

**PERANCANGAN SISTEM *MONITORING* DAN *DATA LOGGING* GAS  
METANA ( $CH_4$ ) UNTUK ANALISIS FLUKS KARBON PADA METODE  
*EDDY COVARIANCE* BERBASIS *INTERNET OF THINGS (IOT)***

**(Skripsi)**

**Oleh**

**MUHAMMAD JUAN PRADANA**



**FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2023**

## ABSTRAK

### PERANCANGAN SISTEM *MONITORING* DAN *DATA LOGGING* GAS METANA ( $CH_4$ ) UNTUK ANALISIS FLUKS KARBON PADA METODE *EDDY COVARIANCE* BERBASIS *INTERNET OF THINGS (IOT)*

Oleh

MUHAMMAD JUAN PRADANA

Kelapa sawit merupakan penghasil minyak untuk makanan maupun bahan bakar. Namun perkebunannya bisa menghasilkan gas metana dan dapat menjadi masalah lingkungan jika tidak dikelola dengan baik. Penelitian ini bertujuan untuk merancang sistem *monitoring* gas metana, termasuk suhu udara, kelembaban udara, kadar air di udara, dan tekanan udara. *Controller* yang digunakan adalah *Arduino Portenta H7* dengan rincian sensor  $CH_4$ , sensor tekanan udara, dan sensor suhu dan kelembaban udara. Pada pengembangan lebih lanjut, data dari perangkat *monitoring* ini bisa disandingkan dengan data arah dan kecepatan angin untuk dilakukan perhitungan *Eddy Covariance*. Dari data tersebut bisa didapat seberapa besar peran perkebunan kelapa sawit dalam kasus meningkatnya pemanasan global. Penelitian ini juga merancang *data logger*, *solar charge controller data parser*, dan *platform Internet of Things*. *Data logger* dan *SCC data parser* menggunakan *Raspberry Pi 4 Model B* dan dibangun pada *platform Node-RED*. *Platform Internet of Things* dibangun menggunakan *database Postgresql*, *Express*, dan *Grafana*. Perangkat *monitoring* gas karbon berhasil melakukan *sensing* data dengan rata-rata data harian 1,8 juta data. *Data logger* mampu menangani kecepatan data masuk sebesar 1 Hz hingga 9 Hz dari masing-masing perangkat *monitoring* gas karbon. *Database Postgresql* mampu melakukan proses data hingga 1,8 juta data lebih. Aplikasi *backend* bisa menangani permintaan masuk dan keluar data serta *Grafana* mampu menampilkan *dashboard* dari data yang diminta.

Kata Kunci: *Monitoring*, Metana, *Arduino Portenta H7*, *Data Logger*, *Raspberry Pi 4B*, *SCC*, *IoT*, *Postgresql*, *Node-RED*, *Grafana*.

## **ABSTRACT**

### **DESIGN OF METHANE (CH<sub>4</sub>) MONITORING AND DATA LOGGING SYSTEM FOR CARBON FLUX ANALYSIS USING THE INTERNET OF THINGS (IOT) BASED EDDY COVARIANCE METHOD**

**By**

**MUHAMMAD JUAN PRADANA**

Oil palm is a source of oil for both food and fuel production. However, its cultivation can result in methane gas emissions and can pose an environmental problem if not managed properly. This research aims to design a methane gas monitoring system, including air temperature, air humidity, air moisture content, and air pressure. The controller used is the Arduino Portenta H7, equipped with CH<sub>4</sub>, air pressure, and Temperature Humidity sensors. In further development, data from these monitoring devices can be combined with wind direction and speed data for Eddy Covariance calculations. This data can provide insights into the role of oil palm plantations in contributing to global warming. The research also designs a data logger, solar charge controller data parser, and an Internet of Things (IoT) platform. The data logger and SCC data parser use a Raspberry Pi 4 Model B and are built on the Node-RED platform. The IoT platform is constructed using a PostgreSQL database, Express, and Grafana. The carbon gas monitoring devices successfully collect data with an average of 1.8 million daily data points. The data logger can handle data input speeds ranging from 1 Hz to 9 Hz from each carbon gas monitoring device. The PostgreSQL database can process over 1.8 million data points. The backend application can manage incoming and outgoing data requests, and Grafana can display requested data on dashboards.

**Keywords:** Monitoring, Methane, Arduino Portenta H7, Data Logger, Raspberry Pi 4B, SCC, IoT, PostgreSQL, Node-RED, Grafana.

**PERANCANGAN SISTEM *MONITORING* DAN *DATA LOGGING* GAS  
METANA ( $CH_4$ ) UNTUK ANALISIS FLUKS KARBON PADA METODE  
*EDDY COVARIANCE* BERBASIS *INTERNET OF THINGS (IOT)***

Oleh

**MUHAMMAD JUAN PRADANA**

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar  
**SARJANA TEKNIK**

Pada

**Program Studi Teknik Elektro  
Fakultas Teknik  
Universitas Lampung**



**FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2023**

Judul Skripsi

**: PERANCANGAN SISTEM MONITORING  
DAN DATA LOGGING GAS METANA (CH<sub>4</sub>)  
UNTUK ANALISIS FLUKS KARBON PADA  
METODE EDDY COVARIANCE BERBASIS  
INTERNET OF THINGS (IOT)**

Nama Mahasiswa

**: Muhammad Juan Pradana**

Nomor Pokok Mahasiswa

**: 1915031038**

Jurusan

**: Teknik Elektro**

Fakultas

**: Teknik**



**1. Komisi Pembimbing**

**Dr.-Ing. Ardian Ulvan, S.T., M.Sc.**  
NIP. 197311281999031005

**Arjanto, S.T., M.T.**  
NIP. 199006212019031011

**2. Mengetahui**

**Ketua Jurusan  
Teknik Elektro**

**Herlinawati, S.T., M.T.**  
NIP. 19710314 199903 2 001

**Ketua Program Studi  
Teknik Elektro**

**Dr. Eng. Nining Purwasih, S.T., M.T.**  
NIP. 19740422 200012 2 001

**MENGESAHKAN**

**1. Tim Penguji**

**Ketua : Dr. Ing. Ardian Ulvan, S.T., M.Sc.**



**Sekretaris : Aryanto, S.T., M.T.**



**Penguji : Mona Arif Muda S.T., M.T**



**Dekan Fakultas Teknik**



**Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. }**  
NIP. 19750928 200112 1 002



**Tanggal Lulus Ujian Skripsi : 4 Agustus 2023**

## SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini , menyatakan bahwa skripsi saya dengan judul “Perancangan Sistem *Monitoring dan Data Logging* Gas Metana ( $CH_4$ ) untuk Analisis Fluks Karbon pada Metode *Eddy Covariance* Berbasis *Internet of Things (IoT)*” dibuat oleh saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung, 12 Oktober 2023

buat pernyataan,



Muhammad Juan Pradana

NPM. 1915031038

## RIWAYAT HIDUP



Penulis dilahirkan di Lampung Timur pada tanggal 29 Desember 2000. Penulis merupakan anak pertama dari dua bersaudara dari pasangan Bapak Aris Suprihartono dan Ibu Sutri Anggraini.

Penulis memulai pendidikan di TK Aisyah desa Bauh Gunung Sari pada tahun 2006-2007, SDN 1 Desa Bauh Gunung Sari pada tahun 2007-2013, SMPN 1 Sekampung Udik pada tahun 2013-2016, dan MAN 2 Bandar Lampung pada tahun 2016-2019.

Penulis menjadi mahasiswa Jurusan Teknik Elektro Universitas Lampung pada tahun 2019 melalui jalur SBMPTN (Seleksi Bersama Masuk Perguruan Tinggi Negeri). Selama menjadi mahasiswa, penulis pernah menjadi anggota Himpunan Mahasiswa Teknik Elektro (HIMATRO) sebagai anggota divisi Media Informasi pada periode 2020 dan periode 2021. Penulis juga menjadi anggota Lembaga riset Unila Robotika dan Otomasi (URO) sejak 2019 sebagai anggota robotika dan sebagai programmer. Penulis tertarik dalam bidang teknologi khususnya web dan otomasi.



## **MOTTO**

“Lakonono ben kelakon!”

**(Penulis)**

“Seni bukan hanya tentang tulisan, rupa, dan indra”

**(mjp\_p.h)**

“Caranya adalah: tidak ada caranya, karena ini terkait dengan naluri dan insting”

**(Guru Gembul)**

“Janganlah kamu berduka cita, sesungguhnya Allah selalu bersama kita.”

**(QS. At-Taubah: 40)**

## **PERSEMBAHAN**

Dengan mengucapkan Alhamdulillah dan syukur kepada Allah SWT atas nikmat serta hidayahnya sehingga skripsi ini dapat terselesaikan dengan baik. Rasa syukur dan bahagia saya persembahkan karya ini kepada:

### **Kedua Orang Tua**

Terima kasih kepada kedua orang tuaku tercinta, Ibu Sutri Anggraini dan Bapak Aris Suprihartono atas segala pengorbanan, doa dan ridho kalian serta dukungannya selama ini. Terima kasih telah memberikan pelajaran berharga tentang makna perjalanan hidup yang sebenarnya sehingga kelak bisa menjadi orang yang bermanfaat bagi semua orang.

### **Dosen Pembimbing dan Pembahas**

Terima kasih kepada dosen pembimbing dan pembahas yang sudah sangat membantu, memberikan motivasi, memberikan arahan serta ilmu yang berharga.

### **Keluarga dan Teman-temanku**

Terimakasih kepada semua orang-orang baik yang telah memberikan pengalaman, semangat, motivasinya, serta doa-doanya dan senantiasa memberikan dukungan dalam hal apapun.

### **Almamater Tercinta**

Universitas Lampung

## SANWACANA

Alhamdulillahirabbil'alamin. Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi/tugas akhir ini dengan judul “Perancangan Sistem *Monitoring* dan *Data Logging* Gas Metana ( $CH_4$ ) untuk Analisis Fluks Karbon pada Metode *Eddy Covariance* Berbasis *Internet of Things* (IoT)”. Dalam pelaksanaan dan pembuatan skripsi/tugas akhir ini penulis menerima dukungan baik secara moril maupun materil yang sangat berharga dari berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih ke semua pihak yang telah membantu, khususnya kepada:

1. Kedua orang tua tercinta dan seluruh keluarga penulis yang tidak hentinya mendoakan serta memberikan dorongan semangat, motivasi serta materi;
2. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung;
3. Ibu Herlinawati, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Universitas Lampung;
4. Bapak Dr. Ing. Ardian Ulvan, S.T., M.Sc. selaku Dosen Pembimbing Utama yang telah memberikan ilmu, bimbingan, bantuan, masukan, dan pandangan serta berbagai pola pikir kepada penulis disetiap kesempatan dengan baik dan menarik;
5. Bapak Aryanto, S.T., M.T. selaku Dosen Pembimbing Pendamping yang telah memberikan ilmu, bimbingan, masukan, motivasi, dan sudut pandang lain serta berbagai pola pikir kepada penulis disetiap kesempatan dengan baik dan menarik;

6. Bapak Mona Arif Muda S.T., M.T. selaku Dosen Penguji yang telah memberikan kritik dan saran serta penyelesaian masalah yang membangun kepada penulis;
7. Bapak Sumadi, S.T., M.T. selaku Pembimbing Akademik yang telah memberikan arahan, nasehat dan bimbingan yang membangun bagi penulis selama menempuh perkuliahan;
8. Segenap dosen di Jurusan Teknik Elektro yang telah memberikan ilmu yang bermanfaat kepada penulis selama menempuh pendidikan perkuliahan;
9. Hana Yunus Putri dan Dimas Danu Saputra selaku tim dari Penelitian Riset Sawi tatas segala dukungan, ilmu, kerja sama dan waktu yang telah diluangkan dalam penelitian ini, sehingga dapat selesai dengan baik dan lancar;
10. Segenap staff di Jurusan Teknik Elektro yang telah membantu penulis baik dalam hal administrasi dan lain-lain;
11. Keluarga Besar Unila Robotika dan Otomasi yang memberikan banyak ilmu, masukan dan saran kepada penulis yang tidak bisa dibayarkan;
12. Keluarga Besar Rumah Baja yang telah membantu, memberi saran, inspirasi, dan canda tawa kepada penulis dalam menyelesaikan skripsi ini;
13. Seorang wanita cantik dari Fakultas MIPA yang selalu memberi semangat kepada penulis serta selalu membantu penulis dalam segala aspek;
14. Lelingku Teknik Elektro'19, terimakasih atas semangat dan kebersamaannya selama menempuh pendidikan di Jurusan Teknik Elektro;
15. Keluarga Besar HIMATRO yang telah memberikan ilmu dan pengalaman kepada penulis selama pendidikan baik secara langsung maupun tak langsung;
16. Seluruh pihak yang telah membantu penulis menyelesaikan skripsi ini.

Penulis mengakui adanya kekurangan dalam skripsi ini dan dengan tulus menerima kritik serta saran yang membangun dari berbagai pihak demi kemajuan bersama. Semoga skripsi ini bermanfaat bagi penulis dan pembaca.

Bandar Lampung, Oktober 2023  
Penulis

Muhammad Juan Pradana

## DAFTAR ISI

	Halaman
<b>DAFTAR TABEL</b> .....	x
<b>DAFTAR GAMBAR</b> .....	xi
<b>I. PENDAHULUAN</b> .....	2
1.1. Latar Belakang dan Masalah .....	2
1.2. Tujuan Penelitian.....	3
1.3. Rumusan Masalah .....	3
1.4. Batasan Masalah.....	4
1.5. Manfaat Penelitian.....	4
1.6. Hipotesis.....	5
1.7. Sistematika Penulisan.....	5
<b>II. TINJAUAN PUSTAKA</b> .....	7
2.1. Penelitian Terdahulu .....	7
2.2. Gas Rumah Kaca.....	9
2.3. Suhu dan Kelembaban Udara.....	11
2.4. <i>Monitoring</i> .....	11
2.5. <i>Eddy Covariance</i> .....	12
2.6. Fluks Karbon .....	14
2.7. <i>Eddy Station</i> .....	15
2.8. <i>Internet of Things (IoT)</i> .....	15

2.9.	<i>Arduino Portenta H7</i> .....	15
2.10.	<i>Portenta Breakoutboard</i> .....	19
2.11.	Sensor Gas Metana.....	21
2.12.	Sensor Suhu dan Kelembaban.....	23
2.13.	Sensor Tekanan Udara .....	24
2.14.	<i>DHT22</i> .....	26
2.15.	<i>Arduino IDE</i> .....	28
2.16.	<i>Raspberry Pi 4 Model B 2 GB Memory</i> .....	28
2.17.	<i>Node JS</i> .....	30
2.18.	<i>Node Red</i> .....	30
2.19.	<i>EPEVER XTRA 4415N</i> .....	30
2.20.	<i>2 Way RS485 to TTL Converter</i> .....	32
2.21.	<i>Zerotier</i> .....	33
2.22.	<i>D-Link DES-1008C</i> .....	34
2.23.	<i>REST API</i> .....	35
2.24.	<i>Postgresql</i> .....	35
2.25.	<i>Expressjs</i> .....	36
2.26.	<i>Github</i> .....	36
2.27.	<i>Dashboard</i> .....	37
2.28.	<i>Grafana</i> .....	37
<b>III.</b>	<b>METODOLOGI PENELITIAN</b> .....	<b>38</b>
3.1.	Waktu dan Tempat Penelitian .....	38
3.2.	Komponen dan Perangkat Lunak Penelitian .....	38

3.3.	Rancangan Perangkat dan Sistem .....	45
3.3.1.	Pengujian Tiap Sensor.....	46
3.3.2.	Perangkat <i>Monitoring</i> Karbon dari Gas Metana .....	50
3.3.3.	Perangkat <i>Data Logger</i> dan <i>SCC Data Parser</i> .....	53
3.3.4.	<i>Platform Internet of Things</i> .....	56
3.4.	Diagram Alir Penelitian .....	56
3.4.1.	Diagram Alir Sistem <i>Monitoring</i> Karbon dari Gas Metana dan <i>SCC Data Parser</i> .....	57
3.4.2.	Diagram Alir Sistem <i>Data Logger</i> .....	58
3.4.3.	Diagram Alir Sistem <i>Platform IoT</i> .....	60
3.5.	Diagram Blok Perangkat dan Sistem .....	61
3.6.	Skenario <i>Monitoring Gas CH<sub>4</sub>, Data Logger, dan Solar System</i> .....	63
3.6.1.	Analisa Kemampuan Latensi Sensor .....	64
3.6.2.	Integrasi Sensor dengan <i>Arduino Portenta H7</i> .....	64
3.6.3.	Pengambilan Data Sensor .....	64
3.6.4.	Pengambilan Data <i>SCC</i> dan Kondisi <i>Data Logger</i> .....	65
3.6.5.	Proses Unggah Data <i>Carbon Monitoring</i> dan Menjaga Kesehatan <i>Data Logger</i> .....	65
3.6.6.	Akses Data pada <i>Data Logger</i> Menggunakan <i>FTP</i> dan <i>Zerotier</i> ...	65
3.6.7.	Penerimaan Data di <i>Platform IoT</i> .....	66
3.6.8.	Visualisasi Data <i>Monitoring</i> .....	66
3.7.	Perancangan Perangkat dan Sistem.....	66
3.7.1.	Perancangan <i>PCB</i> untuk Sensor <i>SHT85</i> .....	66



3.7.2.	Perancangan Perangkat <i>Monitoring</i> Karbon dari Gas Metana.....	67
3.7.3.	Perancangan Perangkat <i>Data Logger</i> .....	68
3.7.4.	Perancangan <i>SCC Data Parser</i> .....	72
3.7.5.	Pengembangan <i>Platform IoT</i> .....	73
<b>IV.</b>	<b>HASIL DAN PEMBAHASAN</b> .....	<b>77</b>
4.1.	Pengujian Latensi Setiap Sensor .....	77
4.1.1.	Sensor gas Metana.....	77
4.1.2.	Sensor Barometrik.....	78
4.1.3.	Sensor Suhu dan Kelembaban.....	79
4.1.4.	<i>DHT22</i> .....	79
4.2.	Pengujian Osiloskop.....	80
4.2.1.	Sensor <i>DHT22</i> .....	80
4.2.2.	Sensor Barometrik.....	81
4.3.	<i>Trial and Error</i> .....	82
4.3.1.	Sensor Suhu dan Kelembaban Memerlukan Komponen Tambahan.....	82
4.3.2.	<i>Memory Portenta H7</i> Terlalu Kecil untuk Membaca <i>Payload</i> .....	84
4.3.3.	<i>RTC Breakout Board</i> Tidak Menjaga Waktu.....	85
4.3.4.	Sensor Barometrik dan Sensor Suhu Kelembaban Gagal Beroperasi Ketika <i>Ground</i> Tidak Disatukan .....	86
4.3.5.	<i>Controller</i> Gagal Melakukan Persiapan.....	88
4.3.6.	<i>Data Logger</i> Tiba-Tiba Tidak Mau Menerima Data.....	88
4.3.7.	<i>Resoure Server 1 Core 2 GB</i> Tidak Memumpuni.....	88
4.4.	Hasil Perancangan Sistem <i>Monitoring</i> Gas Metana.....	90

4.5.	Evaluasi Performa Perangkat <i>Monitoring</i> Gas Metana.....	91
4.5.1.	Performa Sensor.....	91
4.5.2.	Kecepatan <i>Sensing</i> Data.....	94
4.5.3.	Jumlah Data dan Kecocokan <i>Timestamp</i> .....	95
4.6.	<i>Data Logger</i> .....	97
4.6.1.	Pengujian <i>Data Lost Data Logger</i> .....	97
4.6.2.	Implementasi <i>Data Logger</i> .....	98
4.7.	Integrasi Perangkat <i>Monitoring</i> dan <i>Data Logger</i> .....	99
4.8.	Implementasi <i>SCC Data Parser</i> .....	101
4.9.	Pengukuran Kondisi <i>Data Logger</i> .....	103
4.10.	<i>Platform IoT</i> .....	103
4.10.1.	Penggunaan <i>Database Postgresql</i> .....	103
4.10.2.	Pembangunan <i>Backend</i> .....	104
4.10.3.	<i>Dashboard</i> .....	106
<b>V.</b>	<b>KESIMPULAN</b> .....	108
5.1.	Kesimpulan.....	108
5.2.	Saran.....	109
	<b>DAFTAR PUSTAKA</b> .....	110
	<b>LAMPIRAN</b> .....	112

## DAFTAR TABEL

Tabel		Halaman
1.	Spesifikasi Mikrokontroler <i>Arduino Portenta H7</i> .....	18
2.	Spesifikasi <i>Breakout Board Portenta</i> .....	20
3.	Spesifikasi Sensor Gas metana .....	21
4.	Spesifikasi Sensor <i>SHT85</i> .....	24
5.	Spesifikasi Sensor Barometrik.....	25
6.	Spesifikasi Sensor <i>DHT22</i> .....	27
7.	Komponen dan Perangkat Lunak Perangkat <i>Monitoring Gas Metana</i> ...	39
8.	<i>Data Logger dan SCC Data Parser</i> .....	41
9.	<i>Platform IoT</i> .....	44
10.	Komunikasi Perangkat <i>Monitoring Karbon dari Gas Metana</i> .....	52
11.	Pin <i>Wiring SCC Data Parser</i> .....	55
12.	Jumlah Data Harian Perangkat Eddy Station.....	92
13.	<i>Missing Value</i> Data Perangkat <i>Monitoring 1</i> .....	92
14.	<i>Missing Value</i> Data Perangkat <i>Monitoring 2</i> .....	93
15.	Sampel Data Timestamp Kedua Perangkat Eddy Station.....	96
16.	<i>Data Lost</i> .....	97
17.	Sampel Data CSV dari Perangkat <i>Monitoring Gas Karbon 1</i> .....	100

## DAFTAR GAMBAR

Gambar	Halaman
1. Skema Pengukuran Menara Fluks sebagai Bagian dari Metode <i>Eddy Covariance</i> .....	13
2. Fluks Karbon diantara Stok Karbon. ....	14
3. Mikrokontroler <i>Arduino Portenta H7</i> . ....	17
4. <i>Portenta Breakout Board</i> . ....	20
5. Sensor Gas Metana. ....	21
6. Sensor Suhu dan Kelembaban. ....	23
7. Sensor Barometrik. ....	25
8. Sensor <i>DHT22</i> . ....	27
9. <i>Raspberry Pi 4 Model B</i> . ....	29
10. <i>EPEVER XTRA 4415N</i> . ....	31
11. <i>2 Way RS485 to TTL Converter</i> . ....	32
12. <i>D-Link DES-1008C</i> . ....	34
13. Tampilan <i>Grafana</i> . ....	37
14. Rangkaian Pengujian Sensor Gas Metana. ....	47
15. Rangkaian Pengujian Sensor Suhu dan Kelembaban. ....	48
16. Rangkaian Pengujian Sensor Tekanan Udara. ....	49
17. Rangkaian Pengujian Sensor <i>DHT22</i> . ....	49
18. Skema Kelistrikan untuk Komponen Perangkat <i>Monitoring Gas Metana</i> . .....	51
19. Rangkaian Komunikasi Perangkat <i>Monitoring Gas Metana</i> . ....	52
20. Konektivitas <i>Data Logger</i> dengan Sistem <i>Monitoring Karbon dan Internet</i> . ....	54
21. Rangkaian <i>SCC Data Parser</i> . ....	55

22.	Desain Sistem <i>Platform IoT</i> .....	56
23.	Diagram Alir Perangkat <i>Monitoring Karbon dari Gas Metana dan SCC Data Parser</i> .....	58
24.	Diagram Alir Sistem <i>Data Logger</i> .....	59
25.	Diagram Alir Sistem <i>Platform IoT</i> . ....	61
26.	Diagram Blok Sistem <i>Monitoring Karbon dari Gas Metana, Data Logger dan Platform IoT</i> .....	62
27.	Latensi Sensor Gas Metana.....	78
28.	Latensi Sensor Barometrik.....	78
29.	Latensi Sensor Suhu dan Kelembaban.....	79
30.	Pembacaan Osiloskop <i>Idle Condition Sensor DHT22</i> .....	80
31.	Pembacaan Osiloskop <i>Operation Condition Sensor DHT22</i> .....	81
32.	Pembacaan Osiloskop <i>Operation Condition Barometrik SDA</i> .....	82
33.	(a) Desain; (b) Implementasi <i>PCB</i> untuk Sensor Suhu dan Kelembaban. .....	83
34.	Sensor Suhu dan Kelembaban Bekerja dengan <i>PCB</i> . ....	84
35.	Percobaan Pengiriman <i>File</i> dengan <i>Portenta H7</i> .....	85
36.	Fitur <i>RTC Breakout Board Abnormal</i> .....	86
37.	<i>Ground Suhu Kelembaban, Baro dan Portenta H7</i> Tidak Saling Terhubung.....	87
38.	<i>Ground Suhu Kelembaban, Baro, dan Portenta H7</i> Saling Terhubung..	87
39.	<i>Server Overload</i> . ....	89
40.	Hasil Perancangan Sistem <i>Monitoring Gas Metana</i> . ....	90
41.	Jumlah Data Hasil.....	91
42.	Kecepatan <i>Sensing Data Eddy Station</i> . ....	94
43.	Perbedaan Jumlah Data <i>Eddy Station 1 dan 2</i> . ....	95
44.	<i>Flow Data Logger</i> .....	98
45.	Pengunggahan <i>File</i> Terjadwal dari <i>Data Logger</i> .....	99

46.	<i>Flow</i> Pembersihan File Lama. ....	99
47.	Penyimpanan <i>Data Logger</i> . ....	100
48.	<i>Flow Node-RED SCC Data Parser</i> . ....	101
49.	Data dari <i>SCC Data Parser</i> . ....	102
50.	Hasil Instalasi <i>SCC Data Parser</i> . ....	102
51.	Pengukuran Kondisi <i>Data Logger</i> . ....	103
52.	<i>Database</i> pada <i>Server</i> . ....	104
53.	Struktur Proyek <i>Backend</i> . ....	105
54.	Pengujian <i>REST API</i> untuk Mendapatkan Satu Data <i>SCC</i> . ....	106
55.	<i>Dashboard</i> Hasil. ....	107

## I. PENDAHULUAN

### 1.1. Latar Belakang dan Masalah

Indonesia adalah negara penghasil minyak kelapa sawit terbesar pertama di dunia dengan produksi 46,5 juta ton untuk estimasi tahun 2022 [1]. Kelapa sawit merupakan tanaman penting yang dimanfaatkan sebagai penghasil minyak untuk makanan, minyak industri ataupun sebagai bahan bakar *biodisel* (nabati). Produksi CPO (*Crude Palm Oil*) paling banyak diproduksi dari perkebunan besar swasta (PBS) kemudian disusul perkebunan rakyat (PR) serta perusahaan besar negara (PBN). Berdasarkan rata-rata produksi buah kelapa sawit per provinsi di Indonesia pada tahun 2015-2020, ada sembilan provinsi sebagai daerah penghasil kelapa sawit paling besar di Indonesia dengan kalkulasi keseluruhan kontribusi sebesar 87,46% terhadap keseluruhan produksi buah kelapa sawit Indonesia. Pada tahun 2018 Provinsi Riau menduduki peringkat pertama dengan produksi 8,4 juta ton disusul oleh Provinsi Kalimantan Tengah dengan produksi 7,2 ton [2].

Banyaknya jumlah produksi kelapa sawit sebanding dengan luas lahan yang digunakan untuk perkebunan kelapa sawit. Pulau Sumatera menduduki posisi pertama sebagai pulau dengan lahan perkebunan kelapa sawit terbesar di Indonesia dengan luas lahan 7,95 juta hektar disusul oleh pulau Kalimantan dengan luas lahan 5,99 juta hektar [3]. Perkebunan kelapa sawit kerap dikaitkan dengan isu kenaikan pemanasan global, hal ini berkaitan dengan emisi karbon yang dihasilkan baik dari tumbuhan kelapa sawit maupun dari lahan gambut yang dipakai sebagai lahan pertanian.

Ada beberapa gas karbon yang dihasilkan dari emisi karbon kelapa sawit seperti Karbon Dioksida ( $CO_2$ ) dan Metana ( $CH_4$ ). Luas perkebunan kelapa sawit sangat mempengaruhi banyaknya emisi karbon yang dihasilkan.

$CH_4$  termasuk gas rumah kaca (GRK) yang penting karena memiliki daya absorpsi *infrared* yang kuat serta kehadirannya di atmosfer semakin lama semakin meningkat sehingga berkontribusi dalam kenaikan pemanasan global [4]. Gas metana ( $CH_4$ ) adalah gas rumah kaca yang memiliki potensi pemanasan global 28 kali lebih besar dari karbon dioksida ( $CO_2$ ) [5]. Hampir 70% emisi gas metana ( $CH_4$ ) berasal dari sumber-sumber antropogenik serta dua per tiganya asalnya dari aktivitas pertanian [6]. Salah satu sumber utama emisi gas metana adalah dari perkebunan sawit, di mana limbah cair diolah dan menghasilkan gas metana. Emisi gas metana yang dihasilkan oleh perkebunan sawit dapat menjadi masalah lingkungan jika tidak dikelola dengan baik.

*Eddy Covariance* merupakan metode pengukuran *flux* karbon yang umum digunakan untuk mengukur laju pertukaran karbon antara tanah dan atmosfer. Dalam pengukuran *Eddy Covariance*, sensor gas metana digunakan untuk mengukur konsentrasi gas metana di atmosfer, sedangkan sensor *3D Sonic Anemometer* digunakan untuk mengukur kecepatan dan arah angin. Kedua sensor tersebut menghasilkan data yang kemudian diolah oleh sistem untuk menghitung *flux* karbon.

Berdasarkan uraian di atas, penelitian ini merancang sebuah sistem *monitoring* karbon dari gas Metana ( $CH_4$ ) berbasis *Internet of Things*. Sistem ini menggunakan mikrokontroler *Arduino Portenta H7* dengan sensor gas metana sebagai sensor gas dan *Raspberry pi 4 model B* sebagai *data logger*-nya. Data yang didapatkan nantinya bisa dimanfaatkan untuk pengaplikasian metode *Eddy Covariance* pada analisis fluks karbon. Namun, proses pengukuran menggunakan *Eddy Covariance* memiliki beberapa kendala, di antaranya sulitnya memperoleh data secara *real-time* dan *monitoring* yang tidak efisien. Oleh karena itu, perlu dilakukan perancangan sistem *monitoring flux* karbon berbasis *Internet of Things (IoT)* yang dapat mengatasi kendala tersebut.



## 1.2. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah sebagai berikut.

1. Merancang sistem *monitoring* karbon dari gas Metana ( $CH_4$ ) menggunakan sensor gas metana, sensor tekanan udara, dan sensor suhu dan kelembaban.
2. Membuat sistem yang mengintegrasikan sensor gas metana, sensor tekanan udara, dan sensor suhu dan kelembaban dengan mikrokontroler *Arduino Portenta H7* dan sistem *IoT*.
3. Membuat *Data Logger* untuk perangkat *Eddy Station* menggunakan *Raspberry Pi 4B*.
4. Melaporkan dan menganalisa masalah pada perangkat sistem *monitoring* dan *data logger* pada saat perancangan sistem.

## 1.3. Rumusan Masalah

Adapun rumusan masalah pada penelitian ini adalah sebagai berikut.

1. Bagaimana merancang sistem *monitoring* karbon dari gas metana dengan menggunakan sensor gas metana, sensor tekanan udara, dan sensor suhu dan kelembaban?
2. Bagaimana mengintegrasikan sensor gas metana, sensor tekanan udara, dan sensor suhu dan kelembaban dengan mikrokontroler *Arduino Portenta H7* dan sistem *IoT*?
3. Bagaimana merancang suatu sistem *Data Logger* menggunakan *Raspberry Pi 4B* untuk menampung data hasil dari perangkat *Eddy Station*.
4. Bagaimana merancang perangkat sistem *monitoring* dan *data logger* supaya bekerja dengan baik dan dapat dikembangkan lebih lanjut.

#### 1.4. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah sebagai berikut.

1. Mikrokontroler yang digunakan adalah mikrokontroler *Arduino* seri *Portenta H7* bersama *portenta breakoutboard* dengan *ethernet* sebagai media transmisi.
2. Module komponen yang digunakan antara lain sensor gas metana, *DHT22*, sensor suhu dan kelembaban, sensor tekanan udara, *Buck Boost*, *2 Way RS485 to TTL Converter*, *Switch D-Link DES-1008*, *Raspberry Pi 4B*, dan *Power Supply*.
3. Alat yang telah dibuat diuji di sekitar Laboratorium Terpadu Teknik Elektro.
4. Berfokus pada perancangan perangkat *monitoring* dan *data logging*.
5. Hanya membahas hingga operasional perangkat sehingga didapatkan data-data yang diperlukan untuk dilakukan pengukuran fluks karbon tanpa perhitungan fluks karbon.
6. Hanya mengembangkan bagian *monitoring* gas Metana ( $CH_4$ ), bagian *monitoring* gas  $CO_2$  dikembangkan oleh Hana Yunus Putri.

#### 1.5. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah menghasilkan perangkat yang dapat melakukan *monitoring* karbon dari gas Metana ( $CH_4$ ) dengan data hasil *monitoring* yang dapat disimpan pada *data logger* dan *cloud* supaya dimanfaatkan untuk berbagai *platform*. Data hasil *monitoring* bisa digunakan untuk menerapkan metode *Eddy Covariance* sebagai analisis fluks karbon.

## 1.6. Hipotesis

Pada penelitian ini, perangkat dapat melakukan *monitoring* karbon dari gas Metana ( $CH_4$ ) dan menyimpannya pada *data logger* serta *cloud*. Satuan untuk data gas Metana adalah *PPM (Part Per Million)*. Data hasil *monitoring* akan secara otomatis diunggah ke cloud melalui jaringan internet supaya bisa dimanfaatkan lebih lanjut.

## 1.7. Sistematika Penulisan

Adapun sistematika penulisan yang digunakan dalam penelitian ini adalah untuk memberikan gambaran sederhana tentang pembahasan skripsi dan memudahkan pemahaman terhadap materi penelitian yang ditulis dalam beberapa bab, adalah sebagai berikut.

### **I. PENDAHULUAN**

Bab ini menjelaskan tentang latar belakang masalah, tujuan penelitian, perumusan masalah, batasan masalah, manfaat penelitian, hipotesis serta sistematika penulisan pada penelitian ini.

### **II. TINJAUAN PUSTAKA**

Bab ini menjelaskan tentang teori-teori yang mendukungnya, yang merupakan pengantar pemahaman dan mengacu pada bahan penelitian dari berbagai sumber ilmiah seperti buku dan jurnal.

### **III. METODOLOGI PENELITIAN**

Bab ini menjelaskan metodologi penelitian, termasuk waktu dan tempat pelaksanaan, alat dan bahan yang digunakan untuk melakukan penelitian, serta metode dan diagram penelitian yang digunakan untuk melengkapi rancangan akhir penelitian.

#### **IV. HASIL DAN PEMBAHASAN**

Bab ini menjelaskan hasil dan menganalisa data hasil yang dapat dari percobaan ataupun simulasi yang telah dilakukan pada penelitian tugas akhir.

#### **V. KESIMPULAN**

Bab ini berisi kesimpulan yang didapat dari hasil analisis dan pembahasan serta terdapat saran yang dapat diberikan.

## II. TINJAUAN PUSTAKA

### 2.1. Penelitian Terdahulu

Penelitian sebelumnya dilakukan oleh Arfian Dzaki Danurwenda dari Program Studi Teknik Elektro Universitas Pembangunan Panca Budi Medan pada tahun 2020 dengan judul “Rancang Bangun Alat Pengukuran Gas Metana Menggunakan *Arduino Uno* pada *Septic Tank* [7]”. Penelitian ini mengukur kadar gas Metana menggunakan sensor *MQ-4* dengan *Arduino Uno* sebagai mikrokontrolernya. Data pengukuran yang didapatkan hanya ditampilkan pada *LCD (Liquid Crystal Display)* 16x2. Penelitian ini bertujuan untuk membuat dan merancang alat pendeteksi gas Metana pada septic tank menggunakan *Arduino Uno* sebagai pengendali sensor *MQ-4* dan menampilkannya pada *LCD (Liquid Crystal Display)*.

Perbedaan antara penelitian yang dilakukan Arfian Dzaki Danurwenda dan penelitian yang dilakukan penulis adalah pada penelitian ini menggunakan jenis mikrokontroler *Arduino Uno*, sensor *MQ-4* untuk *monitoring* gas Metana, dan menampilkan data pada *LCD (Liquid Crystal Display)* serta perangkat belum terhubung ke internet. Sedangkan pada penelitian yang penulis lakukan menggunakan mikrokontroler *Arduino Portenta H7*, sensor gas metana untuk *monitoring* gas Metana, dan Perangkat berbasis *IoT (Internet of Things)*.

Kesimpulan dari penelitian yang dilakukan oleh Arfian Dzaki Danurwenda adalah telah terealisasi suatu sistem rancang bangun alat pengukuran gas metana pada *septic tank* dengan sensor *MQ-4* dan *Arduino Uno* sebagai sistem kendali dengan keadaan dapat bekerja dengan baik.

Selanjutnya terdapat penelitian yang dilakukan oleh Kevin Diantoro dari Jurusan Teknik Elektro Universitas Singaperbangsa Karawang pada tahun 2020 dengan judul “Implementasi Sensor *MQ 4* dan Sensor *DHT 22* pada Sistem Kompos Pintar Berbasis *IoT* (SIKOMPI) [8]”. Penelitian ini mengukur suhu dan kelembaban tong saat dilakukan pembuatan pupuk kompos menggunakan sensor *DHT 22*, sensor *MQ-4* untuk *monitoring* kadar gas Metana yang diproduksi ketika pembuatan pupuk kompos, dan *Wemos D1 R1* sebagai mikrokontrolernya karena menerapkan *IoT*. Tujuan dari penelitian ini adalah untuk memudahkan pengguna menyiapkan pupuk kompos organik. Alat ini juga dilengkapi dengan teknologi *Internet of Things (IoT)* untuk memungkinkan pengguna memantau suhu, kelembapan, dan kondisi metana dengan mudah selama pengomposan.

Perbedaan antara penelitian Kevin Diantoro dan penelitian yang dilakukan penulis adalah pada penelitian ini menggunakan jenis mikrokontroler *Wemos D1 R1* dan jenis sensor gas Metana yang dipakai adalah sensor *MQ-4* sedangkan pada penelitian yang penulis lakukan menggunakan jenis mikrikontroler *Arduino Portenta H7* dan jenis sensor gas Metana yang dipakai berbeda.

Kesimpulan dari penelitian yang dilakukan oleh Kevin Diantoro ini adalah Sistem kontrol yang dirancang berjalan dengan baik, *error* yang didapatkan masih dibawah batas standar pembacaan error maksimal yaitu 5%.

Terdapat juga penelitian yang dilakukan oleh Abas Sato dari Program Studi Teknik Kimia Universitas WR Supratman Surabaya pada tahun 2021 dengan judul “Penggunaan *MQ-5* sebagai Sensor yang Murah dengan Akurasi Tinggi untuk Pengukuran Konsentrasi Gas Metana dalam Biogas [9]”. Penelitian ini dilakukan di Laboratorium TekniK Kimia Universitas WR Supratman Surabaya. Tujuan dari penelitian ini adalah mendapat alat pengukuran yang lebih murah dan tidak banyak membutuhkan sampel gas dalam jumlah besar jika dibandingkan dengan menggunakan gas kromatografi (*GC*).

Perbedaan antara penelitian Abas Sato dan penelitian yang dilakukan penulis adalah penelitian ini menggunakan mikrokontroler *Arduino Uno* dan sensor gas Metana *MQ-5*. Data gas yang didapat ditampilkan pada layar komputer melalui komunikasi

serial. Sedangkan penelitian yang penulis lakukan menggunakan *Arduino Portenta H7* sebagai mikrokontroler dan sensor gas Metana berbeda. Karena perangkat yang dibuat penulis terhubung dengan internet maka data yang didapat bisa diakses jarak jauh.

Kesimpulan dari penelitian yang dilakukan oleh Abas Sato adalah dengan sensor *MQ-5*, perangkat hemat biaya untuk mengukur kandungan metana dari sampel biogas dapat dirakit. Kelembaban yang dapat mempengaruhi sensitivitas sensor dapat diatasi dengan mengeringkan sampel gas. Alat ini telah terbukti dapat mengukur kandungan metana biogas secara akurat dengan perbedaan kurang dari 5% dibandingkan dengan pengukuran GC.

## **2.2. Gas Rumah Kaca**

Matahari menghasilkan energi dalam jumlah besar, dan sebagian besar energi tersebut dipantulkan kembali ke permukaan bumi dalam bentuk radiasi infra merah. Tidak semua energi yang dipancarkan permukaan bumi kembali ke luar angkasa karena sebagian besar radiasi infra merah diserap kembali oleh gas atmosfer. Karena gas dapat berinteraksi dengan energi, maka gas di atmosfer pada saat terkena energi dari matahari memancarkan kembali radiasi inframerah ke permukaan bumi. Proses ini dikenal sebagai efek rumah kaca. Efek rumah kaca merupakan sebuah kondisi dimana suhu dari permukaan benda langit seperti planet dan bintang, meningkat drastis [10]. Beberapa gas rumah kaca yang dapat mempengaruhi suhu permukaan bumi antara lain:

1.  $CO_2$  (Karbon Dioksida)

$CO_2$  adalah gas rumah kaca utama yang menyebabkan pemanasan global dan terakumulasi di atmosfer akibat aktivitas manusia. Efek dari peningkatan karbon dioksida di atmosfer adalah peningkatan suhu permukaan bumi, kenaikan permukaan laut, anomali iklim, munculnya berbagai penyakit pada manusia dan hewan.

2.  $H_2O$  (Uap Air)

Uap air merupakan penyumbang utama efek rumah kaca. Uap air tidak terlihat dan harus dibedakan dari awan dan kabut, yang terbentuk ketika uap membentuk tetesan air. Jumlah uap air di atmosfer berada di luar kendali manusia dan terutama dipengaruhi oleh suhu bumi. Saat bumi menghangat, jumlah uap air di atmosfer akan meningkat karena meningkatnya laju penguapan.

3.  $CH_4$  (Metana)

Metana terbentuk ketika jenis mikroorganisme tertentu memecah bahan organik dalam kondisi anaerob. Gas ini juga diproduksi secara alami ketika biomassa dipecah di rawa-rawa, oleh karena itu disebut juga gas rawa. Metana mudah terbakar dan menghasilkan karbon dioksida sebagai produk sampingan.

4.  $CFC$  (*Chloro Fluoro Carbon*)

Klorofluorokarbon adalah sekelompok gas buatan manusia. Senyawa CFC tidak mudah terbakar dan tidaklah beracun. CFC sangat stabil sehingga dapat digunakan di berbagai perangkat. Senyawa CFC menghasilkan efek pemanasan hingga seribu kali dibandingkan dengan  $CO_2$ .

5.  $O_3$  (Ozon)

Ozon terjadi secara alami di atmosfer (troposfer, stratosfer). Di troposfer, ozon merupakan produk sampingan polutan yang terbentuk saat sinar matahari bereaksi dengan gas buang mobil, atau bisa juga berasal dari hidrokarbon dan nitrogen dioksida. Ozon di troposfer dapat membahayakan kesehatan manusia, hewan, dan tumbuhan.

6.  $N_2O$  (Nitrogen Oksida)

Nitrogen oksida merupakan gas yang konsentrasinya di atmosfer relatif meningkat akibat meningkatnya kegiatan pertanian. Nitrogen oksida berasal dari bakteri yang memetabolisme berbagai nitrat ( $NO_3^-$ ) dan amonia ( $NH_4^+$ ) dalam pupuk.



### 2.3. Suhu dan Kelembaban Udara

Suhu merupakan ukuran kuantitatif dari temperatur atau panas dan dingin jika diukur menggunakan termometer. Suhu juga menunjukkan derajat panas suatu benda, sehingga jika suhu semakin naik maka benda tersebut akan semakin panas.

Kelembaban adalah konsentrasi uap air di udara. Kelembaban menggambarkan kadar uap air di udara, yang dapat dinyatakan sebagai kelembaban absolut, kelembaban relatif (nisbi), atau defisit tekanan uap air. Kelembaban mutlak adalah konsentrasi uap air (dapat dinyatakan dalam massa atau tekanan uap air) per satuan volume. Kelembaban relatif membandingkan konsentrasi/tekanan uap air aktual dengan keadaan jenuhnya, atau kemampuan udara untuk menahan uap air. Kemampuan udara menahan uap air (saat jenuh) ditentukan oleh suhu udara. Sedangkan defisit tekanan uap air merupakan selisih antara tekanan uap jenuh dengan tekanan uap aktual atau sebenarnya [11].

Tekanan uap air adalah tekanan total (tekanan parsial) uap air di udara, biasanya satuan pengukuran tekanan uap air adalah *pascal* (Pa). Tekanan uap air jenuh adalah tekanan uap air maksimum yang dapat dicapai udara pada suhu tertentu. Tekanan uap air ditentukan oleh *water vapor density* (kerapatan uap air) dan suhu udara. Jumlah uap air yang terkandung per satuan volume udara dapat disebut sebagai massa jenis uap air atau kerapan uap air. Massa total uap air per satuan volume udara disebut *absolute humidity* (kelembaban absolut, biasanya dinyatakan dalam satuan kg/m<sup>3</sup>). Oleh karena itu tekanan uap air sama dengan *absolut himudity* atau sering disebut sebagai *vapor concetration* (konsentrasi uap air) [12].

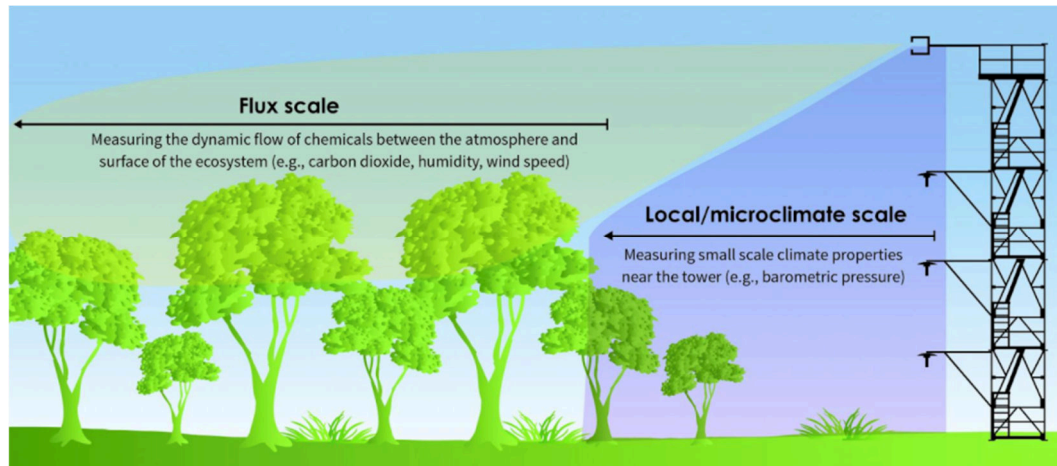
### 2.4. *Monitoring*

*Monitoring* didefinisikan sebagai suatu siklus kegiatan yang mencakup pengumpulan data, verifikasi, pelaporan, dan pelaksanaan informasi suatu proses yang sedang diimplementasikan [13]. *Monitoring* dapat memberikan informasi

tentang kesinambungan proses, sehingga langkah-langkah perbaikan berkelanjutan dapat dibuat. Dalam praktiknya, *monitoring* berlangsung selama proses sedang berjalan.

## 2.5. *Eddy Covariance*

Metode *Eddy Covariance* memberikan pengukuran emisi dan penyerapan gas, dan juga memungkinkan pengukuran fluks dari *sensible heat*, *latent heat*, dan momentum yang terintegrasi di suatu area [14]. *Sensible heat* adalah bentuk energi panas yang diserap atau dilepaskan oleh sistem atau dengan kata lain perubahan temperatur pada sistem disebabkan oleh *sensible heat*. *Latent heat* adalah jumlah energi yang diserap atau dilepas selama terjadinya transisi atau perubahan fase zat yang berlangsung dalam suhu konstan. Metode *Eddy Covariance* telah digunakan secara luas dalam mikrometeorologi selama lebih dari 30 tahun, tetapi sekarang, dengan metodologi yang lebih kuat dan instrumentasi yang lebih maju, metode ini dapat digunakan untuk semua disiplin ilmu, termasuk sains, industri, pemantauan lingkungan, dan inventarisasi. Metode ini bergantung pada pengukuran langsung dan sangat cepat dari transportasi gas aktual dengan kecepatan angin yang arahnya tiga dimensi secara *real time* pada lokasi yang sesungguhnya, menghasilkan perhitungan fluks turbulen di dalam *boundary layer* atmosfer. Daerah yang disebut *boundary layer* adalah daerah yang dimulai dari aliran yang langsung menempel pada permukaan dan berakhir pada titik dimana kecepatan aliran mendekati kecepatan *freestream* [15]. Lebih jelasnya, *boundary layer* adalah lapisan atau penghalang yang tidak terlihat dalam fluida yang digunakan sebagai konsep untuk menyederhanakan perhitungan mekanika fluida. Gambar 1 menunjukkan skema pengukuran menara fluks.



Gambar 1. Skema Pengukuran Menara Fluks sebagai Bagian dari Metode *Eddy Covariance*.

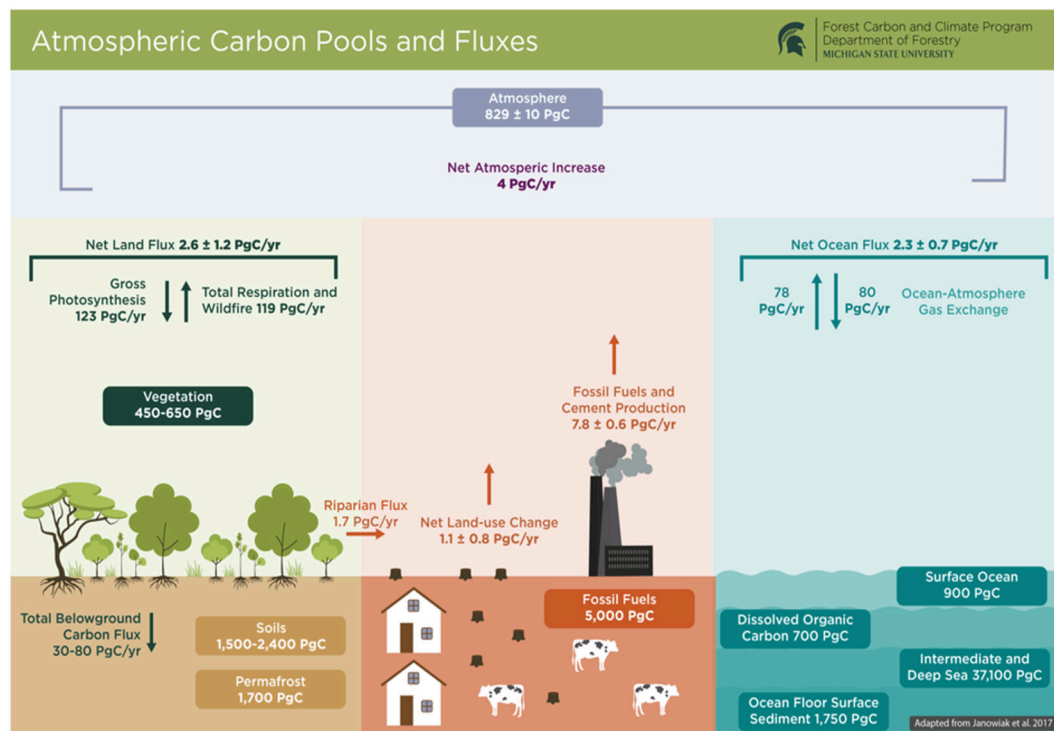
[Sumber: *Difuse*, <https://www.difuse.dartmouth.edu/> ]

*Eddy Covariance* bekerja dengan memonitor turbulensi udara di dekat permukaan tanah, dan dengan demikian mencatat variasi kecepatan angin, suhu, dan konsentrasi gas dalam interval waktu yang sangat kecil [16]. Data dari sensor-sensor ini kemudian diproses menggunakan teknik-teknik matematika dan statistik untuk menghasilkan perkiraan pertukaran gas antara permukaan dan atmosfer.

Supaya bisa diterapkan metode *Eddy Covariance*, maka harus disediakan beberapa data antara lain kecepatan udara tiga dimensi ( $u$ ,  $v$ ,  $w$ ) dan *sonic temperature* dari 3D *sonic anemometer*, konsentrasi gas  $CH_4$ , konsentrasi  $H_2O$  atau uap air, rata-rata suhu udara, kelembaban relatif, tekanan udara, arah angin, presipitasi, dan data lain yang mungkin dibutuhkan [17]. Secara umum, data frekuensi tinggi (10 atau 20 Hz) diukur pada ketinggian tertentu di atas permukaan atau atmosfer dengan menggunakan sensor cepat untuk komponen angin dan suhu udara (*3D Sonic Anemometer*) dan untuk skalar seperti konsentrasi gas (misalnya analisis gas Metana). Aliran *Eddy Covariance* diperoleh dari data frekuensi tinggi melalui beberapa tahap pemrosesan matematis yang kompleks [18].

## 2.6. Fluks Karbon

Fluks karbon mengacu pada jumlah karbon yang dipertukarkan antara stok karbon dengan karbon yang terlepas selama waktu tertentu. Secara sederhana, fluks karbon adalah pergerakan karbon baik di tanah, lautan, atmosfer, ataupun makhluk hidup. Ini menjelaskan mekanisme dibalik stok dan fluks karbon serta menyoroti dampaknya terhadap karbon di hutan. Gambar 2 menjelaskan mengenai fluks karbon hubungannya dengan stok karbon, terdapat karbon yang dilepas dan karbon yang diserap.



Gambar 2. Fluks Karbon diantara Stok Karbon.  
[sumber: Michigan State University, <https://www.canr.msu.edu/>]

## 2.7. *Eddy Station*

*Eddy station* adalah sebutan untuk perangkat *eddy covariance* yang terpasang pada tower di lapangan untuk melakukan pengukuran dan analisis fluks karbon dioksida ( $CO_2$ ) atau metana ( $CH_4$ ) dan uap air ( $H_2O$ ) antara permukaan bumi dan atmosfer. Perangkat ini bekerja dengan prinsip pengukuran pertukaran massa gas antara dua media, yaitu lapisan udara di atas permukaan tanah dan tanah itu sendiri. *Eddy station* dilengkapi dengan sensor khusus yang mampu mendeteksi perubahan cepat dan acak dalam konsentrasi gas di udara, serta kecepatan dan arah angin. Data yang dihasilkan oleh *eddy station* sangat berharga dalam pemahaman terhadap dinamika siklus karbon dan sirkulasi air di ekosistem tertentu, sehingga memberikan wawasan penting dalam studi mengenai perubahan iklim dan peran ekosistem dalam menyimpan karbon.

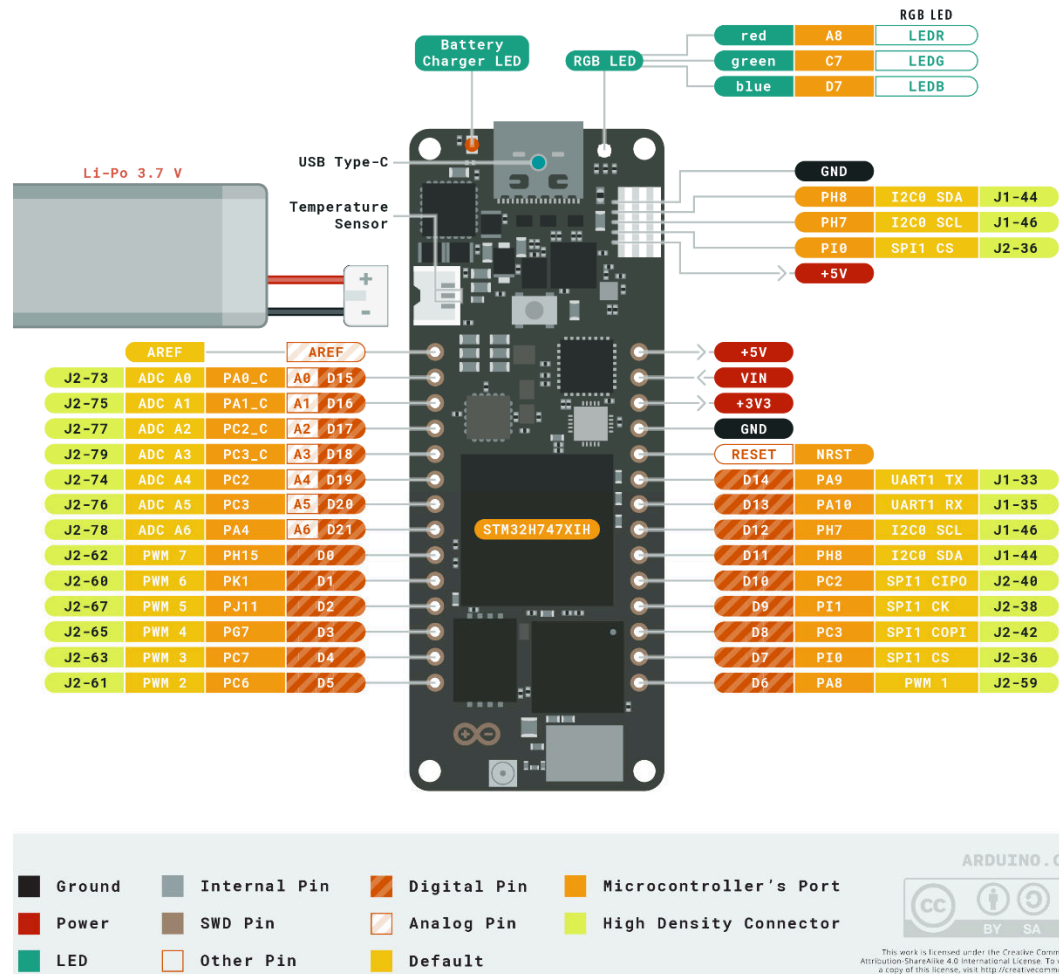
## 2.8. *Internet of Things (IoT)*

*Internet of Things* adalah suatu konsep sebuah jaringan yang luas diseluruh dunia dari suatu objek yang saling terhubung dan dapat dialamatkan secara unik berdasarkan *standard communication protocols* [19]. *IoT* memungkinkan objek dapat melakukan pekerjaan dan terhubung ke dalam jaringan tanpa interaksi langsung antar manusia maupun manusia langsung ke komputer. Penerapan *Internet of Things* untuk peralatan dapat memudahkan proses pengerjaan suatu perintah.

## 2.9. *Arduino Portenta H7*

*Arduino Portenta H7* merupakan sebuah perangkat *microcontroller* yang dikemas dalam bentuk susunan chip *IC (Integrated Circuit)* yang tergabung dalam suatu

*board* dan dirancang untuk melakukan tugas tertentu. *Portenta H7* mengikuti *Arduino MKR form factor*, tetapi ditingkatkan dengan konektor *high-density* keluarga *Portenta* 80 pin. *Portenta H7* bisa diprogram dengan bahasa tingkat tinggi dan menerapkan *Artificial Intelligence* dengan operasi berlatensi rendah pada perangkat kerasnya yang dapat dikustomisasi. *Portenta H7* secara simultan menjalankan kode tingkat tinggi bersama dengan *realtime*, karena memiliki dua prosesor yang dapat menjalankan tugas secara paralel. Misalnya, dimungkinkan untuk mengeksekusi kode yang dikompilasi *Arduino* bersama dengan *MicroPython* satu dan memiliki dua inti untuk berkomunikasi satu sama lain. Fungsionalitas *Portenta* adalah dua kali lipat, dapat berjalan seperti *board* mikrokontroler tertanam lainnya atau sebagai prosesor utama *embedded computer*. Misalnya, menggunakan *Portenta Vision Shield* untuk mengubah H7 menjadi kamera industri yang mampu menjalankan algoritma *realtime machine learning* pada pengambilan video *realtime*. Keunggulan *Portenta H7* jika disbanding dengan *STM32* yaitu *Form Factor* yang Kompatibel dengan *Arduino MKR Form Factor* yang Kompatibel dengan *Arduino MKR*. Selain itu *Portenta H7* mendukung berbagai *library* dan *toolset* yang sudah ada di ekosistem *Arduino*, sehingga memudahkan penggunaan dan pengembangan aplikasi. Dalam penelitian ini penulis menggunakan *Arduino Portenta H7* yang dapat dilihat pada gambar 3.



Gambar 3 Mikrokontroler *Arduino Portenta H7*.  
[sumber: *Arduino Portenta H7*, <https://docs.arduino.cc/>]

Terdapat banyak bagian dari mikrokontroler ini, namun bagian utamanya adalah sebagai berikut:

1. Prosesor Utama, adalah *dual core* Cortex M7 yang berjalan hingga 480 MHz dan *Cortex M4* yang berjalan hingga 240 MHz. Kedua inti berkomunikasi satu sama lain melalui kotak surat yang dibantu perangkat keras tempat *Arduino* mengimplementasikan mekanisme *Remote Procedure Call* yang memungkinkan pemanggilan fungsi pada prosesor lain secara mulus. Memori internal terdiri dari flash 2MB dengan dukungan *read and write* secara bersamaan dan *RAM 1MB*.

2. Modul *Wireless, on board* memungkinkan manajemen konektivitas *WiFi* dan *Bluetooth* secara simultan. Antarmuka *WiFi* dapat dioperasikan sebagai *Access Point*, sebagai Stasiun atau sebagai mode ganda *AP/STA* simultan dan dapat menangani kecepatan transfer hingga 65 Mbps. Antarmuka *Bluetooth* mendukung *Bluetooth Classic* dan *BLE (Bluetooth Low Energy)*.
3. *Crypto Chips, Portenta H7* menyediakan dua pilihan *Crypto Chip opsional* yang berbeda. *ATECC608A* klasik dari *Microchip* memberikan kompatibilitas *low cost backward* dengan aplikasi *Arduino* sementara *SE050C2* yang lebih canggih dari *NXP* memberikan sertifikasi *Common Criteria*, memori penyimpanan yang lebih besar, dan fungsionalitas yang lebih canggih.
4. *Port*, berfungsi sebagai *interface* untuk berbagai perangkat seperti LCD, sensor, memori dan perangkat lainnya ke mikrokontroler.

Untuk spesifikasi lebih lanjut dari mikrokontroler *Arduino Portenta H7* bisa dilihat pada tabel 1.

Tabel 1. Spesifikasi Mikrokontroler *Arduino Portenta H7*

<i>Features</i>	<i>Types</i>	<i>Availabilities</i>
<i>Microcontroller</i>	<i>STM32H747XI dual Cortex®-M7+M4 32bit low power Arm® MCU</i>	
<i>USB connector</i>	<i>USB-C</i>	
<i>Pins</i>	<i>Digital I/O Pins</i>	22
	<i>Analog input pins</i>	8
	<i>PWM pins</i>	10



Tabel 1. (Lanjutan)

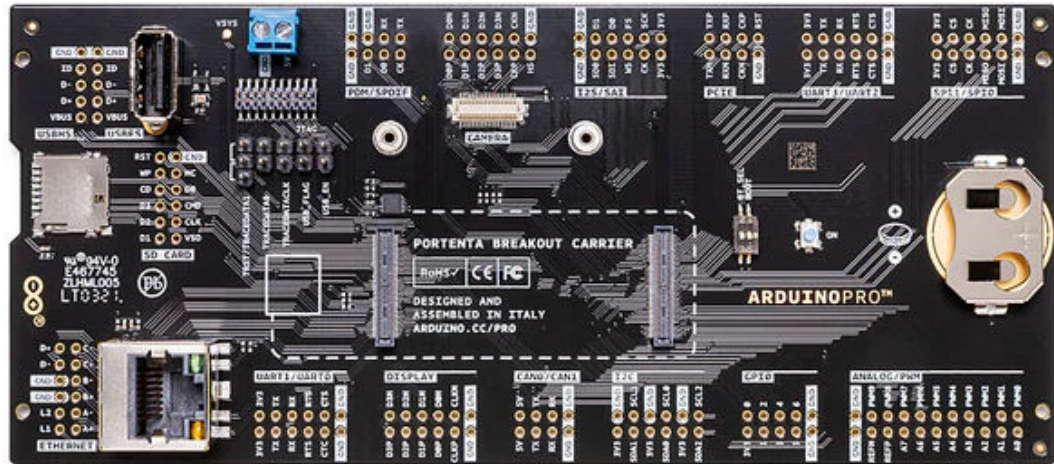
<i>Features</i>	<i>Types</i>	<i>Availabilities</i>
<i>Connectivity</i>	<i>Radio module</i>	<i>Murata 1DX dual WiFi 802.11b/g/n 65 Mbps and Bluetooth® 5.1 BR/EDR/LE</i>
	<i>Secure element</i>	<i>NXP SE0502</i>
<i>Communication</i>	<i>UART</i>	<i>Yes</i>
	<i>I2C</i>	<i>Yes</i>
	<i>SPI</i>	<i>Yes</i>
<i>Power</i>	<i>Circuit operating voltage</i>	<i>3.3V</i>
	<i>Input voltage (VIN)</i>	<i>5V</i>
	<i>DC Current per I/O Pin</i>	<i>8 mA</i>
<i>Clock speed</i>	<i>Main core</i>	<i>480 MHz</i>
	<i>Second core</i>	<i>240 MHz</i>
<i>Memory</i>	<i>ST STM32H747XI</i>	<i>2MB Flash, 1MB RAM</i>
<i>Dimensions</i>	<i>Width</i>	<i>25 mm</i>
	<i>Length</i>	<i>66 mm</i>

Sumber: *Arduino Portenta H7*, <https://docs.arduino.cc/>.

## **2.10. Portenta Breakoutboard**

*Arduino Portenta Breakout board* dirancang untuk membantu *hardware engineer and maker* untuk membuat prototipe dan membantu menguji koneksi perangkat dan kapasitas dalam *board* keluarga *Portenta* (misalnya *Portenta H7*). *Portenta Breakout* membuat semua sinyal konektor *high-density* dapat diakses satu per satu, membuatnya cepat dan mudah untuk menyambungkan dan menguji komponen dan perangkat perangkat keras eksternal seperti biasanya diperlukan selama pengembangan di lab. Penampakan *Breakout board Portenta* dapat dilihat pada

gambar 4. Untuk spesifikasi lebih lanjut dari *Breakout board Portenta* dapat dilihat pada tabel 2.



Gambar 4. *Portenta Breakout Board*.  
[sumber: *Portenta Breakout*, <https://store-usa.arduino.cc/>]

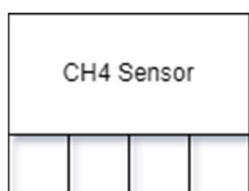
Tabel 2. Spesifikasi *Breakout Board Portenta*

<i>Features</i>	<i>Types</i>	
<i>USB port</i>	<i>USBA</i>	
<i>Ethernet</i>	<i>RJ45 Gbit</i>	
<i>Memory slot</i>	<i>Micro SD card</i>	
<i>Camera connector</i>	<i>OpenMV shutter module</i>	
<i>Debug</i>	<i>MIPI 20T JTAG with trace capability</i>	
<i>Connectors</i>	<i>HD male/female</i>	
<i>RTC power battery</i>	<i>CR2032</i>	
<i>Dimmensions</i>	<i>Weight</i>	69 mm
	<i>Widht</i>	72 mm
	<i>Length</i>	164 mm

Sumber: *Arduino Portenta Breakout*, <https://docs.arduino.cc/>.

## 2.11. Sensor Gas Metana

Sensor adalah suatu alat yang digunakan untuk mengubah perubahan fisika seperti tekanan, kelembaban, suhu dan lain-lain menjadi besaran listrik sehingga dapat dilakukan analisis. *Methane Gas Sensor* merupakan modul sensor yang digunakan untuk pengukuran kadar gas Metana di udara. Sensor gas inframerah adalah sensor miniatur *universal intelligent*, yang mengadopsi teori *NDIR (Nondispersive infrared sensor)* untuk mendeteksi konsentrasi  $CH_4$  di udara dan memiliki selektivitas yang baik, kinerja stabil, umur panjang, juga tidak bergantung pada Oksigen. Sensor suhu bagian dalam dapat digunakan untuk kompensasi suhu. Sensor gas inframerah mini ini dikembangkan dengan integrasi yang erat antara teknologi deteksi gas dengan inframerah, *micro machine workout*, dan desain sirkuit yang unggul. Gambar 5 merupakan penampakan modul sensor gas Metana ini. Diantara sensor *SGX Sensortech MiCS-6814*, *Figaro TGS2611*, *SPEC Sensors Ultra-Low Power Ppb-Level  $CH_4$*  dan *sensor MQ-5*, sensor gas metana yang dipakai penelitian ini adalah sensor yang memiliki pembacaan kadar gas metana yaitu hingga 0 ppm. Spesifikasi lebih lanjut ditampilkan oleh tabel 3.



Gambar 5. Sensor Gas Metana.

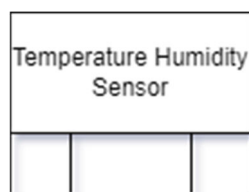
Tabel 3. Spesifikasi Sensor Gas metana

<i>Variables</i>	<i>Types</i>
<i>Detection Gas</i>	<i>Methane (<math>CH_4</math>)</i>

<i>Working Voltage</i>	3.6~5V DC
<i>Average Current</i>	<85mA
<i>Interface Level</i>	3V
<i>Detection range</i>	0~100%vol(selectable) (0-100.0000ppm)
<i>Output Signal</i>	UART
	0.4~2V DC
<i>Warm-up time</i>	3 min
<i>Response time</i>	T90<30s
<i>Working Temperature</i>	-20°C ~60°C
<i>Working Humidity</i>	0~95%RH(no condensation)
<i>Sizes</i>	Φ20×21.4mm
<i>Weight</i>	35g
<i>Lifetime</i>	>5 years
<i>Explosion-proof Mark</i>	Exmb II T6
<i>Defense Grade</i>	IP54

## 2.12. Sensor Suhu dan Kelembaban

Sensor suhu dan kelembaban adalah modul sensor yang digunakan untuk melakukan pengukuran suhu dan kelembaban udara. Sensor ini digunakan sebagai sensor yang mengambil data suhu dan kelembaban udara bebas bersamaan dengan sensor konsentrasi gas Metana. Sensor ini diproduksi oleh *Sensirion*, dengan konektor tipe pin sehingga memudahkan jika dilakukan instalasi atau pelepasan. Sensor suhu dan kelembaban dibangun sangat akurat dengan stabilitas jangka panjang yang merupakan inti dari *platform* kelembaban dan suhu *Sensirion* yang baru. Desain modul yang unik memungkinkan kopling termal terbaik ke lingkungan dan pemisahan dari sumber panas potensial di papan utama. Sensor ini memiliki membran *PTFE* yang didedikasikan untuk melindungi bukaan sensor dari cairan dan debu sesuai dengan *IP67* tanpa memengaruhi waktu respon sinyal kelembaban relatif. Jika dibandingkan dengan sensor *DHT11/DHT22* dan *AM2320*, sensor ini memiliki akurasi yang lebih tinggi, rentang suhu yang lebih luas, dan waktu tanggapan yang lebih cepat, sehingga menjadikannya pilihan yang lebih unggul dalam situasi yang membutuhkan keakuratan dan performa yang tinggi. Perangkat sensor suhu dan kelembaban dapat dilihat pada gambar 6 serta spesifikasinya bisa dilihat pada tabel 4.



Gambar 6. Sensor Suhu dan Kelembaban.

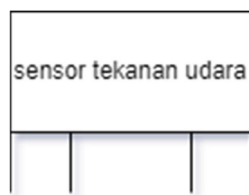
Tabel 4. Spesifikasi Sensor *SHT85*

<i>Variables</i>	<i>Values</i>	<i>Units</i>
<i>Typ. relative humidity accuracy</i>	±1.5	%RH
<i>Operating relative humidity range</i>	0 to 100	%RH
<i>Response time humidity (τ63%)</i>	8	s
<i>Calibration certificate humidity</i>	<i>Factory calibration</i>	
<i>Typ. temperature accuracy</i>	0.1	°C
<i>Operating temperature range</i>	-40 to 105	°C
<i>Response time temperature (τ63%)</i>	5	s
<i>Supply voltage</i>	2.15 - 5.5	V
<i>Average supply current</i>	28	uA
<i>Max. supply current</i>	1000	uA
<i>Interfaces</i>	I <sup>2</sup> C	
<i>Package size (LxWxH)</i>	19 x 5.6 x 3.5	mm <sup>3</sup>

### 2.13. Sensor Tekanan Udara

Sensor tekanan barometrik dilengkapi dengan fungsi pengukuran suhu dan tekanan. Dibandingkan dengan versi sebelumnya, sensor ini menunjukkan konsumsi daya yang lebih rendah, resolusi yang lebih tinggi, dan kecepatan pengambilan sampel yang lebih tinggi. Tekanan barometrik biasanya digunakan untuk mengukur tekanan dan suhu barometrik. Tapi selain itu, kita juga bisa menggunakan sensor

untuk mengukur ketinggian dan tinggi lantai relatif karena ada hubungan tertentu antara ketinggian dan tekanan barometrik. Sensor ini didasarkan pada teknologi sensor tekanan resistif *Piezo Bosch* yang matang yang menampilkan akurasi tinggi serta konsumsi daya yang rendah dan ketahanan *EMC (electromagnetic compatibility)* yang tinggi. Sensor memiliki akurasi sekitar  $\pm 8\text{Pa}$ , yang setara dengan perbedaan ketinggian sekitar  $\pm 0,5\text{m}$ , dan suhu akurasi mutlak  $\pm 0,5^\circ\text{C}$  untuk kisaran suhu antara  $0^\circ\text{C}$  dan  $65^\circ\text{C}$ . Jika dibandingkan dengan sensor *BME280*, sensor ini lebih unggul karena rentang tekanan yang lebih luas (300 hPa hingga 1250 hPa) dibandingkan dengan *BME280* (300 hPa hingga 1100 hPa) dan akurasi tekanan yang lebih tinggi ( $\pm 0.06$  hPa) dibandingkan dengan *BME280* ( $\pm 1$  hPa). Gambar 7 merupakan penampakan dari sensor tekanan udara dan spesifikasi lebih lanjut bisa dilihat pada tabel 5.



Gambar 7. Sensor Barometrik.

Tabel 5. Spesifikasi Sensor Barometrik

<i>Types</i>	<i>Values</i>
<i>Operating Voltage</i>	3.3V-5.5V
<i>Operating Current</i>	0.5mA
<i>Operating Range</i>	300 to 1250 hPa

Tabel 5. (Lanjutan)

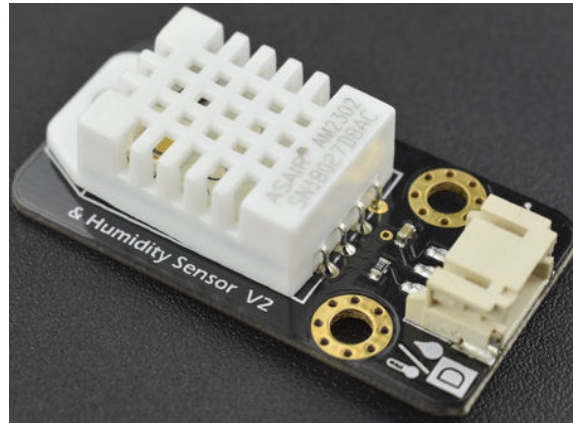
<i>Types</i>	<i>Values</i>
<i>Relative Accuracy</i>	$\pm 8$ Pa ( <i>equivalent to <math>\pm 0.50</math>m @700-900hPa, 25°C-40°C</i> )
<i>Absolute Accuracy</i>	$\pm 50$ Pa ( <i>0°C-65°C@300-1100hPa</i> )
<i>Temperature Coefficient Offset</i>	$\pm 0.75$ Pa/K ( <i>-20°C-65°C@700-1100hPa</i> )
<i>Absolute Accuracy Temperature</i>	$\pm 0.5$ °C ( <i>@0°C-65°C</i> )
<i>Operating Temperature</i>	<i>-40°C~80°C (more accurate in 0°C-65°C)</i>
<i>External Dimension</i>	22mm x 30mm
<i>Mounting Hole Position</i>	15mm
<i>Mounting Hole Dimension</i>	<i>inside diameter 3mm/ outside diameter 6mm</i>
<i>Interface</i>	Gravity-IIC 4Pin or SPI (SPI is only used at 3.3V)

#### 2.14. DHT22

Sensor *DHT22* adalah modul sensor untuk mengukur suhu dan kelembaban udara. Sensor ini berfungsi sebagai sensor untuk memantau suhu sekitar komponen perangkat keras. Dalam implementasinya, perangkat memiliki suhu yang optimal selama beroperasi, jika suhu perangkat melebihi ambang batas maka komponen tidak akan bekerja dengan baik bahkan bisa mati. Oleh karena itu, diperlukan sensor suhu untuk memantau kondisi suhu alat tersebut. Salah satu sensor suhu dan kelembaban yang beredar di pasaran adalah sensor *DHT11* dan *DHT22*. Sensor *DHT22* digunakan dalam penelitian ini, sensor ini dipilih karena sensor *DHT22* lebih akurat dan memiliki jangkauan pengukuran yang lebih besar dibandingkan



*DHT11*. Spesifikasi sensor *DHT22* ditampilkan pada tabel 6. Sensor *DHT22* dapat dilihat pada gambar 8.



Gambar 8. Sensor *DHT22*.

[Sumber: *DFRobot*, <https://www.dfrobot.com/>]

Tabel 6. Spesifikasi Sensor *DHT22*

<i>Types</i>	<i>Values</i>
Tegangan kerja	3.3V – 5V
Arus maksimum	2.5mA
Range pengukuran kelembaban	0 – 100%
Akurasi pengukuran kelembaban	2 – 5%
Range pengukuran suhu	-40°C – 80°C
Akurasi pengukuran suhu	0.5°C
Ukuran	15.1 mm x 25 mm x 7.7 mm

Sumber: Muhammad Habib, <https://www.mahirelektro.com/>.

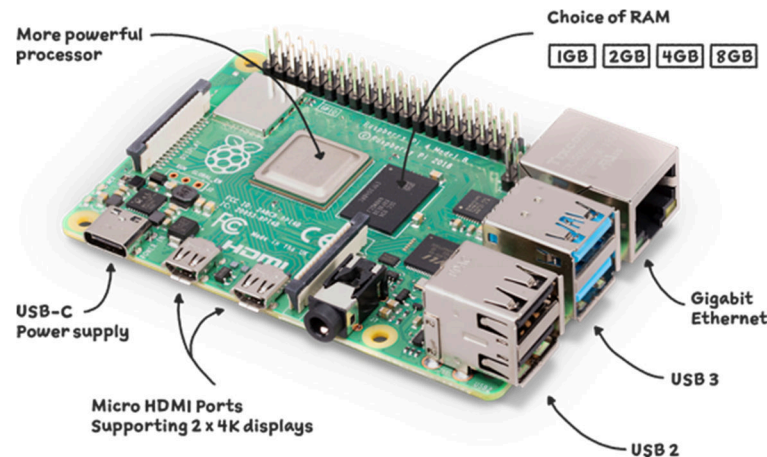
### **2.15. Arduino IDE**

*Arduino IDE (Integrated Development Environment)* adalah program perangkat lunak yang digunakan untuk membuat program yang akan dijalankan pada mikrokontroler. *Software* ini menggunakan sketch pemrograman C untuk membuat programnya. Selain sebagai *editor program*, *Arduino IDE* juga dapat mengkompilasi atau mengontrol program yang ditujukan untuk memudahkan pengembangan program oleh *programmer*.

*Arduino IDE* mempunyai struktur dasar untuk pengembangan sebuah program atau sketch, yaitu *void setup()* dan *void loop()*. *Void setup()* adalah fungsi yang hanya berjalan sekali dalam program. Fungsi ini menyiapkan semua yang dibutuhkan untuk menjalankan program. *Void loop()* adalah isi dari program yang akan dibuat. Fungsi ini berjalan terus menerus (*loop*) hingga perangkat dimatikan.

### **2.16. Raspberry Pi 4 Model B 2 GB Memory**

*Raspberry Pi 4B* merupakan salah satu model dari keluarga *single-board computer (SBC)* yang dikembangkan oleh *Raspberry Pi Foundation*. Model ini dilengkapi dengan kapasitas memori sebesar *2GB*, yang memberikan cukup sumber daya untuk menjalankan berbagai aplikasi dan tugas komputasi. Bentuk fisik dari *Raspberry Pi 4 Model B* bisa dilihat pada gambar 9.



Gambar 9. *Raspberry Pi 4 Model B.*  
 [sumber: *Raspberrypi*, <https://www.raspberrypi.com/> ]

*Raspberry Pi 4B 2GB* memiliki spesifikasi teknis yang menonjol dalam kategori *single-board computer*. Beberapa poin penting dalam spesifikasi teknis tersebut adalah sebagai berikut:

1. *Prosesor: Raspberry Pi 4B 2GB* menggunakan prosesor *Broadcom BCM2711, quad-core Cortex-A72 (ARM v8)* dengan kecepatan *clock* hingga *1.5GHz*. Prosesor ini memberikan kinerja yang lebih tinggi dibandingkan dengan model-model sebelumnya.
2. *Memori: Kapasitas memori sebesar 2GB LPDDR4* memberikan kecepatan *transfer* data yang lebih tinggi dan dukungan untuk *multitasking*.
3. *Konektivitas: Raspberry Pi 4B 2GB* dilengkapi dengan dua *port Micro HDMI*, dua *port USB 3.0*, dua *port USB 2.0*, *port Gigabit Ethernet*, serta slot kartu *microSD* untuk sistem penyimpanan.
4. *Perangkat Keras Lainnya: Selain itu, model ini juga memiliki header GPIO 40-pin*, dua *port MIPI Camera dan Display*, serta *port audio jack 3.5mm*.
5. *Sumber Daya: Raspberry Pi 4B 2GB* dapat ditenagai melalui kabel *USB-C* dengan tegangan *5V* dan arus minimal *3A*.

### 2.17. *Node JS*

*Node.js* merupakan lingkungan *runtime* yang berbasis pada bahasa *JavaScript* yang digunakan untuk menjalankan kode *JavaScript* di sisi *server*. Dengan menggunakan *Node.js*, pengembang dapat membuat aplikasi berbasis *server* yang efisien dan *scalable*.

Kelebihan *Node.js* yaitu performa tinggi, kemampuan *asynchronous*, dan dukungan modul yang kaya. Namun, *Node.js* juga memiliki keterbatasan, seperti *single-threaded nature* yang dapat mempengaruhi kinerja dalam beberapa skenario.

### 2.18. *Node Red*

*Node-RED* adalah platform aliran aliran berbasis visual yang berfungsi sebagai alat penghubung dan otomatisasi antar berbagai perangkat dan layanan. Platform ini berjalan di atas *runtime Node.js* dan menyediakan antarmuka *web* untuk membuat aliran aliran (*flow*) dengan menghubungkan berbagai *node* yang merepresentasikan fungsi atau layanan. *Node-RED* memiliki keunggulan dalam mengembangkan aplikasi berbasis *IoT* karena kemampuannya untuk menghubungkan berbagai perangkat dan layanan. *Node-RED* mendukung integrasi dengan berbagai layanan pihak ketiga, seperti *platform cloud* dan penyedia layanan *web*.

### 2.19. *EPEVER XTRA 4415N*

*EPEVER XTRA 4415N* adalah *regulator charge controller* yang digunakan dalam sistem pengisian baterai pada aplikasi panel surya. *Regulator* ini memiliki peran penting dalam mengatur aliran daya dari panel surya ke baterai untuk menghindari *overcharging* atau *overdischarging* yang dapat merusak baterai. Data dari

perangkat ini bisa diambil melalui *protocol RS485*. Bentuk fisik dari kontroler ini ditampilkan apda gambar 10.



Gambar 10. *EPEVER XTRA 4415N*.  
[Sumber: *Energy Power*, <https://energypower.gr/> ]

*EPEVER XTRA 4415N* memiliki sejumlah spesifikasi teknis yang relevan dalam aplikasi panel surya. Beberapa spesifikasi tersebut antara lain:

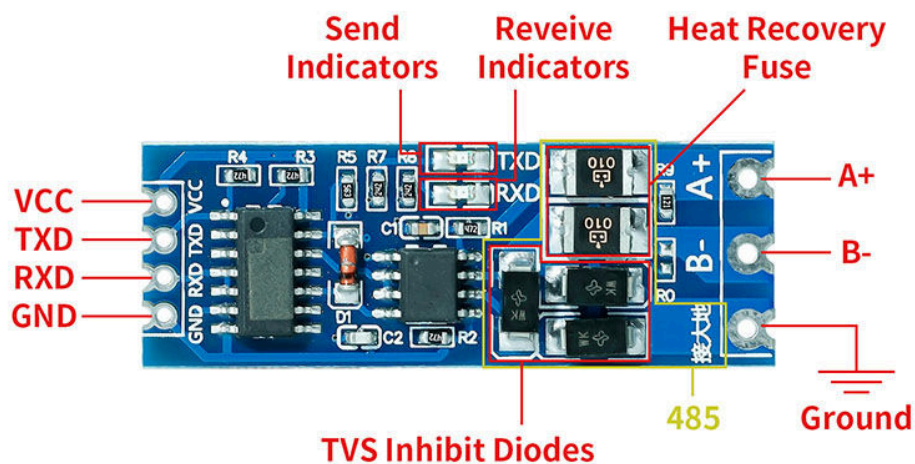
1. Tegangan Masukan: *Regulator* ini mendukung tegangan masukan dari panel surya dalam rentang tertentu, maksimal *VoC* adalah 150V.
2. Tegangan Keluaran: *EPEVER XTRA 4415N* menghasilkan tegangan keluaran sesuai dengan voltase baterai 12V/24V/36V/48V yang digunakan dalam sistem.
3. Arus Maksimum: *Regulator* ini dapat menangani arus maksimum 40A dari panel surya ke baterai.
4. Sistem Proteksi: Dilengkapi dengan sistem proteksi terhadap tegangan tinggi, tegangan rendah, *overcharging*, *overdischarging*, dan lainnya.

5. Tampilan dan Pengaturan: *EPEVER XTRA 4415N* dilengkapi dengan layar LCD untuk memantau status pengisian baterai dan pengaturan parameter tertentu.

### 2.20. 2 Way RS485 to TTL Converter

Modul ini memungkinkan antarmuka *TTL mikrokontroler* untuk diubah menjadi modul *RS485*. Biasanya digunakan untuk otomasi industri. Modul ini dilengkapi dengan desain perlindungan petir dan desain anti-gangguan. Saat digunakan di lapangan dan melakukan transmisi jarak jauh, kita dapat menghubungkan ujung *GND* dari modul ke tanah sehingga perlindungan petir dan anti-gangguan dapat tercapai.

Modul ini mendukung komunikasi antara beberapa mikrokontroler, memungkinkan hingga 128 perangkat dalam satu *bus*. Modul ini dapat dipasang secara *hot-swappable*, untuk mencegah gangguan sinyal, dilakukan area tembaga yang besar, sehingga efeknya sangat baik. Modul ini dapat dilihat pada gambar 11.



Gambar 11. 2 Way RS485 to TTL Converter.  
[Sumber: *Electrow*, <https://www.elecrow.com/>]

### 2.21. *Zerotier*

*Zerotier* adalah sebuah teknologi jaringan terdistribusi yang memungkinkan pembentukan jaringan *virtual* dengan cara yang sangat mudah dan aman. Konsep utama dari *Zerotier* adalah menciptakan *overlay network* yang menghubungkan perangkat secara langsung, tanpa perlu melalui server pusat atau konfigurasi rumit.

*Zerotier* menggunakan pendekatan terdesentralisasi dalam membentuk jaringan. Setiap perangkat yang terhubung ke jaringan *Zerotier* berfungsi sebagai simpul jaringan (*network node*) yang menyebarkan informasi tentang ketersediaan perangkat lain di jaringan.

Setiap perangkat yang ingin bergabung dengan jaringan *Zerotier* harus melewati proses otentikasi dan otorisasi yang ketat. Selain itu, *Zerotier* juga menggunakan enkripsi *end-to-end* untuk melindungi data yang dikirimkan antar perangkat.

*Zerotier* memberikan setiap perangkat yang terhubung ke jaringan sebuah local *IP global* yang unik. Hal ini memungkinkan perangkat-perangkat di jaringan *Zerotier* untuk saling berkomunikasi seperti perangkat yang berada dalam jaringan local yang sama.

*Zerotier* dapat dijalankan pada berbagai *platform*, termasuk *Windows*, *macOS*, *Linux*, *Android*, dan *iOS*. Dengan dukungan lintas *platform* ini, perangkat dari berbagai jenis dan sistem operasi dapat dengan mudah terhubung ke jaringan *Zerotier*.

Meskipun *Zerotier* menggunakan pendekatan terdesentralisasi, tetapi para *administrator* tetap dapat mengelola dan memantau jaringan *Zerotier* melalui sebuah pusat manajemen *web* yang mudah digunakan.

## 2.22. D-Link DES-1008C

*D-Link DES-1008C* adalah sebuah *switch ethernet* yang digunakan untuk menghubungkan beberapa perangkat lokal atau jaringan lokal dalam sebuah *LAN (Local Area Network)*. *Switch* ini memiliki 8 *port ethernet* dengan kecepatan 10/100 Mbps yang memungkinkan transfer data yang cepat antar perangkat dalam jaringan. *Switch* ini termasuk dalam kategori *unmanaged swich*. Bentuk fisik dari *switch* ini ditampilkan pada gambar 12.



Gambar 12. *D-Link DES-1008C*.  
[Sumber: *D-Link*, <https://www.dlink.co.id/> ]

*D-Link DES-1008C* memiliki sejumlah spesifikasi teknis yang relevan dalam lingkungan jaringan. Beberapa spesifikasi tersebut antara lain:

1. Jumlah *Port*: *Switch* ini dilengkapi dengan 8 *port ethernet* yang dapat digunakan untuk menghubungkan perangkat dalam jaringan.
2. Kecepatan Transfer: *DES-1008C* mendukung kecepatan transfer data hingga 10/100 Mbps, memungkinkan transfer data yang cepat antar perangkat.
3. *Auto-Negotiation*: *Switch* ini dilengkapi dengan fitur *auto-negotiation* yang memungkinkan otomatisasi konfigurasi kecepatan dan mode transmisi pada setiap *port*.



4. *Half-Duplex/Full-Duplex*: *DES-1008C* mendukung mode transmisi *half-duplex* dan *full-duplex* yang dapat diatur pada setiap port sesuai dengan kebutuhan jaringan.
5. *Auto-MDI/MDIX*: *Switch* ini memiliki fitur *auto-MDI/MDIX* yang memungkinkan penggunaan kabel *straight-through* atau *crossover* pada setiap *port* tanpa perlu mengatur secara manual.

### **2.23. REST API**

*REST API (Representational State Transfer Application Programming Interface)* adalah sebuah pendekatan dalam arsitektur perangkat lunak yang digunakan untuk berkomunikasi dan bertukar data antara sistem-sistem yang berbeda secara terdistribusi melalui protokol *HTTP*. *REST API* memanfaatkan metode *HTTP* seperti *GET*, *POST*, *PUT*, dan *DELETE* untuk berinteraksi dengan sumber daya (*resources*) yang diidentifikasi dengan *URL (Uniform Resource Locator)*. Pada penelitian ini, *REST API* digunakan sebagai antarmuka komunikasi antara *data logging* karbon dari gas metana ( $CH_4$ ) yang dibangun dengan *Platform IoT*. Dengan menggunakan *REST API*, data dari sensor-sensor yang terhubung ke sistem dapat diakses dan dikirimkan ke database serta *dashboard* secara efisien dan terstruktur.

### **2.24. Postgresql**

*PostgreSQL* adalah sebuah sistem manajemen basis data relasional (*RDBMS*) yang *open-source* dan *powerfull*. *PostgreSQL* menawarkan fitur-fitur lengkap dan mendukung bahasa *query SQL* yang kaya, serta menyediakan kemampuan untuk mengelola data dengan efisien dan aman.

### **2.25. Expressjs**

*Express.js* adalah sebuah *framework web* yang berjalan di atas *platform Node.js* yang digunakan untuk membangun aplikasi *web* dengan mudah dan cepat. *Express.js* menyediakan berbagai fitur dan fungsi yang mempermudah pengembangan aplikasi *web*, seperti *routing*, *middleware*, *template engine*, dan lainnya. Dalam penelitian ini, *Express.js* digunakan sebagai *framework* untuk membangun *backend* dari *platform Internet of Things (IoT)* yang bertanggung jawab dalam mengelola permintaan (*request*) dari berbagai perangkat *IoT* dan mengirimkan respons (*response*) berupa data-data yang diperlukan.

### **2.26. Github**

*GitHub* adalah *platform hosting* kode sumber berbasis *web* yang sangat populer di kalangan pengembang perangkat lunak. *Platform* ini menyediakan repositori *git* publik dan pribadi yang memungkinkan pengguna untuk menyimpan, mengelola, dan berkolaborasi dalam pengembangan perangkat lunak secara terdistribusi. *GitHub* memiliki fitur-fitur kolaborasi yang kuat, yang memungkinkan tim pengembang bekerja bersama dalam proyek perangkat lunak.

## 2.27. Dashboard

*Dashboard* merupakan antarmuka pengguna (*user interface*) dari sistem *monitoring* dan *data logging* gas metana ( $CH_4$ ) serta kondisi *data logger* yang dirancang berbasis *Internet of Things (IoT)*. Pada penelitian ini, *dashboard* dibangun menggunakan *Grafana*, sebuah *platform open-source* yang digunakan untuk visualisasi data secara *real-time*.

## 2.28. Grafana

*Grafana* adalah *platform open-source* yang digunakan untuk melakukan visualisasi dan analisis data dari berbagai sumber yang berbeda. *Platform* ini menyediakan beragam fitur yang memungkinkan pengguna untuk membuat *dashboard* interaktif, diagram, dan laporan berdasarkan data yang diambil dari berbagai *database*, sistem *monitoring*, dan aplikasi lainnya. *Grafana* dapat diintegrasikan dengan berbagai basis data dan sistem *monitoring* untuk mengambil dan mengelola data secara *real-time*. Contoh tampilan *Grafana* ditampilkan pada gambar 13.



Gambar 13. Tampilan *Grafana*.  
[Sumber: *Grafana*, <https://grafana.com/>]

### **III. METODOLOGI PENELITIAN**

#### **3.1. Waktu dan Tempat Penelitian**

Penelitian skripsi ini dilakukan di Laboratorium Terpadu Jurusan Teknik Elektro Universitas Lampung dan di area perkebunan lain. Sedangkan waktu penelitian skripsi ini dimulai dari bulan Agustus 2022 – Agustus 2023.

#### **3.2. Komponen dan Perangkat Lunak Penelitian**

Adapun komponen dan perangkat lunak yang digunakan dan dimanfaatkan pada penelitian ini dapat dilihat pada tabel 7 untuk bagian perangkat *monitoring* gas metana, tabel 8 untuk bagian perangkat *data logger* dan *SCC*, serta tabel 9 untuk *platform IoT*.

Tabel 7. Komponen dan Perangkat Lunak Perangkat *Monitoring* Gas Metana

No.	Nama Komponen dan Perangkat Lunak	Keterangan Penggunaan
1.	<i>Arduino Portenta H7</i>	Sebagai pusat kontrol baik pengolahan data dan pengiriman data.
2.	<i>Portenta Breakout Board</i>	Sebagai <i>expander</i> untuk memanfaatkan pin <i>high-density</i> yang terdapat pada Portenta H7 serta menambah fitur RTC, baca tulis SD Card, dan Ethernet.
3.	Sensor Gas Metana	Sebagai sensor gas Metana.
4.	<i>DHT22</i>	Sebagai sensor suhu dan kelembaban dalam perangkat.
5.	<i>Buck Boost Converter up to 60V</i>	Sebagai komponen untuk menurunkan tegangan listrik.
6.	<i>Arduino IDE V2.0</i>	Sebagai perangkat lunak untuk membangun program yang akan dijalankan pada mikrokontroler.
7.	Sensor Suhu dan Kelembaban	Sebagai sensor suhu dan kelembaban udara luar.
8.	Sensor tekanan Udara	Sebagai sensor tekanan dan suhu <i>barometric</i> .
9.	<i>Arduino_PortentaBreakout.h</i>	<i>Library</i> untuk mendukung fitur-fitur pada <i>Arduino Portenta Breakout Board</i> .
10.	<i>ArduinoHttpClient.h</i>	<i>Library</i> untuk membuat permintaan <i>HTTP</i> dan menerima respons pada <i>Arduino</i> .
11.	<i>ArduinoJson.h</i>	<i>Library</i> untuk memanipulasi data <i>JSON</i> pada <i>Arduino</i> .
12.	<i>DHT.h</i>	<i>Library</i> untuk mengakses sensor <i>DHT (Digital Humidity and Temperature)</i> pada <i>Arduino</i> .

Tabel 7. (Lanjutan)

No.	Nama Komponen dan Perangkat Lunak	Keterangan Penggunaan
13.	<i>Wire.h</i>	<i>Library</i> untuk mengakses komunikasi I2C pada <i>Arduino</i> .
14.	<i>Adafruit_Sensor.h</i>	<i>Library</i> untuk mendukung sensor-sensor dari <i>Adafruit</i> pada <i>Arduino</i> .
15.	<i>Adafruit_BMP3XX.h</i>	<i>Library</i> untuk mengakses sensor <i>BMP3XX</i> ( <i>Barometric Pressure and Altitude</i> ) dari <i>Adafruit</i> pada <i>Arduino</i> .
16.	<i>SHTSensor.h</i>	<i>Library</i> untuk mengakses sensor <i>SHT</i> ( <i>Humidity and Temperature</i> ) pada <i>Arduino</i> .
17.	<i>TimeLib.h</i>	<i>Library</i> untuk mendukung penggunaan waktu dan tanggal pada <i>Arduino</i> .
18.	<i>avr/dtostrf.h</i>	<i>Library</i> untuk mengkonversi angka <i>floating point</i> menjadi string dalam format karakter pada <i>Arduino</i> .
19.	<i>Oscilloscope</i>	Melihat gelombang listrik atau sinyal data.
20.	<i>Multimeter</i>	Mengukur tegangan dan arus.
21.	Kabel <i>UTP CAT 5e + RJ-45</i>	Media komunikasi dengan perangkat lain dan internet.
22.	Kabel <i>nyyhy 1x0.75mm</i>	Kabel daya dari <i>buck boost</i> ke komponen. Kapasitas 300/500V 10A.
23.	Kabel <i>nyyhy 2x0.75mm</i>	Kabel daya dari <i>supply</i> utama ke <i>terminal block</i> . Kapasitas 300/500V 10A.
24.	PCB Bolong	Membagi <i>port IIC</i> .
25.	<i>Eagle</i>	Desain PCB untuk sensor suhu dan kelembaban.

Tabel 7. (Lanjutan)

No.	Nama Komponen dan Perangkat Lunak	Keterangan Penggunaan
26.	<i>Sensor Case</i>	Pelindung sensor untuk luar ruangan.
27.	<i>Box Panel 25x25x12cm</i>	<i>Box busbar</i> dan <i>switch hub</i> .
28.	<i>Box panel 40x60x20cm</i>	<i>Box kontrol</i> sistem <i>monitoring</i> karbon.
29.	<i>Resistor 10k ½ Watt</i>	Komponen tambahan untuk sensor suhu dan kelembaban.
30.	Konektor <i>JST-XH</i>	Konektor untuk <i>PCB</i> sensor suhu dan kelembaban.
31.	Kabel <i>socket DC JST SM</i>	Konektor antara komponen di dalam <i>box panel</i> .
32.	<i>Python</i>	Bahasa pemrograman untuk membantu analisis data.
33.	<i>Pandas</i>	Untuk manipulasi dan analisis data secara efisien dalam <i>Python</i> .
34.	<i>Matplotlib</i>	Untuk membuat visualisasi grafik dan <i>plot</i> data dalam <i>Python</i> .
35.	<i>Seaborn</i>	Untuk membuat visualisasi data statistik yang menarik dan informatif dalam <i>Python</i> .

Tabel 8. *Data Logger* dan *SCC Data Parser*

No.	Nama Komponen dan Perangkat Lunak	Keterangan Penggunaan
1.	<i>Raspberry Pi 4B 2GB</i>	Sebagai <i>Data Logger</i> dan data parser dari <i>SCC</i> .
2.	<i>D-Link DES-1008C</i>	Sebagai pembagi jaringan penghubung ke semua perangkat atau ke layanan internet.

Tabel 8. (Lanjutan)

No.	Nama Komponen dan Perangkat Lunak	Keterangan Penggunaan
3.	<i>Node js</i>	Runtime untuk menjalankan program node red..
4.	<i>Node red</i>	Sebagai <i>platform</i> untuk <i>data logger</i> dan <i>data parser</i> dari <i>SCC</i> .
5.	<i>UFW (Uncomplicated Firewall)</i>	Digunakan untuk mengelola <i>firewall</i> pada sistem operasi <i>Linux</i> .
6.	Modul <i>node-red-contrib-modbus</i>	Komunikasi menggunakan protokol <i>Modbus</i> .
7.	<i>Node function</i>	Untuk memasukkan fungsi <i>javascript</i> pada <i>node-red</i> .
8.	<i>Node sequence join</i>	Untuk menggabungkan <i>object</i> .
9.	<i>Node debug</i>	Untuk menampilkan <i>output</i> atau <i>debugging</i> .
10.	<i>Node http request</i>	Membuat suatu <i>request</i> .
11.	<i>Node inject</i>	Membuat <i>loop</i> atau menambahkan nilai pada suatu objek.
12.	Modul <i>node-red-contrib-cpu</i>	Melihat konsumsi <i>CPU</i> .
13.	Modul <i>node-red-contrib-vcgencmd</i>	Memberikan <i>command</i> ke <i>raspberry pi</i> .
14.	<i>Node tcp in</i>	Menerima koneksi <i>TCP</i> .
15.	<i>Node tcp out</i>	Mengirimkan data melalui <i>protocol TCP</i> .
16.	<i>Node read file</i>	Membaca isi <i>file</i> dan menjadikannya <i>payload</i> .
17.	<i>2 Way RS485 to TTL Converter</i>	Sebagai konverter protokol komunikasi dari <i>RS485</i> ke <i>TTL</i> atau sebaliknya.
18.	<i>Buck Boost Converter up to 60V</i>	Sebagai komponen untuk menurunkan tegangan listrik.



Tabel 8. (Lanjutan)

No.	Nama Komponen dan Perangkat Lunak	Keterangan Penggunaan
19.	<i>Metal Case with Heatsink &amp; Fan For Raspberry Pi 4 Model B</i>	Pelindung <i>raspberry pi</i> serta sebagai pendingin <i>SBC</i> .
20.	Kabel <i>UTP CAT 5e + RJ-45</i>	Media komunikasi dengan perangkat lain dan internet.
21.	Kabel <i>nyyhy 1x0.75mm</i>	Kabel daya dari <i>buck boost</i> ke komponen. kapasitas 300/500V 10A.
22.	<i>Vsftpd</i>	<i>Server FTP</i> yang berguna untuk menyediakan layanan <i>transfer file</i> melalui protokol <i>FTP</i> dengan fokus pada keamanan dan performa yang tinggi.
23.	<i>FileZilla</i>	Aplikasi klien <i>FTP open-source</i> yang memungkinkan pengguna untuk mengelola dan mentransfer <i>file</i> secara mudah melalui protokol <i>FTP</i> , <i>FTPS</i> , dan <i>SFTP</i> .
24.	<i>Zerotier</i>	Layanan jaringan virtual yang memungkinkan pengguna untuk membuat jaringan pribadi yang aman dan terenkripsi melalui internet, sehingga memudahkan akses dan berbagi sumber daya di berbagai perangkat secara global.
25.	<i>Raspberry Pi Imager</i>	Alat untuk menginstall sistem operasi <i>Raspberry Pi</i> .

Tabel 9. Platform IoT

No.	Nama Komponen dan Perangkat Lunak	Keterangan Penggunaan
1.	<i>Postgresql</i>	Sebagai <i>database</i> .
2.	<i>Grafana</i>	Sebagai <i>platform dashboard</i> .
3.	<i>Infinity plugin Grafana</i>	Untuk visualisasi data dari <i>JSON, CSV, XML, GraphQL</i> dan <i>HTML endpoints</i> .
4.	<i>Github</i>	<i>Version Control System</i> serta dokumentasi proyek.
5.	<i>Visual Studio Code</i>	Sebagai <i>editor</i> kode.
6.	<i>Node.js</i>	<i>Runtime</i> untuk menjalankan program <i>backend</i> .
7.	<i>VPS Digital Ocean Ubuntu 22.04 1 Core 2 GB 50 GB</i>	<i>Cloud</i> komputasi dan penyimpanan untuk <i>database, REST API, dan Grafana</i> .
8.	<i>body-parser</i>	Mengurai data yang dikirim melalui permintaan <i>HTTP (request)</i> dalam bentuk <i>body</i> .
9.	<i>cors</i>	Mengaktifkan kebijakan keamanan lintas sumber ( <i>CORS</i> ) dalam aplikasi <i>web server</i> .
10.	<i>csv-parser</i>	Membaca data dari <i>file CSV (Comma-Separated Values)</i> .
11.	<i>dotenv</i>	Membaca variabel lingkungan ( <i>environment variables</i> ) dari <i>file ".env"</i> .
12.	<i>express</i>	<i>Framework web</i> untuk <i>Node.js</i> .
13.	<i>fs</i>	Berinteraksi dengan sistem berkas ( <i>file system</i> ).
14.	<i>moment-timezone</i>	Memanipulasi, memformat, dan menampilkan waktu dan tanggal.
15.	<i>multer</i>	Mengelola pengunggahan ( <i>upload</i> ) berkas dalam aplikasi <i>web server</i> .

Tabel 9. (Lanjutan)

No.	Nama Komponen dan Perangkat Lunak	Keterangan Penggunaan
16.	<i>"pg"</i> dan <i>"pg-hstore"</i>	Modul <i>"pg"</i> adalah klien postgresql untuk Node.js yang memungkinkan koneksi dan kueri ke database postgresql. Sementara itu, <i>"pg-hstore"</i> adalah modul yang menyediakan alat bantu untuk mengonversi data ke format postgresql hstore.
17.	<i>sequelize</i>	<i>ORM (Object-Relational Mapping)</i> untuk <i>Node.js</i> yang bekerja dengan berbagai basis data <i>SQL</i> , termasuk <i>postgresql</i> .
18.	<i>eslint</i>	Proses pemeriksaan kode untuk menemukan dan memberi peringatan tentang potensi masalah atau gaya kode yang tidak sesuai dengan standar tertentu.
19.	<i>postman</i>	Alat pengujian dan pengembangan <i>API</i> yang memudahkan dalam mengirim dan menganalisis permintaan HTTP serta menampilkan hasil responsnya.

### 3.3. Rancangan Perangkat dan Sistem

Penelitian skripsi yang dilakukan penulis merupakan bagian dari riset program Grand Riset Sawit (GRS) “*Monitoring dan Controlling Keseimbangan Emisi Karbon Perkebunan Kelapa Sawit di Lahan Gambut dan Non-Gambut dengan Metode Kecerdasan Buatan*” dengan focus riset adalah pengembangan model sistem *monitoring* keseimbangan emisi karbon pada perkebunan kelapa sawit. Skripsi ini merupakan langkah awal untuk melakukan pengembangan perangkat

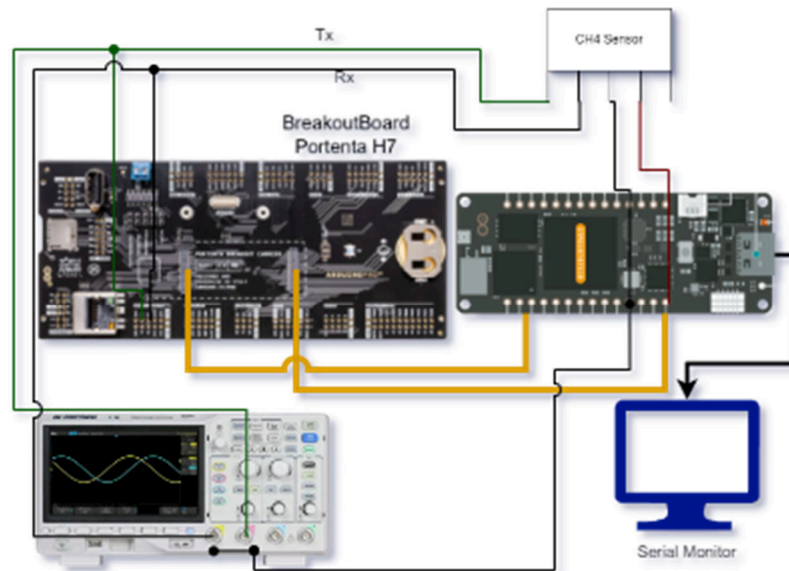
*Eddy Station*. Pada penelitian ini dirancang sebuah perangkat yang dapat mengamati konsentrasi gas Metana, kelembaban udara, suhu udara, tekanan dan suhu barometrik. Selain itu Dirancang juga perangkat data logger menggunakan *raspberry pi*, kemudian diunggah ke *cloud* untuk bisa diolah atau ditampilkan. Data-data yang akan dihasilkan disesuaikan dengan kebutuhan data yang akan digunakan pada metode *Eddy Covariance* sebagai metode perhitungan fluks karbon.

### 3.3.1. Pengujian Tiap Sensor

Pengujian tiap sensor dilakukan untuk mengetahui apakah setiap sensor bisa melakukan pengambilan data seperti suhu, kelembaban, kadar gas Metana dan lainnya. Pengujian sensor dilakukan satu per satu untuk melihat data yang dihasilkan seperti bentuk gelombang dan nilai hasil pembacaan. Khusus untuk sensor suhu dan kelembaban memerlukan komponen berupa dua buah resistor berukuran 10.000 ohm. Supaya lebih memudahkan maka dibuatlah *PCB* untuk menyatukan sensor suhu dan kelembaban dengan komponen tambahannya.

#### 3.3.1.1. Rangkaian Pengujian Sensor Gas Metana

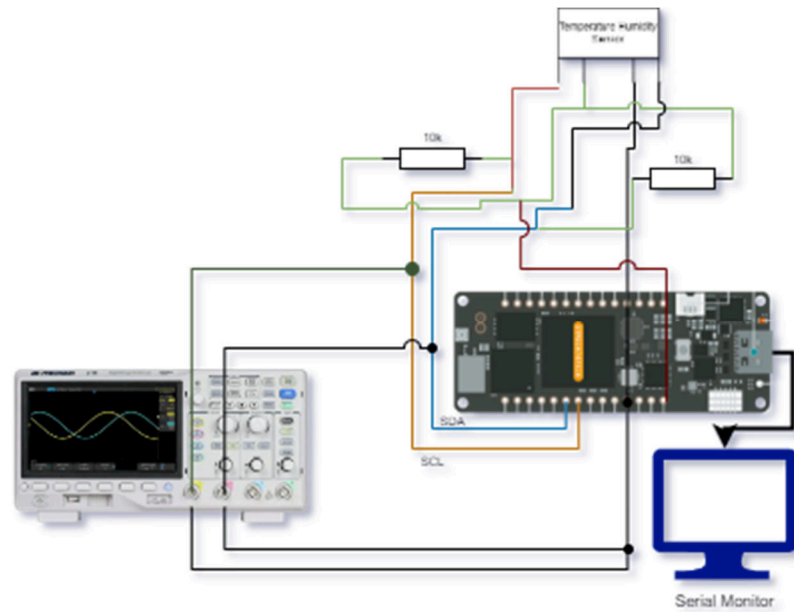
Pengujian ini dilakukan untuk sensor tunggal CH<sub>4</sub> atau gas Metana. Mikrokontroler yang dipakai adalah *Arduino Portenta H7* beserta *Breakout Board* karena pin yang digunakan untuk sensor adalah pin *Serial* dari *Breakout Board*. *Serial monitor* digunakan untuk melihat hasil pembacaan dari sensor. Osiloskop digunakan untuk melihat gelombang data dari sensor. Gambar 14 menampilkan rangkaian pengujian sensor gas Metana.



Gambar 14. Rangkaian Pengujian Sensor Gas Metana.

### 3.3.1.2. Rangkaian Pengujian Sensor Suhu dan Kelembaban

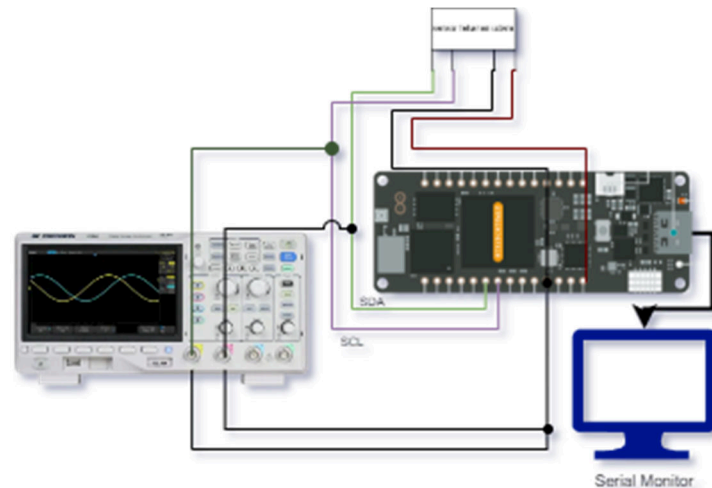
Pengujian ini dilakukan untuk sensor suhu dan kelembaban. Mikrokontroler yang dipakai adalah *Arduino Portenta H7* dengan memakai pin *I2C*. Diperlukan komponen tambahan berupa dua buah resistor ukuran 10.000 ohm untuk pin *SDA* dengan *supply* dan pin *SCL* dengan *supply*. Serial monitor digunakan untuk melihat hasil pembacaan dari sensor. Osiloskop digunakan untuk melihat gelombang data dari sensor. Gambar 15 menampilkan rangkaian pengujian sensor suhu dan kelembaban.



Gambar 15. Rangkaian Pengujian Sensor Suhu dan Kelembaban.

### 3.3.1.3. Rangkaian Pengujian Sensor *Barometrik*

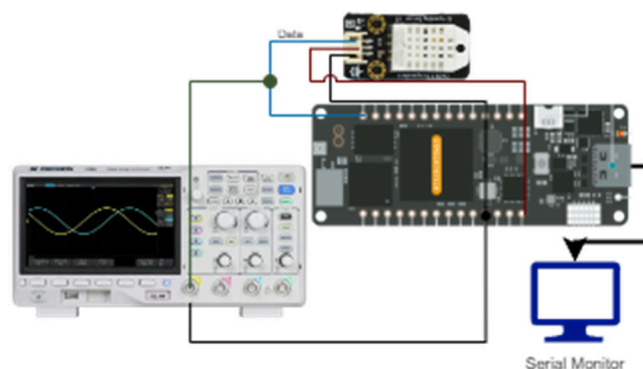
Pengujian ini dilakukan untuk sensor *barometrik*. Mikrokontroler yang dipakai adalah *Arduino Portenta H7* dengan memakai pin *I2C*. *Serial monitor* digunakan untuk melihat hasil pembacaan dari sensor. Osiloskop digunakan untuk melihat gelombang data dari sensor. Gambar 16 menampilkan rangkaian pengujian sensor tekanan udara.



Gambar 16. Rangkaian Pengujian Sensor Tekanan Udara.

#### 3.3.1.4. Rangkaian Pengujian Sensor *DHT22*

Pengujian ini dilakukan untuk sensor *DHT22*. Mikrokontroler yang dipakai adalah *Arduino Portenta H7* dengan memakai pin D5 sebagai komunikasi data. *Serial monitor* digunakan untuk melihat hasil pembacaan dari sensor. Osiloskop digunakan untuk melihat gelombang data dari sensor. Gambar 17 menampilkan rangkaian pengujian sensor *DHT22*.



Gambar 17. Rangkaian Pengujian Sensor *DHT22*.

### 3.3.2. Perangkat *Monitoring* Karbon dari Gas Metana

#### 3.3.2.1. Perancangan Program Sistem Perangkat *Monitoring*

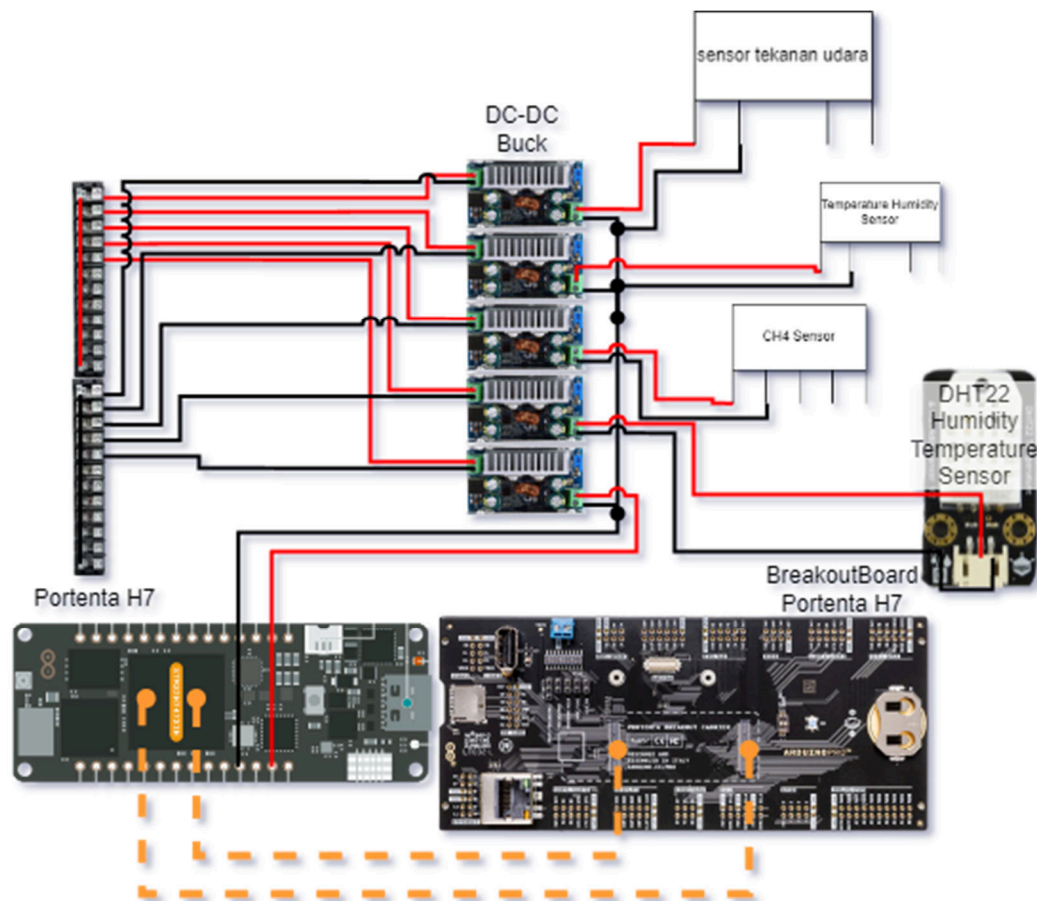
Program yang digunakan untuk menjalankan perangkat dirancang menggunakan *Arduino IDE*. *Library* yang digunakan antara lain *Arduino\_PortentaBreakout*, *ArduinoHttpClient*, *ArduinoJson*, *DHT*, *Wire*, *SPI*, *Adafruit\_Sensor*, *Adafruit\_BMP3XX*, *SHTSensor*, *TimeLib*, dan *avr/dtostrf*. Algoritma yang akan diterapkan yaitu perangkat akan melakukan inisiasi modul *ethernet* lalu mendapatkan waktu dari *NTP server*. Kemudian melakukan inisiasi untuk semua modul sensor serta langsung melakukan *sensing data* dan langsung dikirimkan ke *data logger*. Perangkat akan melakukan siklus *sensing data* selama 30 menit dan istirahat selama 30 menit yang dilakukan terus menerus. Akan dilakukan penyesuaian juga pada perangkat keras jika perangkat lunak tidak dapat bekerja seperti yang diharapkan seperti menggunakan *interface* yang sama untuk sensor suhu kelembaban dan sensor tekanan udara jika penggunaan *Breakout Board* tidak dapat dilakukan.

#### 3.3.2.2. Wiring Perangkat *Monitoring*

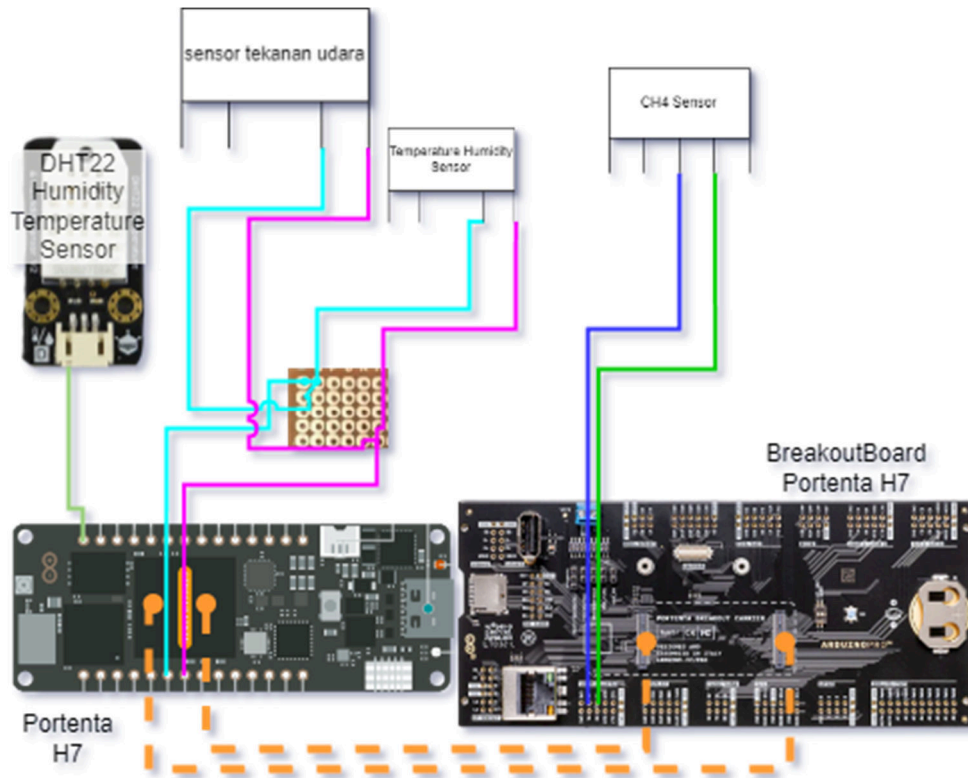
Kelistrikan untuk komponen keseluruhan perangkat menggunakan *DC to DC buck boost* dengan tegangan keluaran 5V. Tujuan dari penggunaan *buck boost* di sini adalah untuk menurunkan tegangan *supply* sebesar 48V dari baterai ke tegangan yang sesuai untuk masing masing komponen. Komponen yang dipakai meliputi 2 *terminal block* 12 pin, 5 *buck boost up to 60V input*, *Portenta H7*, *Breakout board for Portenta H7*, sensor tekanan udara, sensor suhu dan kelembaban, sensor gas Metana, *DHT22*. Kabel yang digunakan untuk menghubungkan antar komponen dengan *buck boost* dan terminal dengan *buck boost* adalah kabel *nyyhy 1x0.75mm*, kabel penghubung daya pusat dengan *terminal block* adalah kabel *nyyhy 2x0.75mm*,



serta kabel *extension* untuk daya sensor tekanan udara, sensor suhu dan kelembaban, sensor gas Metana adalah kabel *UTP CAT 5e*. Mikrokontroler yang dipakai adalah *Portenta H7* karena sudah standar industri. Rangkaian untuk daya *sistem monitoring* karbon dari gas Metana bisa dilihat pada gambar 18.



Gambar 18. Skema Kelistrikan untuk Komponen Perangkat *Monitoring* Gas Metana.



Gambar 19. Rangkaian Komunikasi Perangkat *Monitoring* Gas Metana.

Rangkaian media komunikasi perangkat *monitoring* gas Metana ditampilkan oleh gambar 19. Sensor suhu kelembaban dan sensor tekanan udara menggunakan protokol komunikasi *I2C* dari pin *D11* sebagai *SDA* dan *D12* sebagai *SCL* dari *portenta H7*, sensor *DHT22* menggunakan komunikasi digital dengan pin *D5* pada *Portenta H7* dan sensor gas Metana menggunakan komunikasi *UART* dengan pin *UART1* pada *breakout board*. Untuk lebih jelasnya bisa dilihat pada tabel 10.

Tabel 10. Komunikasi Perangkat *Monitoring* Karbon dari Gas Metana

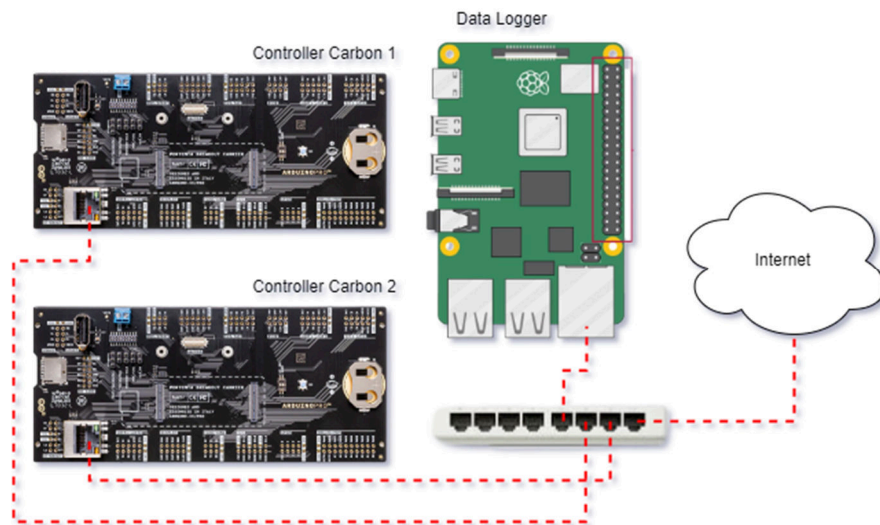
<b><i>Pin Portenta(Pt)/Breakout board(Br)</i></b>	<b><i>Pin Komponen</i></b>
<i>(Pt) D5</i>	<i>DHT22 Data</i>
<i>(Pt) D11 SDA</i>	<i>Baro SDA</i>

Tabel 10. (Lanjutan)

<b>Pin Portenta(Pt)/Breakout board(Br)</b>	<b>Pin Komponen</b>
<i>(Br) UART1 Tx</i>	<i>Methane Rx</i>
<i>(Br) UART1 Rx</i>	<i>Methane Tx</i>
<i>(Pt) D12 SCL</i>	<i>Baro SCL</i>
<i>(Pt) D11 SDA</i>	<i>Temp Humi (SDA)</i>
<i>(Pt) D12 SCL</i>	<i>Temp Humi (SCL)</i>

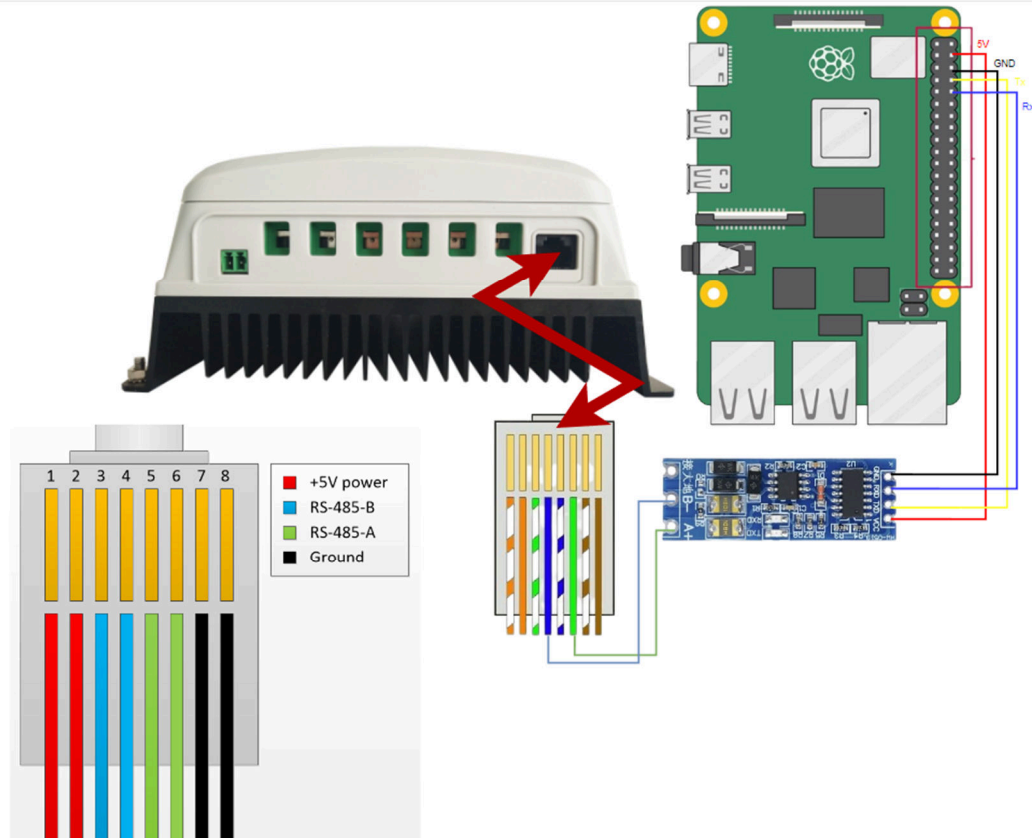
### 3.3.3. Perangkat *Data Logger* dan *SCC Data Parser*

Media komunikasi yang dipakai untuk menghubungkan antara perangkat *data logger* dan *Eddy Station* adalah *wired* melalui *ethernet*. Pembagian jaringan dilakukan oleh *switch D-Link DES-1008C*. *Switch* juga sebagai penghubung ke layanan internet. Pada bagian ini tidak digunakan mikrokontroler melainkan digunakan *SBC (Single Board Computer)* yaitu *Raspberry Pi 4B 2GB*. *Data logger* memanfaatkan *platform node red* dengan memanfaatkan *protocol TCP/IP*. Rangkaian konektivitas antara *data logger* dan *system monitoring* gas karbon serta internet dapat dilihat pada gambar 20.



Gambar 20. Konektivitas *Data Logger* dengan Sistem *Monitoring* Karbon dan Internet.

*Raspberry Pi* juga berfungsi sebagai *SCC data parser* dari *SCC Epever XTRA 4415N* untuk *monitoring* suplai daya. Data akan didapatkan melalui *modbus* sehingga perlu *2 way RS48 to TTL converter* untuk menghubungkan antara *SCC* dengan *Raspberry Pi*. Pin *VCC* pada *converter* dihubungkan ke pin *5V raspberry pi*, pin *GND* dihubungkan ke *GND Raspberry Pi*. Platform yang akan dipakai dalam melakukan parsing adalah *Node red* dengan bantuan modul *node-red-contrib-modbus*. Rangkaian dari *SCC data parser* dapat dilihat pada gambar 21 dan tabel pin *wiring* bisa dilihat pada tabel 11.



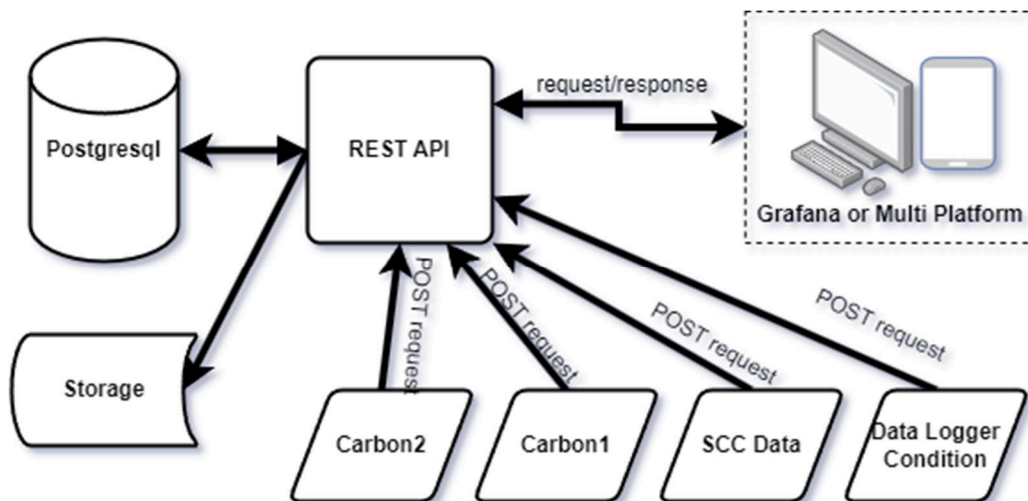
Gambar 21. Rangkaian *SCC Data Parser*.

Tabel 11. Pin *Wiring SCC Data Parser*

<i>Raspberry Pi Pin Board</i>	<i>2 Way RS485 to TTL Pin</i>
4	<i>VCC</i>
6	<i>GND</i>
10	<i>Rx</i>
8	<i>Tx</i>

### 3.3.4. Platform Internet of Things

*Platform IoT* menggunakan *Postgresql* sebagai *database* dan *Grafana* untuk *dashboard*. *Framework web* yang dipakai adalah *express js* dibantu dengan *multer* untuk mengelola pengunggahan berkas pada *web server*. Bagian *backend* akan berjalan pada *runtime Node js*. *Cloud* komputasi memiliki spesifikasi *system operasi Ubuntu 22.04, 1 core, 2 GB RAM, dan 50 GB storage*. Desain sistem dari *platform IoT* secara umum dapat dilihat pada gambar 22.



Gambar 22. Desain Sistem *Platform IoT*.

### 3.4. Diagram Alir Penelitian

Tahapan dari penelitian ini dimulai dari studi literatur sampai dengan pelaporan dalam bentuk penyusunan laporan dalam dipresentasikan dengan penyusunan diagram alir penelitian secara menyeluruh.

### 3.4.1. Diagram Alir Sistem *Monitoring* Karbon dari Gas Metana dan *SCC Data Parser*

Penelitian ini diawali dengan kajian pustaka atau studi literatur, yaitu tahap dimana mencari, mempelajari dan memahami materi dari literatur yang berkaitan dengan penelitian tugas akhir ini. Selama studi literatur ini, terdapat studi bimbingan yang membahas materi dan permasalahan yang ditemukan selama studi literatur. Kemudian melakukan perancangan alat dengan cara merancang semua sistem alat secara utuh hingga tahap perakitan alat. Setelah dilakukan perancangan alat maka akan dilakukan tahap pengujian alat. Jika hasil pengujian alat tidak sesuai dengan yang diharapkan, maka kembali lagi ke tahap perancangan alat untuk mengevaluasi kembali alat supaya alat dapat bekerja sesuai dengan apa yang diharapkan. Jika perangkat *monitoring* gas karbon dan *SCC data parser* berfungsi seperti yang diharapkan, akan berpindah ke langkah berikutnya, yaitu tahap untuk pengumpulan data dari perangkat *monitoring* gas karbon dan *SCC data parser*. Langkah selanjutnya adalah menganalisis data yang didapatkan dari perangkat *monitoring* gas karbon dan *SCC data parser* serta melakukan analisis juga pada perangkat *monitoring* gas karbon dan *SCC data parser*. Setelah menerima hasil analisis, hasil analisis tersebut dicatat secara tertulis dan dilaporkan sebagai hasil penelitian dalam bentuk laporan skripsi. Diagram alir untuk perangkat *monitoring* gas karbon dan *SCC data parser* ditampilkan pada gambar 23.



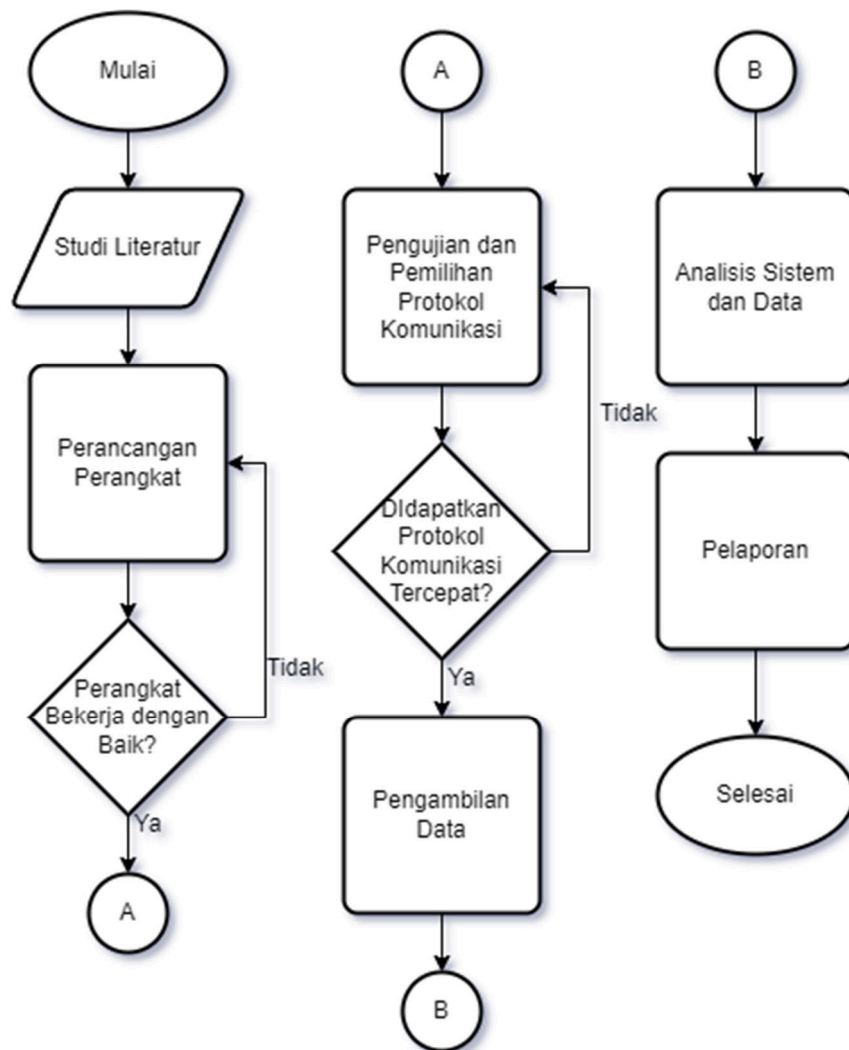
Gambar 23. Diagram Alir Perangkat *Monitoring* Karbon dari Gas Metana dan *SCC Data Parser*.

### 3.4.2. Diagram Alir Sistem *Data Logger*

Penelitian ini diawali dengan kajian pustaka atau studi literatur, yaitu tahap dimana mencari, mempelajari dan memahami materi dari literatur yang berkaitan dengan penelitian tugas akhir ini. Selama studi literatur ini, terdapat studi bimbingan yang membahas materi dan permasalahan yang ditemukan selama studi literatur. Kemudian melakukan perancangan sistem *data logger* dengan cara merancang semua sistem alat secara utuh hingga tahap perakitan alat. Setelah dilakukan perancangan sistem *data logger* maka akan dilakukan tahap pengujian sistem *data logger*. Jika hasil pengujian sistem *data logger* tidak sesuai dengan yang diharapkan, maka kembali lagi ke tahap perancangan sistem *data logger* untuk mengevaluasi kembali alat supaya alat dapat bekerja sesuai dengan apa yang



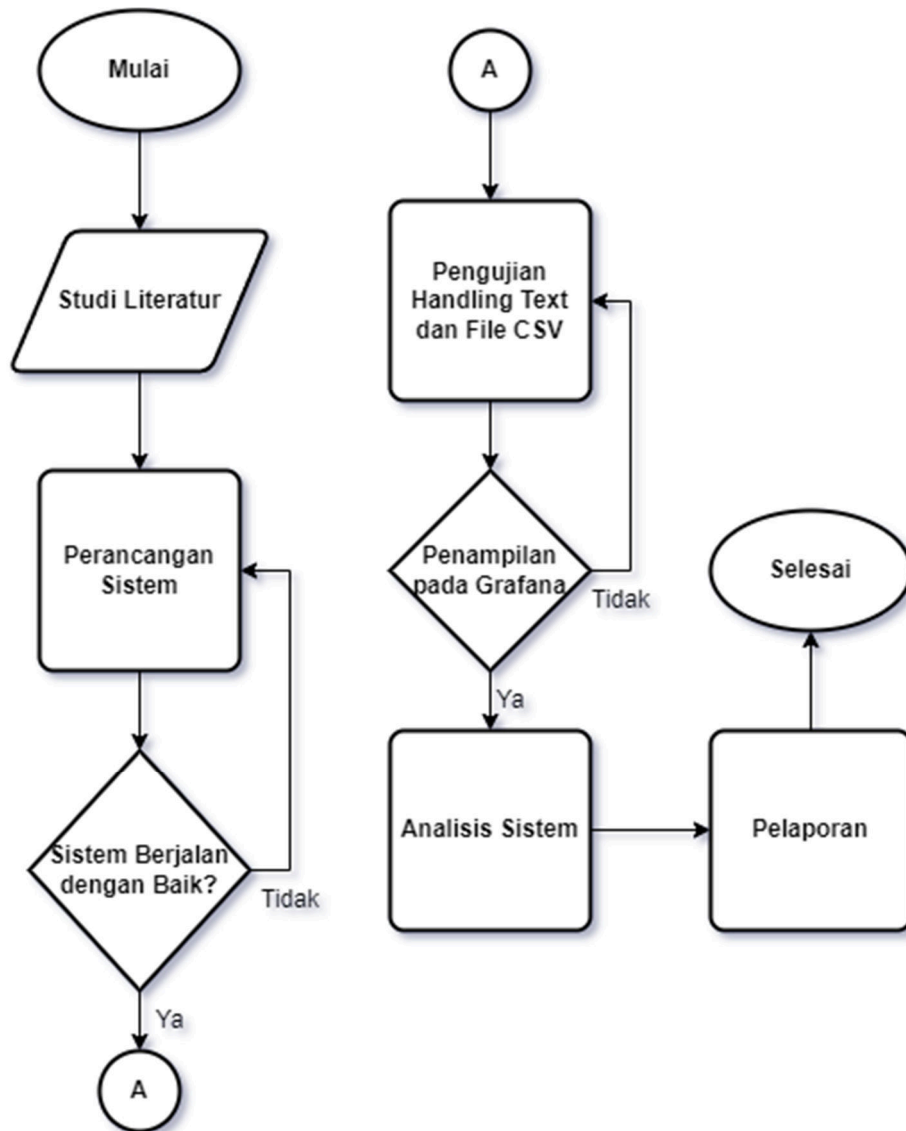
diharapkan. Jika sistem *data logger* berfungsi seperti yang diharapkan, sistem *data logger* akan berpindah ke langkah berikutnya, yaitu tahap untuk pemilihan *protocol* komunikasi tercepat. Protokol komunikasi yang akan diuji adalah *HTTP*, *MQTT* dan *TCP/IP*. Tahap selanjutnya adalah pengumpulan data dari sistem *data logger*. Kemudian menganalisis data yang didapatkan dari sistem *data logger* serta melakukan analisis juga pada sistem. Setelah menerima hasil analisis, hasil analisis tersebut dicatat secara tertulis dan dilaporkan sebagai hasil penelitian dalam bentuk laporan skripsi. Diagram alir untuk sistem *data logger* ditampilkan pada gambar 24.



Gambar 24. Diagram Alir Sistem *Data Logger*.

### 3.4.3. Diagram Alir Sistem *Platform IoT*

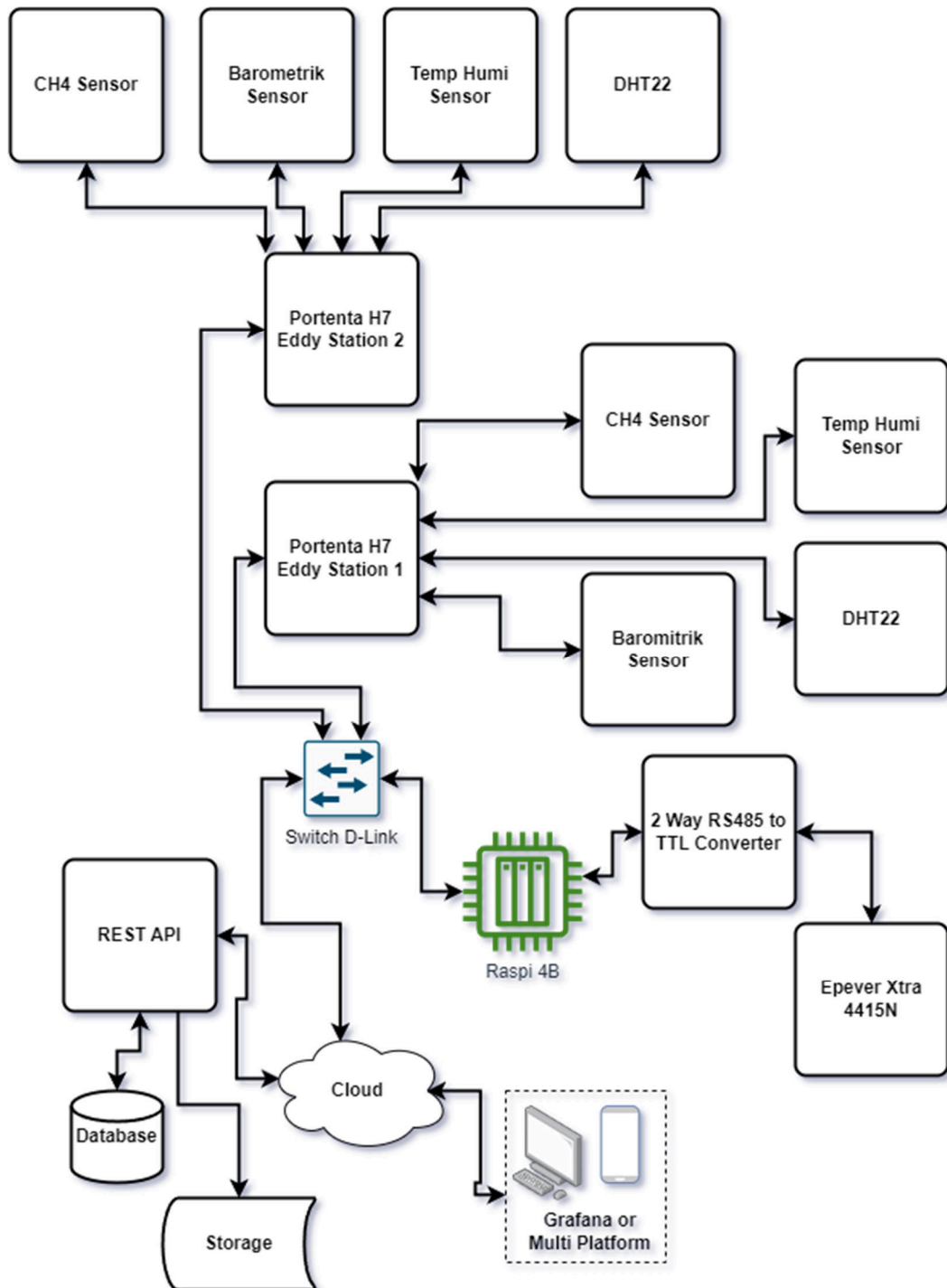
Penelitian ini diawali dengan kajian pustaka atau studi literatur, yaitu tahap dimana mencari, mempelajari dan memahami materi dari literatur yang berkaitan dengan penelitian tugas akhir ini. Selama studi literatur ini, terdapat studi bimbingan yang membahas materi dan permasalahan yang ditemukan selama studi literatur. Kemudian melakukan perancangan sistem dengan cara merancang semua sistem secara utuh hingga tahap pengujian. Setelah dilakukan perancangan sistem maka akan dilakukan tahap pengujian sistem. Jika hasil pengujian sistem tidak sesuai dengan yang diharapkan, maka kembali lagi ke tahap perancangan sistem untuk mengevaluasi kembali sistem supaya sistem dapat bekerja sesuai dengan apa yang diharapkan. Jika sistem berfungsi seperti yang diharapkan, sistem akan berpindah ke langkah berikutnya, yaitu tahap untuk menguji *handling text* dan *file csv* hingga bisa tampil sempurna pada *Grafana*, jika masih belum bekerja maksimal maka akan melakukan pengujian *handling* lagi. Tahap selanjutnya adalah menganalisis sistem. Setelah menerima hasil analisis, hasil analisis tersebut dicatat secara tertulis dan dilaporkan sebagai hasil penelitian dalam bentuk laporan skripsi. Diagram alir untuk pembangunan sistem *platform IoT* ditampilkan pada gambar 25.



Gambar 25. Diagram Alir Sistem Platform IoT.

### 3.5. Diagram Blok Perangkat dan Sistem

Adapun diagram blok perancangan dari perangkat dan sistem dari *monitoring* karbon dari gas Metana, *data logger* dan *platform IoT* ditampilkan pada gambar 26.



Gambar 26. Diagram Blok Sistem *Monitoring Karbon dari Gas Metana, Data Logger dan Platform IoT.*

Seperti yang ditampilkan pada Gambar 26 bahwa perangkat *monitoring* gas karbon menggunakan modul sensor gas Metana untuk mendapatkan kadar gas Metana,

Sensor Barometer sebagai modul sensor pengukur tekanan dan suhu, sensor suhu dan kelembaban sebagai modul sensor suhu dan kelembaban udara yang datanya bisa diolah untuk mendapatkan konsentrasi H<sub>2</sub>O di udara. Sensor *DHT22* digunakan untuk melihat kondisi suhu dan kelembaban udara komponen kontroler yang berada di dalam *box panel*. Keseluruhan sistem dikontrol oleh mikrokontroler *Arduino Portenta H7*. *Portenta H7* ini mengatur semua pemrosesan data baik dalam pengambilan data sensor, hingga melakukan pengiriman data ke *data logger*. *Arduino Portenta H7* menggunakan komponen tambahan yaitu *Breakoutboard* untuk memanfaatkan pin *high-density* pada *Arduino Portenta H7*. Waktu yang dipakai adalah waktu *internal board* yang diatur saat *board* pertama kali menyala dengan memanfaatkan waktu dari *NTP server*. Modul *Ethernet* juga terdapat pada *Breakoutboard* ini dan bisa dimanfaatkan untuk koneksi secara *wired* ke perangkat *data logger*.

*Data logger* menggunakan perangkat *Raspberry Pi 4B*, selain itu *Raspberri Pi* juga bekerja sebagai *SCC data parser* melalui *modbus* dengan bantuan modul *converter RS485 to TTL*. Data pada *data logger* ini bisa diakses melalui *FTP* baik dalam jaringan lokal atau *public* dengan memanfaatkan *Zerotier*. Data dari perangkat *monitoring* gas karbon yang ada pada *data logger* akan dikirim ke *cloud* sehari sekali setiap pukul 00.30 dan pukul 01.00 supaya tidak terlalu membebani *server*. Pada pukul 02.00 akan dilakukan pembersihan data yang berumur lebih dari 14 hari dari data loger untuk menjaga kesehatan *resource*.

*Platform IoT* akan menerima data dari *data logger* dengan layanan *REST API* untuk kemudian data disimpan di *database* atau di *storage*. Data yang disimpan juga akan ditampilkan pada *Grafana* sebagai *dashboard*.

### **3.6. Skenario Monitoring Gas CH<sub>4</sub>, Data Logger, dan Solar System**

Pada pelaksanaan penelitian ini akan dijelaskan skenario dari sistem *monitoring* karbon dari gas metana yang akan diimplementasikan menggunakan sensor gas,

*Arduino Portenta H7*, dan *platform IoT ZeroTier*. Skenario ini merinci langkah-langkah yang dilakukan dalam proses pengambilan data, pengiriman data, dan visualisasi data hasil monitoring gas metana. Dengan implementasi skenario *monitoring gas CH<sub>4</sub>*, *data logger*, dan *solar system* ini, diharapkan sistem dapat memberikan informasi yang tepat waktu dan akurat mengenai tingkat gas metana dalam lingkungan sekitar, kondisi perangkat *data logger* dan *solar system* sehingga dapat mendukung untuk kebutuhan analisa data lebih lanjut.

### **3.6.1. Analisa Kemampuan Latensi Sensor**

Keseluruhan sensor seperti sensor gas Metana, sensor barometer, sensor *suhu kelembaban* dan sensor *DHT22* akan dilakukan pengujian seberapa cepat kemampuan *sensing* data dalam keadaan data hasil *sensing* bisa stabil untuk kemudian bisa disesuaikan dengan kebutuhan data yang diperlukan untuk penggunaan metode *Eddy Covariance*.

### **3.6.2. Integrasi Sensor dengan *Arduino Portenta H7***

Sensor gas metan, sensor barometrik, sensor suhu kelembaban dan sensor *DHT22* akan diintegrasikan dengan *Arduino Portenta H7* menggunakan pin *GPIO* yang sesuai serta pin pada *Breakout Board*. *Arduino Portenta H7* akan bertindak sebagai mikrokontroler yang mengambil data dari sensor secara berkala.

### **3.6.3. Pengambilan Data Sensor**

*Arduino Portenta H7* akan melakukan pengambilan data dari keseluruhan sensor secara periodik. Data dari seluruh sensor akan dikumpulkan dalam format *json* dan

kemudian langsung dikirimkan ke *data logger* melalui protokol yang paling sesuai. Skenario yang dipakai adalah mikrokontroler akan bekerja selama 30 menit kemudian tidur selama 30 menit, lalu bekerja kembali selama 30 menit dan tidur lagi selama 30 menit, siklus ini bekerja secara terus menerus.

#### **3.6.4. Pengambilan Data SCC dan Kondisi Data Logger**

*Raspberry Pi* akan melakukan pengambilan data dari *Solar Charge Controller* secara periodik setiap satu menit dan langsung mengirimkannya ke *platform IoT* dalam bentuk *JSON*. Data yang diambil meliputi arus, tegangan, dan *power* dari baterai maupun dari panel surya. *Raspberry Pi* juga mengirimkan kondisi resourcenya ke *platform IoT* setiap 5 detik.

#### **3.6.5. Proses Unggah Data Carbon Monitoring dan Menjaga Kesehatan Data Logger**

*Data logger* akan mengunggah data ke *platform IoT* dari perangkat *carbon monitoring* 1 pada pukul 00.30, mengunggah data ke *platform IoT* dari perangkat *carbon monitoring* 2 pada pukul 01.00. Untuk menjaga storage tidak penuh, maka akan dilakukan penghapusan data berumur lebih dari 14 hari pada *data logger*.

#### **3.6.6. Akses Data pada Data Logger Menggunakan FTP dan Zerotier**

Data hasil *monitoring* gas karbon, data perangkat *data logger*, dan data baterai serta panel surya akan diakses langsung melalui *protocol FTP* dan dengan tambahan *Zerotier* jika akan diakses menggunakan internet.

### 3.6.7. Penerimaan Data di *Platform IoT*

*Platform IoT* yang ada di server pusat menerima data dari *data logger*. Data akan diolah dan disimpan untuk analisis lebih lanjut seperti melakukan komputasi atau *filter* data ketika menerima *request* dari *client*.

### 3.6.8. Visualisasi Data *Monitoring*

Data *monitoring* gas metana, data kondisi *data logger*, data baterai dan data panel surya yang telah dikumpulkan akan divisualisasikan menggunakan platform *Grafana*. *Grafana* akan menampilkan data dalam bentuk grafik dan *dashboard* yang interaktif untuk memudahkan pemantauan konsentrasi gas metana, kondisi *data logger*, maupun kondisi *solar system*.

## 3.7. Perancangan Perangkat dan Sistem

### 3.7.1. Perancangan *PCB* untuk Sensor *SHT85*

Diperlukan perancangan *PCB* untuk sensor suhu kelembaban karena jarak antar pin hanya 1.27mm dan tidak bisa dilakukan pengujian dengan teknik *jumper*. Perancangan *PCB* menggunakan aplikasi *EAGLE* dari *Autodesk*. Diperlukan komponen tambahan untuk sensor *SHT85* sesuai *datasheet* yaitu dua buah *resistor* ukuran 10k ohm  $\frac{1}{2}$  *Watt* untuk koneksi *VDD* ke *SDA* dan koneksi *VDD* ke *SCL*. Konektor yang dipakai adalah *JST-XH*, serta ukuran diameter *pcb* adalah 24.08mm. Menyediakan dua lubang berukuran 2mm untuk baut.



### 3.7.2. Perancangan Perangkat *Monitoring* Karbon dari Gas Metana

Komponen dan bahan tambahan perlu dipersiapkan sebelum perancangan perangkat *monitoring* gas karbon. Komponen yang dipakai untuk membuat perangkat karbon *monitoring* terbagi menjadi 2 yaitu komponen diluar *box panel* dan di dalam *box panel*. Komponen di luar *box panel* meliputi:

1. *Case sensor* 3 buah
2. Sensor gas Metana
3. Sensor barometrik
4. Sensor suhu dan kelembaban

Komponen di dalam *box panel* meliputi:

1. *Terminal block* 12 pin 2 buah
2. *Buck boost* 5 buah
3. *Portenta H7* dan *Breakout Board*
4. *PCB* bolong
5. Sensor *DHT22*

Tegangan masuk ke seluruh *buck boost* adalah 48V – 56V dan tegangan keluar dari seluruh *buck boost* diatur menjadi 5V. Kabel yang digunakan untuk menghubungkan suplai daya pusat ke *terminal block* adalah kabel *nyyhy 2x0.75mm*. Kabel yang digunakan untuk menghubungkan *terminal block* ke *buck boost* menggunakan kabel *nyyhy 1x0.75mm*. Kabel *DC* konektor *JST SM* digunakan untuk konektor antar komponen maupun dengan kabel *extension* di dalam *box panel* supaya mudah dalam melakukan *maintenance*. Kabel yang dipakai untuk *extension* ke sensor yang berada di luar menggunakan kabel *UTP*.

### 3.7.3. Perancangan Perangkat *Data Logger*

*Raspberry Pi 4 Model B 2 GB RAM* sebagai *data logger*. Perancangan perangkat *data logger* ini diawali dengan instalasi sistem operasi *Raspberry Pi OS*. Instalasi bisa dilakukan menggunakan *Raspberry Pi Imager*. Kemudian menyalakan *Raspberry Pi* untuk dilakukan *basic set up* seperti koneksi ke jaringan internet dan melakukan pengaturan tanggal serta bahasa atau melakukan *update*. Langkah selanjutnya adalah melakukan instalasi *Node-RED* pada *Raspberry Pi* dan menjalankannya.

#### 3.7.3.1. Persiapan *Node-RED*

Biasanya *node-red* akan berjalan pada *port* 1880 sehingga setelah *node-red* berjalan bisa langsung pergi ke browser dan mengunjungi alamat *localhost:1880* untuk masuk ke *platform node-red* dan langsung membuat *flow* baru. Masukkan *node tcp in*, *tcp out*, *function*, dan *debug* ke dalam *flow* masing-masing dua *node* karena perangkat *monitoring* karbon yang akan mengirimkan data terdapat dua buah. Algoritma berjalan dengan logika menerima data *JSON* dari *TCP/IP*, kemudian mengubahnya ke format *csv* untuk disimpan ke dalam *file* yang dikelompokkan perhari. Sistem juga akan mengecek apakah pada data sudah terdapat objek *ts* dan *value*, jika belum maka akan dibuatkan objek *ts* oleh *data logger*. Sistem ini terinspirasi dari *API thingsboard*. Informasi tambahan meliputi:

1. Direktori karbon 1 : '/home/pi/Desktop/data/test\_eddy1'
2. Direktori karbon 2 : '/home/pi/Desktop/data/test\_eddy2'
3. Port karbon 1 : 2001
4. Port karbon 2 : 2002

### 3.7.3.2. Mengaktifkan *FTP*

Pengaktifan *FTP* dimaksudkan untuk memudahkan akses *file data logger* tanpa harus berinteraksi langsung dengan *raspberry pi* maupun menggunakan *remote desktop* seperti *VNC*. Langkah-langkah tersebut meliputi:

1. Membuka terminal
2. Memasukkan perintah: *sudo raspi-config*
3. Memilih: *Interfacing Options*
4. Memilih: *SSH*
5. Memilih “yes” ketika muncul “*Would you like the SSH server to be enabled?*”
6. Menutup terminal dan membuka terminal baru
7. Memasukkan perintah: *sudo apt update*
8. Memasukkan perintah: *sudo apt full-upgrade*
9. Memasukkan perintah: *sudo apt install vsftpd*
10. Memasukkan perintah: *sudo nano /etc/vsftpd.conf*
11. Melakukan penghapusan simbol pagar atau mengganti menjadi baris berikut:
 

```
anonymous_enable=NO
local_enable=YES
write_enable=YES
local_umask=022
chroot_local_user=YES
user_sub_token=$USER
local_root=/home/$USER/FTP
```
12. Menyimpan dan keluar terminal
13. Membuka terminal
14. Membuat direktori *FTP* dengan perintah “*mkdir -p /home/<user>/FTP/files*”, bagian *<user>* disesuaikan, *default* adalah “*pi*”
15. Memasukkan perintah: *sudo service vsftpd restart*

### 3.7.3.3. Mempersiapkan Zerotier

*Zerotier* digunakan untuk mengakses *node-red* ataupun *FTP* menggunakan internet secara aman. Sehingga tanpa menggunakan *VNC*, data pada *data logger* maupun melakukan pengaturan *node-red* bisa dilakukan melalui internet. Langkah mempersiapkan *Zerotier* meliputi:

1. Membuka terminal pada *Raspberry Pi* dan memasukkan perintah: `curl -s https://install.zerotier.com | sudo bash`
2. Memasukkan perintah: `sudo zerotier-cli join NETWORK_ID`. *NETWORK\_ID* disesuaikan dengan *NETWORK\_ID* yang dimiliki
3. Memasukkan perintah: `sudo zerotier-cli listnetworks`. Untuk melihat daftar jaringan yang ada pada *Raspberry Pi*
4. Menambahkan *managed route* ke *zerotier network* pada: `my.zerotier.com/network/NETWORK_ID`, menambahkan izin dan menambahkan rute (*destination* dan *via*)
5. Melakukan pengaktifan *IP forwarding* pada *Raspberry Pi* dengan membuka file `"/etc/sysctl.conf"` dan melakukan penghapusan tanda pagar pada `"net.ipv4.ip_forward"`, selanjutnya lakukan *restart Raspberry Pi*
6. Membuka terminal dan memasukkan perintah `"sudo sysctl -w net.ipv4.ip_forward=1"` untuk mengaktifkan *IP forwarding* jika ternyata setelah *boot* ternyata *IP forwarding* belum hidup
7. Memasukkan `"PHY_IFACE=eth0; ZT_IFACE=zt7nnig26"` pada terminal, sesuaikan dengan yang ada pada *Raspberry Pi*, *eth0* adalah contoh dari *interface* dari *internal* jaringan *raspberry Pi* dan *zt7nnig26* adalah contoh dari *interface* dari *Zerotier* yang berjalan pada *Raspberry Pi*
8. Memasukkan perintah: `sudo iptables -t nat -A POSTROUTING -o $PHY_IFACE -j MASQUERADE`
9. Memasukkan perintah: `sudo iptables -A FORWARD -i $PHY_IFACE -o $ZT_IFACE -m state --state RELATED,ESTABLISHED -j ACCEPT`
10. Memasukkan perintah: `sudo iptables -A FORWARD -i $ZT_IFACE -o $PHY_IFACE -j ACCEPT`
11. Memasukkan perintah: `sudo apt install iptables-persistent`

12. Memasukkan perintah: *sudo bash -c iptables-save > /etc/iptables/rules.v4*

#### 3.7.3.4. Persiapan *Firewall* Menggunakan *UFW*

*Firewall* harus memperbolehkan komunikasi masuk dan keluar untuk *port* yang akan digunakan pada layanan di *data logger* supaya bisa diakses melalui jaringan dan bisa dilakukan interaksi. Langkah tersebut meliputi:

1. Membuka terminal pada *Raspberry Pi* dan memasukkan perintah: *sudo apt update*
2. Memasukkan perintah: *sudo apt full-upgrade*
3. Memasukkan perintah: *sudo apt install ufw*
4. Memasukkan perintah untuk memberi akses ke *node-red*: *sudo ufw allow 1880*
5. Memasukkan perintah untuk memberi akses ke *SSH* dan *SFTP*: *sudo ufw allow 22*
6. Memasukkan perintah untuk memberi akses ke perangkat *monitoring* karbon 1: *sudo ufw allow 2001*
7. Memasukkan perintah untuk memberi akses ke perangkat *monitoring* karbon 2: *sudo ufw allow 2002*
8. Mengaktifkan *UFW* dengan perintah: *sudo ufw enable*. Memilih “y” jika muncul “*Command may disrupt existing ssh connections. Proceed with operation (y|n)?*”

### 3.7.4. Perancangan *SCC Data Parser*

*SCC data parser* digunakan untuk mengambil data tegangan, arus, dan daya baik dari baterai maupun dari panel surya. Data ini didapatkan dari *solar charge controller* melalui *modbus* dan diambil menggunakan *Raspberry Pi*.

#### 3.7.4.1. Persiapan *flow*

Langkah-langkah pengembangan *flow SCC data parser* ini meliputi:

1. Membuka *platform node-red*
2. Membuat *flow* baru
3. Menginstall modul *node-red-contrib-modbus* melalui menu *Manage pallette*
4. Memasukkan satu buah *node Modbus - Read*, tiga buah *Modbus - Getter*, lima buah *function*, satu buah *join*, dua buah *http request*, dan satu buah *debug* ke dalam *pallette*

Algoritma yang dipakai yaitu data akan didapatkan melalui masing-masing alamat *register* kemudian disesuaikan format datanya untuk kemudian dilakukan penggabungan objek menggunakan *join* dan dikirimkan ke *storage Raspberry Pi* dan ke *platform IoT*. Konfigurasi *serial* yang digunakan seperti *port /dev/ttyAMA0* dengan *serial type RTU-BUFFERD* dan *baud rate 115200*. Daftar alamat *register* dan data hasil yang digunakan antara lain:

1. 0x3100 : (quantity 6) Berisi data *pvvoltage*, *pvcurent*, *pvpower*, *bvoltage*, dan *battChargeCurrent*
2. 0x3106 : (quantity 2) Berisi data *battChargePower*
3. 0x310D : (quantity 3) Berisi data *lcurrent* dan *lpower*
4. 0x311A : (quantity 2) Berisi data *bremaining* dan *btemp*
5. 0x331B : (quantity 2) Berisi data *battOverallCurrent*

### 3.7.4.2. Mengaktifkan *Serial Port*

Langkah-langkah pengaktifan *serial port* pada *SCC data parser* ini meliputi:

1. Membuka terminal dan memasukkan perintah: *sudo raspi-config*
2. Memilih: *Interface Options*
3. Memilih: *Serial Port*
4. Memilih “No” jika muncul “*Would you like a login shell to be accessible over serial?*”
5. Memilih “Yes” jika muncul “*Would you like the serial port hardware to be enabled?*”
6. Memulai ulang *Raspberry Pi*
7. Mengubah isi file */boot/config.txt* dengan menambahkan *dtoverlay=uart0* dibawah bagian *enable\_uart=1*

### 3.7.5. Pengembangan *Platform IoT*

*Platform IoT* berperan sebagai *REST API* dan penyimpanan *database* dari perangkat sistem *monitoring* karbon dari gas Metana, data *solar system*, dan kondisi *data logger* serta penampil data pada *dashboard*-nya.

#### 3.7.5.1. Persiapan *Database*

*Backend* diperlukan karena digunakan sebagai penyimpanan data yang sangat besar sehingga data mudah dikelola untuk seterusnya. *Postgresql* dipakai sebagai *database* pada *backend platform IoT* ini. Langkah persiapan *database* meliputi:

1. Menginstall *Postgresql*
2. Masuk ke dalam *database postgres* menggunakan *psql*

3. Membuat *user* baru dengan perintah: *CREATE USER zan WITH PASSWORD 'zan123';*
4. Membuat *database* baru dengan perintah: *CREATE DATABASE zan\_db;*
5. Memberikan izin *database* *zan\_db* ke *user* *zan* dengan perintah: *GRANT ALL PRIVILEGES ON DATABASE zan\_db TO zan;*
6. Melakukan pindah koneksi ke *database* *zan\_db* dengan perintah: *\c zan\_db*
7. Memberikan izin *schema* *public* ke *user* *zan* dengan perintah: *GRANT ALL ON SCHEMA public TO zan;*

### 3.7.5.2. Persiapan *Backend*

*Backend* pada *platform IoT* ini harus memiliki fitur *REST API* karena *protocol* pada *REST API* merupakan *protocol* yang umum dan tidak terlalu kompleks sehingga bisa mudah untuk diintegrasikan ke berbagai aplikasi atau *platform*. Pengembangan *backend* bisa memanfaatkan *text editor* *Visual Studio Code* dan *Github* sebagai *Version Control System*. Sistem operasi yang dipakai untuk menjalankan program *backend* adalah *Ubuntu* versi 22.04. Langkah persiapan *backend* meliputi:

1. Menginstall *Nodejs* versi 18.13.0
2. Melakukan inisiasi proyek dengan perintah: *npm init*
3. Menginstall *package* *body-parser*, *cors*, *csv-parser*, *dotenv*, *express*, *fs*, *moment-timezone*, *multer*, *pg*, *pg-hstore*, dan *sequelize*.
4. Membuat *file* konfigurasi pada *file* *.env* yang berisi informasi untuk masuk ke *database*
5. Membuat model untuk *karbon1*, *karbon2*, kondisi *data logger*, dan *solar system* sehingga bisa didefinisikan oleh *sequelize*
6. Membuat kontroler sederhana seperti mengambil data rata-rata setiap jam untuk data karbon, menerima data dalam bentuk *json* dan memasukkannya ke dalam *database*, atau menerima data dalam bentuk *file csv* kemudian memasukkannya ke dalam *database*
7. Membuat *routes* untuk mengatur *endpoint* dan jenis *request* yang diterima



8. Membuat *file index.js* untuk menjalankan *expres* pada direktori utama proyek

### 3.7.5.3. Persiapan *Dashboard*

*Dashboard* berfungsi sebagai penyedia data yang mudah untuk didapat informasi karena sudah ditampilkan data yang kemungkinan besar akan memberikan *insight* kepada *user* yang melihat. *Grafana* dipilih sebagai *dashboard* karena mudah dalam pengembangannya dan tidak perlu kemampuan pemrograman khusus. Sistem operasi yang dipakai untuk menjalankan program *Grafana* adalah *Ubuntu* versi 22.04. Langkah persiapan *dashboard* meliputi:

1. Membuka terminal dan memasukkan perintah: `sudo apt-get update`
2. Memasukkan perintah: `sudo apt-get install -y apt-transport-https`
3. Memasukkan perintah: `sudo apt-get install -y software-properties-common wget`
4. Memasukkan perintah: `sudo wget -q -O /usr/share/keyrings/grafana.key https://apt.grafana.com/gpg.key`
5. Memasukkan perintah `echo "deb [signed-by=/usr/share/keyrings/grafana.key] https://apt.grafana.com stable main" | sudo tee -a /etc/apt/sources.list.d/grafana.list`
6. Memasukkan perintah: `sudo apt-get update`
7. Memasukkan perintah: `sudo apt-get install Grafana`
8. Melakukan penyesuaian pada file `/etc/grafana/grafana.ini` dari `“;http_port = 3000”` menjadi `“http_port = 4000”`
9. Melakukan penyesuaian pada file `/etc/grafana/grafana.ini` dari:

```
[auth.anonymous]
# enable anonymous access
;enabled = false
```

menjadi:

```
[auth.anonymous]
# enable anonymous access
enabled = true
```

merubah:

```
[auth.anonymous]
# enable anonymous access
;enabled = false
```

menjadi:

```
[auth.anonymous]
# enable anonymous access
enabled = true
```

merubah:

```
[server]
;root_url = %(protocol)s://%(domain)s:%(port)s/
```

menjadi:

```
[server]
root_url = YourDomain.com
```

merubah:

```
[security]
# set to true if you want to allow browsers to render Grafana in a
<frame>, <iframe>, <embed> or <object>. default is false.
;allow_embedding = false
```

menjadi:

```
[security]
# set to true if you want to allow browsers to render Grafana in a
<frame>, <iframe>, <embed> or <object>. default is false.
allow_embedding = true
```

10. Membuka terminal dan memasukkan perintah: `sudo systemctl start grafana-server`
11. Memasukkan perintah: `sudo systemctl status grafana-server`
12. Memasukkan perintah: `sudo systemctl enable grafana-server.service`
13. Membuka browser dan menuju alamat `localhost:4000` dan melakukan *login*
14. Menginstall *plugin Infinity*
15. Membuat *dashboard*

## V. KESIMPULAN

### 5.1. Kesimpulan

Berdasarkan hasil dan pembahasan yang diperoleh pada penelitian “Perancangan Sistem *Monitoring* dan *Data Logging* Gas Metana ( $CH_4$ ) untuk Analisis Fluks Karbon pada Metode *Eddy Covariance* Berbasis *Internet of Things (IoT)*” ini, bisa dilakukan pengambilan kesimpulan sebagai berikut:

1. Perangkat *system monitoring* gas Metana berhasil dirancang dan berhasil melakukan *monitoring* gas Metana, kelembaban udara, suhu udara, tekanan udara, dan kadar uap air di udara dengan kecepatan *sensing* 1 Hz hingga 9 Hz serta mampu mengirimkan data ke *data logger*.
2. *Data logger* dapat menerima data dari dua perangkat *monitoring* karbon melalui *protocol TCP/IP* dengan membedakan *port* untuk masing-masing perangkat *monitoring* karbon dengan rata-rata data perhari untuk satu *file* yaitu 23 *MegaBytes* jika data yang dikirimkan utuh.
3. Perangkat *solar charge controller* berhasil dilakukan pengambilan data secara terus-menerus setiap satu menit.
4. Data dari perangkat *monitoring* karbon, data dari *solar charge controller*, dan data kondisi perangkat *monitoring* berhasil disimpan di *database Postgresql* dengan baik, namun untuk melakukan tulis data sebesar 1,8 juta data harus menggunakan sumber data komputasi standar yaitu 4 *core* dengan *RAM* 8 *GigaBytes*.
5. *REST API* pada *backend* mampu menangani permintaan pengambilan dan penerimaan data dengan baik, serta *dashboard Grafana* dapat menampilkan data yang diminta.

## 5.2. Saran

Berdasarkan penelitian yang dilakukan, terdapat beberapa saran untuk pengembangan selanjutnya yaitu:

1. Melakukan pemisahan jaringan kampus dan perangkat percobaan supaya jika jaringan kampus terjadi gangguan, perangkat percobaan tidak ikut mengalami gangguan.
2. Menginstall ulang *system operasi* pada *data logger* karena terdapat beberapa kesalahan sistem seperti tidak bisa melakukan akses *vnc* secara lokal dan *firewall* yang tiba-tiba tidak memberi akses.
3. Melakukan penambahan sensor *3D Sonic Anemometer* supaya data bisa lengkap untuk dilakukan perhitungan *Eddy Covariance*.
4. Melakukan penambahan sumber daya server supaya fitur data *monitoring* gas karbon berjalan sempurna.
5. Menerapkan metode *Master Slave* untuk membuat *timestamp* serta jumlah data dari kedua perangkat *Eddy Station* menjadi sama.
6. Membedakan sensing sensor *DHT22* menggunakan *millis* supaya tidak mengganggu data yang akan digunakan untuk kebutuhan analisis *Eddy Covariance*.
7. Mengganti sensor *CH<sub>4</sub>* dengan sensor yang lebih teliti.
8. Melakukan kalibrasi untuk keseluruhan sensor.
9. Melakukan pengecekan berkala untuk memastikan perangkat dapat bekerja secara optimal.
10. Mengembangkan lebih lanjut jika ditemukan metode yang jauh lebih efisien dan disempurnakan.

## DAFTAR PUSTAKA

- [1] indexmundi, "Palm Oil Production by Country in 1000 MT," 2022 [Online]. Available: <https://www.indexmundi.com/agriculture/?commodity=palm-oil&graph=production>. [Accessed: Nov. 9, 2022].
- [2] K. P. RI, "Statistik Perkebunan Indonesia 2018-2020," Kementerian Pertanian RI, Jakarta, 2019.
- [3] B.P.S, "Luas Tanaman Perkebunan Menurut Provinsi (Ribuan Hektar), 2019-2021." 2022 [Online]. Available: <https://www.bps.go.id/indicator/54/131/1/luas-tanaman-perkebunan-menurut-provinsi.html>. [Accessed: Nov. 9, 2022].
- [4] H. L. Chang and S. S. Yang, "Measurement of methane emission from soil," *Journal of Chinese Agricultural, Chemical Society*, vol. 35, pp. 475–484, 1997.
- [5] A. K. Klemedtsson et al, "Greenhouse gas emissions from farmed organic soils: a review," *Soil Use and Management*, vol. 13, pp. 245-250, 1997.
- [6] A. Sasmita, I. Isnaini, and R. Zustaka, "Estimasi Gas Rumah Kaca dari Sektor Pertanian, Perkebunan, dan Peternakan di Kabupaten Kampar, Provinsi Riau," *Jurnal Sumberdaya Alam dan Lingkungan*, vol. 8, no. 1, pp. 42–53, 2021.
- [7] D. Dzaki and Alfian, "Rancang Bangun Alat Pengukuran Gas Metana Menggunakan Arduino Uno pada Septic Tank," Universitas Pembangunan Panca Budi Medan, Medan, 2020.
- [8] K. D., Reni R., Ibrahim, "Implementasi Sensor MQ 4 dan Sensor DHT 22 pada Sistem Kompos Pintar Berbasis IoT (SIKOMPI)," Universitas Singaperbangsa Karawang, Karawang, 2020.

- [9] S., Abas, Nyoman S.W., “Penggunaan MQ-5 sebagai Sensor yang Murah dengan Akurasi Tinggi untuk Pengukuran Konsentrasi Gas Metana dalam Biogas,” Universitas WR Supratman Surabaya, Surabaya, 2021.
- [10] R. Pratama, “Efek Rumah Kaca Terhadap Bumi,” Universitas Islam Sumatera Utara, Sumatera Utara, 2019.
- [11] Handoko, “Pengantar Unsur-unsur Cuaca di Stasiun Klimatologi Pertanian, Jurusan Geofisika dan Metereologi,” FMIPA-IPB, Bogor, 1986.
- [12] Pressbooks, UH, “Water Vapor,” [Online]. Available: <http://pressbooks-dev.oer.hawaii.edu/atmo/chapter/chapter-4-water-vapor/>. [Accessed Nov. 21, 2022].
- [13] Mercy Corps, “Design, Monitoring and Evaluation Guidebook,” Mercy Corps, Portland, USA, 2005.
- [14] G. Burba and D. Anderson, “A Brief Practical Guide to Eddy Covariance Flux Measurements,” LI-COR Biosciences, USA, 2010.
- [15] A.M.W., Ardiansyah dan Sutardi, “Studi Numerik Karakteristik Boundary Layer Turbulen pada Pelat Datar dengan Alur Melintang Tipe-D,” Institut Teknologi Sepuluh Nopember (ITS), Surabaya, 2016.
- [16] DD. BALDOCCHI, “Assessing the eddy covariance technique for evaluating carbon dioxide exchange rates of ecosystems: Past, present and future,” *Global Change Biology*, vol. 9, no. 4, pp. 479–492, 2003.
- [17] G. Burba, “Eddy Covariance Method,” LI-COR Biosciences, pp. 34-35, 2013.
- [18] M. Aubinet, T. Vesala, and D. Papale, *Eddy Covariance: A Practical Guide to Measurement and Data Analysis*. London: Springer, 2012.
- [19] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” *Computer networks*, vol. 54, no. 15, pp. 2787-2805, 2010.