

# **SISTEM KOMUNIKASI DAN *MONITORING* PURWARUPA *TRASH SKIMMER***

## **TUGAS AKHIR**

**Karya tulis sebagai salah satu syarat  
untuk memperoleh gelar Sarjana dari  
Institut Teknologi Bandung**

**oleh**

**RIFQI NAFIS MUBAROQ**

**NIM: 10218091**



**PROGRAM STUDI SARJANA FISIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
INSTITUT TEKNOLOGI BANDUNG  
2022**

## ABSTRAK

### SISTEM KOMUNIKASI DAN *MONITORING* PURWARUPA *TRASH SKIMMER*

oleh

**Rifqi Nafis Mubaroq**

**NIM : 10218091**

Menurut data Sistem Informasi Pengelolaan Sampah Nasional (SIPSN) pada 2021 Indonesia menghasilkan timbulan sampah sebesar 28,69 juta ton/tahun dan 15,6% diantaranya merupakan sampah plastik. Jawa Barat sebagai provinsi dengan penduduk terbanyak di Indonesia menjadi penyumbang sampah terbanyak. pengelolaan sampah yang masih belum maksimal mengakibatkan banyaknya penduduk yang membuang sampah ke Sungai Citarum. Pada tahun 2021 sampah yang dikumpulkan dari Sungai Citarum mencapai 15.838 ton/hari dengan persentase sampah plastik sebesar 15,35%. Oleh karena itu dibutuhkan suatu *trash skimmer* yang dapat mengumpulkan sampah-sampah plastik di permukaan air sebagai solusi untuk membersihkan Sungai Citarum. Tujuan perancangan *trash skimmer* agar mampu mengangkut sampah pada permukaan air dalam jumlah besar, namun tidak tenggelam atau kesulitan bergerak. Pada penelitian ini akan dirancang sistem komunikasi berbasis modul radio frekuensi, sistem penggerak yang terdiri dari rangkaian konveyor dan rangkaian propeler, dan sistem *monitoring* berbasis IoT yang terdiri dari rangkaian sistem peringatan kapasitas sampah, rangkaian sistem pemantauan daya, dan rangkaian sistem pelacakan lokasi terkini. Berdasarkan pengujian *trash skimmer*, didapatkan data sistem komunikasi *trash skimmer* berupa rentang keberhasilan komunikasi modul RF *transmitter-receiver* yakni 0-30 m. kualitas komunikasi modul RF dipengaruhi oleh jarak dan penghalang yang dibuktikan dengan semakin kecilnya jumlah sinyal yang diterima ketika jarak semakin jauh dan atau penghalang yang semakin besar. Sistem *monitoring trash skimmer* berbasis IoT dibuat dengan menggunakan *database* InfluxDB dengan *program* Node-Red menggunakan *data processor* Raspberry Pi. Data yang ditampilkan berupa pemantauan persentase daya, sistem peringatan kapasitas sampah, dan lokasi terkini menggunakan API *dashboard* Grafana.

Kata kunci: Grafana, Modul RF, *Monitoring*, Sampah, *Trash Skimmer*.

## **ABSTRACT**

### **COMMUNICATION AND MONITORING SYSTEM ON TRASH SKIMMER PROTOTYPE**

by

**Rifqi Nafis Mubaroq**

**NIM : 10218091**

According to National Waste Management Information System (SIPSN) data in 2021, Indonesia generated 28.69 million tons of garbage per year and 15.6% of it is plastic waste. West Java as the province with the largest population in Indonesia is the largest contributor to waste. Waste management that is not good enough made residents throwing their garbage into the Citarum River. In 2021 the waste collected from the Citarum River reached 15,838 tons/day with a percentage of plastic waste of 15.35%. Therefore we need a trash skimmer that can collect plastic waste on the surface of the water as a solution to clean the Citarum River. The purpose of designing the trash skimmer is to be able to transport large amounts of garbage on the surface of the water, and also not sink or have difficulty in moving. In this study, a radio frequency module-based communication system will be designed, a drive system consisting of a conveyor circuit and a propeller circuit, and an IoT-based monitoring system consisting of a series of waste capacity warning systems, a series of power monitoring systems, and a series of the latest location tracking systems. Based on the trash skimmer test, the trash skimmer communication system data obtained in the form of a communication success range of the RF transmitter-receiver module, which is 0-30m. The communication quality of the RF module is affected by distance and obstructions as evidenced by the smaller the number of signals received when the distance and/or obstructions are getting thicker. An IoT-based trash skimmer monitoring system was created using the InfluxDB database with the Node-Red program using the Raspberry Pi data processor. The data displayed is in the form of monitoring power presentations, waste capacity warning systems, and current locations using the Grafana dashboard API.

**Keywords:** Garbage, Grafana, Monitoring, RF Module, Trash Skimmer.

**SISTEM KOMUNIKASI DAN *MONITORING* PURWARUPA  
*TRASH SKIMMER***

oleh  
**Rifqi Nafis Mubaroq**  
**NIM : 10218091**

**Program Studi Sarjana Fisika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Institut Teknologi Bandung**

Menyetujui

Dosen Pembimbing 1

Dosen Pembimbing 2

Tanggal .....

Tanggal .....

---

Prof. Dr. Ing. Mitra Djamal  
NIP. 19600522 198503 1 002

---

Dr. Nina Siti Aminah, S.Si., M.Si.  
NIP. 19830415 2012122 004

Tim Penguji:

1. Fahdzi Muttaqien, S.Si., M.Si., M.Sc., M.Eng, Ph.D. (Penguji Dalam KK)
2. Harry Mahardika, S.Si., M.Si., Ph.D. (Penguji Luar KK)

## KATA PENGANTAR

Alhamdulillah, segala puji bagi Allah SWT atas rahmat-Nya sehingga penulisan Tugas Akhir yang berjudul “Sistem Komunikasi dan Monitoring Purwarupa Trash Skimmer” dapat selesai sebagai salah satu persyaratan kelulusan Progran Sarjana (S1) Prodi Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam (FMIPA), Institut Teknologi Bandung.

Selama penyusunan tugas akhir ini terdapat beberapa hambatan yang dihadapi mulai dari kurangnya wawasan dan kemampuan, waktu pengiriman alat yang cukup lama karena dipesan secara online dari luar negeri, sampai perangkat yang kurang menunjang. Meskipun begitu, berkat bimbingan, bantuan, nasehat dan dukungan dari berbagai pihak tugas akhir ini dapat diselesaikan dengan baik. Oleh karena itu, disampaikan terima kasih kepada pihak-pihak berikut.

1. Kedua orang tua yang telah melahirkan dan membesarkan penulis serta selalu memberikan bantuan secara lahir dan batin dan selalu berdoa untuk keselamatan dan kesuksesan penulis.
2. Prof. Dr.Ing Mitra Djamal, selaku dosen pembimbing satu yang telah membantu pelaksanaan tugas akhir agar berjalan dengan lancar.
3. Dr. Nina Siti Aminah, S.Si., M.Si., selaku dosen pembimbing kedua yang telah memberikan arahan dan dukungan dari mulai menjadi dosen pembimbing mata kuliah studi mandiri hingga tugas akhir.
4. Beasiswa ADik (Afirmasi Dikti) 3T yang telah banyak memberikan fasilitas dan dukungan kepada penulis sehingga bisa fokus dalam menjalani perkuliahan dan membantu meningkatkan *soft skill* melalui kegiatan-kegiatan yang diadakan.
5. Dosen serta staf di lingkungan FMIPA khususnya Prodi Fisika yang telah memberikan ilmu selama perkuliahan kepada penulis hingga dapat menyelesaikan tugas akhir ini dengan baik.
6. Syifa, Wulan, dan Aflaha sebagai rekan satu proyek tugas akhir, terima kasih atas bantuan dan kerja sama yang diberikan selama penyelesaian proyek tugas akhir.

7. Keluarga Muncak (Dara, Mujahid, Fakhrol, Rifqa, Nita, Isya dan Hapis), keluarga ADik 3T, keluarga TPB (Bakti, Salma dan Yolanda), keluarga Ekfis (Deni dan Darul), dan keluarga Viel Gluck(Aldha dan Armalinda) yang telah memberikan dukungan serta kebersamaan kepada penulis selama kuliah di ITB.
8. Teman-teman Fisika angkatan 2018 Widyandra Gamayasa, serta teman-teman Forum Mahasiswa Afirmasi (FMA) yang telah berbagi pengalaman selama di ITB.

Penelitian ini diharapkan dapat berguna bagi kemajuan teknologi khususnya dibidang Fisika Instrumentasi di Indonesia dan membantu mengatasi permasalahan sampah di permukaan perairan. Tugas akhir ini masih memiliki banyak kekurangan, maka dari itu diharapkan kritik dan saran yang membangun dari berbagai pihak agar penelitian ini dapat disempurnakan di masa mendatang.

Bandung, 25 Juli 2022

Rifqi Nafis Mubaroq  
NIM 10218091

## **PEDOMAN PENGGUNAAN BUKU TUGAS AKHIR**

Buku Tugas Akhir Sarjana ini tidak dipublikasikan, namun terdaftar dan tersedia di Perpustakaan Institut Teknologi Bandung. Buku ini dapat diakses umum, dengan ketentuan bahwa penulis memiliki hak cipta dengan mengikuti aturan HaKI yang berlaku di Institut Teknologi Bandung. Referensi kepustakaan diperkenankan dicatat, tetapi pengutipan atau peringkasan hanya dapat dilakukan seizin penulis, dan harus disertai dengan kebiasaan ilmiah untuk menyebutkan sumbernya.

Memperbanyak atau menerbitkan sebagian atau seluruh buku Tugas Akhir harus atas izin Program Studi Sarjana Fisika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi Bandung.

*Dipersembahkan kepada Papah, Mamah, Teteh, keluarga  
besar, sahabat tercinta serta semua orang beriman.*



# DAFTAR ISI

<b>ABSTRAK .....</b>	<b>ii</b>
<b>ABSTRACT .....</b>	<b>iii</b>
<b>KATA PENGANTAR.....</b>	<b>v</b>
<b>PEDOMAN PENGGUNAAN BUKU TUGAS AKHIR.....</b>	<b>vii</b>
<b>DAFTAR ISI.....</b>	<b>ix</b>
<b>DAFTAR TABEL .....</b>	<b>xi</b>
<b>DAFTAR GAMBAR DAN ILUSTRASI.....</b>	<b>xii</b>
<b>DAFTAR SINGKATAN.....</b>	<b>xiii</b>
<b>BAB I    PENDAHULUAN .....</b>	<b>1</b>
I.1    Latar Belakang.....	1
I.2    Rumusan Masalah.....	2
I.3    Tujuan .....	2
I.4    Ruang Lingkup Penelitian .....	2
I.5    Metodologi Penelitian.....	3
I.6    Sistematika Penulisan .....	4
<b>BAB II   TINJAUAN PUSTAKA .....</b>	<b>5</b>
II.1   Aplikasi Hukum Archimedes pada Benda.....	5
II.2   Sistem Penggerak <i>Trash Skimmer</i> .....	5
II.2.1   Motor DC <i>Brushless</i> .....	5
II.2.2 <i>Electronic Speed Controller</i> (ESC) .....	6
II.2.3   Motor DC <i>Gearbox</i> .....	7
II.2.4   Modul <i>Driver</i> L298N.....	7
II.2.5   Modul <i>Joystick</i> .....	8
II.3   Sistem Komunikasi <i>Trash Skimmer</i> .....	8
II.3.1   Arduino Uno .....	8
II.3.2   Modul Radio Frekuensi .....	9
II.4   Sistem Monitoring <i>Trash Skimmer</i> .....	9
II.4.1   NodeMCU .....	9
II.4.2   Modul GPS NEO-6M .....	10
II.4.3   Sensor Level Air .....	11
II.4.4   Sensor Ultrasonik Modul HC-SR04 .....	11

II.4.5	Sensor Tegangan DC .....	12
II.4.6	<i>Received Signal Strenght Indicator</i> (RSSI) .....	12
II.5	Sistem Internet of Things (IoT) .....	13
II.5.1	<i>Message Query Transport Telemetry</i> (MQTT) .....	13
II.5.2	InfluxDB .....	13
II.5.3	Node-RED .....	14
II.5.4	Grafana .....	14
<b>BAB III</b>	<b>METODOLOGI PENELITIAN.....</b>	<b>15</b>
III.1	Perancangan Sistem Komunikasi <i>Trash Skimmer</i> .....	15
III.2	Perancangan Sistem Penggerak <i>Trash Skimmer</i> .....	16
III.2.1	Sistem Penggerak- <i>Transmitter</i> .....	16
III.2.2	Sistem Penggerak- <i>Receiver</i> .....	16
III.3	Perancangan Sistem Monitoring <i>Trash Skimmer</i> .....	18
<b>BAB IV</b>	<b>HASIL DAN PEMBAHASAN.....</b>	<b>21</b>
IV.1	Data Ukuran Prototipe <i>Trash Skimmer</i> .....	21
IV.2	Data Kualitas Komunikasi Modul RF 433 MHz .....	22
IV.3	Sistem <i>Monitoring Trash Skimmer</i> .....	23
IV.3.1	Data Kualitas GPS NEO-6M dan Sistem Jangkauan <i>Monitoring Trash Skimmer</i> .....	24
IV.3.2	Data Sistem Peringatan Kapasitas Sampah .....	27
IV.3.3	Data Sistem Pemantauan Daya <i>Trash Skimmer</i> .....	29
<b>BAB V</b>	<b>KESIMPULAN .....</b>	<b>32</b>
V.1	Kesimpulan .....	32
V.2	Saran .....	33
<b>LAMPIRAN A</b>	<b>: Data Sensor Sistem <i>Monitoring Trash Skimmer</i>.....</b>	<b>36</b>
A.1	Data Nilai RSSI Terhadap Perubahan Jarak .....	36
<b>LAMPIRAN B</b>	<b>: Program Sistem Komunikasi &amp; <i>Monitoring</i>.....</b>	<b>37</b>
B.1	Program Sistem Komunikasi Penggerak- <i>Transmitter</i> .....	37
B.2	Program Sistem Komunikasi Penggerak- <i>Receiver</i> .....	38
B.3	Program Sistem <i>Monitoring Trash Skimmer</i> .....	40

## DAFTAR TABEL

Tabel IV.1 Data keberhasilan komunikasi Modul RF 433 MHz .....	22
Tabel IV.2 Data Kualitas Komunikasi Modul RF 433 MHz .....	23
Tabel IV.3 Data Sistem Peringatan Kapasitas Sampah .....	28
Tabel IV.4 Data Sistem Pemantauan Daya .....	30
Tabel IV.5 Data Nilai RSSI NodeMCU.....	26
Tabel IV.6 Data Sistem Peringatan Jangkauan Monitoring Trash Skimmer .....	27

## DAFTAR GAMBAR DAN ILUSTRASI

Gambar II.1 Motor Brushless KV500 .....	6
Gambar II.2 ESC 60A <i>Waterproof</i> .....	6
Gambar II.3 Motor DC <i>Gearbox</i> 12V .....	7
Gambar II.4 Modul <i>Driver</i> L298N .....	7
Gambar II.5 Modul <i>Joystick</i> .....	8
Gambar II.6 Arduino Uno R3 .....	9
Gambar II.7 Modul RF 433 MHz .....	9
Gambar II.8 NodeMCU ESP8266.....	10
Gambar II.9 Modul GPS NEO-6M .....	10
Gambar II.10 Sensor level air .....	11
Gambar II.11 Sensor ultrasonik modul HC-SR04 .....	12
Gambar II.12 Modul sensor tegangan.....	12
Gambar III.1 Skema sistem komunikasi trash skimmer. ....	15
Gambar III.2 Rangkaian sistem penggerak-transmitter. ....	16
Gambar III.3 Rangkaian sistem penggerak-receiver.....	17
Gambar III.4 Rangkaian sistem <i>monitoring trash skimmer</i> .....	18
Gambar III.5. <i>Flowchart</i> sistem pengiriman data <i>monitoring trash skimmer</i> . ....	19
Gambar III.6. <i>Flowchart</i> sistem <i>monitoring trash skimmer</i> .....	19
Gambar IV.1 Foto trash skimmer.....	21
Gambar IV.2 Sistem <i>monitoring trash skimmer</i> pada program Grafana. ....	24
Gambar IV.3 Konfigurasi <i>node</i> dan program pemisahan data ( <i>parse data</i> ) pada <i>node function</i> sistem jangkauan <i>monitoring trash skimmer</i> . ....	25
Gambar IV.4 Data <i>track location</i> .....	25
Gambar IV.5 Sistem Jangkauan <i>monitoring</i> pada program Grafana .....	27
Gambar IV.6 Konfigurasi <i>node</i> dan program pemisahan data ( <i>parse data</i> ) pada <i>node function</i> sistem peringatan kapasitas sampah. ....	28
Gambar IV.7 Sistem peringatan kapasitas sampah .....	29
Gambar IV.8 Konfigurasi <i>node</i> dan program pemisahan data ( <i>parse data</i> ) pada <i>node function</i> sistem pemantauan daya <i>trash skimmer</i> .....	30
Gambar IV.9 Sistem pemantauan daya trash skimmer pada Grafana.....	31

## DAFTAR SINGKATAN

Singkatan	Arti
A	<i>Ampere</i>
API	<i>Application Programming Interface</i>
BLDC	<i>Brushless DC</i>
dBm	<i>Decibel-milliwatts</i>
DC	<i>Direct Curent</i>
GPS	<i>Global Positioning System</i>
IC	<i>Integrated Ciruit</i>
IDE	<i>Integrated Development Environtment</i>
IoT	<i>Internet of Things</i>
LED	<i>Light-Emiting Diode</i>
LiPo	<i>Lithium-Polymer</i>
RC	<i>Radio-controlled</i>
RF	<i>Radio Frekuensi</i>
RSSI	<i>Received Signal Strengty Indication</i>
TTL	<i>Transistor-transistor logic</i>
V	<i>Volt</i>
WiFi	<i>Wireless Fidelity</i>

# BAB I PENDAHULUAN

## I.1 Latar Belakang

Sampah plastik di perairan merupakan salah satu masalah yang dialami oleh setiap negara di dunia ini. Menurut data Sistem Informasi Pengelolaan Sampah Nasional (SIPSN) negara Indonesia menghasilkan timbulan sampah sebesar 28,69 juta ton/tahun dan 15,6% diantaranya merupakan sampah plastik (SIPSN 2021).

Jawa Barat sebagai provinsi dengan penduduk terbanyak di Indonesia menjadi penyumbang sampah terbanyak akibat banyaknya aktivitas manusia. Sistem pengelolaan sampah yang masih belum maksimal mengakibatkan banyaknya penduduk yang membuang sampah ke tempat yang tidak seharusnya. Salah satu tempat tersebut merupakan Sungai Citarum. Sungai Citarum sering masuk kedalam daftar sungai paling tercemar di dunia baik diberitakan oleh media nasional maupun internasional. Menurut *website* Citarum Harum pada tahun 2021 sampah yang dikumpulkan dari Sungai Citarum mencapai 15.838 ton/hari dengan persentase sampah plastik sebesar 15,35%. (KLHK 2021)

Pada tahun 2019 prototipe mengenai teknologi untuk membersihkan sampah yang ada di perairan telah pernah dibuat sebelumnya di ITB oleh Khayima Arnisti pada penelitian tugas akhir yang berjudul “Pengaruh Perubahan Beban terhadap Gaya Apung Robot Pembersih Sampah”. Prototipe tersebut mengaplikasikan konsep dasar dari hukum Archimides mengenai konsep terapung, namun memiliki beberapa kendala dan perbaikan untuk meningkatkan performa dari prototipe alat tersebut. Kendala tersebut diantaranya yakni motor propeler yang tidak dapat berfungsi dengan baik sehingga *trash skimmer* tidak dapat bergerak di atas permukaan air. Selanjutnya sistem konveyor yang belum dapat berfungsi dengan baik sehingga tidak dapat melakukan sistem pengangkutan sampah. Selain perbaikan tersebut, terdapat beberapa ide dan gagasan yang dirasa dapat meningkatkan kualitas *trash skimmer* tersebut. Ide tersebut yaitu penambahan sistem pemantauan daya baterai yang digunakan *trash skimmer*, sistem peringatan kapasitas *trash skimmer*, dan juga sistem *monitoring* lokasi terkini. Sistem-sistem ini dapat dipantau secara *realtime* dan berbasis IoT sehingga dapat diakses dimana saja selama ada jaringan internet.

Oleh karena itu penulis ingin melanjutkan pembuatan *trash skimmer* sehingga dapat digunakan dengan baik serta membangun sistem *monitoring* berbasis IoT.

## **I.2 Rumusan Masalah**

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, terdapat beberapa rumusan masalah sebagai berikut:

1. Bagaimana merancang sistem gerak *trash skimmer* agar bisa berfungsi dengan baik?
2. Bagaimana merancang sistem komunikasi *trash skimmer* menggunakan modul radio frekuensi *transmitter-receiver*?
3. Bagaimana merancang sistem *monitoring trash skimmer* berbasis IoT?

## **I.3 Tujuan**

Tujuan penelitian pada tugas akhir ini adalah sebagai berikut:

1. Merancang sistem gerak *trash skimmer* agar bisa berfungsi dengan baik.
2. Merancang sistem komunikasi *trash skimmer* menggunakan modul radio frekuensi *transmitter-receiver*.
3. Merancang sistem *monitoring trash skimmer* berbasis IoT.

## **I.4 Ruang Lingkup Penelitian**

Pada tugas akhir ini dibatasi dengan ruang lingkup penelitian berupa aspek-aspek berikut ini.

1. Pengujian prototipe dilakukan pada kondisi air yang tenang dengan target sampah yang diambil merupakan sampah yang terletak pada permukaan air.
2. Data yang akan ditampilkan pada sistem *monitoring* adalah data persentase daya baterai, kapasitas penampungan sampah berupa data level air (keterapungan robot), data lokasi terkini menggunakan modul GPS, dan data jangkauan komunikasi sistem *monitoring* menggunakan RSSI NodeMCU.

## I.5 Metodologi Penelitian

Metode yang digunakan pada penulisan tugas akhir ini adalah sebagai berikut:

### 1. Tinjauan Pustaka

Tinjauan pustaka dilakukan pada perangkat dan sistem yang digunakan untuk membuat *trash skimmer* melalui studi terhadap komponen yang digunakan melalui *datasheet*, jurnal, buku, dan halaman web yang memuat metode perancangan sistem gerak benda di permukaan air. Selanjutnya melakukan evaluasi terhadap prototipe sebelumnya.

### 2. Analisis dan Perancangan

Menganalisis spesifikasi daya motor yang dibutuhkan agar *trash skimmer* dapat bergerak dengan baik. Selanjutnya dirancang sistem gerak *trash skimmer* yang terdiri dari sistem komunikasi *trash skimmer* menggunakan modul radio frekuensi *transmitter-receiver*, sistem penggerak *trash skimmer* yang terdiri dari rangkaian *joystick*, rangkaian motor propeler, dan rangkaian konveyor. Serta membangun sistem *monitoring trash skimmer* berbasis IoT.

### 3. Eksperimen

Pada tahap ini dilakukan perancangan sistem gerak *trash skimmer*. melakukan proses pembuatan rangkaian sistem gerak *trash skimmer*, mekanik *trash skimmer*, dan perangkat lunak *trash skimmer*. Selanjutnya eksperimen dilakukan dengan menguji program *sketch* “Arduino” untuk mengirim perintah oleh *transmitter* menuju *receiver* pada modul RF. Menguji perintah deteksi pada sistem sensor level air, menguji perintah gerak pada sistem rangkaian konveyor dan rangkaian propeler, menguji sensor tegangan untuk menghitung daya, menguji sistem GPS untuk *tracking trash skimmer*, dan pembuatan *dashboard* API Grafana sebagai sistem monitoring dan analisis.

### 4. Analisis Hasil

Dilakukan analisis data yang diperoleh dari eksperimen pengujian alat. Proses analisis hasil dilakukan untuk menjawab permasalahan pada *trash skimmer*.



## 5. Evaluasi

Evaluasi dilakukan sebelum menyelesaikan *trash skimmer* agar dihasilkan prototipe yang sesuai dengan rencana dan minim kesalahan. Evaluasi juga dilakukan agar kekurangan pada penelitian ini dapat digunakan sebagai acuan untuk pengembangan pada penelitian selanjutnya.

## I.6 Sistematika Penulisan

Sistematika penulisan pada penelitian ini dibagi dalam lima bab dengan rincian sebagai berikut:

### 1. BAB I - Pendahuluan

Bab ini membahas latar belakang, perumusan masalah, tujuan penelitian tugas akhir, batasan masalah, metode tugas akhir, dan sistematika penulisan.

### 2. BAB II - Teori Dasar

Bab ini membahas tentang berbagai pustaka terkait komponen yang digunakan pada penelitian.

### 3. BAB III – Perancangan Sistem dan Metodologi Penelitian

Pada bab ini membahas tentang sistem komunikasi *trash skimmer*, sistem penggerak *trash skimmer*, dan sistem *monitoring trash skimmer*.

### 4. BAB IV - Hasil dan Pembahasan

Bab ini menampilkan hasil uji coba prototipe *trash skimmer* dan data yang diperoleh dari hasil pengujian dalam bentuk tabel, gambar, dan pembahasan dari data yang diperoleh.

### 5. BAB V - Kesimpulan dan Saran

Bab ini berisi kesimpulan yang diperoleh dari hasil analisis penelitian dan saran untuk pengembangan penelitian selanjutnya.

## BAB II TINJAUAN PUSTAKA

### II.1 Aplikasi Hukum Archimedes pada Benda

Hukum Archimedes menjelaskan hubungan antara gaya berat dan gaya apung pada suatu benda jika dimasukkan ke dalam fluida. Gaya apung terjadi karena tekanan pada fluida di sekitar benda semakin besar jika kedalamannya meningkat. Sehingga tekanan pada bagian bawah benda lebih besar dari tekanan bagian atas dan menyebabkan benda mengalami gaya angkat. Gaya angkat ini disebut dengan gaya apung (Halliday, 2010). Persamaan gaya apung ditunjukkan pada persamaan II.1 berikut.

$$F_b = -\rho g V \quad (\text{II. 1})$$

Dengan  $F_b$  adalah gaya apung,  $\rho$  adalah massa jenis fluida,  $g$  adalah percepatan gravitasi, dan  $V$  adalah volume fluida.

Hukum Archimedes diaplikasikan terhadap *trash skimmer* agar dapat tetap bekerja di permukaan air dan tidak tenggelam. Perubahan ketinggian air pada *trash skimmer* yang mengapung di permukaan air juga menjadi suatu parameter untuk menghitung sistem peringatan kapasitas sampah yang dibuat.

### II.2 Sistem Penggerak *Trash Skimmer*

#### II.2.1 Motor DC *Brushless*

Motor DC *brushless* memiliki magnet permanen pada bagian rotor dan memiliki elektromagnet pada bagian stator. Penggunaan Motor DC *brushless* (BLDC) memiliki beberapa keuntungan yakni biaya perawatan mesin yang rendah, kecepatan rotasi motor yang tinggi, dan konfigurasi dengan perangkat lain yang sederhana. Motor BLDC juga telah menggantikan mesin induksi dengan magnet permanen yang memiliki energi tinggi, sehingga secara signifikan mengurangi berat total magnet (Mariano, 2017). Bentuk Motor DC *brushless* dapat dilihat pada Gambar II.1 berikut.



Gambar II.1 Motor Brushless KV500.

BLDC terdiri dari *controller*, *driver* dan *inverter*, *stator*, *rotor*, *axle*, dan sensor Hall. Stator dan rotor akan menghasilkan medan magnetik sehingga akan berputar dengan frekuensi yang sama (Yedamale, 2003).

### II.2.2 *Electronic Speed Controller (ESC)*

*Electronic Speed Controller (ESC)* merupakan komponen elektronik yang berfungsi sebagai pengatur kecepatan rotasi motor RC. ESC bekerja dengan mengubah daya yang berasal dari baterai menjadi sinyal berurutan yang diteruskan pada tiga keluaran kabel pada motor *brushless*. Urutan sinyal tersebut mengendalikan laju putaran pada motor. Kecepatan yang diinginkan pada setiap motor dikirim oleh pengendali melalui tiga sinyal kabel ESC (Kurnia Rahman, Supriyanto, and Meizinta 2019). Bentuk ESC dapat dilihat pada Gambar II.2.

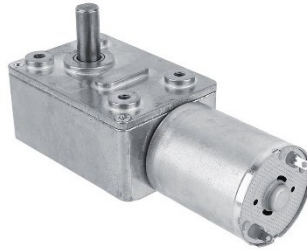


Gambar II.2 ESC 60A *Waterproof*.

ESC perlu memiliki kapasitas arus yang lebih besar dibandingkan motor yang digunakan. Jika ESC memiliki kapasitas arus yang lebih kecil dibandingkan motor, maka terdapat kemungkinan ESC mengalami kenaikan suhu dan terbakar akibat permintaan arus yang terlalu besar dari motor (Hibban 2015). ESC yang digunakan untuk *trash skimmer* adalah *ESC Program Card Brushless 60A*.

### II.2.3 Motor DC *Gearbox*

Motor DC merupakan komponen elektronik yang dapat mengubah energi listrik menjadi energi mekanik. Motor DC yang digunakan untuk *trash skimmer* bagian konveyor adalah motor DC *gearbox* 12V. Motor torsi tinggi berbentuk toroid dan memiliki diameter besar serta sumbu pendek. Bentuk motor DC *gearbox* dapat dilihat pada Gambar II.3.

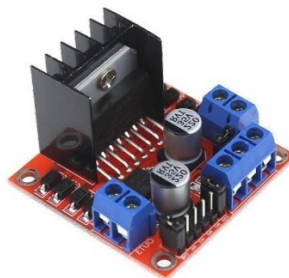


Gambar II.3 Motor DC *Gearbox* 12V.

Prinsip kerja motor DC berdasarkan pada penghantar yang membawa arus ditempatkan dalam suatu medan magnet. Penghantar akan mengalami gaya dan menimbulkan torsi yang akan menghasilkan rotasi mekanik sehingga motor akan berputar (Rahayuningtyas 2009).

### II.2.4 Modul *Driver* L298N

Modul *driver* L298N merupakan komponen elektronik yang berfungsi sebagai pengatur arah rotasi motor DC *gearbox*. Menurut lembar data L298, tegangan operasinya adalah 5V hingga 46V, dan arus maksimum yang diizinkan sebesar 3A. Modul L298N memiliki dua input yang disediakan untuk mengaktifkan atau menonaktifkan perangkat secara independen dari sinyal input. Bentuk Modul *Driver* L298N dapat dilihat pada Gambar II.4.

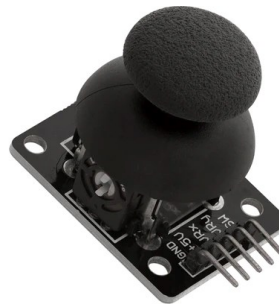


Gambar II.4 Modul *Driver* L298N.

Driver motor dua arah ini didasarkan pada IC L298. Modul L298N mampu mengontrol dua motor sekaligus secara masing-masing hingga 2A dengan 2 arah. Modul ini sangat ideal untuk aplikasi robot dan sangat cocok untuk koneksi ke mikrokontroler yang hanya membutuhkan beberapa jalur kontrol per motor. Modul ini juga dapat dihubungkan dengan sakelar manual sederhana, gerbang logika TTL, relai, dan lain-lain (Rittenberry 2005).

### II.2.5 Modul *Joystick*

Modul *joystick* merupakan komponen elektronik yang berfungsi untuk mengatur arah gerak perangkat lain. Modul ini sama seperti *joystick* analog pada *gamepad*. Modul ini memiliki 2 buah potensiometer dengan sudut 90°. Potensio terhubung dengan stik pendek yang berpusat pada pegas. Jika modul *joystick* dihubungkan dengan Arduino Uno maka nilainya akan terbaca sebesar 512 pada posisi di tengah atau netral., lalu jika digerakkan maka nilainya akan berubah antara 0-1023 (Agustina 2017). Bentuk Modul *Joystick* ditunjukkan oleh Gambar II.5.



Gambar II.5 Modul *Joystick*.

## II.3 Sistem Komunikasi *Trash Skimmer*

### II.3.1 Arduino Uno

Arduino adalah perangkat elektronik yang bersifat *opensource* berdasarkan perangkat keras dan perangkat lunak yang mudah digunakan. Forum Arduino dapat mengamati input sensor, *joystick*, atau pesan Twitter dan mengubahnya menjadi output yang mengaktifkan motor, menyalakan LED, mengirimkan sesuatu secara online. Arduino Uno menggunakan bahasa pemrograman Arduino dan perangkat lunak Arduino (IDE), yang sepenuhnya didasarkan pada Pemrosesan (Ingle et al. 2020). Bentuk perangkat Arduino Uno dapat dilihat pada Gambar II.6.



Gambar II.6 Arduino Uno R3.

### II.3.2 Modul Radio Frekuensi

Modul radio frekuensi (modul RF) merupakan komponen elektronik yang berfungsi untuk mengirim atau menerima pesan antara dua perangkat melalui sinyal radio. Modul RF dapat digunakan untuk pengendalian jarak jauh tanpa menggunakan kabel (*wireless*). Modul RF berfungsi untuk kirim-terima data antara *transmitter* dan *receiver* Pada *trash skimmer* sehingga bisa mengendalikan gerak propeler dan konveyor. Jenis modul RF yang digunakan pada penelitian ini adalah 433 MHz. Berikut bentuk Modul RF 433 MHz ditunjukkan oleh Gambar II.7.

RF433Mhz Module  
Transmitter & Receiver



Gambar II.7 Modul RF 433 MHz.

## II.4 Sistem Monitoring *Trash Skimmer*

### II.4.1 NodeMCU

NodeMCU merupakan salah satu mikrokontroler yang bersifat *arduino compatible development board* dan dirancang khusus untuk keperluan IoT. NodeMCU menggunakan cip WiFi yang cukup terkenal saat ini yaitu ESP8266. Cukup banyak

modul WiFi yang menggunakan SoC ESP8266 (Mulyono, Qomaruddin, and Syaiful Anwar 2018). NodeMCU berfungsi untuk menerima dan mengirimkan data sensor yang diperoleh dari *trash skimmer* menuju *database MQTT mosquitto*. Bentuk NodeMCU ESP8266 ditunjukkan oleh Gambar II.8.



Gambar II.8 NodeMCU ESP8266.

#### II.4.2 Modul GPS NEO-6M

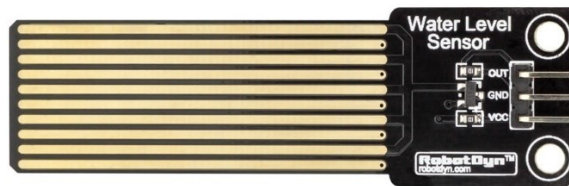
Modul GPS NEO-6M merupakan komponen elektronik yang berfungsi untuk melacak posisi secara langsung di permukaan bumi relatif terhadap satelit. Modul ini dapat melacak hingga 22 satelit pada 50 saluran dan hanya menggunakan arus suplai 45 mA. Terdapat LED pada Modul GPS NEO-6M yang menunjukkan status *Position Fix*. LED akan berkedip setiap satu detik jika GPS telah menemukan cukup banyak satelit atau dapat menentukan posisi secara akurat sedangkan LED tidak berkedip meunjukkan bahwa GPS sedang mencari satelit atau belum dapat menentukan posisi secara akurat. Rentang tegangan operasi Modul NEO-6M adalah 2,7 hingga 3,6 V. Modul ini dilengkapi dengan *regulator 3V3 dropout* ultra-rendah MIC5205 dari MICREL. Pin logika juga toleran 5 volt, sehingga dapat dengan mudah menghubungkannya ke Arduino atau mikrokontroler logika 5V tanpa menggunakan konverter level logika (Suryana 2021). Bentuk Modul GPS NEO-6M dapat dilihat pada Gambar II.9 sebagai berikut.



Gambar II.9 Modul GPS NEO-6M.

### II.4.3 Sensor Level Air

Sensor level air merupakan komponen elektronik yang berfungsi untuk mengetahui ketinggian air pada jarak tertentu. Sensor ini memanfaatkan sifat konduktivitas bahan di dalam air. Sensor level air terdiri atas konduktor yang disusun secara paralel yang bekerja dengan prinsip resistor variabel. Nilai resistansi pada sensor level air akan menurun jika semakin banyak permukaan sensor yang terkena oleh air, begitu pula sebaliknya. Hal ini karena sifat konduktor yang jika terkena air maka konduktivitas akan meningkat sehingga resistansi akan turun. Data yang terbaca pada mikrokontroler merupakan tegangan keluaran sesuai dengan nilai resistansi, sehingga bisa digunakan untuk menentukan ketinggian air. Bentuk Sensor level air dapat dilihat pada Gambar II.10 berikut.



Gambar II.10 Sensor level air.

### II.4.4 Sensor Ultrasonik Modul HC-SR04

Sensor ultrasonik HR-SC04 merupakan sensor yang dapat berfungsi sebagai pengukur jarak benda. sensor ini memiliki kemampuan dalam mengukur objek non-kontak. Sensor ini mampu mengukur objek non-kontak sampai 4 m dengan akurasi jangkauan hingga 3 mm. Prinsip dasar kerja sensor ini menggunakan pemicu IO untuk 10us sinyal tingkat tinggi sehingga secara otomatis mengirimkan delapan 40kHz dan akan mendeteksi jarak benda jika ada sinyal pulsa kembali. jika sinyal kembali, waktu durasi IO adalah waktu dari mengirim ultrasonik hingga kembali. Dari waktu tempuh tersebut maka kita dapat menentukan jarak antara benda dan sensor (Alief, 2017). Bentuk Modul HR-SC04 dapat dilihat pada Gambar II.11 sebagai berikut.





Gambar II.11 Sensor ultrasonik modul HC-SR04.

#### II.4.5 Sensor Tegangan DC

Sensor tegangan DC merupakan komponen elektronik yang berfungsi untuk mendeteksi dan mengukur tegangan DC suatu alat atau rangkaian. Modul sensor tegangan DC menggunakan pembagi potensial yang mengurangi tegangan input dengan faktor 5. Modul Sensor Tegangan mampu mengukur tegangan sampai 25V sehingga dapat membantu mikrokontroler untuk memantau tegangan yang jauh lebih tinggi dari tegangan maksimal mikrokontroler seperti Arduino Uno. Bentuk Modul sensor tegangan ditunjukkan oleh Gambar II.12.



Gambar II.12 Modul sensor tegangan.

#### II.4.6 *Received Signal Strength Indicator (RSSI)*

RSSI merupakan teknologi yang digunakan untuk mengukur indikator kekuatan sinyal yang diterima oleh sebuah perangkat wireless. Nilai RSSI dapat dihasilkan dari perangkat-perangkat yang memiliki kemampuan menerima sinyal atau gelombang elektromagnetik. Salah satu perangkat penerima nilai RSSI dan pengiriman ke MQTT yang banyak digunakan adalah NodeMCU (Esp8266) sedangkan Salah satu perangkat yang memiliki kemampuan mentransmisikan sinyal WiFi adalah gawai atau router (Djamal, 2020).

## II.5 Sistem Internet of Things (IoT)

### II.5.1 *Message Query Transport Telemetry* (MQTT)

Protokol adalah peraturan atau prosedur untuk mengirimkan data pada perangkat elektronik. setiap komputer berkomunikasi dengan komputer lainnya menggunakan sebuah protokol. Ada banyak jenis protokol, beberapa yang populer adalah *User Datagram Protocol* (UDP), *File Transfer Protocol* (FTP), *Transmission Control Protocol* (TCP). *Internet Control Message Protocol* (ICMP) dan salah satu yang sering dipakai untuk komunikasi *Internet of Things* (IoT) adalah MQTT. MQTT merupakan salah satu protokol yang dapat digunakan untuk komunikasi pada sistem komputer.

Sistem kerja MQTT menggunakan *Publish* dan *Subscribe* data. Perangkat akan terhubung pada sebuah *broker* (server) dan mempunyai suatu *Topic* tertentu. *Broker* pada MQTT berfungsi untuk mengelola data *publish* dan *subscribe* dari berbagai *device*, Beberapa contoh dari *broker* adalah Mosquitto, HiveMQ dan Mosca.

*Publish* merupakan cara suatu *device* untuk mengirimkan datanya ke *subscribers*. Biasanya suatu *publisher* terhubung dengan sensor tertentu untuk mendapatkan data. *Subscribe* merupakan cara suatu device untuk menerima berbagai data dari *publisher*. *Subscriber* dapat berupa aplikasi *monitoring* sensor dan sebagainya yang akan meminta data dari *publisher*. *Topic* merupakan serial unik untuk pengelompokan data disuatu kategori tertentu. Pada sistem kerja protokol MQTT, *topic* wajib dibuat agar dapat membedakan jalur data tertentu sesuai kebutuhan. Pada setiap transaksi data antara *Publisher* dan *Subscriber* harus memiliki suatu *topic* tertentu (Djamal, 2020).

### II.5.2 InfluxDB

InfluxDB adalah layanan *database open source* yang menyediakan platform time series untuk mengukur, mempelajari, dan melakukan otomatisasi sistem. InfluxDB dapat menangani jutaan data per detik dan disimpan dengan memadatkan data secara otomatis untuk meminimalkan ruang penyimpanan. Terdapat fitur permintaan secara terus menerus atau *continuous queries* (CQ) yang dapat membantu mengotomatiskan data terkini. Pada sistem ini, InfluxDB digunakan

untuk menyimpan data sensor pada sistem *monitoring trash skimmer* (Ichwanda, 2020).

### **II.5.3 Node-RED**

Node-RED adalah sebuah tool berbasis browser untuk membuat aplikasi *Internet of Things* (IoT) dengan lingkungan pemrograman secara visual sehingga mempermudah penggunaanya untuk membuat aplikasi sebagai “flow”. *Flow* ini terbentuk dari *node-node* yang saling berhubungan di mana tiap *node* melakukan tugas tertentu. Node-RED menyediakan berbagai jenis *node* seperti menampilkan *node* input dan *node* output seperti *publish* dan *subscribe* dari topik MQTT dari sebuah *broker* (Mulyono, Qomaruddin, and Syaiful Anwar 2018).

### **II.5.4 Grafana**

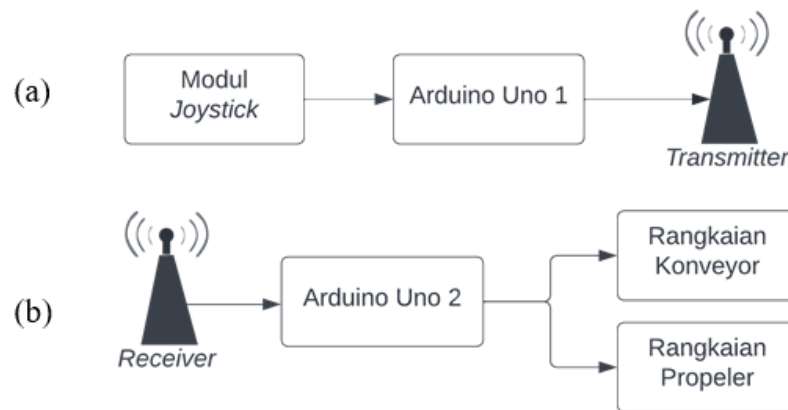
Grafana merupakan sebuah *software opensource* yang didesain untuk membaca data metrik dan mengubah data-data tersebut menjadi sebuah grafik, panel, atau sebuah data tertulis. *Software* ini banyak sekali digunakan untuk melakukan analisis data dan *monitoring*. Grafana juga dapat digunakan sebagai aplikasi *monitoring* berbasis IoT dengan *database* yang berasal dari InfluxDB yang sudah terintegrasi.

## BAB III METODOLOGI PENELITIAN

Pada penelitian ini metodologi penelitian dibagi menjadi tiga, yakni fokus membahas perancangan sistem komunikasi *trash skimmer*, perancangan sistem penggerak *trash skimmer*, dan perancangan sistem *monitoring trash skimmer* secara elektrik.

### III.1 Perancangan Sistem Komunikasi *Trash Skimmer*

Sistem komunikasi *trash skimmer* berperan dalam melakukan pengiriman sinyal dari *transmitter* menuju *receiver*. Sistem ini berfungsi menyambungkan sistem penggerak secara *wireless* dari rangkaian penggerak-*transmitter* ke rangkaian penggerak-*receiver*. Selanjutnya karakter perintah penggerak yang diterima oleh *receiver* digunakan sebagai perintah aksi untuk sistem penggerak *trash skimmer*. Skema sistem komunikasi *trash skimmer* dapat dilihat pada Gambar III.1.



Gambar III.1 Skema sistem komunikasi trash skimmer (a) bagian pengirim sinyal (transmitter), (b) bagian penerima sinyal (reciever).

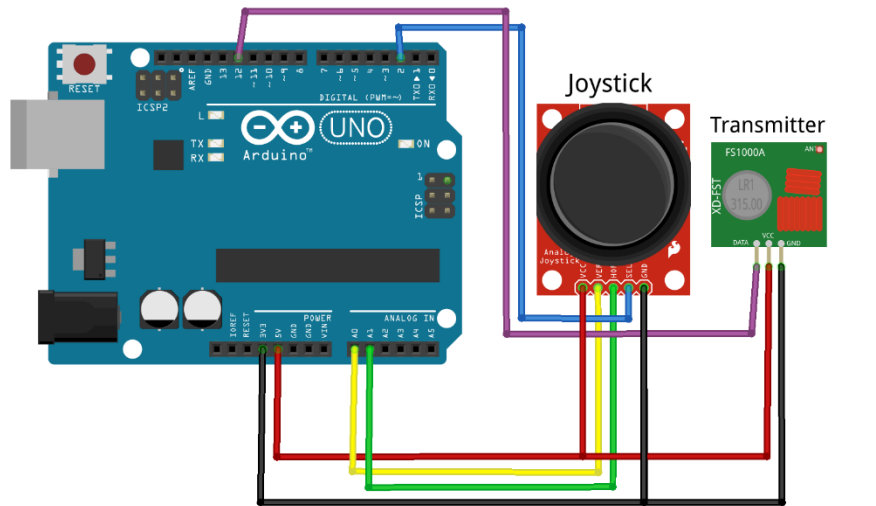
Gambar III.1 bagian (a) menunjukkan komponen sistem komunikasi *trash skimmer* bagian pengirim sinyal terdiri atas rangkaian *joystick* dan *transmitter* serta Gambar III.1 bagian (b) menunjukkan sistem komunikasi *trash skimmer* bagian penerima sinyal yang terhubung dengan sistem penggerak *trash skimmer* yang meliputi rangkaian propeler dan rangkaian konveyor.

## III.2 Perancangan Sistem Penggerak *Trash Skimmer*

Sistem penggerak *trash skimmer* dibagi menjadi dua bagian besar, yakni sistem penggerak pada *transmitter* yang terdiri dari rangkaian *joystick*, serta rangkaian sistem penggerak pada *receiver* yang terdiri dari rangkaian propeler dan rangkaian konveyor.

### III.2.1 Sistem Penggerak-Transmitter

Sistem ini berperan dalam mengontrol gerak propeler dan konveyor pada *trash skimmer*. Sistem Penggerak-Transmitter menggunakan modul *Joystick*. Input analog dari *joystick* diolah menjadi karakter, selanjutnya dikirimkan menuju *receiver* melalui *transmitter*. Berikut adalah gambar rangkaian sistem penggerak-transmitter:



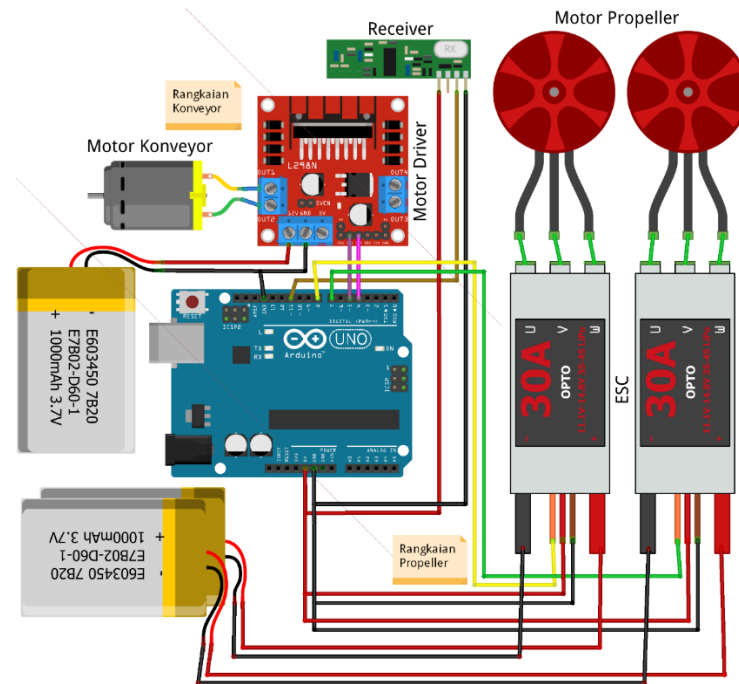
Gambar III.2 Rangkaian sistem penggerak-transmitter yang terdiri dari Arduino Uno, modul joystick, dan modul RF-Transmitter.

Berdasarkan Gambar III.2 *Joystick* mengirimkan karakter menuju Arduino. Karakter tersebut berupa perintah maju ke depan, maju ke kanan, maju ke kiri, serta menyalakan konveyor. Karakter yang diterima Arduino Uno akan diteruskan menuju *transmitter* dan selanjutnya akan dikirimkan secara nirkabel menuju rangkaian penggerak-*receiver*.

### III.2.2 Sistem Penggerak-Receiver

Sistem penggerak-*receiver trash skimmer* adalah sistem yang berfungsi untuk menjalankan badan *trash skimmer* di permukaan air (rangkain propeler) dan menggerakkan konveyor untuk pengangkutan sampah pada *trash skimmer*

(rangkaian konveyor). Rangkaian propeler dan konveyor terhubung pada sistem komunikasi *trash skimmer* (*receiver*). Sistem penggerak-*receiver* terdiri dari dua ESC 60A, dua motor *brushless* SSS 56114 KV500, dan dua propeler. Sumber daya yang digunakan untuk sistem penggerak-*receiver* berasal dari baterai LiPo.

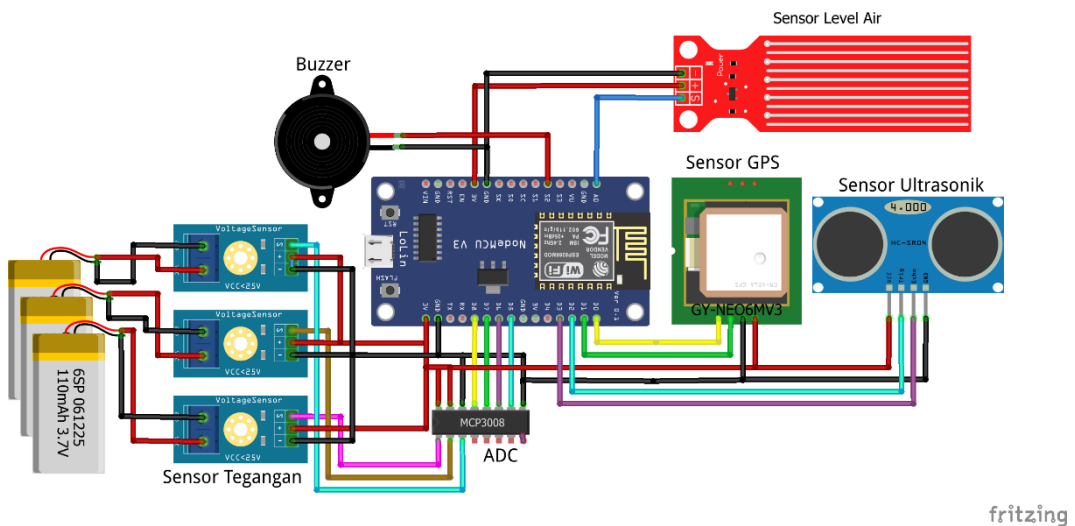


Gambar III.3 Rangkaian sistem penggerak-*receiver* yang terdiri dari Arduino Uno, Modul RF-Receiver, motor konveyor, motor propeler, ESC, dan Modul L298N.

Berdasarkan Gambar III.3 modul RF-*receiver* akan menerima pesan dari modul RF-*transmitter* yang telah dikirimkan. Perintah maju ke depan dari *transmitter* akan menjalankan motor propeler secara bersamaan sehingga *trash skimmer* akan berjalan ke depan. Perintah maju ke kanan akan menjalankan motor propeler bagian kiri sedangkan motor propeler bagian kanan akan mati sehingga *trash skimmer* akan berbelok ke kanan. Perintah maju ke kiri akan menjalankan motor propeler bagian kanan sedangkan motor propeler bagian kiri akan mati sehingga *trash skimmer* akan berbelok ke kiri. Perintah menyalakan konveyor akan menggerakkan motor konveyor sehingga dapat mengangkat sampah dari permukaan air menuju ke tempat penampungan sampah pada *trash skimmer*.

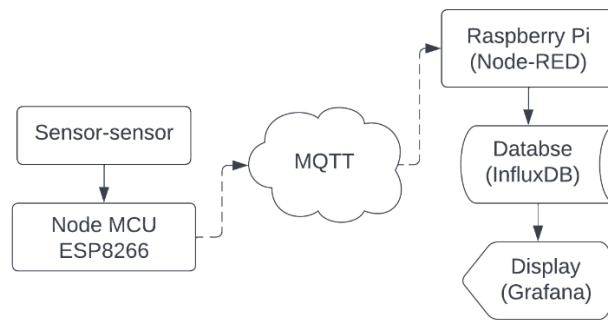
### III.3 Perancangan Sistem *Monitoring Trash Skimmer*

Sistem *monitoring trash skimmer* dibuat untuk mengukur parameter kondisi robot saat digunakan. Sistem ini terdiri dari sistem pemantauan daya *trash skimmer* saat digunakan, sistem peringatan kapasitas pengangkutan sampah, sistem *monitoring* posisi *trash skimmer*, serta sistem jangkauan *monitoring trash skimmer*. Komponen yang digunakan untuk sistem ini adalah mikrokontroler NodeMCU ESP8266, ADC modul MCP3008, sensor tegangan DC 0-25 V, sensor level air, sensor ultrasonik modul HC-SR04, Buzzer, sensor GPS Modul GPS NEO-6M. Sistem *monitoring* tersebut dirangkai sebagai berikut.



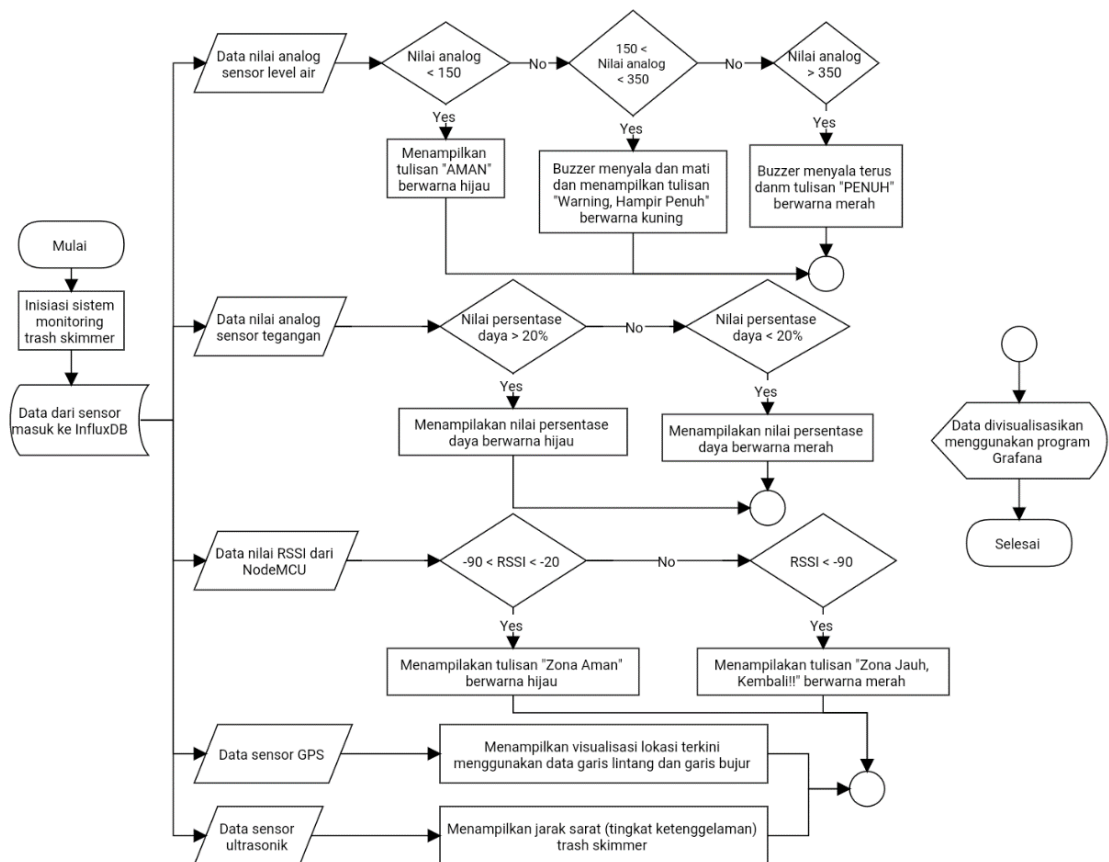
Gambar III.4 Rangkaian sistem *monitoring trash skimmer*.

Berdasarkan Gambar III.4 Sensor-sensor dipasang ke NodeMCU ESP8266. Pada rangkaian tersebut terdapat penambahan ADC MCP3008 sebagai penambahan pin analog karena pada penelitian ini sensor analog yang digunakan terdapat 4 buah sedangkan pin analog *default* dari NodeMCU hanyalah satu. Setelah itu rangkaian akan dinyalakan dengan menyambungkan NodeMCU ke sumber tegangan 5V. Selanjutnya proses pengiriman data sensor menuju *database* dan divisualisasikan menggunakan program Grafana. Proses pengiriman tersebut dilakukan sesuai dengan *flowchart* berikut.



Gambar III.5. Flowchart sistem pengiriman data *monitoring trash skimmer*.

Berdasarkan Gambar III.5 sensor akan mengirimkan data melalui NodeMCU lalu NodeMCU akan mengirimkan data tersebut menuju server menggunakan protokol MQTT dengan broker Mosquitto. Selanjutnya Raspberry Pi akan bertindak sebagai *subscriber* atau penerima data menggunakan program Node-RED. Pada program Node-RED data dikirim ke *database* InfluxDB. Selanjutnya sistem *monitoring* menggunakan program Grafana dengan databse dari InfluxDB dibuat.



Gambar III.6. Flowchart sistem *monitoring trash skimmer*.



Berdasarkan Gambar III.6 data yang diperoleh dari sensor akan masuk ke *database* 'trashskimmeritb' dengan nama *measurement* 'waterlevel', 'location', dan 'power'. Data yang ditampilkan berupa sarat *trash skimmer* (bagian yang belum terendam air), persentase daya baterai yang sedang dipakai, lokasi terkini *trash skimmer* saat dijalankan, dan data nilai RSSI NodeMCU sebagai parameter jangkauan sumber jaringan WiFi.. Selain ditampilkan melalui API *dashboard* Grafana, data yang diperoleh dari sensor juga digunakan sebagai *conditional statement* untuk menyalakan *buzzer* sebagai alarm agar pengguna bisa mengetahui kondisi *trash skimmer* di permukaan air.

## BAB IV HASIL DAN PEMBAHASAN

### IV.1 Data Ukuran Prototipe *Trash Skimmer*

*Trash skimmer* yang telah dibuat memiliki ukuran dan dimensi sebagai berikut:

Panjang	: 100 cm
Lebar	: 82 cm
Tinggi	: 60 cm
Sarat trash skimmer	: 13-17,2 cm
Massa	: 41,81 kg

Berikut adalah gambar prototipe dan dimensi *trash skimmer*:



(a)

(b)

Gambar IV.1 Foto trash skimmer (a) prototipe, (b) dimensi lebar, panjang dan tinggi.

Berdasarkan **Error! Reference source not found.** bagian (a) terlihat *trash skimmer* sedang dijalankan di kolam kelautan ITB. *Trash skimmer* dapat berjalan sesuai dengan perintah pada rangkaian penggerak-transmitter. Sistem rangkaian penggerak yang dibuat dapat bekerja dengan baik mulai dari rangkaian propeler yang dapat membelokkan *trash skimmer* sampai rangkaian konveyor yang dapat mengangkat sampah dari permukaan air menuju ke tempat penampungan sampah pada *trash skimmer*. Kecepatan *trash skimmer* yang didapatkan sebesar 0,21 m/s.

## IV.2 Data Kualitas Komunikasi Modul RF 433 MHz

Berdasarkan metode yang dilakukan pada penelitian ini, yaitu dengan menggunakan modul RF. Maka dilakukan pengambilan data keberhasilan komunikasi sistem *transmitter-receiver* pada *trash skimmer*. Data yang diambil berdasarkan program *delay* 1 detik dengan perlakuan beberapa kondisi. Jika data yang dikirimkan oleh *transmitter* diterima oleh *receiver*, maka percobaan tersebut diberi tanda "Berhasil" dan diberi tanda "Gagal" jika *receiver* tidak menerima data *transmitter* sama sekali. Data dapat dilihat pada Tabel IV.1.

Tabel IV.1 Data keberhasilan komunikasi *transmitter-receiver* Modul RF 433 MHz.

Kondisi	Batas Jarak yang ditentukan (m)	Keberhasilan komunikasi data
Dalam Ruangan	0	Berhasil
	10	
Luar Rungan Tanpa penghalang	0	
	10	
	20	
	30	
Luar rungan dengan penghalang pohon	0	
	10	
	20	
	30	

Berdasarkan Tabel IV.1 diketahui bahwa komunikasi *transmitter-receiver* modul RF pada *trash skimmer* memenuhi batas jarak dibuktikan dengan keterangan "Berhasil". Hal ini memungkinkan untuk pengendalian *trash skimmer* secara jarak jauh hingga 30 meter. Oleh karena lebar Sungai Citarum hanya berkisar 5-10 meter, maka pengendalian *trash skimmer* di Sungai Citarum dapat dilakukan secara leluasa menggunakan modul RF 433 MHz. Pada uji coba *trash skimmer* didapati fenomena *delay* komunikasi antara modul RF *transmitter-receiver*. Hal ini dibuktikan dengan uji coba *trash skimmer* saat menggunakan atap panel surya dengan ukuran yang berbeda. Gerak *trash skimmer* saat menggunakan panel surya yang lebih besar mengalami *delay* yang lebih lama. Oleh karena itu maka dilakukan juga pengukuran

jumlah sinyal yang masuk ke *receiver* dalam 1 menit terhadap proram pengiriman sinyal setiap 1 detik sebagai parameter *delay* komunikasi *transmitter-receiver* pada modul RF 433 MHz. Berikut data kualitas komunikasi modul RF 433 MHz.

Tabel IV.2 Data Kualitas Komunikasi Modul RF 433 MHz.

Batas Jarak yang ditentukan (m)	Kondisi	Rata-rata Jumlah Sinyal per Detik
Jarak 3 m	Tanpa penghalang	55 sinyal / detik
	Penghalang pintu	51 sinyal / detik
Jarak 10 m	Tanpa penghalang	50 sinyal / detik
	Penghalang pintu	46 sinyal / detik

Berdasarkan Tabel IV.2 diketahui bahwa penerimaan sinyal per detik pada *transmitter-receiver* modul RF pada trash skimmer dipengaruhi oleh jarak dan juga penghalang sehingga menyebabkan *delay* atau penundaan komunikasi yang cukup signifikan. Pengaruh jarak terhadap komunikasi modul RF yakni semakin jauh *transmitter* dengan *receiver* maka sinyal yang diterima akan semakin sedikit begitu pula sebaliknya. Sedangkan pengaruh penghalang pada modul RF adalah semakin besar penghalang maka sinyal yang diterima akan semakin sedikit dan begitu pula sebaliknya.

### IV.3 Sistem *Monitoring Trash Skimmer*

Sistem *monitoring trash skimmer* terdiri dari sistem pemantauan daya yang tersisa untuk menjalankan trash skimmer, sistem peringatan kapasitas pengangkutan sampah, sistem *monitoring* posisi, serta sistem jangkauan *monitoring trash skimmer*. Berikut hasil tampilan sistem *monitoring trash skimmer* pada program Grafana.

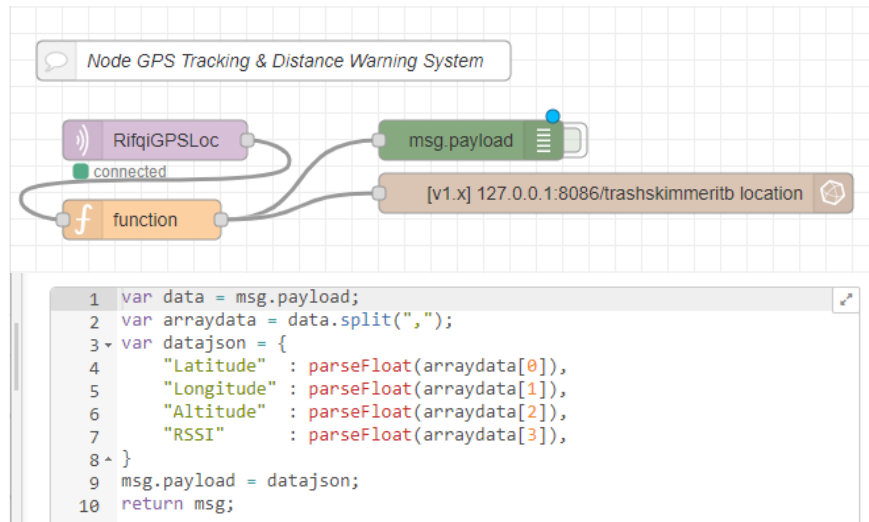


Gambar IV.2 Sistem *monitoring trash skimmer* pada program Grafana.

Berdasarkan Gambar IV.2 terlihat tampilan sistem *monitoring trash skimmer* menggunakan program Grafana. Tampilan program terdiri dari pelacakan lokasi terkini menggunakan modul GPS NEO-6M, sistem peringatan kapasitas sampah, sistem jangkauan *monitoring trash skimmer*, serta sistem pemantauan daya *trash skimmer*.

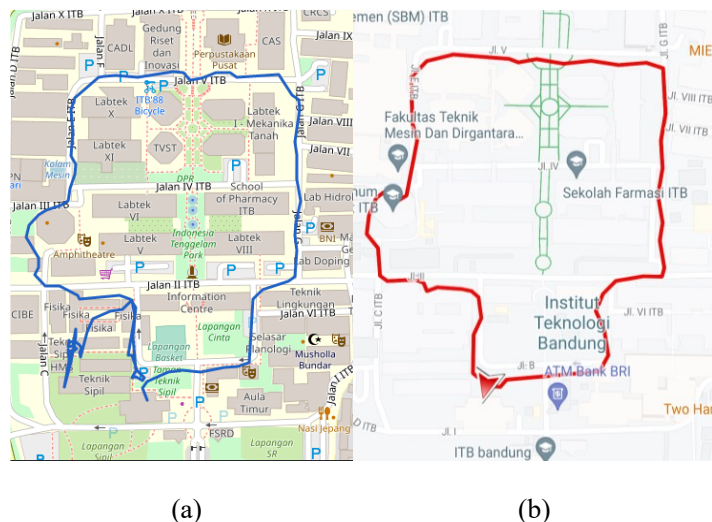
#### IV.3.1 Data Kualitas GPS NEO-6M dan Sistem Jangkauan *Monitoring Trash Skimmer*

Data GPS yang didapatkan melalui modul GPS NEO-6M kemudian dikirim menuju *database* InfluxDB melalui protokol MQTT dengan server Mosquitto. *Topic* MQTT yang digunakan pada program pengiriman data GPS adalah "RifqiGPSLoc". Data GPS yang dikirim merupakan data garis lintang (*latitude*), garis bujur (*longitude*), ketinggian (*altitude*), dan nilai RSSI NodeMCU. Selanjutnya dibuat konfigurasi *node* pada program Node-RED untuk menyimpan data GPS ke *database* InfluxDB. Berikut gambar konfigurasi *node* dan program pemisahan data (*parse data*) pada *node function*.



Gambar IV.3 Konfigurasi *node* dan program pemisahan data (*parse data*) pada *node function* sistem jangkauan *monitoring trash skimmer*.

Berdasarkan Gambar IV.3 *node function* pada gambar berfungsi untuk memisahkan data (*parsing*) dari bentuk pesan huruf menjadi data angka. Selanjutnya pengambilan data kualitas GPS dilakukan dengan membandingkan *track location* modul GPS NEO-6M dengan aplikasi GPS Tracker. Aplikasi GPS Tracker merupakan aplikasi gratis yang tersedia di google playstore dan telah diunduh lebih dari 5 juta kali dengan rating 4,6 yakni tergolong tinggi dibanding aplikasi serupa sehingga dipilih sebagai pembandingan. Lintasan yang diambil yakni jalan di dalam kampus ITB. Berikut data hasil perbandingannya.



Gambar IV.4 Data *track location* (a) Modul GPS NEO-6M da (b) aplikasi GPS *Tracker*.

Berdasarkan Gambar IV.4 hasil *tracking* yang diberikan oleh Modul GPS NEO-6M memiliki kemiripan yang cukup tinggi dengan aplikasi GPS tracker meskipun pada Modul GPS NEO-6M memiliki galat yang cukup besar ketika dibawa ke ruangan yang tertutup. Hasil ini dibuktikan dengan data GPS yang diberikan ketika menuju Gedung Fisika terdapat bias yang cukup besar, meskipun begitu modul GPS NEO-6M tetap berfungsi dengan baik di bawah pepohonan mengacu pada kondisi jalan di sekitar kampus ITB. Hal ini menandakan bahwa modul GPS NEO-6M yang digunakan cocok dipakai untuk *tracking* perjalanan *trash skimmer* dan *monitoring* lokasi terkini.

Selanjutnya dilakukan pembuatan sistem jangkauan *monitoring trash skimmer* yang berfungsi untuk memberikan peringatan pada *trash skimmer* ketika koneksi WiFi NodeMCU (untuk rangkaian *monitoring*) terhadap gawai atau router akan terputus. Jika koneksi WiFi NodeMCU terputus maka rangkaian *monitoring* tidak bisa mengirimkan data menuju *database* InfluxDB. Oleh karena itu sistem ini mencegah agar tidak terjadi putus komunikasi saat *trash skimmer* digunakan. Sistem ini dibuat berdasarkan nilai RSSI dari NodeMCU yang akan berubah seiring dengan bertambah atau berkurangnya jarak terhadap sumber WiFi. Berikut Data RSSI NodeMCU yang didapat berdasarkan perubahan jarak.

Tabel IV.3 Data Nilai RSSI NodeMCU terhadap perubahan jarak Sumber WiFi.

Jarak Sumber WiFi (m)	Rentang Nilai RSSI (dBm)	Keterangan
0	-25 sampai -90	Terhubung
10		
20		
30		
40		
50	-90 sampai -98	Terputus
> 58	tidak ada data	

Berdasarkan Tabel IV.3 dapat dilihat perubahan nilai RSSI terhadap pertambahan jarak sumber WiFi. Berdasarkan data tersebut kemudian dibuat sistem peringatan jangkauan *monitoring trash skimmer* dengan beberapa kondisi. Berikut tabel data sistem peringatan jangkauan *monitoring trash skimmer*.

Tabel IV.4 Data Sistem Peringatan Jangkauan Monitoring Trash Skimmer.

Rentang Nilai RSSI (dBm)	Tampilan pada Grafana
-20 sampai -90	Tulisan "Zona Aman" dengan <i>background</i> panel berwarna hijau
Kurang dari -90	Tulisan "Zona Jauh, Kembali!!" dengan <i>background</i> panel berwarna merah

Tabel IV.4 menunjukkan sistem peringatan jangkauan *monitoring trash skimmer* berdasarkan perubahan nilai RSSI. Nilai RSSI kurang dari -90 dBm diambil agar memberikan ruang pada *trash skimmer* untuk melakukan perjalanan kembali menuju titik asal atau berpindah ke area dekat sumber WiFi. Berikut hasil tampilan sistem jangkauan *monitoring trash skimmer* pada program Grafana.

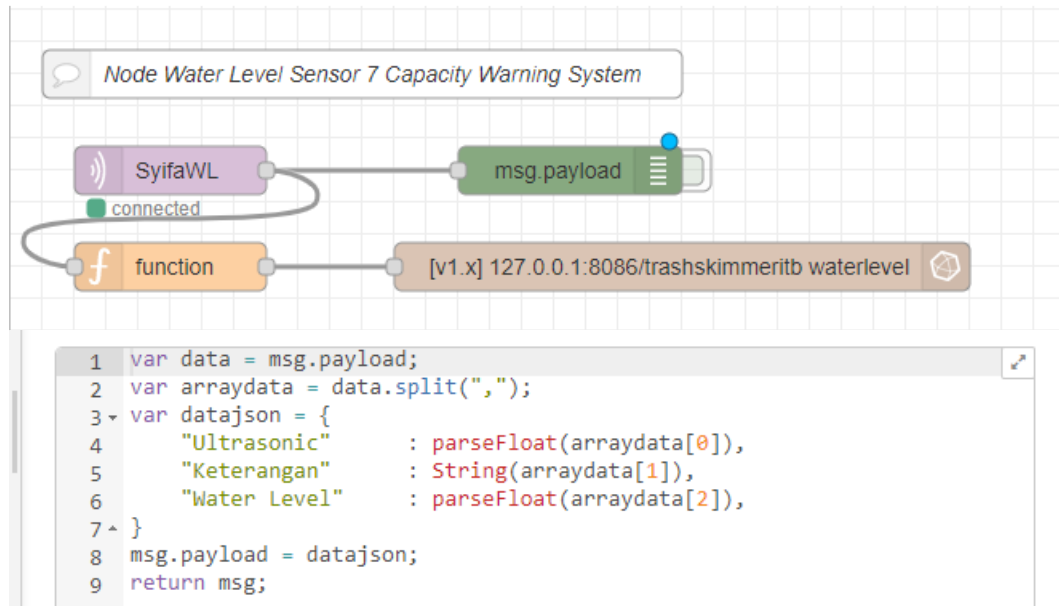


Gambar IV.5 Sistem Jangkauan *monitoring* pada program Grafana: (Atas) Kondisi zona aman & (Bawah) kondisi zona jauh.

### IV.3.2 Data Sistem Peringatan Kapasitas Sampah

Data sistem peringatan kapasitas sampah yang didapatkan melalui sensor level air dan sensor ultrasonik modul HC-SR04 dikirim menuju *database* InfluxDB melalui protokol MQTT dengan server Mosquitto. Topik MQTT yang digunakan pada program pengiriman data adalah "SyifaWL". Data sensor yang dikirim merupakan data analog sensor level air dan jarak air dengan batas sarat *trash skimmer*. Kemudian dibuat konfigurasi *node* pada program Node-RED untuk menyimpan data sensor ke *database* InfluxDB. Berikut gambar konfigurasi *node* dan program pemisahan data (*parse data*) pada *node function*.





Gambar IV.6 Konfigurasi *node* dan program pemisahan data (*parse data*) pada *node function* sistem peringatan kapasitas sampah.

Gambar IV.6 menunjukkan konfigurasi *node* untuk sistem peringatan kapasitas sampah. *Node function* pada gambar berfungsi untuk memisahkan data (*parsing*) dari bentuk pesan huruf menjadi data angka. Selanjutnya dibuat sistem peringatan kapasitas sampah yang berfungsi untuk memberikan informasi kepada pengguna ketika *trash skimmer* sudah dipenuhi oleh sampah. Sistem ini dibuat dengan menghitung jarak sarat pada *trash skimmer*. Data sistem peringatan kapasitas sampah ditunjukkan oleh tabel berikut.

Tabel IV.5 Data Sistem Peringatan Kapasitas Sampah

Nilai Analog Sensor Level air	Tampilan pada Grafana	Kondisi / Keterangan
Nilai analog < 150	Tulisan "AMAN" dengan <i>background</i> panel berwarna hijau	Buzzer peringatan tidak berbunyi
$150 \leq$ Nilai analog $\leq 350$	Tulisan ""WARNING, HAMPIR PENUH" dengan <i>background</i> panel berwarna kuning	Buzzer peringatan berbunyi setiap 1 detik
Nilai analog > 350	Tulisan "PENUH!!!" dengan <i>background</i> panel berwarna merah	Buzzer peringatan terus berbunyi

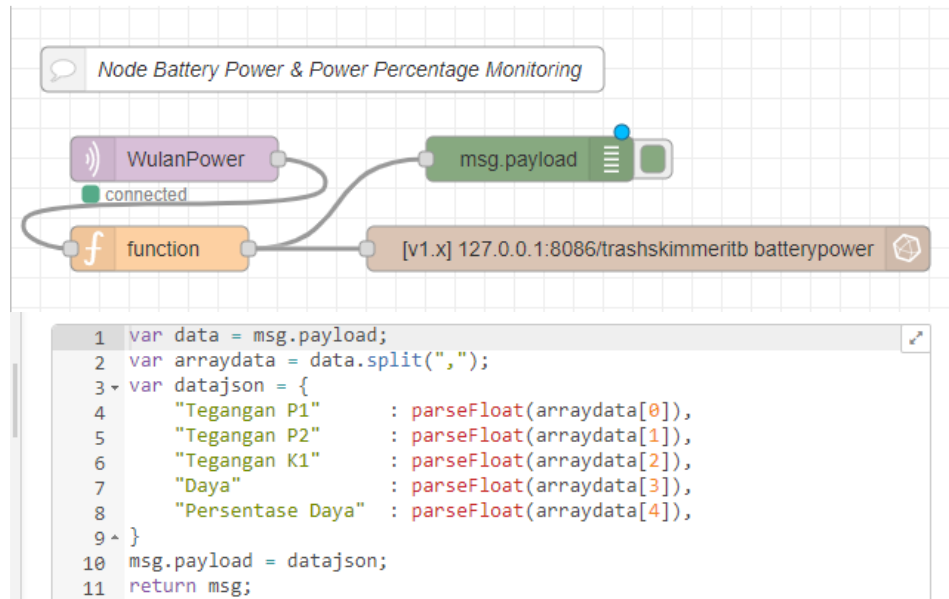
Berdasarkan Tabel IV.5 dapat dilihat sistem peringatan kapasitas sampah pada beberapa kondisi berdasarkan nilai analog sensor level air yang diterima pada database InfluxDB. Sedangkan data sensor ultrasonik modul HC-SR04 akan langsung ditampilkan secara realtime menuju program Grafana. Berikut hasil tampilan sistem jangkauan monitoring trash skimmer pada program Grafana.



Gambar IV.7 Sistem peringatan kapasitas sampah: (a) kondisi aman, (b) kondisi *warning*, & (c) kondisi penuh.

#### IV.3.3 Data Sistem Pemantauan Daya *Trash Skimmer*

Data sistem pemantauan daya *trash skimmer* yang didapatkan melalui sensor tegangan dikirim menuju *database* InfluxDB melalui protokol MQTT dengan server Mosquitto. Topik MQTT yang digunakan pada program pengiriman data adalah "WulanPower". Data sensor yang dikirim merupakan data analog sensor tegangan yang terhubung dengan seluruh baterai LiPo pada *trash skimmer*. Kemudian dibuat konfigurasi *node* pada program Node-RED untuk menyimpan data sensor tegangan ke *database* InfluxDB. Berikut gambar konfigurasi *node* dan program pemisahan data (*parse data*) pada *node function*.



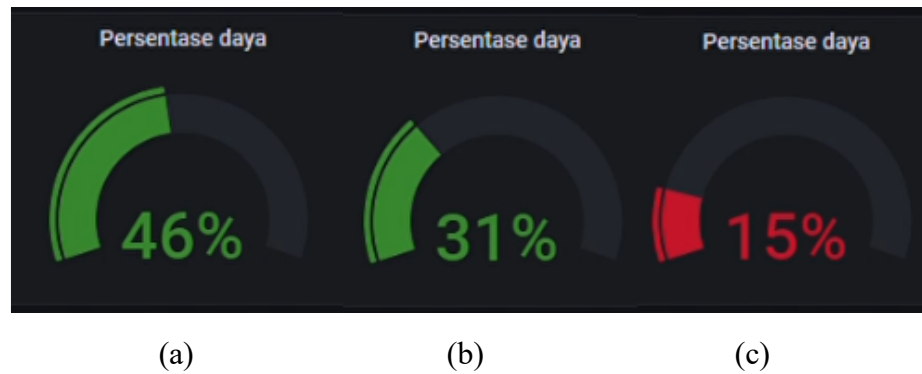
Gambar IV.8 Konfigurasi *node* dan program pemisahan data (*parse data*) pada *node function* sistem pemantauan daya *trash skimmer*.

Gambar IV.8 menunjukkan konfigurasi *node* untuk sistem pemantauan daya *trash skimmer*. *Node function* pada gambar berfungsi untuk memisahkan data (*parsing*) dari bentuk pesan huruf menjadi data angka. Selanjutnya dibuat sistem pemantauan daya *trash skimmer* yang berfungsi untuk memantau sisa persentase daya baterai yaang tersedia ketika *trash skimmer* sedang digunakan. Daya yang dihitung merupakan daya total yang berasal dari baterai LiPo yang digunakan untuk menghidupkan sistem penggerak *trash skimmer* yang terdiri dari rangkaian propeler dan rangkaian konveyor. Nilai daya diambil berdasarkan sensor tegangan yang masuk melalui NodeMCU dengan arus konstan karena kecepatan motor yang digunakan konstan.

Tabel IV.6 Data Sistem Pemantauan Daya.

Sensor Tegangan	Tampilan pada Grafana	Kondisi / Keterangan
persentase daya > 20%	Persentase daya dengan <i>background</i> panel berwarna hijau	Buzzer tidak berbunyi
Persentase daya < 20%	Persentase daya dengan <i>background</i> panel berwarna merah	Buzzer berbunyi kontinu

Tabel IV.6 menunjukkan sistem pemantauan dan peringatan sistem daya pada 2 kondisi. Batas sisa daya 20% dipilih karena dianggap mampu memulangkan *trash skimmer* dari tengah sungai ke titik awal. Berikut hasil tampilan sistem pemantauan daya *trash skimmer* pada program Grafana.



Gambar IV.9 Sistem pemantauan daya trash skimmer pada Grafana: (a) saat 46%, (b) saat 30%, & (c) saat 15%.

## BAB V KESIMPULAN

### V.1 Kesimpulan

Berdasarkan metode yang digunakan serta hasil dan pembahasan pada penelitian ini, maka dapat disimpulkan sebagai berikut:

1. Seluruh sistem gerak trash skimmer telah berhasil dibuat. *Trash skimmer* mampu bergerak diperairan tenang dengan kecepatan 0,21 m/s. *Trash skimmer* bergerak sesuai instruksi pengguna melalui joystick pada sistem komunikasi trash skimmer.
2. Sistem komunikasi *trash skimmer* telah berhasil dibuat. Data kualitas komunikasi telah diambil dan didapatkan hasil rentang 0-30m keberhasilan komunikasi dengan beberapa kondisi. Lalu didapatkan pengaruh jarak dan penghalang terhadap penerimaan sinyal *receiver* yakni semakin jauh jarak *transmitter* dan atau semakin tebal penghalang maka semakin sedikit sinyal yang diterima.
3. Sistem *monitoring trash skimmer* telah berhasil dibuat. Sistem tersebut terdiri dari sistem pemantauan daya menggunakan rangkaian sensor tegangan, sistem peringatan kapasitas sampah menggunakan rangkaian sensor level air dan sensor ultrasonik, sistem pelacakan lokasi terkini menggunakan rangkaian modul GPS, dan sistem jangkauan *monitoring* menggunakan nilai RSSI NoveMCU. Aplikasi sistem *monitoring trash skimmer* dibuat menggunakan program Grafana serta dapat diakses menggunakan laptop dan gawai

## V.2 Saran

Berdasarkan eksperimen yang telah dilakukan, maka untuk pengembangan prototipe *trash skimmer* terdapat beberapa saran yakni sebagai berikut:

1. Sistem komunikasi *trash skimmer* perlu dikembangkan lebih lanjut salah satunya dengan mencari modul komunikasi lain yang mampu melakukan komunikasi data dengan frekuensi yang lebih tinggi agar sistem komunikasi lebih lancar dan stabil.
2. Untuk sistem peringatan kapasitas pengangkutan sampah, sensor yang dipakai cukup menggunakan sensor ultrasonik modul HC-SR04, karena panjang sarat *trash skimmer* melebihi panjang sensor level air yang digunakan. Fungsi alarm dari sensor level air juga dapat digunakan dengan sensor ultrasonik modul HC-SR04.
3. Untuk Efisiensi pembuatan alat, sistem komunikasi trash skimmer dapat dilakukan menggunakan sistem IoT dengan rangkaian pengontrol menggunakan Gawai serta penerima sinyal menggunakan NodeMCU. Dengan membuat sistem komunikasi IoT, maka rangkaian arduino meggunakan *joystick* tidak perlu dibuat dan penggantian sistem penerimaan sinyal atau sistem penggerak dapat diganti menggunakan NodeMCU yang relatif lebh murah dibandingkan Arduino Uno.
4. Dapat dibuat sistem router yang terintegrasi langsung pada *trash skimmer* sehingga sistem jangkauan *monitoring trash skimmer* tidak perlu dibuat karena sistem jarak komunikasi IoT tidak berubah.

## DAFTAR PUSTAKA

- Agustina, Nova. 2017. "Perancangan Digital Wireless Remote Stick Commander Untuk Pengendali Camera Crane Dan Pan Tilt Head Berbasis Sensor Accelerogyro," 105. <http://repository.its.ac.id/46808/>. diakses pada tanggal 2 Juli 2022
- Alief, Hafizul. 2017. "DC MOTOR SPEED CONTROL USING MEASURED DISTANCE BY ULTRASONIC SENSOR." 549: 40–42
- Djamal, Daryanda Dwiammardi. 2020. "PENGEMBANGAN APLIKASI NAVIGASI BERBASIS RSSI-WIFI PADA SMARTPHONE ANDROID." *Tugas Akhir ITB* 4-15.
- Halliday, D., Resnick, R., & Walker, J. 2010. *Fisika Dasar Edisi 7 Jilid 1*. Jakarta: Penerbit Erlangga.
- Hibban, Hudaibiy. 2015. "Desain Kontroler Fuzzy PID Gain Scheduling Untuk Pengaturan Kecepatan Motor DC Tanpa Sikat." *Jurnal Teknik ITS* 4 (2): 2–5.
- Ichwanda, Arsharizka Syahadati. 2020. "PENGEMBANGAN SISTEM PELACAKAN POSISI DI DALAM RUANGAN BERBASIS WIFI-RSSI MENGGUNAKAN SMARTPHONE." *Tugas Akhir ITB* 5-10.
- Ingle, Kshitija A, Akash G Bhatkar, Rahul S Tarmale, Tejashri D Ingle, and Mohan S Bawaskar. 2020. "A Review of River Cleaning Robot Using Solar Power," no. 7.
- KLHK. 2021. "Citarum Harum Juara." *SAMPAH DAS CITARUM CAPAI 15.838 TON/HARI*. <https://citarumharum.jabarprov.go.id/sampah-das-citarum-capai-15-838-ton-hari/>. diakses pada tanggal 2 Juli 2022
- Kurnia Rahman, Aji, Hadi Supriyanto, and Tika Meizinta. 2019. "Rancang Bangun Dan Implementasi Sistem Kendali Quadcopter Melalui Jaringan Internet Berbasis Lokasi Menggunakan Smartphone Android," 307–18. <https://doi.org/10.5614/sniko.2018.35>. diakses pada tanggal 2 Juli 2022

- Mariano, M. Scicluna, K. & Scerri, J. 2017. "Modelling of a sensorless rotor Flux oriented BLDC machine." *International Conference on Electical Drives and Power Electronics* 194-199.
- Mulyono, Sri, Muhammad Qomaruddin, and Muhammad Syaiful Anwar. 2018. "Penggunaan Node-RED Pada Sistem Monitoring Dan Kontrol Green House Berbasis Protokol MQTT." *Jurnal Transistor Elektro Dan Informatika (TRANSISTOR EI* 3 (1): 31–44.
- Rahayuningtyas, Ari. 2009. "Pembuatan Sistem Pengendali 4 Motor DC Penggerak 4 Roda Secara Independent Berbasis Mikrokontroler AT89C2051." *Jurnal Fisika Himpunan Fisika Indonesia* 9 (2): 24–33.
- Rittenberry, R. 2005. "Hands-on Technology: L298N Dual H-Bridge Motor Driver." *Occup Health Saf* 74 (2): 24.  
[http://www.handsontec.com/dataspecs/L298N Motor Driver.pdf](http://www.handsontec.com/dataspecs/L298N_Motor_Driver.pdf). diakses pada tanggal 1 Juli 2022
- SIPSN. 2021. "CAPAIAN KINERJA PENGELOLAAN SAMPAH." *Sistem Informasi Pengelolaan Sampah Nasional (SIPSN)* .  
<https://sipsn.menlhk.go.id/sipsn/>. diakses pada tanggal 1 Juli 2022
- Suryana, T. 2021. "Antarmuka Ublox Neo-6m Gps Module Dengan Nodemcu Esp8266."  
[https://repository.unikom.ac.id/68725/%0Ahttps://repository.unikom.ac.id/68725/1/Antarmuka ublox NEO-6M GPS Module dengan NodeMCU ESP8266.pdf](https://repository.unikom.ac.id/68725/%0Ahttps://repository.unikom.ac.id/68725/1/Antarmuka_ublox_NEO-6M_GPS_Module_dengan_NodeMCU_ESP8266.pdf). diakses pada tanggal 2 Juli 2022
- Yedamale, P. 2003. *Brushless DC (BLDC) Motor Fundamentals: Author: Padmaraja Yedamale Microchip Technology Inc.*



## **LAMPIRAN A : Data Sensor Sistem *Monitoring Trash Skimmer***

### **A.1 Data Nilai RSSI Terhadap Perubahan Jarak**

<b>No</b>	<b>Jarak (m)</b>	<b>RSSI (dBm)</b>
1	1	-15
2	3	-35
3	6	-48
4	9	-59
5	12	-69
6	15	-73
7	18	-75
8	21	-80
9	24	-79
10	27	-82
11	30	-85
12	33	-86
13	36	-87
14	39	-89
15	42	-90
16	45	-93
17	48	-95
18	51	-97
19	54	-99
20	57	Tidak ada
21	60	Tidak ada

## LAMPIRAN B : Program Sistem Komunikasi & *Monitoring*

### B.1 Program Sistem Komunikasi Penggerak-*Transmitter*

```
// Include RadioHead Amplitude Shift Keying Library
#include <RH_ASK.h>
// Include dependant SPI Library
#include <SPI.h>

#define joyX A0
#define joyY A1
int xValue = 0;
int yValue = 0;

// Create Amplitude Shift Keying Object
RH_ASK rf_driver;

void setup() {
    Serial.begin(9600);
    // Initialize ASK Object
    rf_driver.init();
}

void loop() {
    xValue = analogRead(joyX);
    yValue = analogRead(joyY);

    if((xValue>=450)&&(xValue<=520)&&(yValue<=2)){
        const char *msg = "W";
        Serial.println("W");
        rf_driver.send((uint8_t*)msg, strlen(msg));
        rf_driver.waitPacketSent();
    }
    else if((yValue>=460)&&(yValue<=540)&&(xValue<=2)){
        const char *msg = "A";
        Serial.println("A");
        rf_driver.send((uint8_t*)msg, strlen(msg));
        rf_driver.waitPacketSent();
    }
    else if((yValue>=460)&&(yValue<=540)&&(xValue>900)){
        const char *msg = "D";
        Serial.println("D");
        rf_driver.send((uint8_t*)msg, strlen(msg));
        rf_driver.waitPacketSent();
    }
}
```

```

        else if((xValue>=460)&&(xValue<=540)&&(yValue>900)){
            const char *msg = "S";
            Serial.println("S");
            rf_driver.send((uint8_t*)msg, strlen(msg));
            rf_driver.waitPacketSent();
        }
        else
    if((xValue>=460)&&(xValue<=540)&&(yValue>=460)&&(yValue<=540
    )){
        const char *msg = "C";
        Serial.println("C");
        rf_driver.send((uint8_t*)msg, strlen(msg));
        rf_driver.waitPacketSent();
    }

    delay(500);
}

```

## **B.2 Program Sistem Komunikasi Penggerak-Receiver**

```

#include <RH_ASK.h>
#include <SPI.h>
#include <Servo.h>

int motor1pin1 = 2;
int motor1pin2 = 3;
int motor2pin1 = 4;
int motor2pin2 = 5;

RH_ASK rf_driver;
Servo ESC1;
Servo ESC2;

int pos = 5;
int potValue;
int on = 0;

void setup(){
    pinMode(motor1pin1, OUTPUT);
    pinMode(motor1pin2, OUTPUT);
    pinMode(motor2pin1, OUTPUT);
    pinMode(motor2pin2, OUTPUT);
    rf_driver.init();
    Serial.begin(9600);
    delay(1000);
    ESC1.attach(7,1000,2000);
    ESC2.attach(8,1000,2000);
}

```

```

}

int switchpos(){
  // Set buffer to size of expected message
  uint8_t buf[1];
  uint8_t buflen = sizeof(buf);
  // Check if received packet is correct size
  if (rf_driver.recv(buf, &buflen)){
    // Message received with valid checksum
    const char *msg = (char*)buf;
    Serial.print("Message Received: ");
    Serial.println(msg);

    if(strcmp(msg,"W") == 0){
      return 1;
    }
    else if(strcmp(msg,"A") == 0){
      return 2;
    }
    else if(strcmp(msg,"D") == 0){
      return 3;
    }
    else if(strcmp(msg,"S") == 0){
      return 4;
    }
    else if(strcmp(msg,"C") == 0){
      return 5;
    }
  }
}

void loop(){
  pos = switchpos();
  Serial.println(pos);
  //delay(100);
  switch(pos) {
    case 1:
      Serial.println("Masuk 1");
      //potValue = map(1000, 0, 1023, 0, 180);
      ESC1.write(105);
      ESC2.write(100);
      break;
    case 2:
      Serial.println("Masuk 2");
      //potValue = map(0, 0, 1023, 0, 180);
      ESC1.write(105);
      ESC2.write(91);

```

```

break;
case 3:
    Serial.println("Masuk 3");
    //potValue = map(0, 0, 1023, 0, 180);
    ESC1.write(91);
    ESC2.write(100);
break;
case 4:
    Serial.println("Masuk 4");
    //potValue = map(0, 0, 1023, 0, 180);
    ESC1.write(91);
    ESC2.write(91);
    digitalWrite(motor2pin1, HIGH);
    digitalWrite(motor2pin2, LOW);
break;
case 5:
    Serial.println("Masuk 5");
    //potValue = map(0, 0, 1023, 0, 180);
    ESC1.write(91);
    ESC2.write(91);
break;}
delay(500);
digitalWrite(motor2pin1, LOW);
digitalWrite(motor2pin2, LOW);
pos = 5;
}

```

### **B.3 Program Sistem *Monitoring Trash Skimmer***

```

#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include <PubSubClient.h>
#include <ESP8266WiFi.h>
#include <MCP3008.h> // ADC Extention
#include <Wire.h> // sensor voltage
#include "credentials.h"

// Choose two Arduino pins to use for software serial
// nilai RSSI
int quality;
int getQualityRSSI;
//sensor tegangan
float Vmodul1 = 0.0;
float Vmodul2 = 0.0;
float Vmodul3 = 0.0;
float VmodulTot = 0.0;
int value1 = 0;

```

```

int value2 = 0;
int value3 = 0;
int persentase;
int arus = 20;
int Daya;
int hitung;
// Define pin analog
#define CS_PIN D5
#define CLOCK_PIN D8
#define MOSI_PIN D6
#define MISO_PIN D7
MCP3008 adc(CLOCK_PIN, MOSI_PIN, MISO_PIN, CS_PIN);
int Channel;
// Water Level Sensor Pin
int sensorPin = A0;
int buzzerPin = 10;
int sensorValue = 0;
int triggerPin = D3;
int echoPin = D4;
String WL;
float Jarak;
float jarak;
//GPP Pin
int RXPin = D2;
int TXPin = D1;
int GPSBaud = 9600;
const char* ssid = "Rifqi Nafis";
const char* password = "12345678";
const char* mqttServer = "test.mosquitto.org";
const char* mqttUsername = "Username";
const char* mqttPassword = "Password";
char *pubTopic[] =
{"RifqiGPSLoc", "SyifaWL", "WulanPower"}; //payload[0]
will control/set LED
char payload[] = "Hello";
TinyGPSPlus gps; // Create a TinyGPS++ object
SoftwareSerial gpsSerial(RXPin, TXPin); // Create a software
serial port called "gpsSerial"
WiFiClient espClient;
PubSubClient client(espClient);

void setup_wifi()
{
    delay(10);
    // We start by connecting to a WiFi network
    Serial.println();

```

```

Serial.print("Connecting to ");
Serial.println(ssid);
WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
}
randomSeed(micros());
Serial.println("");
Serial.println("WiFi connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());
}

void reconnect()
{
    // Loop until we're reconnected
    while (!client.connected())
    {
        Serial.print("Attempting MQTT connection...");
        String clientId = "TrashSkimmerITB";          // Create a
random client ID
        clientId += String(random(0xffff), HEX);
        if (client.connect(clientId.c_str(), mqttUsername,
mqttPassword)) { // Attempt to connect
            Serial.println("connected");
        }
        else {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" try again in 5 seconds");
            delay(5000);          // Wait 5 seconds before retrying
        }
    }
}

void setup()
{
    // Water Level Setup
    pinMode(triggerPin, OUTPUT);
    pinMode(echoPin, INPUT);
    pinMode (sensorPin, INPUT);
    pinMode (buzzerPin, OUTPUT);
    // GPS setup
    gpsSerial.begin(GPSBaud); // Start the software serial
port at the GPS's default baud
    //Esp setup

```

```

    Serial.begin(9600);
    Serial.println();
    setup_wifi();
    client.setServer(mqttServer, 1883);
}

void loop()
{
    // This sketch displays information every time a new
    sentence is correctly encoded.
    if (!client.connected()){
        reconnect();
    }
    client.loop();
    while (gpsSerial.available() > 0)
        if (gps.encode(gpsSerial.read()))
        {
            displayInfoGPS();
            displayInfoWL();
            displayInfoWL2();
            displayInfoDaya();
            client.subscribe("RifqiGPSLoc"); // Ensure that we
are subscribed to the topic
            client.subscribe("SyifaWL");
            client.subscribe("WulanPower");
            String GPS = String(gps.location.lat(), 7) + "," +
String(gps.location.lng(), 7) + "," +
String(gps.altitude.meters())+ "," + String(WiFi.RSSI());
            String Waterlevel = String(Jarak) + "," +
String(WL) + "," + String(sensorValue);
            String dayabaterai = String(Vmodul1) + "," +
String(Vmodul2) + "," + String(Vmodul3) + "," + String(Daya)
+ "," + String(persentase);
            if(client.publish("RifqiGPSLoc",GPS.c_str()) &&
client.publish("SyifaWL",Waterlevel.c_str()) &&
client.publish("WulanPower",dayabaterai.c_str())) ){
                delay(1000);
                Serial.println("Publish message success");
                Serial.println();
                Serial.println();
            }
            else {
                Serial.println("Could not send message :(");
                reconnect();
            }
            delay(2000);
        }
}

```



```

    }
    if (millis() > 5000 && gps.charsProcessed() < 10) {
        Serial.println("No GPS detected");
        while(true);
    }
}

void displayInfoGPS()
{
    if (gps.location.isValid()){
        Serial.print("Latitude: ");
        Serial.println(gps.location.lat(), 7);
        Serial.print("Longitude: ");
        Serial.println(gps.location.lng(), 7);
        Serial.print("Altitude: ");
        Serial.println(gps.altitude.meters());
    }
    else{
        Serial.println("Location: Not Available");
    }
}

void displayInfoWL()
{
    int sensorValue = analogRead(A0);
    Serial.print("Sensor Water Level: ");
    Serial.print(sensorValue);
    if (sensorValue < 250) {
        digitalWrite(buzzerPin, LOW);
        Serial.println(" AMAN");
        WL = "AMAN";
    }
    else if (sensorValue > 250 and sensorValue < 460) {
        digitalWrite(buzzerPin, HIGH);
        delay(100);
        digitalWrite(buzzerPin, LOW);
        delay(100);
        Serial.println(" WARNING");
        WL = "WARNING";
    }
    else if (sensorValue > 500) {
        digitalWrite(buzzerPin, HIGH);
        Serial.println(" PENUH!!!");
        WL = "PENUH!!!";
    }
}

```

```

void displayInfoWL2() {
    long duration, jarak;
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(2);
    digitalWrite(triggerPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(triggerPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    jarak = (duration/2) / 29.1;
    Serial.print("jarak :");
    Serial.print(jarak);
    Serial.println(" cm");
    Jarak = jarak;
}

void displayInfoDaya(){
    value1 = adc.readADC(0);
    value2 = adc.readADC(1);
    value3 = adc.readADC(2);
    Vmodul1 = (value1 * 7.4) / 1024.0;
    Vmodul2 = (value2 * 7.4) / 1024.0;
    Vmodul3 = (value3 * 7.4) / 1024.0;
    VmodulTot = Vmodul1 + Vmodul2 + Vmodul3;
    Daya = VmodulTot*arus;
    hitung = Daya*100;
    persentase = hitung/(148*3); // karena ada 3 baterai
    Serial.print("Daya = ");
    Serial.println(Daya);
    Serial.print("Tegangan = ");
    Serial.print(Vmodul1);
    Serial.print(" ");
    Serial.print(Vmodul2);
    Serial.print(" ");
    Serial.println(Vmodul3);
    Serial.print("Persentase sisa daya = ");
    Serial.println(persentase);
}

```