Diagram Sistem Kendali Loop Tertutup

2.3. Mikrokontroler

Mikrokontroler adalah sebuah komponen dalam bentuk *chip*, mikrokontroler memiliki kemampuan untuk menerima, memproses, dan merespons sinyal sesuai dengan program yang telah diprogram di dalamnya. Fungsinya melibatkan penerimaan sinyal masukan dari sensor yang terhubung dengan lingkungan sekitar, serta pengiriman sinyal keluaran ke aktuator yang dapat mengubah kondisi lingkungan. Ketika merujuk pada perangkat atau yang dapat berinteraksi dengan lingkungannya, kita sering mendengar istilah "otak" di belakangnya dikaitkan dengan pemanfaatan algoritma khusus yang dikenal sebagai mikrokontroler.[10].

Mikrokontroler sebagai elemen dari unit kendali sistem, dianggap sebagai "komputer mini" karena lokasinya yang berada di atas IC (*chip*). Selain itu, mikrokontroler mengandung pengontrol yang dapat diprogram, memungkinkan penggunaan program khusus sesuai dengan kebutuhan sistem. Dengan adanya mikrokontroler memungkinkan sistem untuk beroperasi secara inovatif membuka jalan bagi solusi otomatisasi yang efisien.

2.3.1. Arduino Mega 2560

Arduino mega 2560 adalah salah satu papan pengembangan mikrokontroler yang memiliki kemampuan yang sangat baik untuk menangani aplikasi yang kompleks. Dalam penelitian ini, Arduino Mega2560 dipilih sebagai otak dari sistem pemantauan dan pengendalian untuk mengatur suhu dan kadar oksigen dalam model tambak udang[11].

Arduino mega2560 memiliki arsitektur yang kuat, terdapat memori, terdapat 54 pin digital I/O dan 16 pin analog input, arduino mega2560 memberikan fleksibilitas yang cukup untuk mengembangkan berbagai sensor, seperti sensor suhu, sensor oksigen, dan aktuator dengan mudah.[12] Dengan total 54 pin digital I/O yang disediakan, Arduino Mega 2560 memiliki fleksibilitas tinggi dalam menghubungkan dan mengendalikan berbagai perangkat elektronik. Ini memungkinkan peneliti untuk mengintegrasikan sensor, seperti sensor suhu dan sensor oksigen, yang menjadi kunci dalam pengelolaan lingkungan tambak udang. Tabel 2.2, Merupakan spesifikasi arduino mega 2560.



Gambar 2. 2 Arduino Mega 2560 [12]

Tabel 2.2 Spesifikasi Arduino Mega 2560

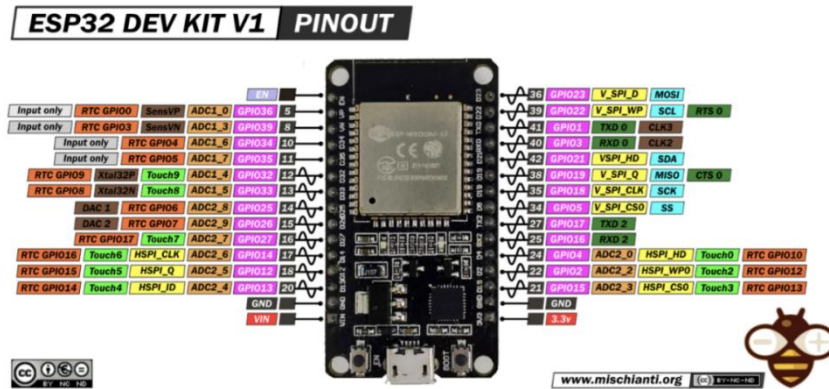
Keterangan	Spesifikasi
Tegangan Operasi	5V
Tegangan masukan(disarankan)	7V-12V
Tegangan masukan (Batas)	6V-20V
Pin I/O Digital	54 Pin 9 (15 diantaranya menyediakan PWM)
Pin Masukan Analog	16 Pin
Arus DC per Pin I/O	20mA
Arus DC untuk Pin 3.3V	50mA
Flash Memory	256 KB dimana 8 KB digunakan oleh bootloader
SRAM	8 KB
EEPROM	4 KB
Dimensi	101,52mm(p) X 53,3mm (l)
Clock Speed	16 Mhz
Berat	37 gram

2.3.2. NodeMCU ESP 32

NodeMCU ESP32 adalah papan mikrokontroler berbasis ESP32, sebuah *System on Chip (SoC)* yang dilengkapi dengan modul *WiFi* dan *bluetooth* serta jumlah *General Purpose Input-Output (GPIO)* yang banyak [10-11]. ESP32 merupakan penerus dari ESP8266 dirancang dengan daya ultra rendah TSMC 40 nm teknologi. Modul ini dapat diprogram dengan bahasa pemrograman C atau C++, artinya juga dapat dijalankan dengan *compiler* Arduino menggunakan *software* Arduino IDE [12-13]. ESP32 sangat cocok digunakan dalam proyek IoT sehingga dapat diimplementasikan untuk aplikasi *controlling* maupun *monitoring* jarak jauh. Tabel spesifikasi Node MCU ESP32 dapat dilihat pada Tabel 2.3

Tabel 2.3. Spesifikasi NodeMCU ESP 32

Keterangan	Spesifikasi
Digital I/O Pin	30 Pin
12 bit ADC Pin	18 Channel
Operating Voltage	2,3V-3,6V
Operating current	3,0V-3,6V
WiFi Frequency Range	2,4GHz-2,5GHz
Clock Speed	40 MHz
USB Port	Micro USB



Gambar 2.3. ESP 32 [6]

2.3.3. Arduino IDE

Arduino IDE (*Integrated Development Environment*) merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan [19]. Melalui perangkat lunak ini pemrograman arduino dapat dilakukan untuk mengimplementasikan fungsi-fungsi tertentu dalam perangkat tersebut.



Gambar 2. 4 Arduino IDE

Pada Gambar 2.4 ditunjukkan tampilan jendela perangkat lunak arduino IDE [12]. Berikut beberapa fungsi yang dapat diakses melalui toolbar pada perangkat lunak arduino IDE:

1. *Verify*, untuk melakukan verifikasi atau kompilasi program.
2. *Upload*, untuk mengunggah program ke papan arduino.
3. *New Sketch*, membuat lembar kerja baru.

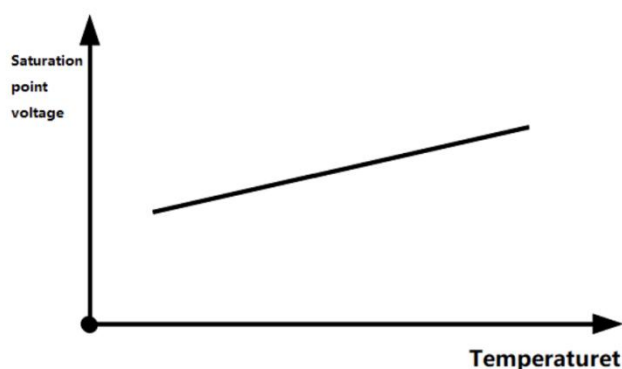
4. *Open Sketch*, membuka program yg telah disimpan sebelumnya.
5. *Save Sketch*, menyimpan lembar kerja yang telah dikerjakan.
6. Serial Monitor, membuka *interface* untuk komunikasi serial.
7. Keterangan Aplikasi, menampilkan pesan-pesan yang dilakukan aplikasi.
8. *Text Editor*, tempat untuk menuliskan program.
9. *Console Log*, menampilkan informasi tentang program.
10. *Line Number Sketch*, menampilkan nomor pada setiap baris kode.
11. Informasi *Board* dan *Port*, menginformasikan port yang dipakai oleh board Arduino.

2.4. Sensor

Sensor adalah suatu alat yang digunakan untuk menangkap sekaligus mendeteksi berbagai perubahan besaran fisik seperti besaran listrik. Sensor yang digunakan untuk penelitian ini adalah sensor oksigen dan sensor suhu DS18B20.

2.4.1. Sensor Dissolved Oxygen (DO)

Sensor DO atau Sensor oksigen terlarut adalah alat yang digunakan untuk mengukur jumlah oksigen yang terlarut langsung di dalam air. Sensor ini berperan penting untuk memantau kualitas air, terutama dalam konteks akuakultur. Sensor oksigen SKU SEN0237 bekerja berdasarkan prinsip elektrokimia. Sensor ini memiliki dua elektroda yang terendam dalam larutan elektrolit. Ketika oksigen masuk ke sensor, terjadi reaksi elektrokimia yang menghasilkan arus listrik. Arus ini kemudian dikonversi menjadi tegangan yang sebanding dengan konsentrasi oksigen di lingkungan. Sensor ini membutuhkan kalibrasi eksternal untuk menyesuaikan keluaran tegangan analog, tegangan harus dikalibrasi agar tegangan linier dengan konsentrasi oksigen [13].



Gambar 2. 5 Hubungan temperatur dengan tegangan kalibrasi

Tabel 2. 1 Nilai *setpoint* kadar oksigen terlarut berdasarkan suhu.

T °C	DO mg/L	T °C	DO mg/L	T °C	DO mg/L
0	14.6	16	9.86	32	7.3
1	14.22	17	9.64	33	7.17
2	13.8	18	9.47	34	7.06
3	13.44	19	9.27	35	6.94
4	12.76	20	9.09	36	6.84
5	12.44	21	8.91	37	6.72
6	12.11	22	8.74	38	6.6
7	11.83	23	8.57	39	6.52
8	11.56	24	8.41	40	6.4
9	11.229	25	8.25	41	6.33
10	11.29	26	8.11	42	6.23
11	11.04	27	7.96	43	6.13
12	10.76	28	7.83	44	6.06
13	10.54	29	7.68	45	5.97
14	10.31	30	7.56	46	5.88
15	10.06	31	7.43	47	5.79



Gambar 2.6. Sensor Dissolved Oxygen (DO)

Tabel 2.4. Spesifikasi Sensor DO SKU SEN O237

Keterangan	Spesifikasi
Rentang Pengukuran	0-20mg/l
Suhu kerja	0-40°C
Tegangan masukan	3,3V-5,5 V
Jenis Sensor	Oksigen Elektrokimia
Keluaran	Analog
Tipe	Probe Galvanic
Panjang Kabel	2 Meter
Dimensi	42mm x 32mm

2.4.2. Sensor Suhu DS18B20

Sensor suhu DS18B20 adalah sensor digital yang memiliki 12-bit ADC internal dan memiliki tingkat akurasi yang cukup tinggi yaitu 0,5 derajat celcius. Sensor ini dapat membaca perubahan temperatur lingkungan dan mengkonversikan temperatur tersebut menjadi sebuah sinyal digital. Sensor ini bekerja menggunakan protokol komunikasi 1-wire (*one-wire*) dan memiliki tiga kaki yaitu Vcc, Ground, dan Data. Sensor ini dapat digunakan untuk mengukur suhu pada rentang -55 sampai 125 derajat celcius [14].



Gambar 2.7. Sensor Suhu DS18B20 [14]

Tabel 2.5. Spesifikasi Sensor Suhu DS18B20 [14]

Spesifikasi	Keterangan
Tegangan Operasi	3,3V-5,5 V
Resolusi	9-bit hingga 12-bit
Protokol komunikasi	1-Wire
Rentang Suhu	-55°C -125°C
Akurasi	±0.5°C pada rentang -10°C hingga +85°C
Dimensi	Diameter 4,8mm, Panjang 12,5mm

2.5. Aktuator

Aktuator adalah perangkat mekanis yang digunakan untuk mengubah energi listrik menjadi gerakan fisik. Aktuator berperan dalam menjalankan perintah atau kontrol dari mikrokontroler untuk menggerakkan motor. Aktuator yang digunakan pada penelitian ini adalah aerator dan motor Pompa air 12V [16].

2.5.1. Pompa Air

Pompa adalah sebuah perangkat mekanis yang berfungsi untuk mengangkat cairan dari lokasi rendah ke lokasi yang lebih tinggi, atau untuk mengalirkan cairan dari area

bertekanan rendah ke area bertekanan tinggi. Pompa juga difungsikan sebagai perangkat untuk meningkatkan kecepatan aliran dalam suatu sistem jaringan perpipaan.[17] Pompa air di penelitian ini menggunakan pompa air 12V yang digunakan untuk memompa cairan pendingin dan cairan pemanas saat kondisi bak uji saat terjadi keadaan abnormal. Bentuk fisik pompa air pompa air 12V dilihat pada Gambar 2.7. berikut ini.



Gambar 2.8. Pompa Air DC 12V[17].

2.5.2. Aerator

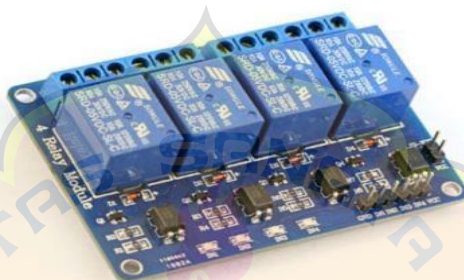
Aerator adalah alat yang digunakan dalam kolam atau akuarium untuk menghasilkan gelembung udara, yang berfungsi dengan menggerakkan air di dalam agar melepaskan oksigen yang ada di udara ke dalam air aerator ini sangat penting untuk menjaga kesehatan ikan di dalam akuarium, karena menyediakan oksigen yang dibutuhkan oleh ikan.[18]. Aerator yang digunakan pada penelitian ini dapat dilihat pada Gambar 2.8. berikut.



Gambar 2.9. Aerator[18].

2.5.3. Relay

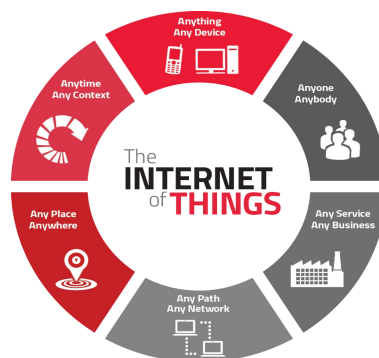
Modul *Relay* adalah modul yang dapat menjalankan fungsi logika mikrokontroler arduino, mengendalikan tegangan tinggi dengan menggunakan tegangan rendah meminimalkan terjadinya penurunan tegangan, memungkinkan penggunaan fungsi penundaan waktu atau *time delay function*, melindungi komponen lainnya dari kelebihan tegangan penyebab korsleting, menyederhanakan rangkaian agar lebih ringkas [18]. Jenis *Relay* berdasarkan jumlah *channel* yaitu modul *relay 1 channel, 2 channel, 4 channel, 8 channel, 16 channel, 32 channel*.



Gambar 2.10. *Relay empat Channel.*

2.6. Internet Of Things

Internet of Things (IoT) adalah sebuah konsep teknologi yang menghubungkan perangkat elektronik dan sensor ke internet, sehingga memungkinkan perangkat tersebut untuk saling berkomunikasi dan bertukar data secara otomatis. IoT sering digunakan untuk mengembangkan sistem monitoring dan pengendalian yang efisien dalam berbagai aplikasi, seperti *smart trash can, smart home*, dan sistem kantor pintar. IoT memungkinkan pengguna untuk mengontrol dan memantau perangkat dari jarak jauh melalui koneksi internet, sehingga memudahkan pengguna dalam mengelola dan memantau perangkat secara efektif.[20]

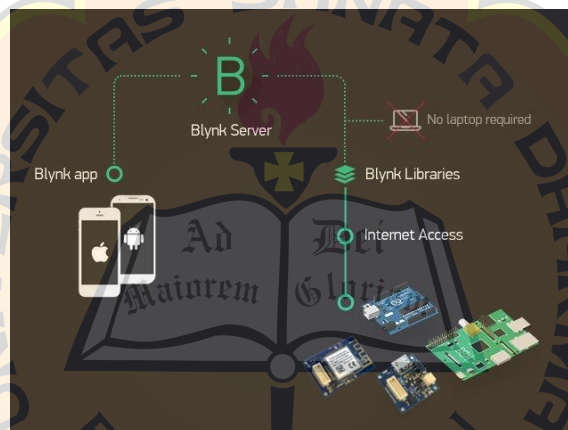


Gambar 2.11. IoT [20].

2.6.1. Blynk Apps

Blynk apps adalah sebuah platform *Internet of Things* (IoT) yang menyediakan beragam fitur untuk menghubungkan perangkat keras IoT dengan aplikasi *mobile*, serta memungkinkan pengguna untuk mengontrol dan memantau perangkat keras dari jarak jauh. Platform ini dapat digunakan untuk mengendalikan berbagai jenis perangkat keras seperti Arduino, NodeMCU, Raspberry Pi, ESP8266, dan sejenisnya melalui internet.

Blynk juga menyediakan berbagai fitur diantaranya pembuatan antarmuka grafis untuk proyek, penyimpanan data dari sensor, visualisasi data, serta fitur-fitur lainnya yang mendukung pengembangan proyek IoT. Platform *Blynk* tersedia secara gratis (*open source*) maupun berbayar, dan dapat diakses melalui website serta aplikasi *mobile* untuk Android dan iOS.[21].



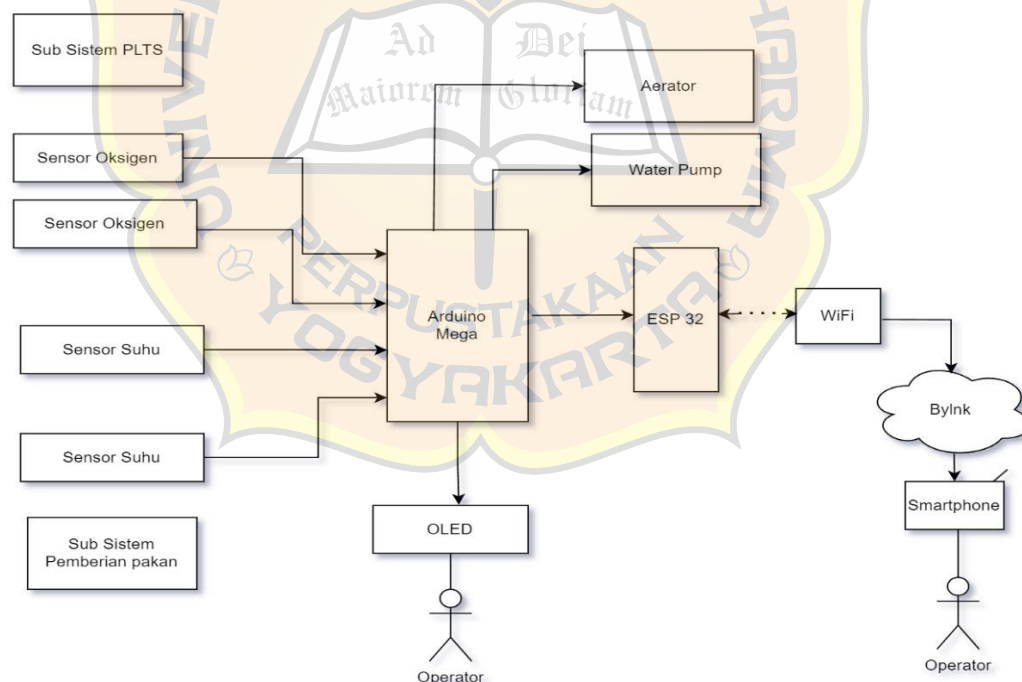
Gambar 2. 12. Blynk Server [21]

BAB III RANCANGAN PENELITIAN

Pada bab ini membahas perancangan “Sistem Pemantauan dan Pengendalian Suhu dan Kadar Oksigen pada Model Tambak Udang dengan Teknologi IoT”. Perancangan perangkat keras mekanis, perancangan perangkat keras elektronis, perancangan perangkat lunak, serta perancangan desain GUI pada aplikasi Blynk. Hasil akhir pada perancangan ini akan menampilkan suhu dan kadar oksigen termonitor jarak jauh melalui *smartphone*.

3.1. Diagram Blok Sistem

Perancangan alat terdiri dari perancangan perangkat keras dan perangkat lunak. Arduino Mega berperan sebagai otak sistem yang berfungsi sebagai mikrokontroler yang berkomunikasi dengan modul NodeMCU menggunakan komunikasi serial untuk saling bertukar data. NodeMCU diaplikasikan sebagai modul tambahan untuk Arduino Mega agar dapat terhubung dengan internet untuk mengirim data ke *Cloud/Blynk*.



Gambar 3.1. Diagram Blok Sistem Secara Umum.

Gambar 3.1 merupakan diagram blok sistem secara umum. Arduino mega menerima data masukan dari sensor, setelah itu arduino mega memproses data dan mengirimkan perintah ke pompa air dan aerator jika terjadi kondisi abnormal. Data yang diolah Arduino

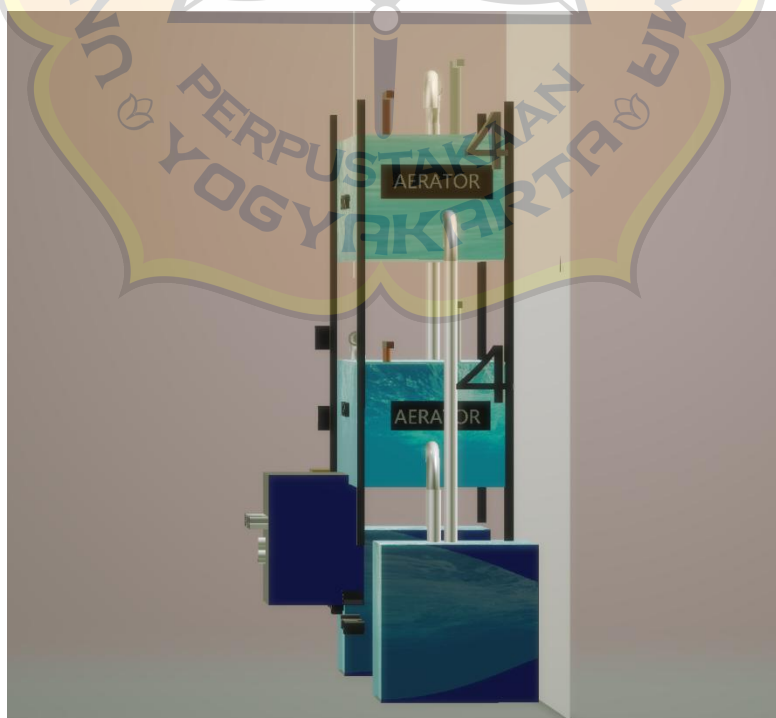
mega diteruskan ke esp 32 dan esp 32 mengirimkan data ke *blynk* untuk pemantauan jarak jauh melalui *smartphone*.

3.2. Perancangan Perangkat Keras Mekanis

Rancangan pengujian yang akan dibuat pada tugas akhir ini dapat dilihat pada Gambar 3.2. dan pada Gambar 3.3. Rancangan prototipe kolam dua tingkat dengan ukuran Panjang 60cm, Lebar 30cm, Tinggi 35 cm.



Gambar 3.2. Perancangan Kolam Dua Tingkat Tampak Depan.



Gambar 3.3. Perancangan Kolam Dua Tingkat Sisi Samping.

Penjelasan komponen-komponen perancangan pada Gambar 3.2 dan 3.3 sebagai berikut :

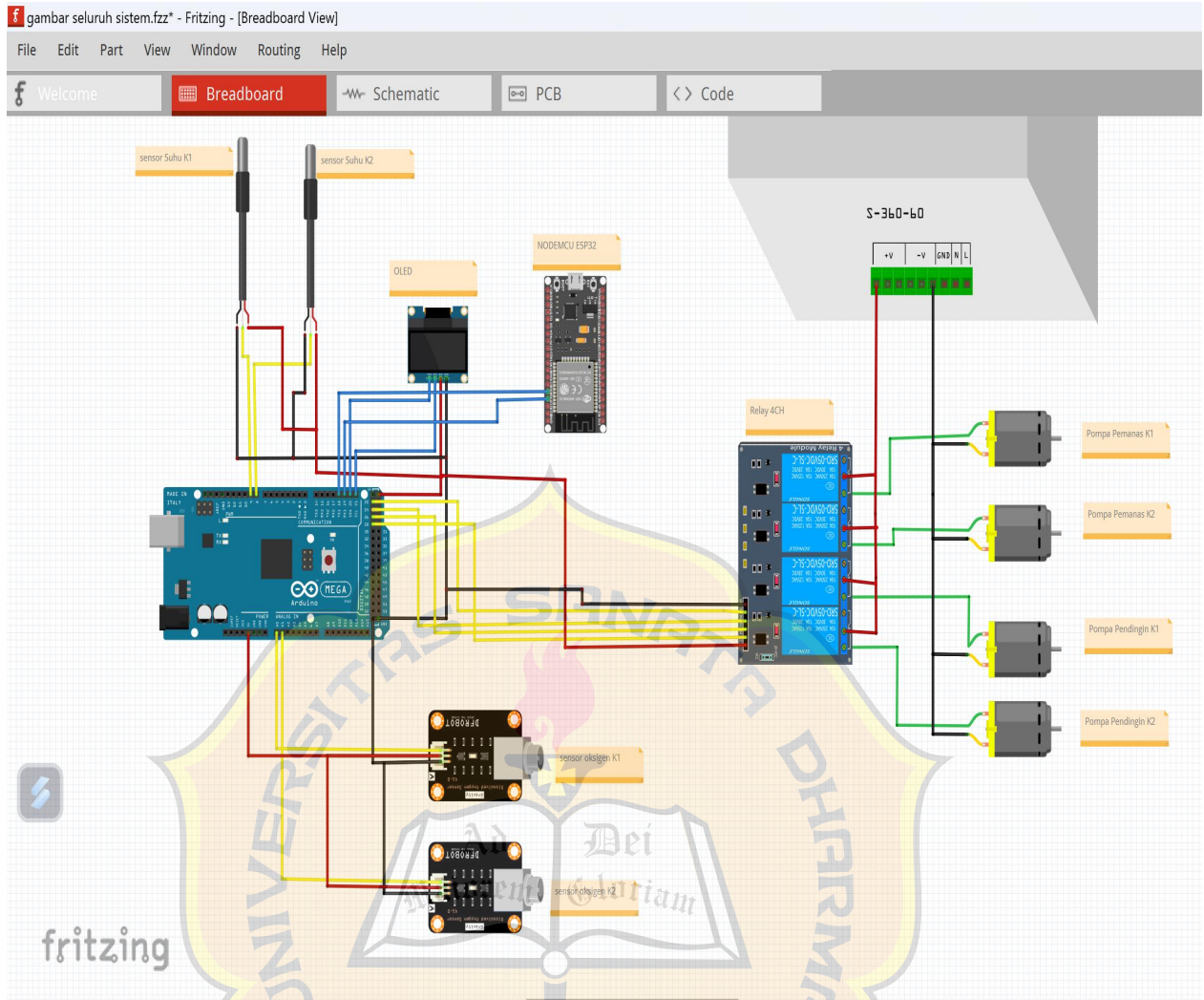
1. Sensor oksigen terlarut digunakan untuk mengukur kadar oksigen pada air prototipe kolam 2 tingkat.
2. Sensor suhu digunakan untuk memantau kondisi suhu air prototipe kolam 2 tingkat.
3. Box Mikrokontroler, di dalam box terdapat Arduino Mega, ESP 32, *Power Supply* 12V dan *Relay*.
4. Aerator digunakan untuk memompa oksigen kedalam air.
5. Pompa air digunakan untuk memompa air ke dalam kolam.
6. *Tank Supply* air digunakan untuk cadangan air apabila suhu pada kolam perancangan terindikasi abnormal, maka pompa akan mulai mensirkulasikan air kolam sampai kondisi suhu stabil.
7. *Pipa Supply* air digunakan untuk menyalurkan air dari *tank supply* ke kolam perancangan.

3.3. Perancangan Perangkat Elektronis

Perancangan perangkat elektronis merupakan langkah penting dalam pengembangan sistem pengawasan dan pengendalian suhu dan kadar oksigen yang termonitor melalui *Internet of Things* (IoT). Dalam konteks ini, sistem ini mengadopsi mikrokontroler Arduino Mega sebagai inti utama, didukung oleh berbagai perangkat elektronika yang mendukung fungsi dan mekanisme operasional sistem. Secara keseluruhan, rangkaian perangkat keras elektronis untuk sistem ini dapat ditemukan pada Gambar 3.4

Mikrokontroler Arduino Mega berperan sebagai otak pusat sistem, mengendalikan operasi dan interaksi antara komponen-komponen elektronis. Berdasarkan Keputusan yang diambil, mikrokontroler akan mengaktifkan atau menonaktifkan actuator untuk menjaga *set point* yang sudah ditentukan, data hasil pemrosesan pada Arduino mega dilanjutkan ke *blynk*.

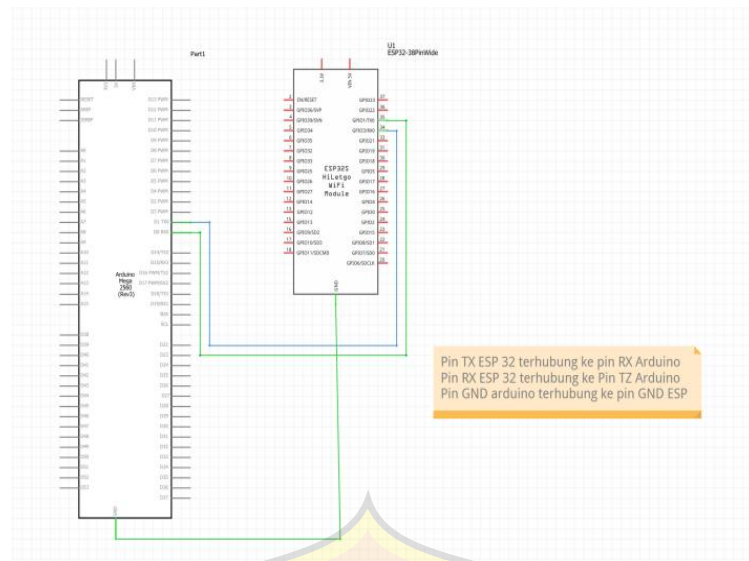
Aktuator untuk pengendalian kondisi air menggunakan pompa air 12V yang diatur oleh *relay* untuk memompa suplai air panas dan air dingin yang sudah disiapkan pada bak air umpan. Aerator digunakan untuk menambah oksigen terlarut dalam air ketika kadar oksigen dalam air berkurang dari *set point* yang sudah ditentukan.



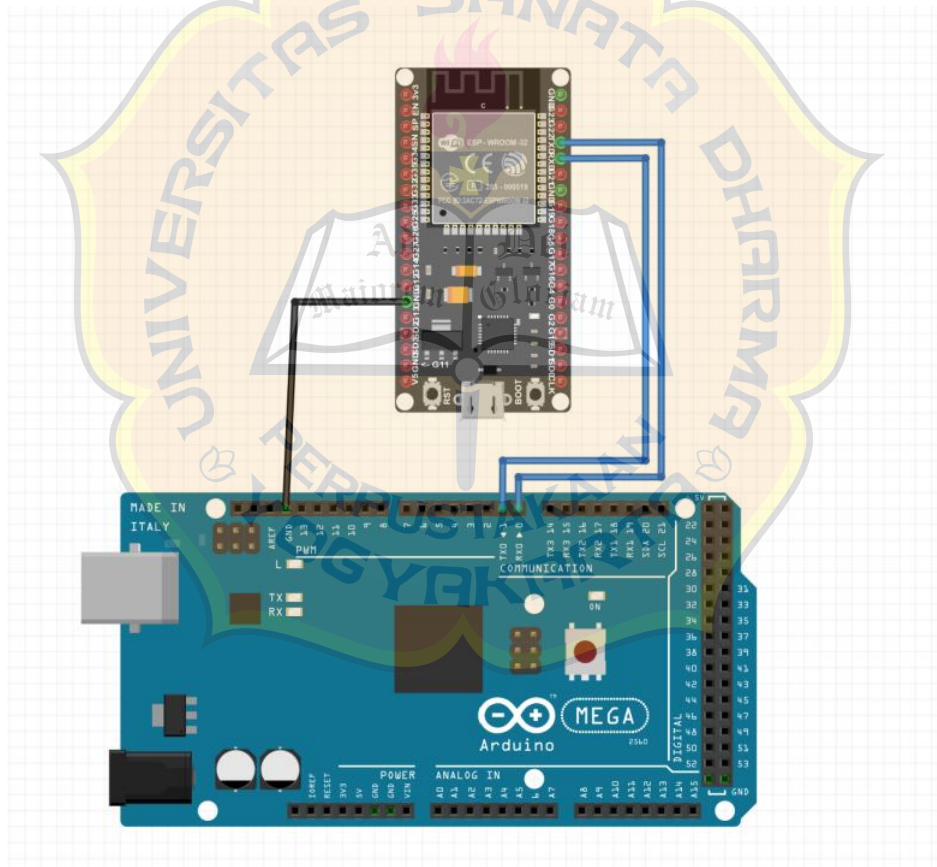
Gambar 3.4. Rangkaian Elektronik Keseluruhan Sistem.

3.3.1. Rangkaian ESP 32 dengan Arduino Mega

Dalam rangkaian ESP32 dengan Arduino Mega, ESP32 berperan sebagai modul *WiFi* yang mengirim dan menerima data untuk diproses oleh mikrokontroler Arduino Mega 2560. Pemasangan pin melibatkan koneksi TX dan RX untuk komunikasi serial antara keduanya. Keduanya juga terhubung melalui pin digital untuk keperluan kontrol tambahan, sementara pin 3V3 ESP32 dihubungkan ke pin 3.3V Arduino Mega untuk tegangan operasional. Dengan konfigurasi ini, ESP32 dapat berfungsi sebagai perangkat *WiFi* eksternal yang dikendalikan oleh Arduino Mega untuk aplikasi IoT.



Gambar 3.5. Schematic Rangkaian ESP 32 dengan Arduino Mega.



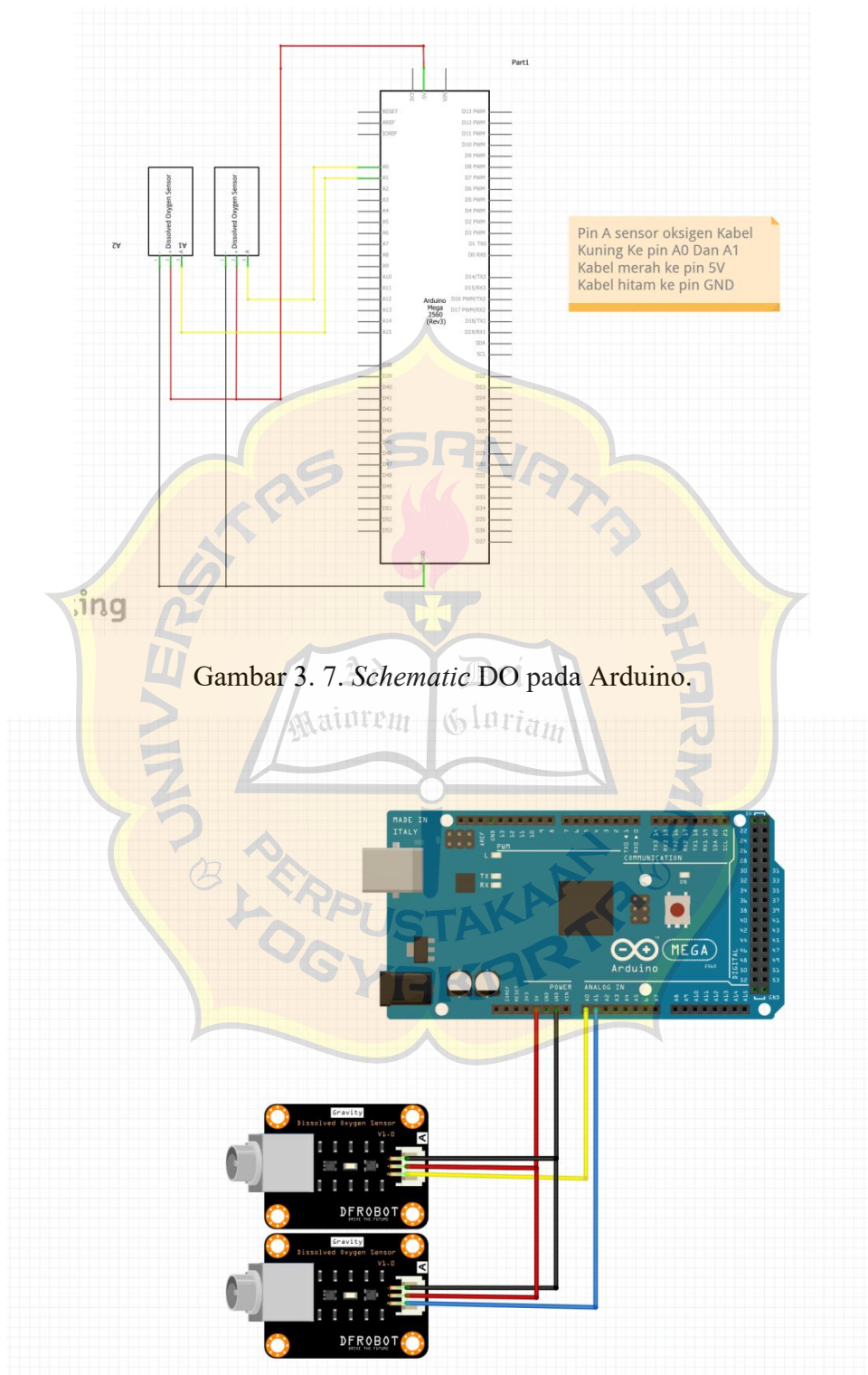
Gambar 3. 6. Pemasangan Pin ESP pada Arduino Mega.

Tabel 3.1. Alamat Pin ESP dengan Arduino Mega

ESP 32	Arduino Mega
TX	RX
RX	TX
GND	GND

3.3.2. Sensor DO (SKU SEN0237)

Pemasangan Pin Sensor DO dapat dilihat pada Gambar 3.7.



Gambar 3. 7. Schematic DO pada Arduino.

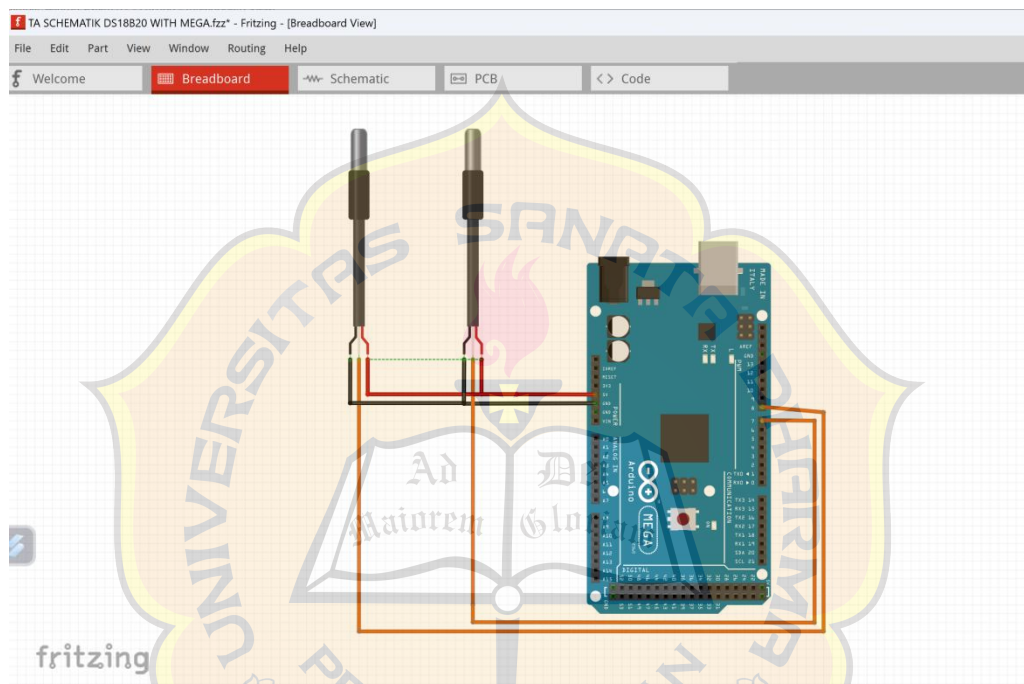
Gambar 3.8. Pemasangan Pin Sensor pada Arduino Mega.

Tabel 3. 2. Pemasangan Pin Sensor DO pada Arduino Mega.

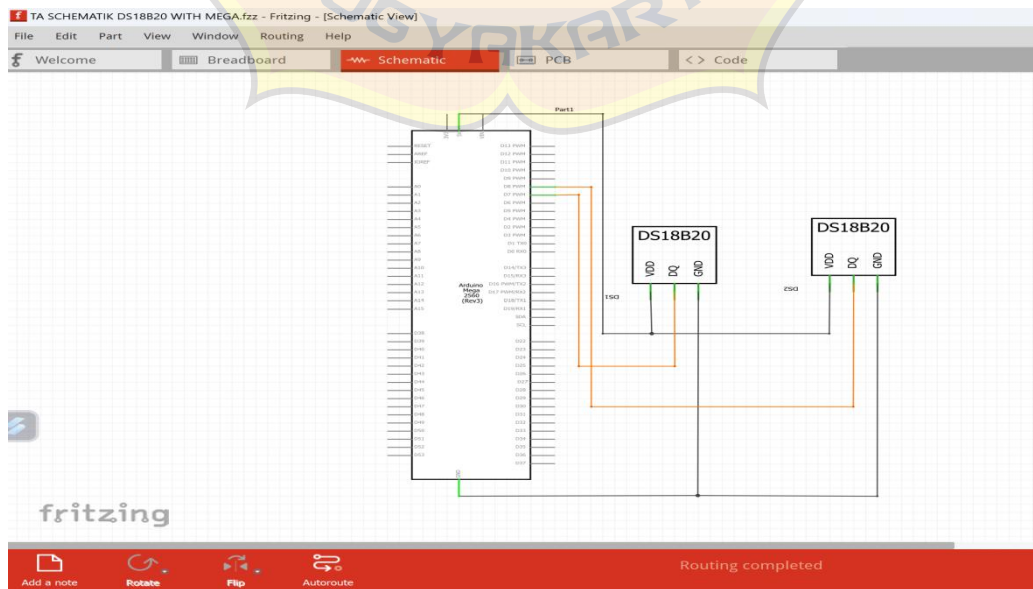
Sensor DO	Arduino Mega
Pin Positif (+)	5V
Pin Negatif (-)	GND
Pin A	A0, A1

3.3.3. Sensor Suhu (DS18B20)

Pemasangan Pin pada Sensor Suhu dapat dilihat pada Gambar 3.9. berikut ini:



Gambar 3. 9. Pemasangan Pin Sensor Suhu pada Arduino Mega.



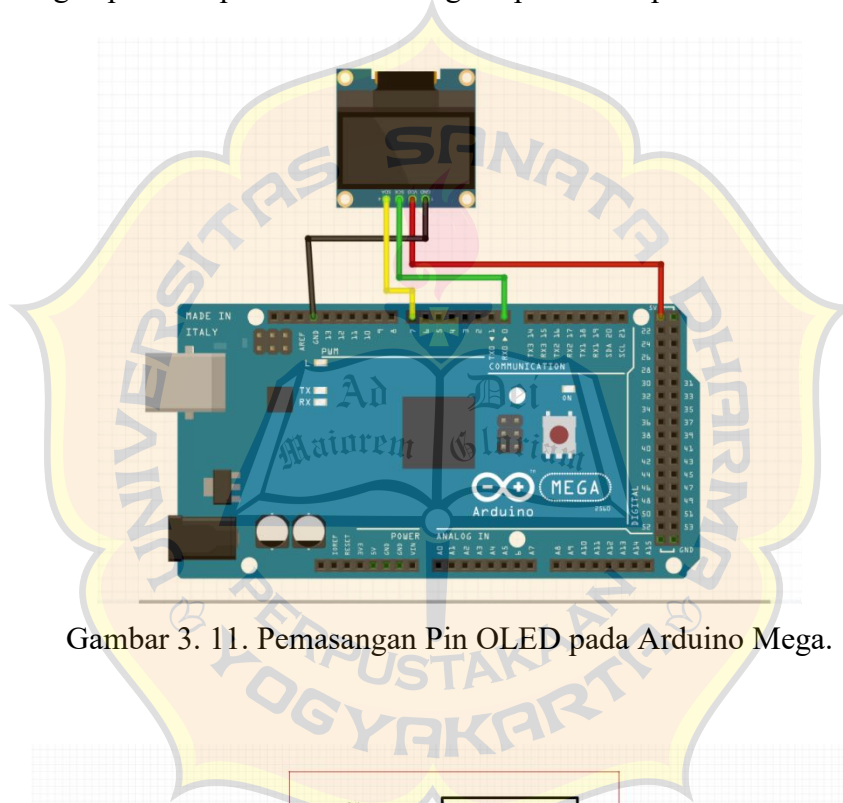
Gambar 3.10. Schematic Rangkaian Sensor Suhu dan Arduino Mega.

Tabel 3.3. Pemasangan Pin Sensor Suhu pada Arduino Mega.

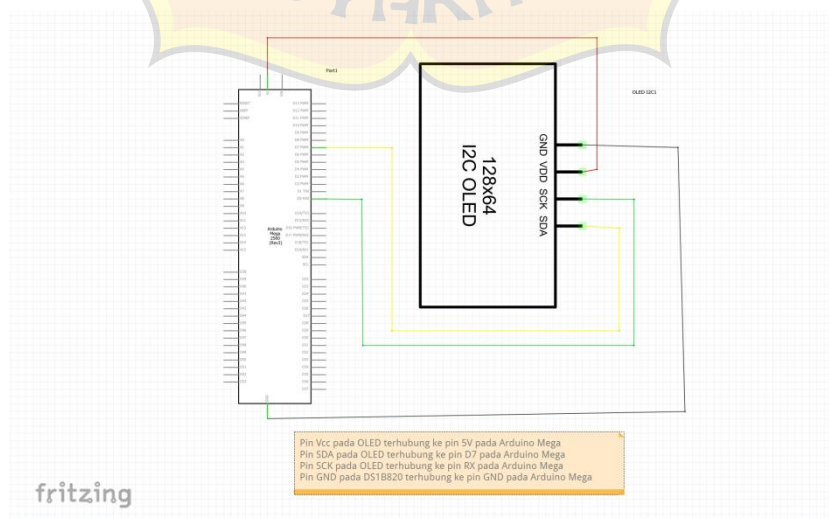
Sensor Suhu (DS18B20)	Arduino Mega
VCC	5V
DQ	D7, D8
GND	GND

3.3.4. Perancangan OLED

Oled digunakan untuk menampilkan data keluaran dari arduino agar bisa dilakukan pemantauan langsung di lokasi. Oled bertugas menampilkan data sensor DO dan sensor suhu. Pemasangan pin oled pada Arduino Mega dapat dilihat pada Gambar 3.10.



Gambar 3. 11. Pemasangan Pin OLED pada Arduino Mega.



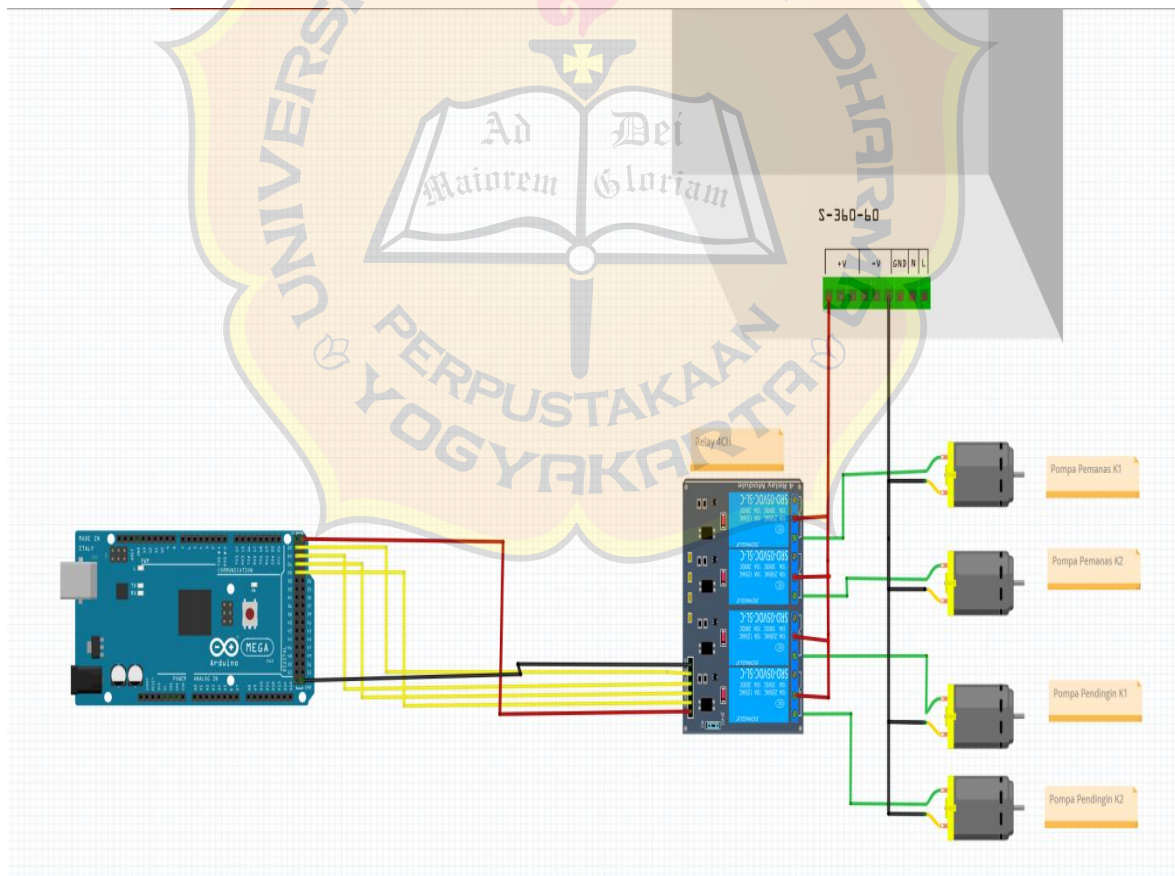
Gambar 3. 12. Schematic OLED dan Arduino Mega.

Tabel 3. 4. Pemasangan Pin OLED pada Arduino Mega.

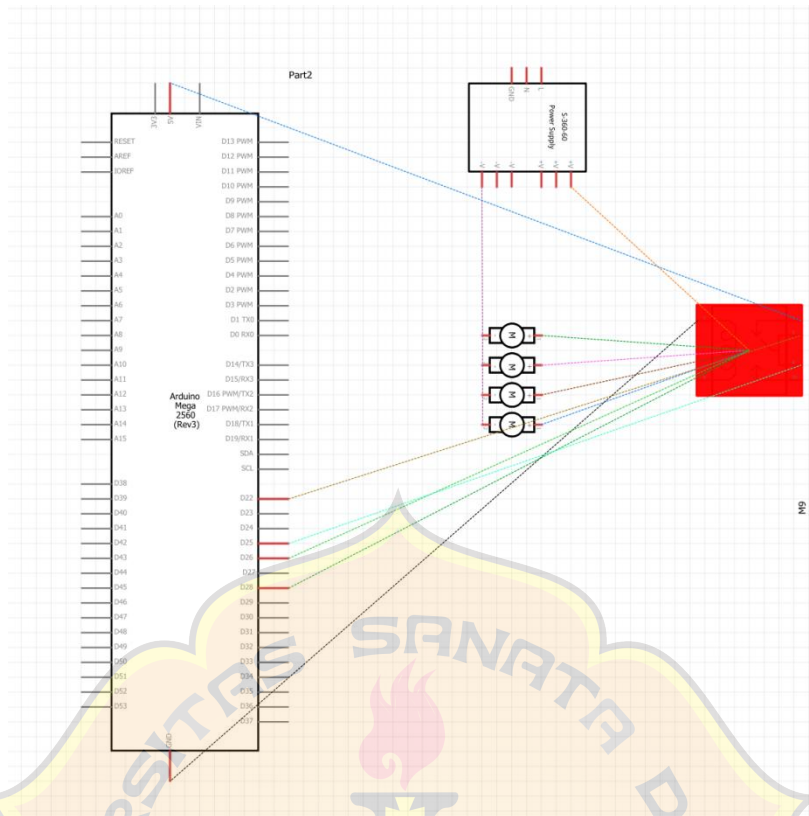
OLED	Arduino Mega
VCC	5V
SCK	SCL
SDA	SDA
GND	GND

3.4. Perancangan Pompa Air

Pompa air digunakan untuk memompa air panas dan air dingin kedalam bak uji untuk mensirkulasikan air sampai pada kondisi normal. Pompa air nantinya akan dikontrol oleh *Relay*. Gambar 3.13 merupakan rangkaian Arduino Mega, *Relay*, Pompa DC dan *Power Supply* sedangkan Gambar 3.14 merupakan diagram perkabelan Arduino Mega, *Relay*, Pompa DC dan *Power Supply*. Berikut konfigurasi pemasangan *Relay* ke Arduino Mega dapat dilihat pada Tabel 3.5, *Relay* ke *Power Supply* dapat dilihat pada Tabel 3.6 dan *Relay* ke Pompa DC dapat dilihat pada Tabel 3.7



Gambar 3.13. Pemasangan Pin pada *Relay* ke Arduino Mega; *Relay* ke Motor; *Relay* ke *Power Supply*.



Gambar 3. 14 diagram perkabelan Arduino Mega, Relay, Pompa DC dan Power Supply

Tabel 3.5. Pemasangan Pin Relay ke Arduino Mega

Relay	Arduino Mega
IN1	D12
IN2	D13
IN3	D14
IN4	D15

Tabel 3.6. Pemasangan Pin Relay ke Power Supply

Relay	Power Supply
COM1	V+
COM2	V+
COM3	V+
COM4	V+

Tabel 3.7. Pemasangan Pin Relay ke Pompa DC

Relay	Pompa DC
NO1	PIN (+)
NO2	PIN (+)
NO3	PIN (+)
NO4	PIN (+)

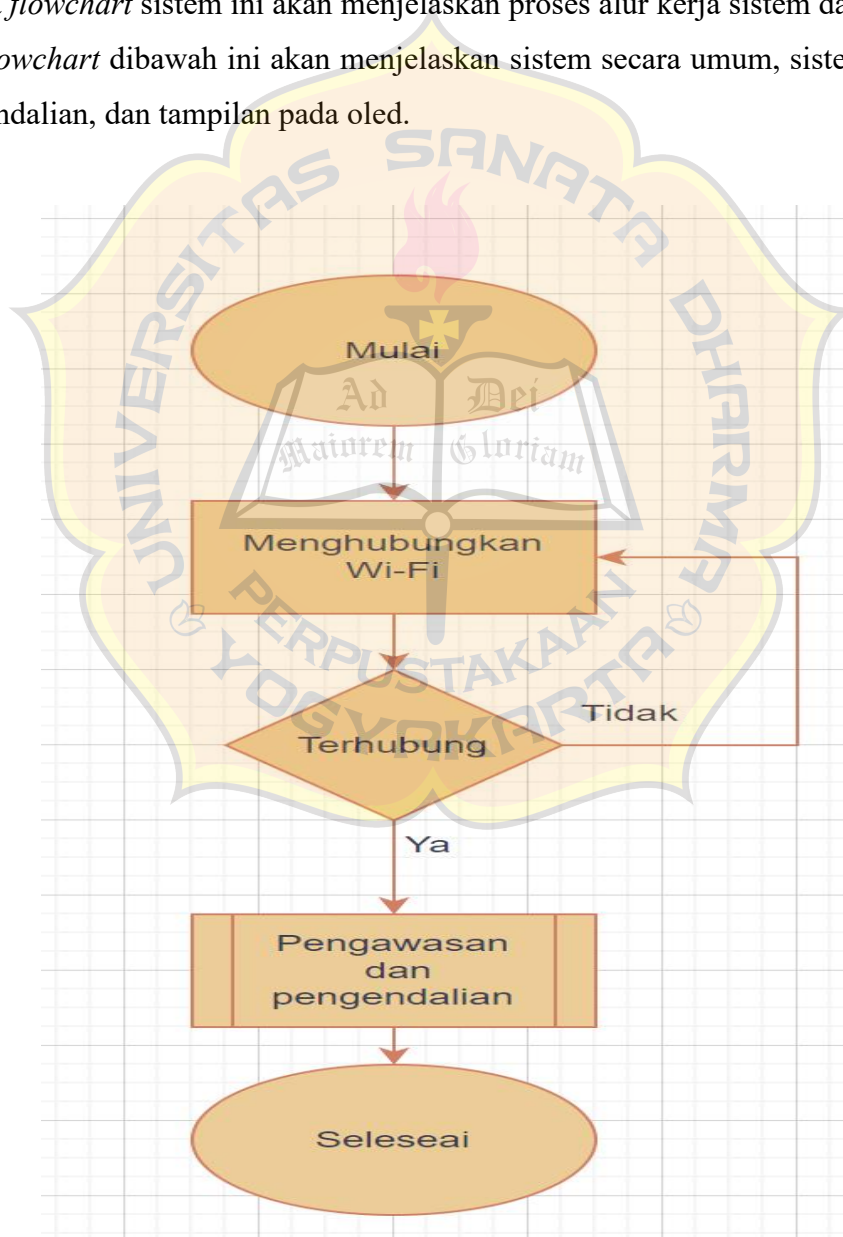
3.5. Perancangan Perangkat Lunak

Perancangan perangkat lunak akan membahas tentang alur proses kerja sistem monitoring dan pengendalian air bekerja, proses ini akan dijelaskan menggunakan beberapa Flowchart, yaitu:

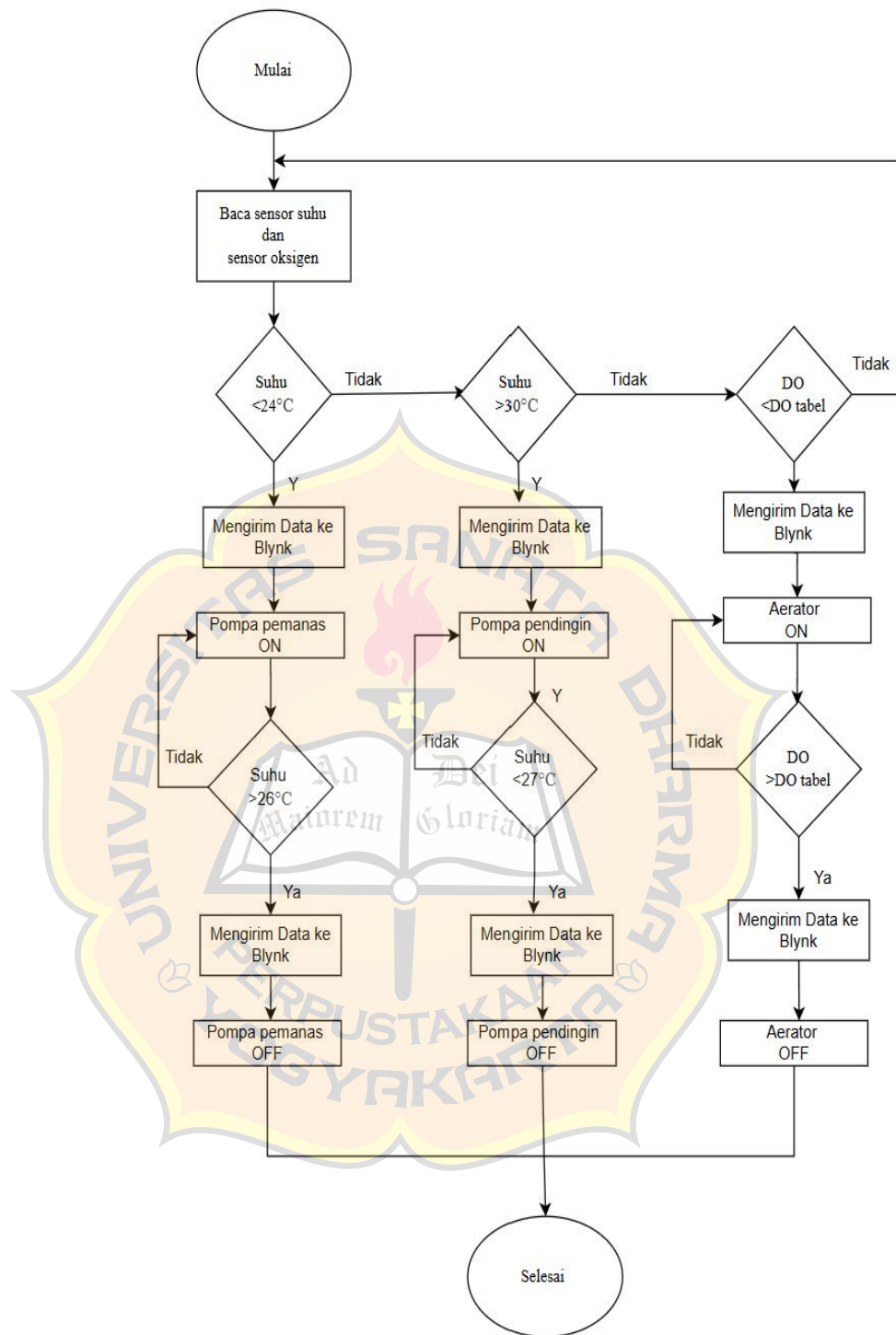
1. *Flowchart* sistem secara umum.
2. *Flowchart* sistem pemantauan dan pengendalian.
3. *Flowchart Monitoring di Oled dan Blynk*.

3.5.1. *Flowchart* Sistem

Pada *flowchart* sistem ini akan menjelaskan proses alur kerja sistem dari awal sampai selesai. *Flowchart* dibawah ini akan menjelaskan sistem secara umum, sistem pemantauan dan pengendalian, dan tampilan pada oled.



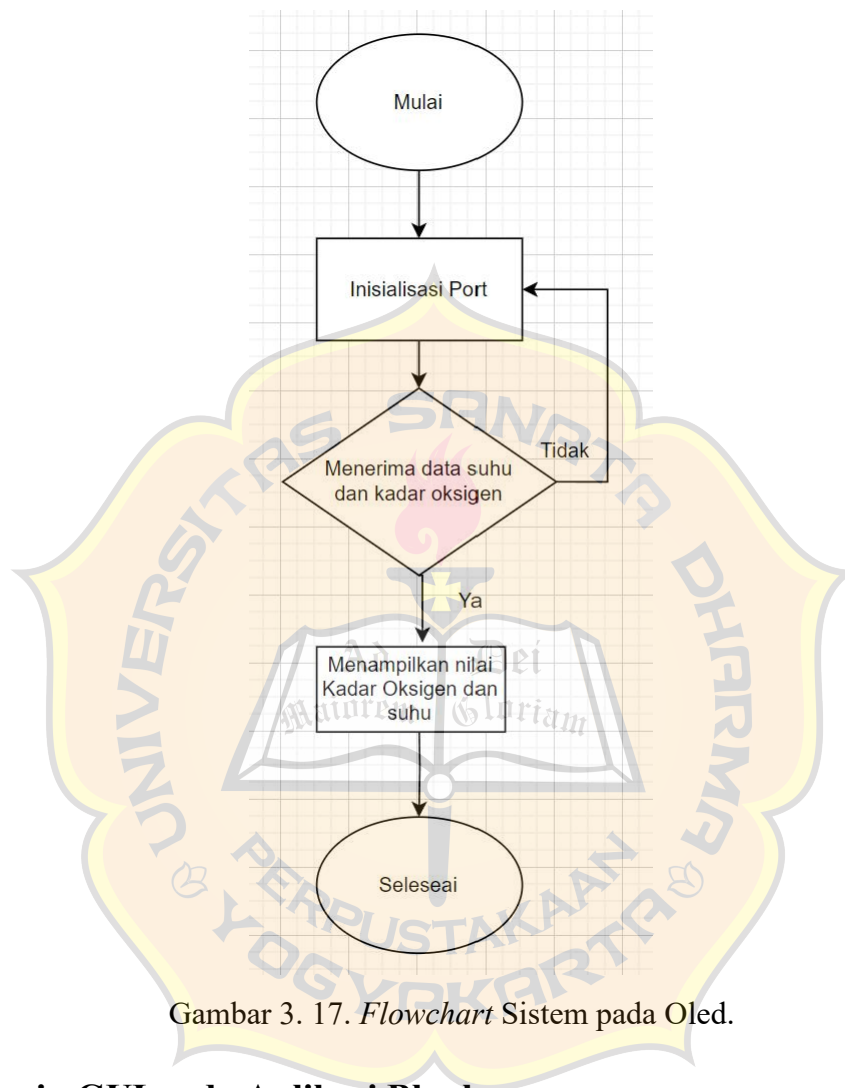
Gambar 3.15. *Flowchart* Sistem secara umum.



Gambar 3.16 Flowchart Pengendalian pada Arduino Mega

3.5.2. Flowchart Sistem pada Oled

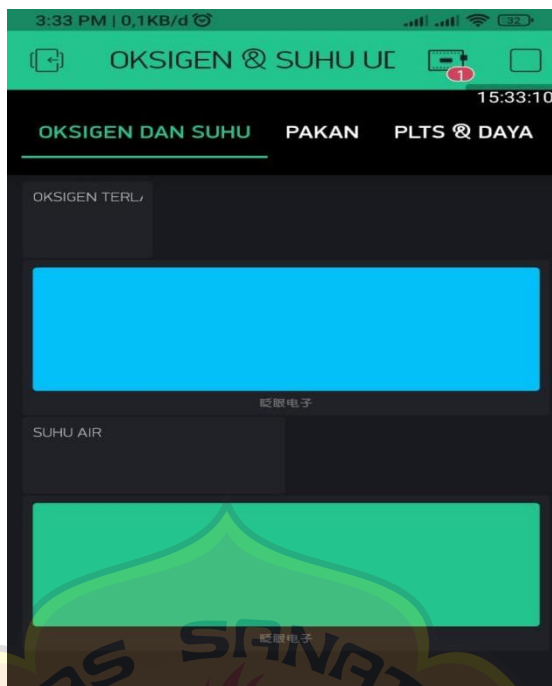
Pada diagram alir ini oled menerima data dari sensor yang mendeteksi lingkungan. Oled akan menampilkan keterangan sesuai dengan data yang diterima dari sensor. *Flowchart* oled dapat dilihat pada Gambar 3.16.



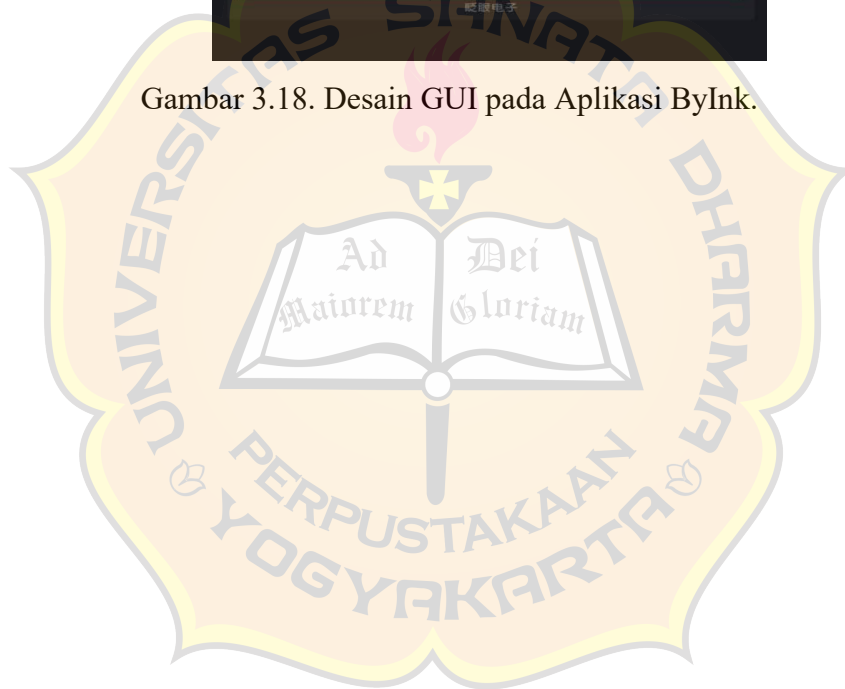
Gambar 3. 17. *Flowchart* Sistem pada Oled.

3.5.3. Desain GUI pada Aplikasi Blynk

Desain GUI pada aplikasi Blynk terdapat tiga tab terpisah yang dimana masing-masing menjalankan fungsinya, desain GUI terdiri dari dua tampilan, pada bagian pertama terdapat indikator yang nantinya akan menampilkan data oksigen terlarut yang dibaca oleh sensor DO, pada tampilan yang kedua nantinya akan menampilkan data suhu yang dibaca oleh sensor DS18B20. Gambar desain GUI dapat dilihat pada Gambar 3.17.



Gambar 3.18. Desain GUI pada Aplikasi ByInk.

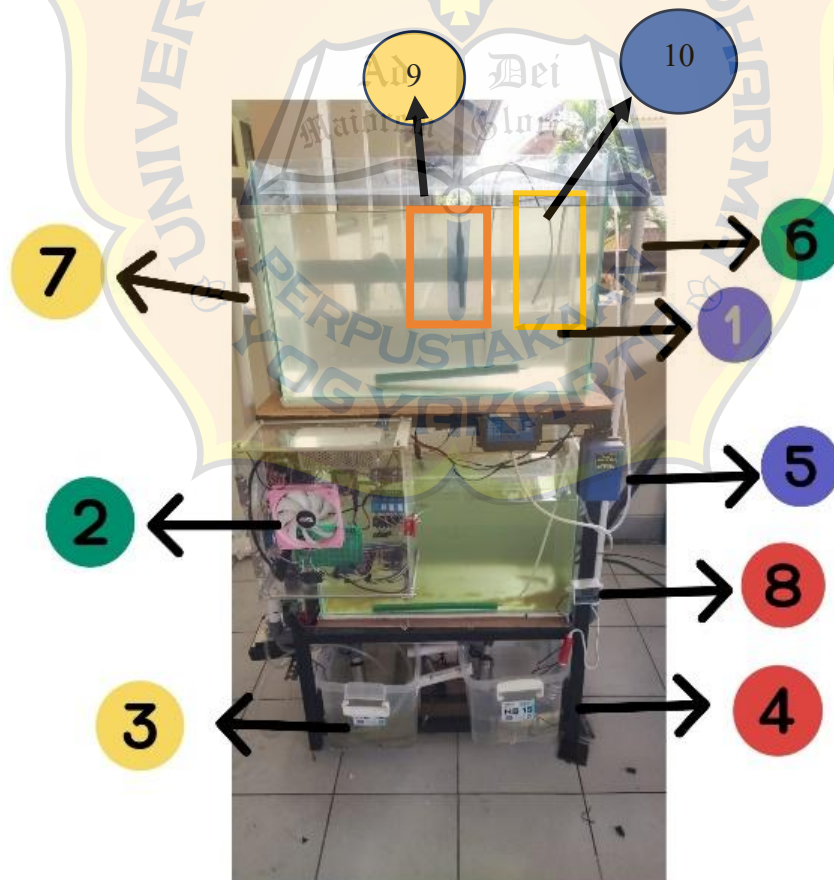


BAB IV HASIL DAN PEMBAHASAN

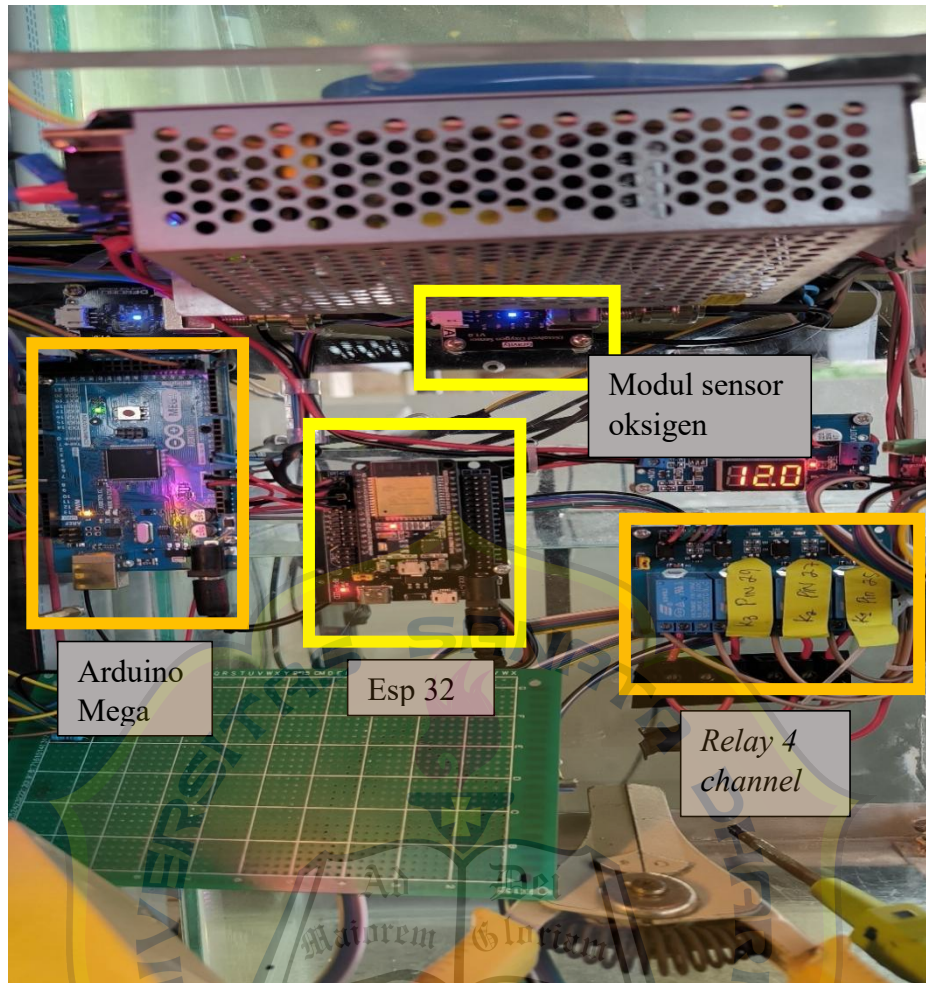
Bab ini membahas implementasi hasil perancangan “Sistem Pemantauan dan Pengendalian Suhu dan Kadar Oksigen pada Model Tambak Udang Vaname dengan Teknologi IoT”. Pengujian sistem dilakukan melalui percobaan pada kondisi suhu yang berbeda, baik dalam kondisi suhu panas sampai kondisi suhu stabil dan suhu dingin sampai pada suhu stabil. Selain itu dilakukan juga pengujian kadar oksigen saat nilai DO (*Dissolved Oxygen*) kurang dari nilai DO-tabel hingga mencapai nilai yang sesuai dengan yang tercantum dalam dasar teori.

4.1 Implementasi Perangkat Keras

Implementasi perangkat keras sistem pemantauan dan pengendalian suhu dan kadar oksigen akan dijelaskan bentuk fisik prototipe alat. Bentuk fisik prototipe dapat dilihat pada Gambar 4.1 dan Gambar 4.2. Keterangan pada alat dapat dilihat pada Tabel 4.1.



Gambar 4. 1 Keseluruhan Sistem.



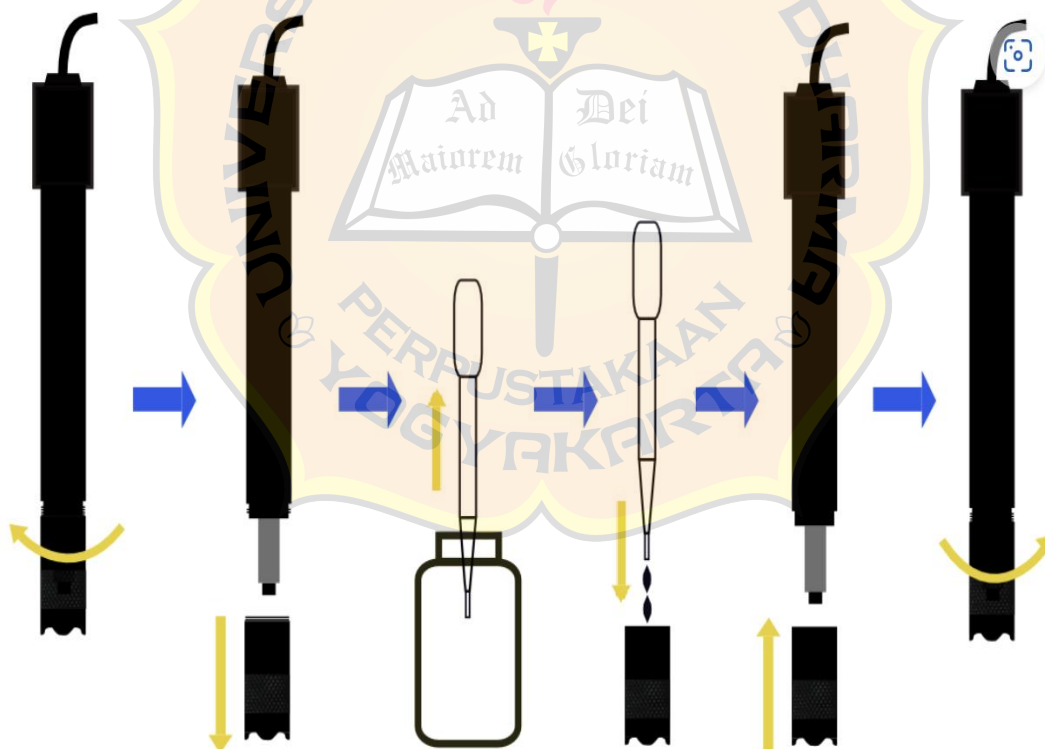
Gambar 4.1 Kotak Panel

Tabel 4. 1 Keterangan Bagian Perangkat

No	Keterangan
1	Model kolam tambak udang.
2	Kotak Panel.
3	Bak penampung 1 terdapat 2 pompa air untuk memompa cairan pendingin ke kolam 1 dan kolam 2.
4	Bak penampung 2 terdapat 2 pompa air untuk memompa cairan pemanas ke kolam 1 dan kolam 2.
5	Aerator digunakan untuk memompa oksigen.
6	Pipa pembuangan air.
7	Pipa masukan air.
8	Thermostat.
9	Sensor oksigen.
10	Sensor suhu.

4.1.1 Kalibrasi Sensor

Proses kalibrasi sensor suhu DS18B20 tidak dilakukan kalibrasi eksternal, karena sensor ini telah dikalibrasi secara presisi dengan ketelitian pembacaan $\pm 0,5^{\circ}\text{C}$ dalam rentang suhu -10°C hingga $+85^{\circ}\text{C}$, nilai perbandingan dapat dilihat pada Tabel 4.2. Sensor oksigen terlarut SKU SEN-0237 dilakukan kalibrasi eksternal, karena sensor ini memiliki sensitifitas pengukuran yang berbeda tiap sensor. Kalibrasi sensor dilakukan dengan cara menyiapkan dua wadah air yang berisi suhu dengan suhu berbeda, wadah pertama berisi cairan panas yang nantinya akan menjadi titik kalibrasi pada suhu panas, hasil nilai kalibrasi pada suhu panas dapat dilihat pada Gambar 4.4, wadah kedua berisi cairan dingin yang nantinya akan menjadi titik kalibrasi suhu dingin. Hasil Kalibrasi Sensor Dapat Dilihat Pada Gambar 4.5. Langkah awal sebelum melakukan kalibrasi adalah dengan memasukkan cairan *electrolyte* kedalam probe sensor. Proses pengisian cairan *electrolyte* dapat dilihat pada Gambar 4.3 dan nilai kalibrasi sensor dapat dilihat pada Gambar 4.4



Gambar 4.3 Proses Pengisian Cairan *Electrolyte*

Proses pengisian cairan dilakukan dengan cara membuka tutup membran dari probe, dilanjutkan dengan pengisian cairan *electrolyte* kedalam tutup membran dan pastikan

membran terisi penuh, tutup kembali tutup membran dan cek kembali permukaan probe untuk memastikan tidak ada udara terperangkap dalam probe.

```
kalbrasitanggal12.ino
1 #include <Arduino.h>
2 #define DO_PIN A5
3 #define VREF 4900
4 #define ADC_RES 1024
5 #define TWO_POINT_CALIBRATION 0
6 #define READ_TEMP (34)
7 #define CAL1_V (1464) //tegangan kalibrasi
8 #define CAL1_T (34) //suhu kalibrasi
9
10
11
12
13 #define CAL2_V (392) //mv

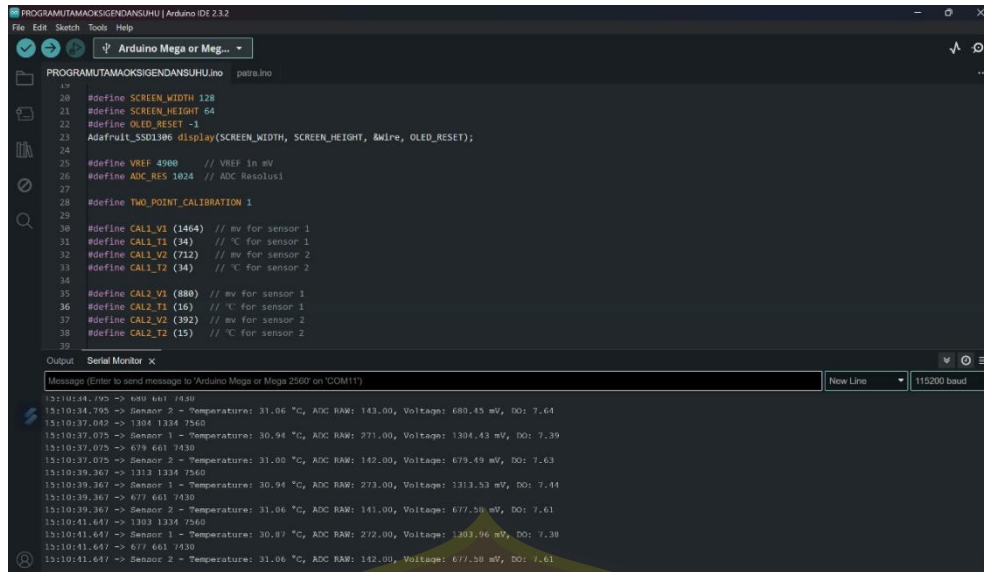
Output Serial Monitor x
Message (Enter to send message to 'Arduino Mega or Mega 2560' on 'COM11')
13:51:13.797 -> Temperatur: 34 ADC RAW: 307 ADC Voltage: 1469 DO: 7094
13:51:14.808 -> Temperatur: 34 ADC RAW: 306 ADC Voltage: 1464 DO: 7070
13:51:15.772 -> Temperatur: 34 ADC RAW: 306 ADC Voltage: 1464 DO: 7070
13:51:16.787 -> Temperatur: 34 ADC RAW: 307 ADC Voltage: 1469 DO: 7094
13:51:17.781 -> Temperatur: 34 ADC RAW: 306 ADC Voltage: 1464 DO: 7070
13:51:18.795 -> Temperatur: 34 ADC RAW: 306 ADC Voltage: 1464 DO: 7070
13:51:19.794 -> Temperatur: 34 ADC RAW: 306 ADC Voltage: 1464 DO: 7070
13:51:20.780 -> Temperatur: 34 ADC RAW: 306 ADC Voltage: 1464 DO: 7070
```

Gambar 4.4 Kalibrasi Pada Suhu Panas

```
File Edit Sketch Tools Help
Arduino Mega or Meg...
kalbrasitanggal12.ino
1 #include <Arduino.h>
2 #define DO_PIN A5
3 #define VREF 4900
4 #define ADC_RES 1024
5 #define TWO_POINT_CALIBRATION 0
6 #define READ_TEMP (16)
7 #define CAL1_V (880) //tegangan kalibrasi
8 #define CAL1_T (16) //suhu kalibrasi
9
10

Output Serial Monitor x
Message (Enter to send message to 'Arduino Mega or Mega 2560' on 'COM11')
11:44:16.592 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:16.592 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:16.592 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:17.487 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:18.480 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:19.449 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:20.452 -> Temperatur: 16 ADC RAW: 186 ADC Voltage: 890 DO: 9972
11:44:21.478 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:22.480 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:23.475 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:24.480 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:25.475 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:26.487 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:27.482 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:28.479 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:29.487 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:30.488 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:31.459 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:32.468 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:33.475 -> Temperatur: 16 ADC RAW: 185 ADC Voltage: 885 DO: 9916
11:44:34.505 -> Temperatur: 16 ADC RAW: 184 ADC Voltage: 880 DO: 9860
11:44:35.498 -> Temperatur: 16 ADC RAW: 185 ADC Voltage: 885 DO: 9916
```

Gambar 4.5 Kalibrasi Pada Suhu Dingin



Gambar 4. 6 Nilai Hasil Kalibrasi

Nilai ADC Raw adalah nilai digital mentah yang dihasilkan oleh ADC saat mengkonversi sinyal analog (tegangan) menjadi sinyal digital. ADC mengambil tegangan input analog dan mengubahnya menjadi nilai pada rentang ADC 10-bit. Nilai Voltage (mV) adalah tegangan dalam milivolt yang diukur oleh sensor, Tegangan ini adalah representasi dari sinyal analog yang diukur oleh sensor sebelum di konversi menjadi nilai digital oleh ADC. Nilai tegangan ini dapat dihitung dari nilai ADC raw menggunakan tegangan referensi dan resolusi ADC. Nilai

$$\text{Voltage (mV)} = \left(\text{Raw} \times \frac{\text{Tegangan Referensi}}{\text{ADC Resolusi}} \right) \tag{4.1}$$

$$\text{Nilai Voltage (mV) pada suhu } 34^{\circ}\text{C} = \left(306 \times \frac{4900}{1024} \right) = 1464,25 \text{ mV}$$

$$\text{Nilai Voltage (mV) pada suhu } 16^{\circ}\text{C} = \left(184 \times \frac{4900}{1024} \right) = 880,46 \text{ mV}$$

Pembacaan nilai tegangan pada pada Gambar 4.4 dan Gambar 4.5 sudah sesuai dengan perhitungan dengan rumus 4.1. Sehingga menghasilkan nilai DO dengan dengan rumus 4.2. Pengujian kesesuaian nilai kadar oksigen terbaca oleh sensor dengan data acuan nilai kadar oksigen dapat dihitung dengan rumus

$$\text{DO} = \left(\text{Voltage mV} \times \frac{\text{DO referensi}}{\text{Voltage mV}} \right) \tag{4.2}$$

$$\text{DO} = \left(1313,53 \times \frac{7,43}{1313,53} \right) = 7,43 \text{ mg/L}$$

Pengujian kesesuaian pembacaan nilai kadar oksigen oleh sensor dengan data acuan nilai kadar oksigen dapat dihitung dengan rumus 4.2.

4.2 Hasil Pengujian Sistem

Hasil pengujian sistem dilakukan untuk mengetahui tingkat keberhasilan sistem dalam mengolah sensor, mengukur, dan mengendalikan nilai sensor pada “Sistem Pemantauan dan Pengendalian Kadar Oksigen dan Suhu pada Model Tambak Udang Vaname dengan Teknologi IoT”

4.2.1 Pengukuran Nilai pada Sensor Oksigen dan Sensor Suhu

Sub bab ini membahas tentang pengukuran nilai kadar oksigen menggunakan modul sensor oksigen dan pengukuran nilai suhu menggunakan sensor suhu. Perbandingan nilai pembacaan sensor suhu dilakukan dengan membandingkan nilai dari sensor dengan alat ukur pembanding, yaitu thermometer. Hasil perbandingan pengukuran sensor suhu dengan alat pembanding thermometer dapat dilihat pada Tabel 4.2. perbandingan nilai pembacaan sensor oksigen dilakukan dengan membandingkan nilai dari sensor dengan dengan nilai *setpoint* kadar oksigen. Hasil perbandingan pengukuran sensor oksigen dengan *setpoint* berdasarkan suhu dapat dilihat pada Tabel 4.3.

Tabel 4.2 Perbandingan Pengukuran Sensor Suhu dengan Alat Ukur Pembanding.

No.	Sensor Suhu DS18B20 (°C)	Thermometer (°C)	Error (%)
1	23.62	24	1.58
2	24.7	25	1.20
3	25.68	26	1.23
4	26.61	27	1.44
5	27.56	28	1.57
Rata-rata <i>error</i>			1.41

Tabel 4.3 Perbandingan Pengukuran Sensor Oksigen dengan *Setpoint* Berdasarkan Suhu

No	Suhu °C	Pengukuran Sensor	Nilai <i>Setpoint</i> Oksigen Terlarut Berdasarkan Suhu	Error (%)
1	27	7.84	7.96	1.5
2	28	7.71	7.83	1.53
3	29	7.57	7.68	1.43
4	30	7.41	7.56	1.98
5	31	7.27	7.43	2.15
Rata-rata nilai <i>error</i>				1.71

Pengujian data perbandingan antara nilai sensor dan alat ukur pembanding yang dilakukan sebanyak lima kali, persentase *error* dapat dihitung dengan rumus berikut.

$$error = \left| \frac{Exact\ Value - Approximate\ Value}{Exact\ Value} \right| \times 100\% \tag{4.3}$$

4.2.2 Pengujian Suhu Panas ke Suhu Normal

Pengujian ini dilakukan untuk mengamati *respons* sistem ketika suhu berubah dari kondisi panas ke kondisi normal Pada kolom uji pertama. *Setpoint* suhu ditetapkan pada 27°C. Data pengujian kondisi suhu panas sampai kondisi normal dapat dilihat pada Tabel 4.4. Hasil respon sistem terhadap perubahan suhu dari kondisi panas ke kondisi normal ditunjukkan pada Gambar 4.6.

$$Setpoint = 27; X = 26,81$$

$$Ess = \left| \frac{Setpoint - X}{Setpoint} \right| \times 100\%$$

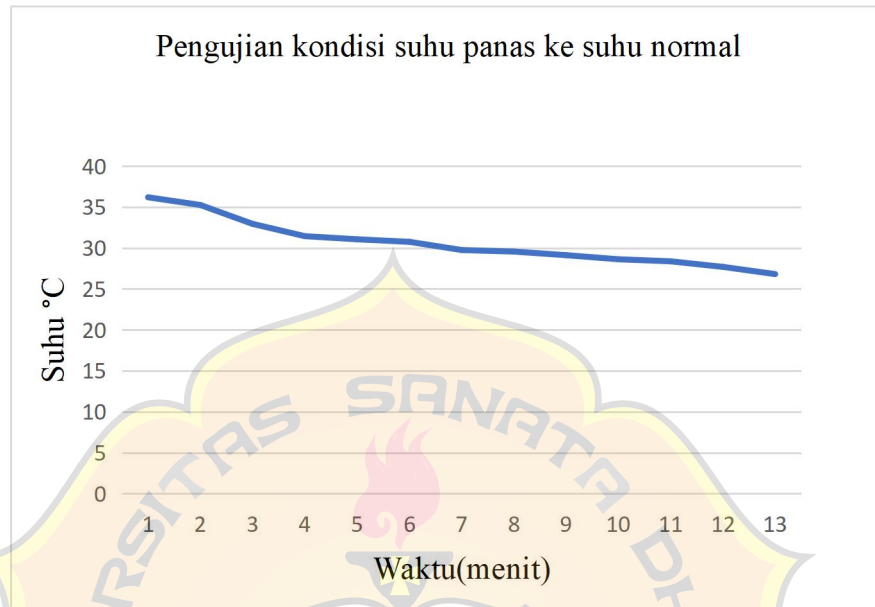
$$Ess = \left| \frac{27 - 26,81}{27} \right| \times 100\%$$

$$Ess = 0,703 \%$$

Tabel 4.4 Pengujian Kondisi Panas ke Kondisi Normal

pengujian 1					
Kondisi Panas ke Kondisi Normal pada Kolam 1					
waktu (menit)	Suhu (°C)	Kondisi Air	Pompa Pendingin		Keterangan Pengujian
			Kondisi Pengujian	Kondisi Seharusnya	
1	36.19	PANAS	ON	ON	Berhasil
2	35.25	PANAS	ON	ON	Berhasil
3	32.94	PANAS	ON	ON	Berhasil
4	31.44	PANAS	ON	ON	Berhasil
5	31.06	PANAS	ON	ON	Berhasil
6	30.75	PANAS	ON	ON	Berhasil
7	29.75	PANAS	ON	ON	Berhasil
8	29.56	PANAS	ON	ON	Berhasil
9	29.12	PANAS	ON	ON	Berhasil
10	28.62	PANAS	ON	ON	Berhasil
11	28.37	PANAS	ON	ON	Berhasil
12	27.69	PANAS	ON	ON	Berhasil
13	26.81	NORMAL	OFF	OFF	Berhasil

Kondisi saat menit pertama, suhu berada pada 36,19°C dan kondisi pompa pendingin *on* sampai suhu 27,69°C. Ketika suhu sudah mencapai 26,81°C pompa dalam kondisi *off*. Pompa pendingin berfungsi dengan baik dengan proses penurunan suhu membutuhkan waktu \pm 13 menit.



Gambar 4.6 Grafik Penurunan Suhu

4.2.3 Pengujian Suhu Dingin ke Suhu Normal

Pengujian ini dilakukan untuk mengamati respons sistem ketika suhu berubah dari kondisi dingin ke kondisi normal. *Setpoint* suhu ditetapkan pada 26°C. Proses pengujian dimulai dengan suhu awal air pada kondisi dingin pada suhu 22,06°C. Sistem merespon dengan mengaktifkan pompa air pemanas untuk meningkatkan suhu air hingga mencapai *setpoint* yang ditetapkan. Data hasil pengujian dapat dilihat pada Tabel 4.5, Grafik perubahan suhu selama pengujian ditunjukkan pada Gambar 4.7.

$$\text{Setpoint} = 26$$

$$X = 26,12$$

$$Ess = \left| \frac{\text{Setpoint} - X}{\text{Setpoint}} \right| \times 100\%$$

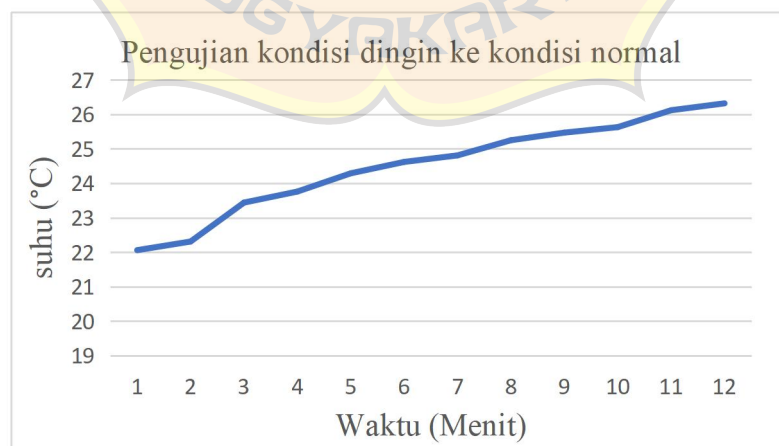
$$Ess = \left| \frac{26 - 26,12}{26} \right| \times 100\%$$

$$Ess = 0,461 \%$$

Tabel 4.5 Pengujian Suhu Dingin ke Suhu Normal

Pengujian 2					
Kondisi Dingin ke Kondisi Normal pada Kolam 2					
waktu (Menit)	suhu (°C)	Kondisi Air	Pompa Pemanas		Keterangan Pengujian
			Kondisi Pengujian	Kondisi Seharusnya	
0	22.06	DINGIN	ON	ON	Berhasil
1	22.31	DINGIN	ON	ON	Berhasil
2	23.44	DINGIN	ON	ON	Berhasil
3	23.94	DINGIN	ON	ON	Berhasil
4	24.19	DINGIN	ON	ON	Berhasil
5	24.62	DINGIN	ON	ON	Berhasil
6	24.81	DINGIN	ON	ON	Berhasil
7	25.25	DINGIN	ON	ON	Berhasil
8	25,37	DINGIN	ON	ON	Berhasil
9	25,62	DINGIN	ON	ON	Berhasil
10	26,12	DINGIN	OFF	OFF	Berhasil

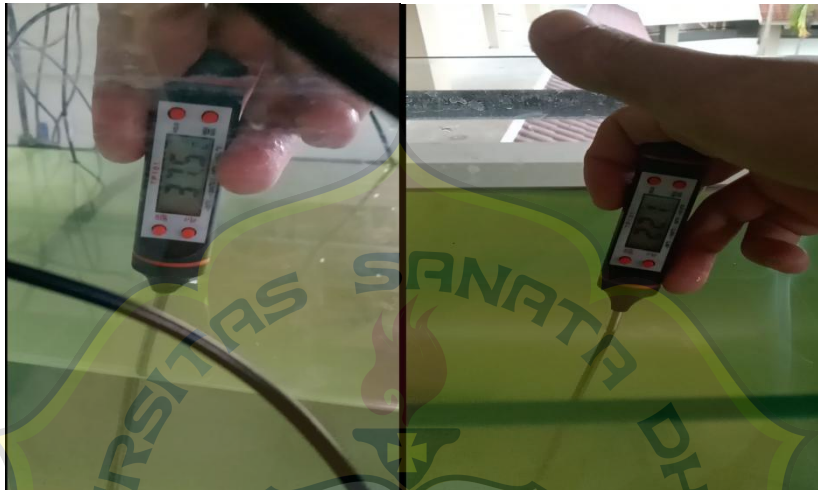
Pada kondisi pertama, suhu berada pada 22,06°C, menunjukkan kondisi air yang dingin. Pada kondisi ini, pompa pemanas berada dalam keadaan on untuk menaikkan suhu air sampai suhu menuju *setpoint* 26°C. Pada menit kesepuluh, suhu air mencapai 26,12°C yang sudah mencapai *setpoint* suhu normal. Pada kondisi ini, pompa pemanas secara otomatis beralih ke kondisi *off*. Proses kenaikan suhu sampai suhu Kembali normal membutuhkan waktu ± 10 menit. Sistem menunjukkan respon yang efektif dalam menjaga suhu air pada level yang ditetapkan sesuai dengan perancangan alat.



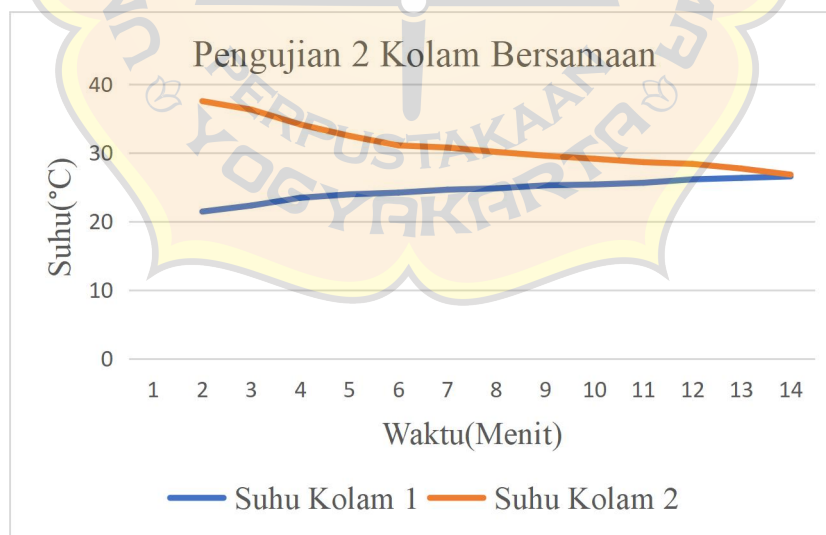
Gambar 4.7 Grafik Kenaikan Suhu

4.2.5 Pengujian Dua Kolam Bersamaan

Pengujian ini dilakukan untuk mengamati respon sistem ketika kedua bak uji dalam kondisi abnormal dan posisi sensor berada pada permukaan kolam dengan kondisi suhu kolam satu dalam kondisi panas dan suhu kolam dua dalam kondisi dingin. Gambar 4.8. Menunjukkan kondisi kolam pada saat pengujian, Grafik pengujian kedua kolam dapat dilihat pada Gambar 4.9.



Gambar 4.8 Suhu pada Kolam Satu dan Kolam Dua saat Pengujian.



Gambar 4.9 Grafik Pengujian Dua Kolam Dengan Kondisi Panas dan Kondisi Dingin.

Pengujian awal kolam pertama pada kondisi suhu awal $22,11^{\circ}\text{C}$ *relay* pompa pemanas dan kolam kedua pada kondisi suhu $37,51^{\circ}\text{C}$ mengaktifkan *relay* pompa pendingin. Pengujian ini membutuhkan waktu ± 14 menit untuk mencapai kondisi normal.

4.2.6 Pengujian Sensor Oksigen

Pengujian ini dilakukan untuk mengamati respons sistem ketika kadar oksigen terlarut turun dari kondisi rendah dan sampai kondisi normal pada suhu konstan 27°C. Berdasarkan nilai Do-tabel di kondisi suhu 27°C, nilai *setpoint* DO adalah sebesar 7,96 mg/L. Data hasil pengujian sensor oksigen dapat dilihat pada Tabel 4.6.

$$\text{Setpoint} = 7,96 \text{ mg/L}$$

$$X = 8,13$$

$$Ess = \left| \frac{\text{Setpoint} - X}{\text{Setpoint}} \right| \times 100\%$$

$$Ess = \left| \frac{7,96 - 8,13}{7,96} \right| \times 100\%$$

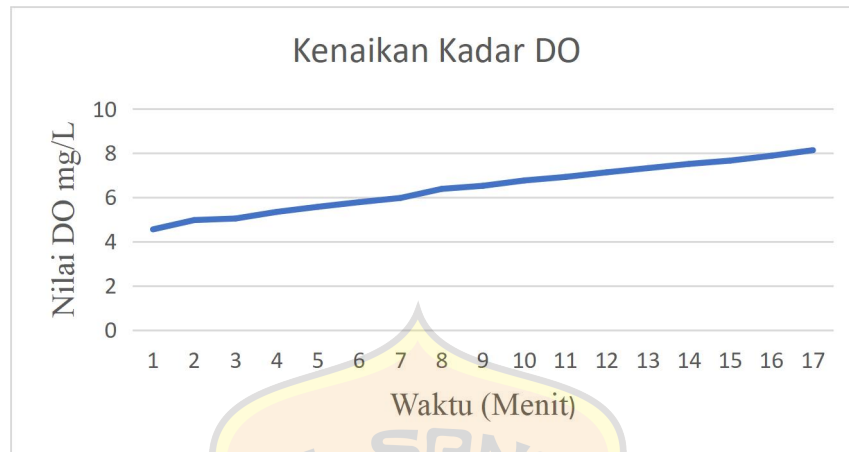
$$Ess = 2,135\%$$

Tabel 4.6 Pengujian Sensor Oksigen

Waktu (Menit)	Suhu °C	Nilai DO	Kondisi Aerator		Keterangan Pengujian
			Kondisi Pengujian	Kondisi Seharusnya	
0	27	4,55	ON	ON	Berhasil
1	27	4,97	ON	ON	Berhasil
2	27	5,04	ON	ON	Berhasil
3	27	5,34	ON	ON	Berhasil
4	27	5,57	ON	ON	Berhasil
5	27	5,78	ON	ON	Berhasil
6	27	5,97	ON	ON	Berhasil
7	27	6,38	ON	ON	Berhasil
8	27	6,52	ON	ON	Berhasil
9	27	6,76	ON	ON	Berhasil
10	27	6,92	ON	ON	Berhasil
11	27	7,13	ON	ON	Berhasil
12	27	7,32	ON	ON	Berhasil
13	27	7,51	ON	ON	Berhasil
14	27	7,66	ON	ON	Berhasil
15	27	7,88	ON	ON	Berhasil
16	27	8.13	OFF	OFF	Berhasil

Pengujian kadar oksigen dengan kondisi awal 4,55 mg/L. Sistem merespon dengan menyalakan aerator untuk meningkatkan kadar oksigen terlarut di kolam. Kenaikan kadar oksigen membutuhkan waktu ± 16 menit sampai mencapai nilai kadar oksigen yang ditetapkan. Hasil pengujian menunjukkan bahwa sistem mampu meningkatkan kadar

oksigen dari kondisi rendah hingga mencapai nilai *setpoint* yang ditetapkan. Grafik kenaikan kadar oksigen selama pengujian dapat dilihat pada Gambar 4.10, yang menunjukkan perubahan kadar oksigen dari kondisi rendah hingga mencapai *setpoint*.



Gambar 4.10 Kenaikan Kadar Oksigen Terlarut

4.3 Implementasi Perangkat Lunak

Implementasi perangkat lunak menjelaskan Bagian ini akan menjelaskan tentang hasil pengamatan sistem pada perangkat lunak, perangkat yang digunakan yaitu Arduino. Perangkat lunak Arduino *IDE* digunakan untuk memprogram Arduino Mega 2560 dan ESP 32 untuk menampilkan data pada aplikasi *Blynk*.

4.3.1 Program pada Arduino Mega

Sistem ini menggunakan enam *library* yang digunakan dalam pemrograman Arduino Mega. *Library* yang digunakan yaitu, *onewire.h*, *library* ini digunakan untuk sensor suhu DS18B20 agar dapat berkomunikasi melalui satu jalur bus data. *DallasTemperature.h* merupakan *library* untuk sensor suhu DS18B20. *Adafruit_GFX.h* dan *Adafruit_SSD1306.h*. adalah *library* untuk menggunakan Oled. *Library* keseluruhan sistem dapat dilihat pada Gambar 4.11.

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
```

Gambar 4.11 Program Library Arduino Mega 2560

Program untuk deklarasi pin pada Arduino Mega dapat dilihat pada Gambar 4.12. Program untuk mendeklarasikan pin sensor oksigen, sensor suhu, *relay* satu sampai *relay* enam dan deklarasi untuk tampilan pada OLED.

```
#define DO_PIN1 A5
#define DO_PIN2 A6
#define ONE_WIRE_BUS_1 8
#define ONE_WIRE_BUS_2 9

#define RELAY1 22 // pompa pemanas kolam1
#define RELAY2 24 // pompa pendingin kolam1
#define RELAY3 26 // pompa pemanas k2
#define RELAY4 28 // pompa pendingin k2
#define RELAY5 4 // aerator k1
#define RELAY6 5 // aerator k2

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1
Adafruit_SSD1306_display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);
```

Gambar 4.12 Deklarasi pin pada Arduino Mega

Program pada Gambar 4.13. merupakan program untuk memulai serial bawaan dari Arduino dengan *setting* kecepatan data 115200 bps dan setting kecepatan data 9600 bps untuk pengiriman data komunikasi ke NodeESP 32 serta terdapat pemanggilan program *persiapan_patra* untuk memulai program utama.

```
void setup() {
  Serial.begin(115200);
  Serial1.begin(9600); // komunikasi ke esp 32
  persiapan_patra();
}
```

Gambar 4.13 Void Setup

Program pada Gambar 4.14. merupakan program utama dari sistem pemantauan dan pengendalian suhu dan kadar oksigen pada model tambak udang. Pertama dengan mendeklarasikan kondisi awal untuk masing-masing *relay*, pembacaan sensor suhu dan juga sensor oksigen. Data dari sensor suhu dan sensor oksigen akan ditampilkan pada OLED dan dikirimkan ke ESP 32 untuk menampilkan data di aplikasi *blynk*. Gambar 4.15 terdapat pengendalian *relay* yang digunakan saat suhu pada air kurang dari *setpoint* atau

lebih dari *setpoint* yang sudah ditentukan serta pengendalian *relay* yang digunakan saat kadar oksigen air kurang atau lebih dari *setpoint* yang sudah ditentukan.

```
void persiapan_patra() {
  sensors1.begin();
  sensors2.begin();

  pinMode(RELAY1, OUTPUT);
  pinMode(RELAY2, OUTPUT);
  pinMode(RELAY3, OUTPUT);
  pinMode(RELAY4, OUTPUT);
  pinMode(RELAY5, OUTPUT);
  pinMode(RELAY6, OUTPUT);

  digitalWrite(RELAY1, HIGH);
  digitalWrite(RELAY2, HIGH);
  digitalWrite(RELAY3, HIGH);
  digitalWrite(RELAY4, HIGH);
  digitalWrite(RELAY5, HIGH);
  digitalWrite(RELAY6, HIGH);

  if (!sensors1.getAddress(tempSensor1, 0)) Serial.println("Gagal membaca alamat 1 on Bus 1");
  if (!sensors2.getAddress(tempSensor2, 0)) Serial.println("Gagal membaca alamat 2 on Bus 2");

  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println(F("SSD1306 allocation failed"));
    for (;;)
      ;
  }
  display.display();
  delay(2000);
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(SSD1306_WHITE);
}
```

Gambar 4.14 Void Setup.

Program Pada Gambar 4.15. merupakan program untuk mengontrol *relay* berdasarkan pembacaan suhu dan kadar oksigen dari dua sensor suhu dan dua sensor DO. Program ini diaktifkan sesuai dengan kondisi air pada model kolam tambak udang

```
if (temperature1 < 24) {
  digitalWrite(RELAY1, LOW);
  relay1Active = true;
} else if (relay1Active && temperature1 >= 26) {
  digitalWrite(RELAY1, HIGH);
  relay1Active = false;
}

if (temperature1 > 30) {
  digitalWrite(RELAY2, LOW);
  relay2Active = true;
} else if (relay2Active && temperature1 <= 27) {
  digitalWrite(RELAY2, HIGH);
  relay2Active = false;
}

if (temperature2 < 24) {
  digitalWrite(RELAY3, LOW);
  relay3Active = true;
} else if (relay3Active && temperature2 >= 26) {
  digitalWrite(RELAY3, HIGH);
  relay3Active = false;
}

if (temperature2 > 30) {
  digitalWrite(RELAY4, LOW);
  relay4Active = true;
} else if (relay4Active && temperature2 <= 27) {
  digitalWrite(RELAY4, HIGH);
  relay4Active = false;
}

if (DO1 < float(DO_Table[temp1]) / 1000 - 0.8) {
  digitalWrite(RELAY5, LOW);
  relay5Active = true;
} else if (relay5Active && DO1 >= float(DO_Table[temp1]) / 1000 - 0.2) {
```

Gambar 4.15 Program On Off Relay

Program pada Gambar 4.16. merupakan program untuk melakukan pembacaan nilai pada sensor suhu dan sensor oksigen. Fungsi dalam program bertujuan untuk mengambil, menghitung rata-rata, dan menampilkan data dari sensor secara berkelanjutan. Program dimulai dengan pembacaan suhu pada bus satu dan bus dua. Setiap pembacaan suhu dan nilai analog dikonversi menjadi tegangan nilai-nilai ini kemudian ditambahkan untuk dihitung nilai rata-ratanya.

```
void looping_patra() {
  sensors1.requestTemperatures();
  sensors2.requestTemperatures();
  for (int i = 0; i < 10; i++) {
    float temperature1 = sensors1.getTempC(tempSensor1);
    tot_temp1 = tot_temp1 + temperature1;
    ADC_Raw1 = analogRead(DO_PIN1);
    ADC_Voltage1 = float(VREF) * ADC_Raw1 / ADC_RES;
    tot_adc1 = tot_adc1 + ADC_Voltage1;
  }
  for (int i = 0; i < 10; i++) {
    float temperature2 = sensors2.getTempC(tempSensor2);
    tot_temp2 = tot_temp2 + temperature2;
    ADC_Raw2 = analogRead(DO_PIN2);
    ADC_Voltage2 = float(VREF) * ADC_Raw2 / ADC_RES;
    tot_adc2 = tot_adc2 + ADC_Voltage2;
  }
  float temperature1 = tot_temp1 / 10;
  uint8_t temp1 = temperature1;
  tot_temp1 = 0;
  ADC_Voltage1 = tot_adc1 / 10;
  tot_adc1 = 0;
  DO1 = readDO(ADC_Voltage1, temperature1, CAL1_V1, CAL1_T1, CAL2_V1, CAL2_T1) / 1000.0;

  Serial.print("Sensor 1 - Temperature: ");
  Serial.print(temperature1);
  Serial.print(" C, ADC RAW: ");
  Serial.print(ADC_Raw1);
  Serial.print(", Voltage: ");
  Serial.print(ADC_Voltage1);
}
```

Gambar 4.16 Program Pembacaan Nilai Sensor.

Pada Gambar 4.17. merupakan program untuk menampilkan nilai suhu dan kadar oksigen terlarut dari sensor pada oled.

```
display.clearDisplay();
display.setCursor(0, 0);
display.print("Temp 1: ");
display.print(temperature1);
display.print(" C");

display.setCursor(0, 10);
display.print("DO 1: ");
display.print(DO1);
display.print(" mg/L");

display.setCursor(0, 20);
display.print("Temp 2: ");
display.print(temperature2);
display.print(" C");

display.setCursor(0, 30);
display.print("DO 2: ");
display.print(DO2);
display.print(" mg/L");

display.display();
delay(1000);
}
```

Gambar 4.7 Tampilan pada Oled

4.3.2 Program pada ESP 32

Program pada Gambar 4.18. Merupakan program pada ESP32 untuk menghubungkan ke jaringan wifi yang sudah ditentukan dan terdapat token otentikasi

untuk ke platform *blynk*. Deklarasi virtual pin bertujuan untuk menampilkan data pada *blynk* sesuai dengan virtual pin yang sudah ditentukan pada *blynk apps*.

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>

char auth[] = "I20KZZx0qpsuLxOdNg7BuKof0QwPkIDh";
char ssid[] = "tralalala";
char pass[] = "tobrut123#";

// Blynk virtual pins
#define VPIN_T1 V1
#define VPIN_DO1 V2
#define VPIN_T2 V3
#define VPIN_DO2 V4

void setup()
{
  Serial.begin(9600); // Ckomunikasi serial mega
  Blynk.begin(auth, ssid, pass, "iot.serangkota.go.id", 8080); // server blynk
}
```

Gambar 4.18 Program deklarasi pin dan library pada ESP 32

Program pada Gambar 4.19. merupakan program untuk mengirim data sensor yang diterima ESP 32 dilanjutkan pengiriman ke *blynk apps*.

```
Blynk.virtualWrite(VPIN_T1, t1);
Blynk.virtualWrite(VPIN_DO1, do1);
Blynk.virtualWrite(VPIN_T2, t2);
Blynk.virtualWrite(VPIN_DO2, do2);
}
Blynk.run();
}
```

Gambar 4.19 Program untuk mengirim data ke *blynk apps*.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari hasil pengujian dan pengambilan data yang telah dilakukan terhadap “Sistem Pemantauan dan Pengendalian Suhu Kadar Oksigen pada Model Tambak Udang Vaname dengan Teknologi IoT”, beberapa kesimpulan dapat diambil sebagai berikut:

1. Alat yang telah dibuat sudah bekerja dengan baik.
2. Pemantauan secara langsung melalui oled dan secara jarak jauh melalui smartphone sudah berjalan dengan baik sesuai perancangan.
3. Kedua sensor sudah bekerja dengan baik. Rata-rata eror pembacaan pada sensor suhu sebesar 1,41% dan rata-rata eror pembacaan pada sensor oksigen sebesar 1,86%.
4. Pompa pendingin sudah bekerja dengan baik dengan proses penurunan suhu dari kondisi panas ke suhu normal membutuhkan waktu \pm 13 menit. Sistem menunjukkan respon yang efektif dalam menjaga suhu air pada kondisi yang sudah ditetapkan sesuai dengan perancangan alat dengan nilai *Ess* sebesar 0,703%.
5. Pompa pemanas sudah bekerja dengan baik dengan proses kenaikan suhu dari suhu dingin ke suhu normal membutuhkan waktu \pm 10 menit. Sistem menunjukkan respon yang efektif dalam menjaga suhu air pada kondisi yang sudah ditetapkan sesuai dengan perancangan alat dengan nilai *Ess* sebesar 0,461%.

5.2. Saran

1. Penambahan sistem Filterasi air agar air dapat digunakan secara berkelanjutan.

DAFTAR PUSTAKA

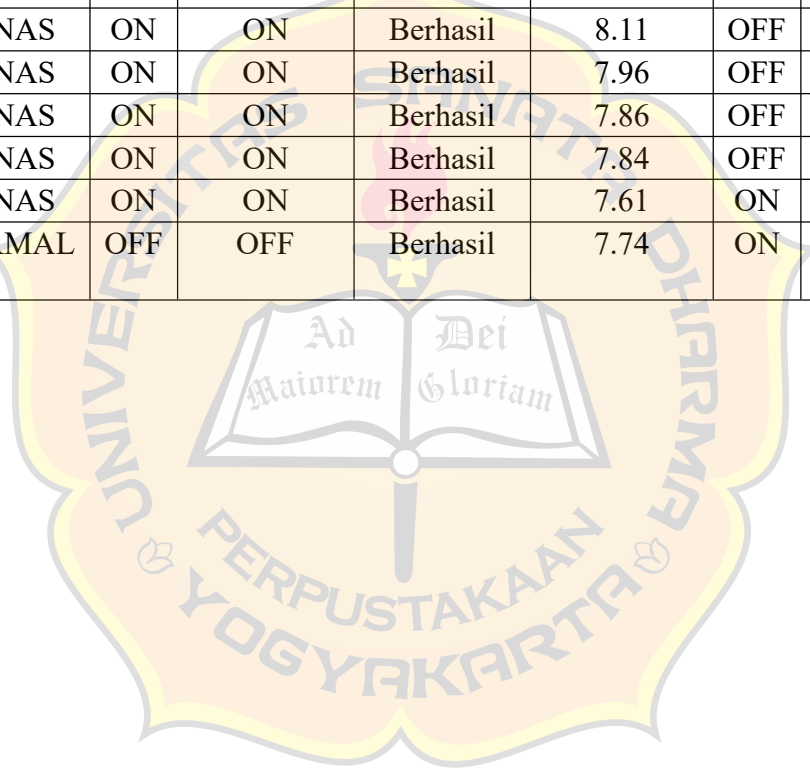
1. Andhika Rakhmanda, Naila Husnayain, 2021, Membangun Keberlanjutan Pangan dan Perikanan: Ciptakan Udang Segar, Sehat dan Berkualitas melalui Budidaya Udang Berkelanjutan.
2. Halim, A.M., Krisnawati, M. and Fauziah, A., 2021, Dinamika Kualitas Air Pada Pembesaran Udang Vannamei (*Litopenaeus vannamei*) Secara Intensif Di Pt. Andulang Shrimp Farm Desa Andulang Kecamatan Gapura Kabupaten Sumenep Jawa Timur. *Jurnal Penelitian Chanos Chanos*, 19(2), pp.143-153.
3. Detik.com, 2022, Dampak Perubahan Iklim, Udang Tambak di Brebes Gagal Dipanen, diakses 10 Oktober 2023.
4. fenza, 2023, Penyebab Gagal Panen di Tambak Udang & Cara Mengatasinya, diakses 10 Oktober 2023.
5. Stevanus Hari Wijatmika. 2014, Alat Ukur Kadar Oksigen Terlarut Dengan Metode Elektrolisis Berbasis ATmega8535, Tugas Akhir, Jurusan Teknik Elektro, FST, Universitas Sanata Dharma, Yogyakarta.
6. Putri, Novia Indriani. 2020, Sistem Monitoring Dan Pengendalian Jarak Jauh Tingkat Keasaman Dan Salinitas Kolam Ikan Koi, Tugas Akhir, Jurusan Teknik Elektro, FST, Universitas Sanata Dharma, Yogyakarta.
7. Halim, A. M., Fauziah, A., & Aisyah, N. 2022, Kesesuaian Kualitas Air pada Tambak Udang Vannamei (*Litopenaeus vannamei*) di CV. Lancar Sejahtera Abadi, Probolinggo, Jawa Timur. *Jurnal Penelitian Chanos Chanos*, 20(2), 77-88.
8. Bangsa, Puja Cikal, Razali Daud, and Nuzul Asmilia. 2015, Pengaruh Peningkatan Suhu terhadap Jumlah Eritrosit Ikan Nila (*Oreochromis niloticus*), *Jurnal Medika Veterinaria* 9, no. 1.
9. Ma'arif, Alfian., 2021 Dasar sistem kendali pemodelan, pengendalian, analisis, simulasi, dan implementasi. Uad Press.
10. Abarca, Roberto Maldonado., 2021, Sistem Mikro Kontroler., *Nuevos Sist. Comun. e Inf* 4, no. 1.

11. Suciyati, S.W. and Gurum A Pauzi, G.A.P., 2021, *An analysis of data acquisition system of temperature, oxygen, and carbon dioxide in refrigerator with arduino mega 2560*. Jurnal Ilmiah Pendidikan Fisika Al-Biruni, 10(1), pp.119-127.
12. Labelektronika., ARDUINO MEGA 2560 MIKROKONTROLER ATmega2560, diakses 10 Oktober 2023.
13. DFROBOT,. Gravity: Analog Dissolved Oxygen Sensor SKU:SEN0237, diakses 10 Oktober 2023.
14. Ikhsan, Rizqy Nurul, and Niken Syafitri., Pemanfaatan Sensor Suhu DS18B20 sebagai Penstabil Suhu Air Budidaya Ikan Hias, In Prosiding Seminar Nasional Energi, Telekomunikasi dan Otomasi (SNETO), pp. 18-26. 2021.
15. Wu, Yan Xiang, Dan Liu, and Xing Hong Kuang., 2011 *A Temperature detecting system based on DS18B20*, *Advanced Materials Research* 328.
16. Misel, 2023., Mengenal Definisi, Jenis, dan Fungsi Aktuator otomatisasi, diakses 20 Desember 2023.
17. Ariansyah, Muhammad Dwi, and Sariman Sariman., 2021 Analisa Performa Pompa Air DC 12V 42 Watt terhadap Variasi Kedalaman Pipa Menggunakan Baterai dengan Sumber Energi dari Matahari, Jurnal Syntax Admiration 2, no. 6: 1083-1102.
18. N.Lathif., Rekomendasi Pompa Akuarium Terbaik, diakses 20 Desember 2023.
19. Yuliyanti., 2023 Rancang Bangun Sistem Aerator dan Kontrol Kualitas Air untuk Budidaya Udang Vaname Berbasis Arduino,” PP. 1-48.
20. Y. Efendi., 2018, Internet of Things (IoT) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile, Jurnal Ilmu Ilmu Komputer., vol. 4, no. 2, pp. 21–27, doi: 10.35329/jiik.v4i2.41.
21. N. Nordin., 2020 *Integration of IoT on Power Monitoring and Control for Housing Electrical System*, vol. 2, no. 2, pp. 1–50.



LAMPIRAN 1

Kondisi panas ke normal									
waktu (menit)	suhu (°C)	Kondisi Air	pompa pendingin		Keterangan	Nilai DO (mg/L)	Kondisi Aerator		Keterangan
			Real	Seharusnya			Real	Seharusnya	
1	36.19	PANAS	ON	ON	Berhasil	5.34	ON	ON	Berhasil
2	35.25	PANAS	ON	ON	Berhasil	6.73	ON	ON	Berhasil
3	32.94	PANAS	ON	ON	Berhasil	6.73	ON	ON	Berhasil
4	31.44	PANAS	ON	ON	Berhasil	7.15	ON	ON	Berhasil
5	31.06	PANAS	ON	ON	Berhasil	7.12	ON	ON	Berhasil
6	30.75	PANAS	ON	ON	Berhasil	7.33	ON	ON	Berhasil
7	29.75	PANAS	ON	ON	Berhasil	8.17	OFF	OFF	Berhasil
8	29.56	PANAS	ON	ON	Berhasil	8.11	OFF	OFF	Berhasil
9	29.12	PANAS	ON	ON	Berhasil	7.96	OFF	OFF	Berhasil
10	28.62	PANAS	ON	ON	Berhasil	7.86	OFF	OFF	Berhasil
11	28.37	PANAS	ON	ON	Berhasil	7.84	OFF	OFF	Berhasil
12	27.69	PANAS	ON	ON	Berhasil	7.61	ON	ON	Berhasil
13	26.81	NORMAL	OFF	OFF	Berhasil	7.74	ON	ON	Berhasil



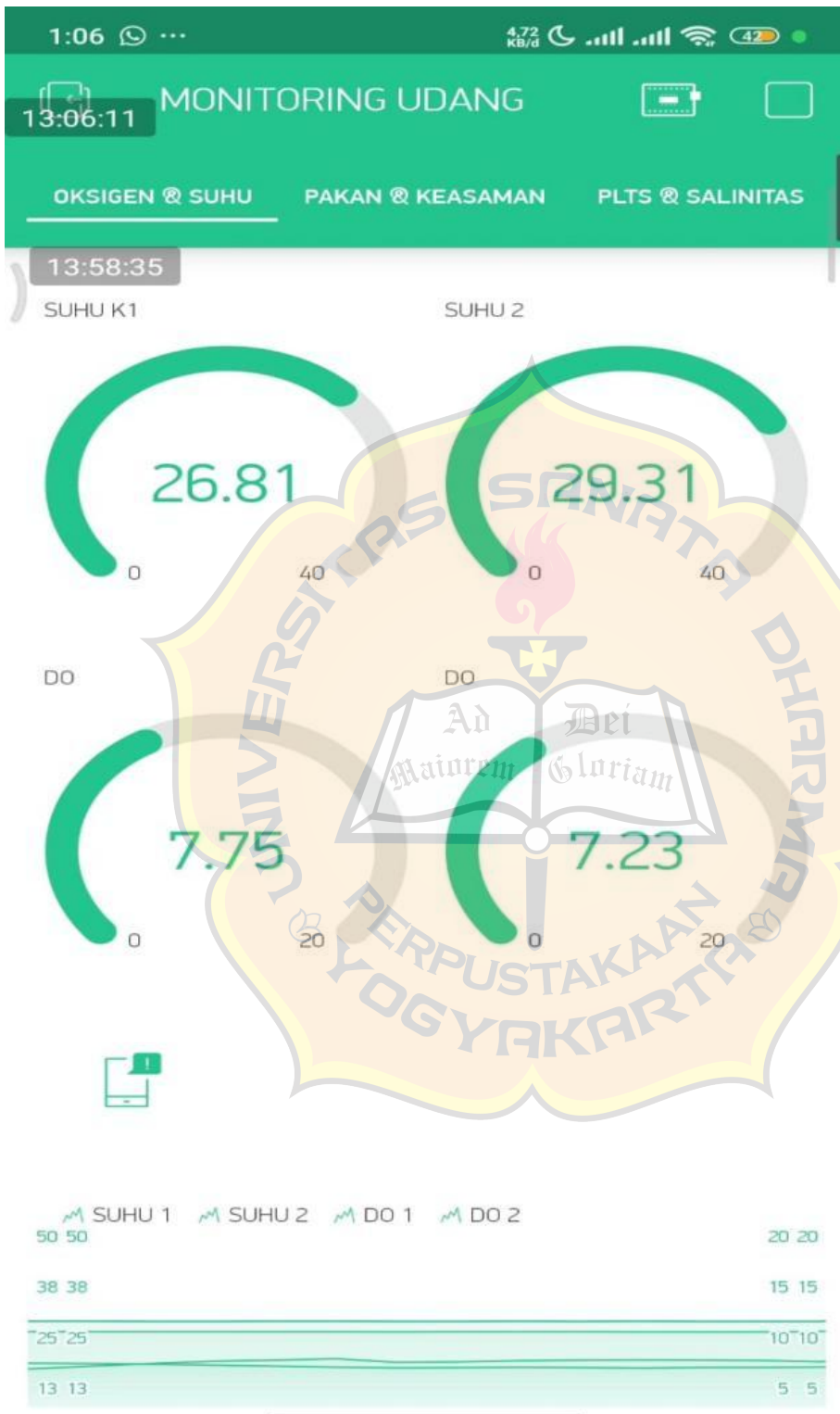
LAMPIRAN 2

Cairan electrolyte



LAMPIRAN 3

Tampilan Pada blynk



LAMPIRAN 4

Tampilan pada Oled



LAMPIRAN 5

Listing program arduino mega

```

#include <Arduino.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define DO_PIN1 A5
#define DO_PIN2 A6
#define ONE_WIRE_BUS_1 8
#define ONE_WIRE_BUS_2 9

#define RELAY1 22 // temp 1
#define RELAY2 24 // temp1
#define RELAY3 26
#define RELAY4 28
#define RELAY5 4
#define RELAY6 5

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET -1
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RESET);

#define VREF 4900 // VREF in mV
#define ADC_RES 1024 // ADC Resolution

#define TWO_POINT_CALIBRATION 1

#define CAL1_V1 (1095) // mv for sensor 1
#define CAL1_T1 (35) // °C for sensor 1
#define CAL1_V2 (569) // mv for sensor 2
#define CAL1_T2 (35) // °C for sensor 2

#define CAL2_V1 (580) // mv for sensor 1
#define CAL2_T1 (16) // °C for sensor 1
#define CAL2_V2 (138) // mv for sensor 2
#define CAL2_T2 (16) // °C for sensor 2

const uint16_t DO_Table[41] = {
  14460, 14220, 13820, 13440, 13090, 12740, 12420, 12110, 11810, 11530,
  11260, 11010, 10770, 10530, 10300, 10080, 9860, 9660, 9460, 9270,
  9080, 8900, 8730, 8570, 8410, 8250, 8110, 7960, 7820, 7690,
  7560, 7430, 7300, 7180, 7070, 6950, 6840, 6730, 6630, 6530, 6410
};

```

```

OneWire oneWire1(ONE_WIRE_BUS_1);
OneWire oneWire2(ONE_WIRE_BUS_2);

DallasTemperature sensors1(&oneWire1);
DallasTemperature sensors2(&oneWire2);

DeviceAddress tempSensor1, tempSensor2;

float ADC_Raw1, ADC_Raw2;
float ADC_Voltage1, ADC_Voltage2;
float DO1, DO2;
float tot_temp1 = 0;
float tot_adc1 = 0;
float tot_temp2 = 0;
float tot_adc2 = 0;

float readDO(uint32_t voltage_mv, uint8_t temperature_c, float CAL1_V, float CAL1_T,
float CAL2_V, float CAL2_T) {
  uint16_t V_saturation = (int16_t)((int8_t)temperature_c - CAL2_T) *
((uint16_t)CAL1_V - CAL2_V) / ((uint8_t)CAL1_T - CAL2_T) + CAL2_V;
  Serial.print(voltage_mv);
  Serial.print(" ");
  Serial.print(V_saturation);
  Serial.print(" ");
  Serial.println(DO_Table[(int)temperature_c]);
  return (voltage_mv * DO_Table[temperature_c] / V_saturation);
}

bool relay1Active = false;
bool relay2Active = false;
bool relay3Active = false;
bool relay4Active = false;
bool relay5Active = false;
bool relay6Active = false;

void setup() {
  Serial.begin(115200);
  Serial1.begin(9600); // Initialize Serial1 for communication with ESP32
  persiapan_patra();
}

void loop() {
  looping_patra();
}

void persiapan_patra() {
  sensors1.begin();
  sensors2.begin();

```

```

pinMode(RELAY1, OUTPUT);
pinMode(RELAY2, OUTPUT);
pinMode(RELAY3, OUTPUT);
pinMode(RELAY4, OUTPUT);
pinMode(RELAY5, OUTPUT);
pinMode(RELAY6, OUTPUT);

digitalWrite(RELAY1, HIGH);
digitalWrite(RELAY2, HIGH);
digitalWrite(RELAY3, HIGH);
digitalWrite(RELAY4, HIGH);
digitalWrite(RELAY5, HIGH);
digitalWrite(RELAY6, HIGH);

if (!sensors1.getAddress(tempSensor1, 0)) Serial.println("Unable to find address for
Sensor 1 on Bus 1");
if (!sensors2.getAddress(tempSensor2, 0)) Serial.println("Unable to find address for
Sensor 2 on Bus 2");

if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
  Serial.println(F("SSD1306 allocation failed"));
  for (;;)
    ;
}
display.display();
delay(2000);
display.clearDisplay();
display.setTextSize(1);
display.setTextColor(SSD1306_WHITE);
}

void looping_patra() {
  sensors1.requestTemperatures();
  sensors2.requestTemperatures();
  for (int i = 0; i < 10; i++) {
    float temperature1 = sensors1.getTempC(tempSensor1);
    tot_temp1 = tot_temp1 + temperature1;
    ADC_Raw1 = analogRead(DO_PIN1);
    ADC_Voltage1 = float(VREF) * ADC_Raw1 / ADC_RES;
    tot_adc1 = tot_adc1 + ADC_Voltage1;
  }
  for (int i = 0; i < 10; i++) {
    float temperature2 = sensors2.getTempC(tempSensor2);
    tot_temp2 = tot_temp2 + temperature2;
    ADC_Raw2 = analogRead(DO_PIN2);
    ADC_Voltage2 = float(VREF) * ADC_Raw2 / ADC_RES;
    tot_adc2 = tot_adc2 + ADC_Voltage2;
  }
}

```



```

float temperature1 = tot_temp1 / 10;
uint8_t temp1 = temperature1;
tot_temp1 = 0;
ADC_Voltage1 = tot_adc1 / 10;
tot_adc1 = 0;
DO1 = readDO(ADC_Voltage1, temperature1, CAL1_V1, CAL1_T1, CAL2_V1,
CAL2_T1) / 1000.0;

```

```

Serial.print("Sensor 1 - Temperature: ");
Serial.print(temperature1);
Serial.print(" °C, ADC RAW: ");
Serial.print(ADC_Raw1);
Serial.print(", Voltage: ");
Serial.print(ADC_Voltage1);
Serial.print(" mV, DO: ");
Serial.println(DO1);

```

```

//float temperature2 = sensors2.getTempC(tempSensor2);
//ADC_Raw2 = analogRead(DO_PIN2);
//ADC_Voltage2 = float(VREF) * ADC_Raw2 / ADC_RES;
float temperature2 = tot_temp2 / 10;
uint8_t temp2 = temperature2;
tot_temp2 = 0;
ADC_Voltage2 = tot_adc2 / 10;
tot_adc2 = 0;
DO2 = readDO(ADC_Voltage2, temperature2, CAL1_V2, CAL1_T2, CAL2_V2,
CAL2_T2) / 1000.0;

```

```

Serial.print("Sensor 2 - Temperature: ");
Serial.print(temperature2);
Serial.print(" °C, ADC RAW: ");
Serial.print(ADC_Raw2);
Serial.print(", Voltage: ");
Serial.print(ADC_Voltage2);
Serial.print(" mV, DO: ");
Serial.println(DO2);

```

```

// Relay logic for Sensor 1
if (temperature1 < 24) {
  digitalWrite(RELAY1, LOW);
  relay1Active = true;
} else if (relay1Active && temperature1 >= 26) {
  digitalWrite(RELAY1, HIGH);
  relay1Active = false;
}
if (temperature1 > 30) {
  digitalWrite(RELAY2, LOW);
  relay2Active = true;
} else if (relay2Active && temperature1 <= 27) {

```

```

digitalWrite(RELAY2, HIGH);
relay2Active = false;
}
if (temperature2 < 24) {
digitalWrite(RELAY3, LOW);
relay3Active = true;
} else if (relay3Active && temperature2 >= 26) {
digitalWrite(RELAY3, HIGH);
relay3Active = false;
}
if (temperature2 > 30) {
digitalWrite(RELAY4, LOW);
relay4Active = true;
} else if (relay4Active && temperature2 <= 27) {
digitalWrite(RELAY4, HIGH);
relay4Active = false;
}
if (DO1 < float(DO_Table[temp1]) / 1000 - 0.8) {
digitalWrite(RELAY5, LOW);
relay5Active = true;
} else if (relay5Active && DO1 >= float(DO_Table[temp1]) / 1000 - 0.2) {
digitalWrite(RELAY5, HIGH);
relay5Active = false;
}
if (DO2 < float(DO_Table[temp2]) / 1000 - 0.8) {
digitalWrite(RELAY6, LOW);
relay6Active = true;
} else if (relay6Active && DO2 >= float(DO_Table[temp2]) / 1000 - 0.2) {
digitalWrite(RELAY6, HIGH);
relay6Active = false;
}
Serial1.print("T1:");
Serial1.print(temperature1);
Serial1.print(",DO1:");
Serial1.print(DO1);
Serial1.print(",T2:");
Serial1.print(temperature2);
Serial1.print(",DO2:");
Serial1.println(DO2);

// Update the OLED display
display.clearDisplay();
display.setCursor(0, 0);
display.print("Temp 1: ");
display.print(temperature1);
display.print(" C");

display.setCursor(0, 10);
display.print("DO 1: ");

```

```

display.print(DO1);
display.print(" mg/L");

display.setCursor(0, 20);
display.print("Temp 2: ");
display.print(temperature2);
display.print(" C");

display.setCursor(0, 30);
display.print("DO 2: ");
display.print(DO2);
display.print(" mg/L");

display.display();

delay(1000);
}

```

Listing program esp 32

```

#define BLYNK_PRINT Serial
// Include necessary libraries
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <softwareserial.h>
char auth[] = "I2OKZZx0qpsuLxOdNg7BuKof0QwPkIDh";
char ssid[] = "tralalala";
char pass[] = "tobrut123#";

// Blynk virtual pins
#define VPIN_T1 V1
#define VPIN_DO1 V2
#define VPIN_T2 V3
#define VPIN_DO2 V4

void setup()

```

```
{
  Serial.begin(9600); // Ckomunikasi serial mega
  Blynk.begin(auth, ssid, pass, "iot.serangkota.go.id", 8080); // server blynk
}

void loop()
{
  if (Serial.available())
  {
    String data = Serial.readStringUntil('\n');
    float t1 = data.substring(data.indexOf("T1:") + 3, data.indexOf(",DO1:")).toFloat();
    float do1 = data.substring(data.indexOf("DO1:") + 4, data.indexOf(",T2:")).toFloat();
    float t2 = data.substring(data.indexOf("T2:") + 3, data.indexOf(",DO2:")).toFloat();
    float do2 = data.substring(data.indexOf("DO2:") + 4).toFloat();

    // Display data on Serial Monitor
    Serial.print("Received data -> ");
    Serial.print("T1: ");
    Serial.print(t1);
    Serial.print(" °C, DO1: ");
    Serial.print(do1);
    Serial.print(" mg/L, T2: ");
    Serial.print(t2);
    Serial.print(" °C, DO2: ");
    Serial.println(do2);
    Serial.println(" mg/L");

    // Send data to Blynk
    Blynk.virtualWrite(VPIN_T1, t1);
    Blynk.virtualWrite(VPIN_DO1, do1);
```

```
Blynk.virtualWrite(VPIN_T2, t2);  
Blynk.virtualWrite(VPIN_DO2, do2);  
}  
Blynk.run();  
}
```



Lampiran 6

Peletakan sensor pada saat pengujian







Lampiran 7

Kalibrasi sensor oksigen



