

# Sistem Monitoring Baterai Berbasis IoT Menggunakan Blynk IoT

## IOT BASED BATTERY MONITORING SYSTEM USING BLYNK IOT

Alifvio Arya Sapta Pradana  
Applied Science School  
Telkom University  
Bandung, Indonesia  
[alifvioaryasp270901@gmail.com](mailto:alifvioaryasp270901@gmail.com)

Dr. Nina Hendrarini S.T., M.T.  
School of Applied Science  
Telkom University  
Bandung, Indonesia  
[ninahendrarini@tass.telkomuniversity.ac.id](mailto:ninahendrarini@tass.telkomuniversity.ac.id)

Devie Ryana Suchendra  
S.T.,M.T.  
Applied Science School  
Telkom University  
Bandung, Indonesia  
[deviesuchendra@tass.telkomuniversity.ac.id](mailto:deviesuchendra@tass.telkomuniversity.ac.id)

**Abstract**—*Battery Management System (BMS)* memerlukan pemantauan *realtime* secara cepat, oleh karena itu membutuhkan aplikasi dan jaringan untuk pencatatan jarak jauh untuk memantau aktivitas baterai. BMS jika tidak ada sistem monitor akan menyulitkan sistem pemantauan. Solusinya adalah membuat sistem yang mendukung pemantauan BMS dengan berbasis IOT. Proyek akhir ini bertujuan memakai aplikasi blynk untuk mempermudah user untuk memantau kondisi baterai. Hasil skenario melalui aplikasi blynk dapat menampilkan nilai rata-rata sensor tegangan=3V, arus=25073.28mA, dan soc=30% dan user dapat memantau dari *smartphone* yang sudah ter integrasi oleh aplikasi blynk.

**Kata kunci**— *Sistem Monitoring, Aplikasi Web, IoT*

### I. PENDAHULUAN

Beberapa tahun terakhir, *Internet Of Things (IoT)* telah menjadi daya tarik utama untuk pemantauan dan kontrol baterai. Baterai adalah komponen terpenting untuk perangkat apa pun karena memberi daya pada seluruh sistem. Fungsi utama sistem manajemen baterai berbasis IoT mencakup fasilitas pencatatan data jarak jauh untuk memantau aktivitas baterai. Karena itu, baterai penting untuk memantau tingkat voltase baterai karena pengisian/pengosongan yang tidak tepat atau berlebihan dapat menyebabkan kerusakan pada Baterai. *Battery Management System (BMS)* pada kendaraan listrik memerlukan sistem yg mempermudah monitor proses pengelolaan penggunaan baterai. Sistem yang dimaksud tersebut adalah berupa aplikasi monitoring yang diintegrasikan dengan perangkat *Battery Management System (BMS)*. Aplikasi tersebut akan menampilkan pemantauan kinerja BMS saat mengelola baterai. Aplikasi tersebut menampilkan hasil kalau diimplementasikan pada kendaraan listrik dapat mempengaruhi umur pada baterai dan sejauh mana BMS dapat meningkatkan performa serta keamanan kendaraan. Dengan mengetahui masalah ini lebih awal, seseorang dapat mengambil langkah-langkah perbaikan atau penggantian baterai sebelum masalahnya menjadi lebih parah. Oleh sebab itu, pengembangan BMS harus dilakukan pengawasan rutin tanpa terbatas waktu dan tempat.

Memonitor kondisi baterai sangat sulit maka dari itu untuk membangun dan mengembangkan suatu sistem monitoring ini membutuhkan sebuah Aplikasi. Salah satu kinerja BMS saat mengelola tegangan dan arus baterai kalau diimplementasikan pada kendaraan listrik. Aplikasi yang digunakan dalam membangun monitor pada baterai yaitu Blynk IoT. Kalau sistem ini diimplementasikan di kendaraan listrik dapat mempengaruhi umur pada baterai dan sejauh mana BMS dapat meningkatkan performa serta keamanan kendaraan. Oleh sebab itu, pengembangan BMS harus dilakukan pengawasan rutin tanpa terbatas waktu dan tempat. Permasalahan tersebut maka rumusan masalah dari proyek akhir ini adalah bagaimana cara melakukan pengawasan terhadap baterai dengan mudah melalui internet. Memakai aplikasi blynk untuk mempermudah user untuk memantau kondisi baterai. Menampilkan nilai sensor tegangan, arus dan soc di aplikasi blynk. Dapat memeriksa status baterai di *smartphone* yang sudah terintegrasi oleh aplikasi blynk.

### II. KAJIAN TEORI

#### 2.1 Tinjauan Pustaka

Melakukan penulisan Proyek Akhir mengenai IoT dalam sebuah sistem monitoring sudah pernah diterapkan sebelumnya. Beberapa di antaranya menerapkan IoT dalam sistem monitoring pada baterai. Sistem monitoring pada penelitian.[1] didukung oleh dua perangkat utama yaitu perangkat keras dan perangkat lunak. Perangkat keras yang menjadi komponen utama yaitu sensor voltage dan sensor arus (ACS712). Perangkat lunak digunakan untuk menampilkan data informasi yang dikirim mikrokontroler dalam sebuah tampilan web dan aplikasi *smartphone*. Metode yang sama juga diterapkan pada penelitian.[2] tentang monitoring arus dan tegangan dari baterai serta menganalisis selisih dari tegangan dan arus yang didapatkan dari sistem yang dirancang terhadap alat ukur lain. Sistem monitoring serupa juga dilakukan oleh pada penelitian. [3] Pemantauan jarak jauh baterai mengacu pada kemampuan untuk mengawasi dan memantau status baterai suatu perangkat atau sistem dari lokasi yang jauh, biasanya melalui jaringan atau koneksi internet. Pemantauan baterai jarak jauh memungkinkan manajemen daya yang lebih baik.

Pengguna dapat melihat seberapa efisien baterai digunakan dan mengambil tindakan untuk memperpanjang masa pakai baterai atau meningkatkan efisiensi energi.[4] Penelitian ini diawali dengan pengambilan nilai sensor arus (ACS712) dan nilai voltage. Nilai-nilai ini akan diterima oleh mikrokontroler (Arduino Uno Wifi R3) dikirimkan ke halaman web dan bisa juga diterima di aplikasi smartphone melalui koneksi internet.[5]

## 2.2 Dasar Teori

### A. Mikrokontroler Arduino Uno Wifi R3 + ESP8266

Arduino Uno R3 adalah papan pengembangan mikrokontroler yang berbasis chip ATmega328P. Arduino Uno memiliki 14 digital pin *input / output* (atau biasa ditulis I/O, Dimana 14 pin diantaranya dapat digunakan sebagai *output* PWM antara lain pin 0 sampai 13), 6 pin *input* analog, menggunakan crystal 16 MHz antara lain pin A0 sampai A5, koneksi USB, jack listrik, header ICSP dan tombol reset. [6] Hal tersebut adalah semua yang diperlukan untuk mendukung sebuah rangkaian mikrokontroler.

### B. Sensor Arus ACS712

ACS712 memiliki kemampuan untuk mengukur arus dalam rentang tertentu dan menghasilkan *output* tegangan sebanding dengan arus yang diukur. Sensor ini umumnya digunakan dalam aplikasi seperti pemantauan daya, kendali motor, pengukuran konsumsi energi, dan sistem proteksi. Penggunaan pin pada sensor ini menggunakan 3 pin yaitu pin VCC (Voltage Supply): Ini adalah pin untuk memberikan tegangan suplai ke sensor.

### C. Sensor Voltage

Modul sensor tegangan ini memanfaatkan konsep pembagi tegangan yang mampu meminimalkan tegangan masukan hingga 5 kali lipat dari nilai sebenarnya. Tegangan *input* maksimum untuk Arduino adalah 5V. Oleh karena itu, modul ini hanya mampu menerima input Tegangan maksimal 25V. [7]. Modul ini penggunaan sensor tegangan hanya digunakan untuk menghasilkan keluaran data tegangan baterai.

### D. Baterai Li-Ion

Baterai Li-ion, atau baterai ion litium, adalah jenis baterai isi ulang yang menggunakan litium-ion sebagai elektrolit untuk menggerakkan aliran listrik antara elektroda positif (katoda) dan elektroda negatif (anoda). Baterai ini dikenal karena memiliki kepadatan energi yang tinggi, umur pakai yang relatif lama, dan kurangnya efek memori. [8].

### E. Aplikasi Blynk

Blynk merupakan sebuah platform *Internet of Things* (IoT) dengan tujuan untuk mengendalikan dan memonitoring perangkat keras secara jarak jauh menggunakan internet atau intranet (jaringan LAN). Blynk adalah dashboard digital dengan fasilitas antarmuka grafis dalam pembuatan proyeknya. Penambahan komponen pada aplikasi Blynk dengan cara Drag and Drop sehingga memudahkan dalam penambahan komponen *Input* atau *output* tanpa perlu kemampuan pemrograman Android maupun iOS. Selain itu menggunakan platform ini kita dapat menyimpan data-data dari sensor serta dapat menampilkan hasil pengukuran data secara visual baik menggunakan angka, warna atau grafis yang memudahkan dalam pembuatan proyek di bidang *Internet Of Think*. [9].

### F. Aplikasi Arduino IDE

Arduino IDE (*Integrated Development Environment*) adalah sebuah *software* pengembangan perangkat lunak yang dirancang khusus untuk pemrograman mikrokontroler dan papan pengembangan Arduino. [13] Aplikasi Arduino IDE ini dibutuhkan untuk menulis sebuah program dan mengubahnya ke dalam bentuk kode biner lalu meng-uploadnya ke dalam memori papan Arduino. [14] Arduino IDE menjalankan bahasa pemrograman C++ untuk menjalankan eksekusi yang dikehendaki pada mikrokontroler yang hendak dijalankan.

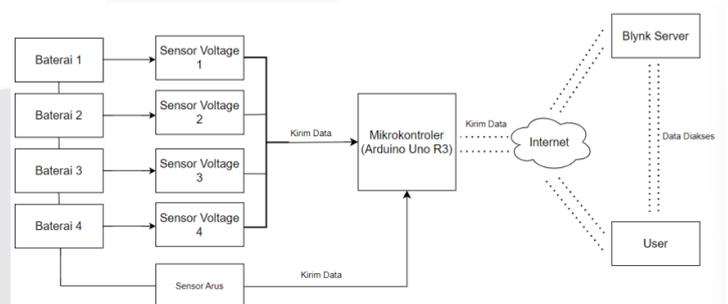
### G. Aplikasi Wireshark

Wireshark adalah sebuah perangkat lunak yang digunakan untuk menganalisis jaringan dan pemantauan lalu lintas data di jaringan komputer. Dengan kemampuannya untuk menangkap dan memeriksa paket-paket data yang melintas di jaringan, Wireshark memungkinkan pengguna untuk memahami lalu lintas jaringan, mendiagnosis masalah jaringan, menemukan kelemahan keamanan, dan melakukan *debugging* pada protokol jaringan.[15]

## III. ANALISIS DAN PERANCANGAN

### 3.1 Gambaran Sistem Usulan

Gambaran Sistem berikut merupakan gambaran yang telah direncanakan pada proyek akhir ini. Proses memonitoring baterai menggunakan beberapa sensor dan peralatan lainnya. Data baterai yang diambil melalui sensor akan secara otomatis masuk ke dalam aplikasi Blynk. Pengguna dapat melihat secara *realtime* pengukuran hasil baterai menggunakan smartphone melalui aplikasi Blynk tersebut. Pada bagian ini sensor yang digunakan adalah sensor arus dan sensor tegangan. Setelah data baterai didapat oleh sensor lalu dikirimkan ke mikrokontroler berupa Arduino UNO R3 WiFi (diubah ke mode NodeMCU). Gambaran sistem saat ini dalam pembahasan proyek akhir adalah sebagai berikut:



Gambar 1 Gambaran Sistem Usulan

Spesifikasi perangkat keras dari perancangan sistem monitoring baterai adalah :

- 1) Sensor voltage berfungsi untuk mengukur tegangan dari suatu rangkaian baterai. Sensor tegangan dapat memberikan keluaran analog atau digital tergantung pada tipe dan modelnya.
- 2) Sensor ACS712 merupakan sensor modul yang digunakan untuk mengukur arus yang mengalir melalui rangkaian dan tegangan pada titik tertentu. Dengan demikian, sensor ini cocok digunakan oleh baterai.

Spesifikasi perangkat lunak dari perancangan sistem monitoring baterai adalah :

1) Software yang digunakan pada Proyek Akhir ini menggunakan Aplikasi Blynk untuk monitoring sebuah data pada baterai.

2) Bahasa pemrograman C++ pada mikrokontroler digunakan untuk pemuatan program yang berisi pengambilan data pada sensor dan diteruskan ke Arduino Uno R3 (diubah ke mode Node MCU ESP8266).

3) Pada Arduino Uno R3 juga menggunakan bahasa pemrograman C++ yang digunakan sebagai penerima data dari sensor lalu data tersebut dikirim ke server Blynk.

### 3.4 Perancangan Sistem

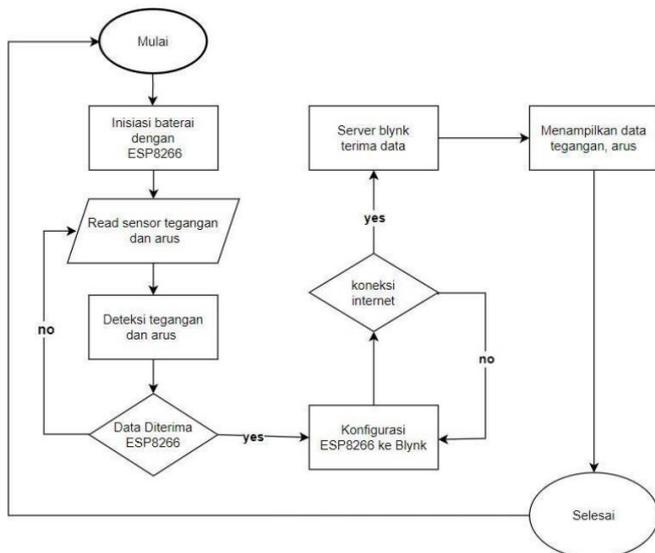
Sistem dibagi menjadi dua bagian yaitu bagian *hardware* dan *software*. Didalam sistem monitoring ini terdapat tiga tahapan sistem kerja. Berikut merupakan uraian dari tiga tahapan sistem kerja :

1. *Input*, dibagian ini sensor arus dan sensor voltage yang telah dipasang pada baterai yang dilakukan pengukuran akan mengeluarkan nilai. Nilai dari keluaran sensor akan disimpan lalu diteruskan ke mikrokontroler.

2. Mikrokontroler, peran dari bagian ini sebagai pengambil nilai sensor yang dilakukan oleh Arduino UNO WiFi R3 (diubah ke mode NodeMCU ESP8266). Selanjutnya nilai yang telah diambil NodeMCU yang menerima nilai sensor akan mendata apakah nilai semua sensor sudah terima, jika sudah diterima maka nilai akan diteruskan ke server Blynk.

3. *Output*, disini merupakan keluaran nilai yang telah diproses dan dianalisis sehingga menghasilkan data ulang valid sesuai dengan nilai yang telah dikirimkan oleh sensor. Aplikasi Blynk akan mengambil data yang telah diolah pada NodeMCU. Nilai akan ditampilkan dalam *smartphone* atau PC

### 3.5 Flowchart

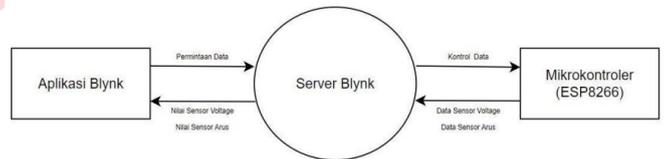


Gambar 2 Flowchart

Gambar 2 menggambarkan inisiasi program Arduino IDE agar dapat menghubungkan NodeMCU Esp8266 ke Wi-Fi seperti yang telah diatur pada program Arduino IDE. NodeMCU Esp8266 yang sudah terhubung dengan Wi-Fi akan membaca data dari sensor. Selanjutnya sensor akan mendeteksi tegangan dan arus yang akan dikirimkan ke server Blynk melalui NodeMCU Esp8266. Kemudian user masuk ke aplikasi Blynk yang terkoneksi dengan internet untuk melihat hasil data yang keluar dari NodeMCU Esp8266 tersebut.

### 3.6 Pemodelan DFD (Data Flow Diagram)

Data Flow Diagram (DFD) adalah DFD digunakan dalam analisis sistem dan perancangan perangkat lunak untuk menggambarkan alur data melalui berbagai proses. DFD membantu dalam memahami bagaimana data diproses di dalam suatu sistem dan bagaimana informasi berpindah antara berbagai komponen sistem.



Gambar 3 DFD Level 0

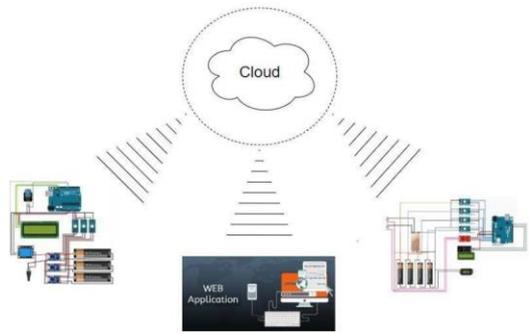
Pada sistem ini merupakan DFD level 0. Menunjukkan bahwa data melalui masukan eksternal ke dalam server Blynk, data ini berasal dari satu input eksternal yaitu Mikrokontroler. Setelah system menerimanya, data tersebut diolah hingga dapat dikirim data output menuju Aplikasi Blynk.



Gambar 4 DFD Level 1

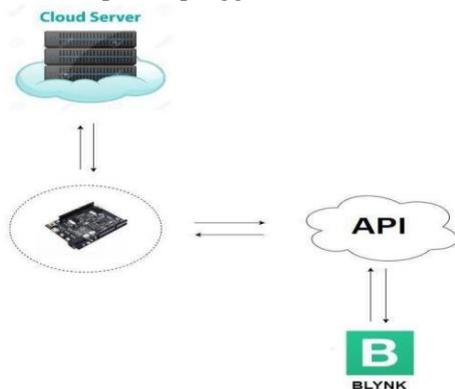
Pada gambar 4 menampilkan Data Flow Diagram (DFD) level 1, Dalam system ini input setiap data masukan akan melalui beberapa proses sampai akhirnya menghasilkan output berupa Aplikasi Blynk pada smartphone.

### 3.7 Perancangan Grand Topologi



Gambar 5 Grand Topologi

Pada gambar 5 menunjukkan bahwa Grand Topologi merupakan sistem untuk berjalannya sebuah sistem monitoring. Sistem yang saling integrasi antara hardware dan software agar berjalan dengan baik dan menyediakan antarmuka untuk aplikasi pengguna.



Gambar 6 Cloud Server

Pada gambar 6 menunjukkan sebuah sistem rancangan cloud server yang bertujuan untuk proses pengambilan data atau pengiriman data yang disimpan di cloud server. Dan selanjutnya data yang disimpan, pengguna dapat mengakses cloud server yang terhubung dengan internet untuk menampilkan hasil data yang diterima melalui aplikasi Blynk.

### 3.8 Metode Penelitian

Metode penelitian disini untuk memecahkan suatu masalah yang membahas tentang pengujian Parameter delay, jitter, throughput dan packet loss. Diuji berdasarkan hasil pengamatan menggunakan software wireshark. Melakukan pengujian menggunakan provider dari Indosat.

### 3.9 QoS (Quality Of Service)

Untuk menguji performa sistem saat terjadinya pengiriman data ke jaringan internet maka akan diuji menggunakan software Wireshark. Aplikasi ini umum digunakan sebagai alat troubleshoot pada jaringan yang bermasalah, selain itu juga bisa digunakan untuk pengujian software karena kemampuannya untuk membaca konten dari tiap paket trafik data. Parameter untuk menguji sistem pengiriman data pada Perangkat mikrokontroler ESP8266 adalah :

1. Delay
2. Jitter
3. Throughput
4. Packet Loss

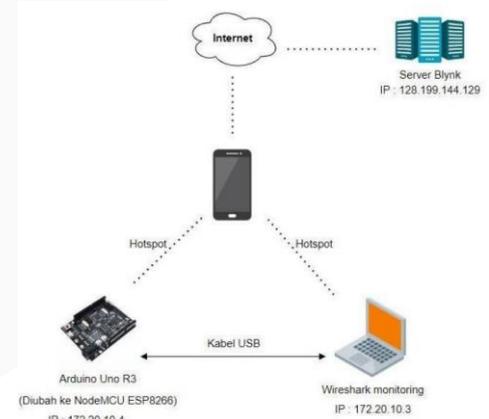
## IV. IMPLEMENTASI DAN PENGUJIAN

### 4.1 Implementasi

Implementasi pada proyek ini membahas tentang hasil pengujian dan analisa sistem yang dirancang dan dibuat, yang bertujuan untuk mengetahui sistem, pengecekan sistem yang dibuat serta menerapkan sistem telah memenuhi kriteria yang diinginkan.

#### 1. Skenario Pengujian Pengiriman Data

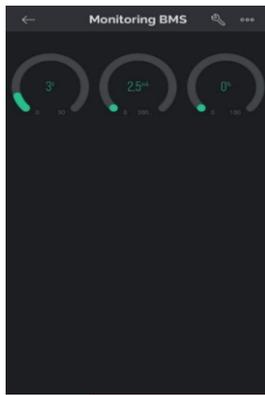
Pengujian dilakukan dengan skenario dengan transport data yaitu menggunakan jaringan LTE Indosat. Traffic data yang dipantau yaitu pada bagian hardware. Adapun skema hirarki pengujian pengiriman data ini terdapat pada gambar berikut.



Gambar 10 Hirarki Pengujian Pengiriman Data

#### 2. Perancangan Sistem Monitoring

Berikut tampilan hasil rancangan perangkat lunak untuk sistem monitoring menggunakan smartphone IOS versi 16.7.4 A11 Bionic Iphone.



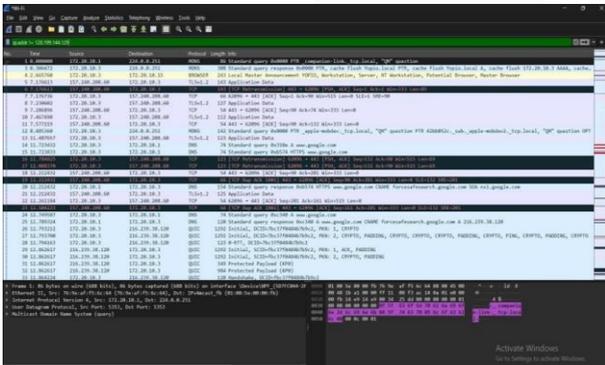
Gambar 11 Tampilan Sistem Monitoring

#### 4.2 Pengujian

Pengujian dilakukan terhadap sistem *monitoring* dengan alat lalu membandingkan nilai yang dibaca oleh sensor dengan alat pengukur yang ada. Setelah itu mengirimkan data pembacaan nilai sensor kemudian dikirimkan pada *server* Blynk yang diteruskan ke aplikasi Blynk yang terhubung dengan template.

##### 1. Pengujian Hasil Pengiriman Data

Laptop yang sudah terinstall *software* wireshark dijadikan gateway untuk traffic data Node MCU agar memudahkan dalam memonitoring traffic datanya tanpa harus memecahkan enkripsi ketika di dihubungkan dalam satu network wifi. Parameter yang menjadi standar pengukuran integritas dari pengiriman data diantaranya adalah delay, jitter, throughput dan Packet Loss. Contoh proses capture traffic menggunakan wireshark seperti gambar berikut



Gambar 12 Tampilan Capture Wireshark

Data yang sudah dicapture kemudian diolah menggunakan microsoft excel untuk dihitung masing masing nilai dari keempat parameter dengan dua skenario transport data. Berdasarkan TIPHON (Telecommunications and Internet Protocol Harmonization Over Network) yang merupakan standar penilaian parameter QoS yang dikeluarkan oleh badan standar ETSI (European Telecommunications Standards Institute).

Delay dapat dihitung menggunakan rumus:

$$\text{Rata - rata Delay} = \frac{\text{Total Delay}}{\text{Total Paket yang diterima}}$$

Untuk Standarisasi nilai jitter berdasarkan TIPHON adalah sebagai berikut.

Jitter dapat dihitung dengan rumus.

$$\text{Jitter} = \frac{\text{Total variasi Delay}}{\text{Total Paket yang diterima} - 1}$$

Total variasi delay diperoleh dari penjumlahan :

$$(\text{delay } 2 - \text{delay } 1) + (\text{delay } 3 - \text{delay } 2) + \dots (\text{delay } n - \text{delay } (n-1))$$

Untuk standarisasi nilai Packet Loss berdasarkan TIPHON adalah sebagai berikut.

$$\text{Packet Loss} = \frac{\text{Paket data yang dikirim} - \text{paket data yang diterima}}{\text{Paket data yang dikirim}} \times 100$$

Sedangkan Throughput bisa dihitung menggunakan rumus.

$$\text{Throughput} = \frac{\text{Jumlah Bytes}}{\text{Time Span}}$$

Pengamatan dilakukan kurang lebih selama 5 menit atau 300 detik pada saat jam 11-12 malam. Data yang dihasilkan saat capture traffic ada pada tabel 1.

Tabel 1 Data Hasil Capture Traffic pada Hardware

Transport	LTE INDOSAT	
Sumber	Server Blynk	NodeMCU
Tujuan	NodeMCU	Server Blynk
IP Source	128.199.144.129	172.20.10.4
IP Destination	172.20.10.4	128.199.144.129
Jumlah Packet Data yang dikirim	3587	3456
Jumlah packet Data yang diterima	3587	3456
Total Delay (s)	208.474	206.062
Total Variasi Delay (s)	0,288729036	295,947262
Packet Data yang diterima (bytes)	4373029	4339570

Untuk perhitungan delay, jitter, throughput dan packet loss saat pengiriman data dari *server* blynk ke nodeMCU menggunakan transport LTE Indosat. Menggunakan data yang ada pada tabel 1. Untuk mendapatkan rata rata delay per packet dihitung, maka didapatkan rata-rata delay untuk pengiriman data dari server blynk ke Node MCU adalah

Tabel 2 Kategori Hasil Delay

Sesi pengujian	Waktu Pengujian (menit)	Delay (ms)
Pagi	5	152,06
Sore	5	118,18
Malam	5	58,11

Dari hasil perhitungan di dapatkan sesi pertama untuk masing-masing parameter menghasilkan rata-rata nilai delay kurang dari 10 menit.



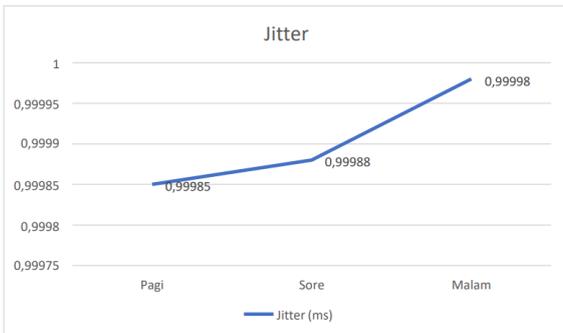
Gambar 13 Grafik Delay

Sedangkan nilai Jitter maka didapat nilai Jitter selama proses capture traffic adalah

Tabel 3 Kategori Hasil Jitter

Sesi pengujian	Waktu Pengujian (menit)	Jitter (ms)
Pagi	5	0,99985
Sore	5	0,99988
Malam	5	0,99998

Dari hasil perhitungan di dapatkan sesi kedua untuk masing-masing parameter menghasilkan rata-rata nilai delay kurang dari 10 menit.



Gambar 14 Grafik Jitter

Nilai packet loss dihitung menggunakan rumus maka didapat nilai packet loss selama proses capture traffic adalah.

Tabel 4 Kategori Hasil Packet Loss

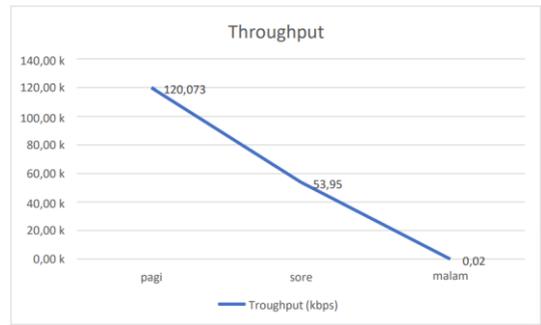
Sesi pengujian	Waktu Pengujian (menit)	Packet Loss (%)
Pagi	5	0
Sore	5	0
Malam	5	0

Nilai throughput dihitung menggunakan rumus maka didapat nilai throughput selama proses capture traffic adalah

Tabel 5 Kategori Hasil Throughput

Sesi pengujian	Waktu Pengujian (menit)	Throughput (kbps)
Pagi	5	120,073
Sore	5	0,04316
Malam	5	0,02097

Dari hasil perhitungan di dapatkan sesi ketiga untuk masing-masing parameter menghasilkan rata-rata nilai delay kurang dari 10 menit.



Gambar 15 Grafik Throughput

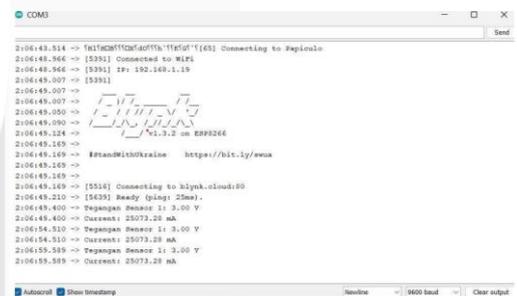
Data hasil pengukuran traffic data lalu kemudian dihitung untuk mencari nilai QoS dihasilkan pada table berikut.

Tabel 6 Hasil Pengukuran QoS

Transport	LTE INDOSAT
Sumber	NodeMCU
Tujuan	Server Blynk
IP Sumber	128.199.144.129
IP Tujuan	172.20.10.4
Delay (ms)	58.11
Jitter (ms)	0.99
Throughput (kbps)	0.02
Packet Loss (%)	0

## 2. Pengujian Hasil Temperature Dan Arus

Pengujian sensor voltage dan sensor arus yang dilakukan adalah dengan membandingkan nilai yang telah dibaca oleh sensor yang di serial monitor dengan gauge display pada aplikasi Blynk.



Gambar 16 Hasil Tegangan Dan Arus Pada Serial Monitor

Berikut dari gambar 16 ditampilkan hasil pengujian pada serial monitor pada software Arduino IDE terlihat menunjukkan informasi nilai sensor tegangan di angka 3.00 V, sedangkan nilai yang di sensor arus menunjukkan di angka 25073.28 mA.

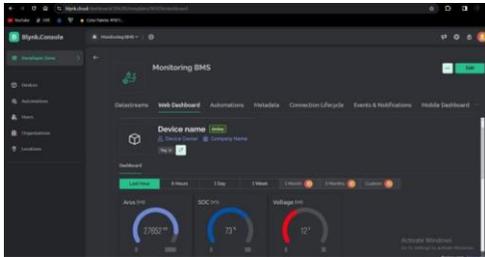


Gambar 17 Hasil Tegangan Dan Arus Pada Gauge Display

Berikut dari gambar 17 ditampilkan hasil dari pengujian yang ada di template Blynk, menunjukkan nilai hasil pengukuran dari yang di serial monitor tadi data yang diambil dengan angka yang sama yaitu sensor tegangan yang di angka 3 V, sedangkan yang di sensor arus di angka 2.5 mA.

### 3. Integrasi NodeMCU ESP8266 ke Blynk

Untuk mengintegrasikan NodeMCU ESP8266, terlebih dahulu harus dibuat template pada Blynk. Template yang dibikin harus sesuai untuk menampilkan output yang diterima dari NodeMCU ESP8266, yaitu untuk menampilkan data dari tegangan dan arus. Untuk membuat template nya bisa pada link Blynk console.



Gambar 18 Template Blynk Console

Setelah template selesai dibuat, Blynk akan memberikan Auth Token berupa kode yang akan dimasukkan ke pemrograman seperti gambar 19 berikut ini.

```
#define BLYNK_TEMPLATE_ID "TMPL6jRYriHdy"
#define BLYNK_TEMPLATE_NAME "Monitoring BMS"
#define BLYNK_AUTH_TOKEN "3Sj9z5WCRbhvmO-EhRVsr2mSc9tLUv6v"
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
```

Gambar 19 Auth Token pada Blynk Console

### 4. Menghubungkan Internet dan Blynk

Pada aplikasi Blynk untuk menghubungkan NodeMCU dengan internet menggunakan jaringan wifi, untuk menghubungkannya memerlukan kode program dengan ketik ssid dan password wifi yang mau dihubungkan. Pada kode program juga terdapat kode token yang berfungsi untuk menghubungkan NodeMCU ke akun aplikasi Blynk pengguna smartphone, yang tiap pengguna mempunyai token yang berbeda.

```
char auth[] = "KODE_TOKEN_BLYNK";
char ssid[] = "NAMA_WIFI";
char pass[] = "PASSWORD_WIFI";
```

Gambar 20 Kode Program Menghubungkan Internet dan Blynk

### 5. Pengujian Hasil Data

Tabel 7 Hasil Data Voltage

Data Ke-	Data Voltage pada Serial Monitor	Data Voltage pada Blynk	soc	Keterangan
1	3.00 V	3.00 V	30%	Berhasil
2	3.00 V	3.00 V	30%	Berhasil
3	3.00 V	3.00 V	30%	Berhasil
4	2.85 V	2.85 V	30%	Berhasil
5	1.52 V	1.52 V	30%	Berhasil
6	3.65 V	3.65 V	30%	Berhasil
7	1.30 V	1.30 V	30%	Berhasil
8	2.00 V	2.00 V	30%	Berhasil
9	2.33 V	2.33 V	30%	Berhasil
10	3.89 V	3.89 V	30%	Berhasil
11	1.02 V	1.02 V	30%	Berhasil
12	1.48 V	1.48 V	30%	Berhasil
13	2.01 V	2.01 V	30%	Berhasil
14	3.22 V	3.22 V	30%	Berhasil
15	2.30 V	2.30 V	30%	Berhasil
16	1.43 V	1.43 V	30%	Berhasil
17	2.41 V	2.41 V	30%	Berhasil
18	1.90 V	1.90 V	30%	Berhasil
19	2.73 V	2.73 V	30%	Berhasil
20	3.33 V	3.33 V	30%	Berhasil

Tabel 8 Hasil Data Current

Data Ke-	Data Current pada Serial Monitor	Data Current pada Blynk	soc	Keterangan
1	25073.28 mA	25073.28 mA	30%	Berhasil
2	25073.28 mA	25073.28 mA	30%	Berhasil
3	25073.28 mA	25073.28 mA	30%	Berhasil
4	24253.46 mA	24253.46 mA	30%	Berhasil
5	23043.22 mA	23043.22 mA	30%	Berhasil
6	21174.08 mA	21174.08 mA	30%	Berhasil
7	24063.27 mA	24063.27 mA	30%	Berhasil
8	25239.84 mA	25239.84 mA	30%	Berhasil
9	26604.28 mA	26604.28 mA	30%	Berhasil
10	22005.50 mA	22005.50 mA	30%	Berhasil
11	20073.70 mA	20073.70 mA	30%	Berhasil
12	25093.29 mA	25093.29 mA	30%	Berhasil
13	21001.93 mA	21001.93 mA	30%	Berhasil
14	23074.16 mA	23074.16 mA	30%	Berhasil
15	25073.78 mA	25073.78 mA	30%	Berhasil
16	25137.81 mA	25137.81 mA	30%	Berhasil
17	24015.32 mA	24015.32 mA	30%	Berhasil
18	22443.47 mA	22443.47 mA	30%	Berhasil
19	21394.05 mA	21394.05 mA	30%	Berhasil
20	26023.68 mA	26023.68 mA	30%	Berhasil

## REFERENSI

### V. KESIMPULAN

Kesimpulan etelah dilakukan implementasi dan pengujian sistem monitoring di atas, dapat disimpulkan sebagai berikut:

1. Telah memudahkan user untuk memantau baterai dari mana saja memakai aplikasi blynk
2. aplikasi blynk berhasil menampilkan nilai rata-rata sensor tegangan=3V , arus=25073.28mA, dan soc=30%.
3. User bisa memantau kondisi baterai dari smartphone melalui aplikasi blynk.

Setelah proyek ini dikerjakan terdapat beberapa saran yang dapat diterapkan dalam penelitian berikutnya.

1. Pada pengembangan selanjutnya dapat difokuskan lebih pada bagian proteksi terhadap overcharging, overdischarging, overheating, dan perlindungan arus lebih.
2. Pada penggunaan sensor tegangan dan arus perlu dipertimbangkan lagi agar pengukuran sensor dapat lebih presisi.
3. Pada tampilan aplikasi bisa ditambahkan dengan fitur-fitur seperti notifikasi informasi baterai.

- [1] M. S. Ismail, *RANCANG BANGUN SMOKE DETECTOR BERBASIS MIKROKONTROLER ARDUINO UNO R3*, Jakarta: Institut Teknologi Telkom Jakarta, D3 Teknik Telekomunikasi, 2015, p. 11.
- [2] T. Suryana, *MENGIRIM DATA HASIL PENGUKURAN HUMIDITY DAN TEMPERATURE SENSOR DHT11 DENGAN ARDUINO UNO WIFI R3 ATMEGA328P ESP8266*, Bandung: Universitas Komputer Indonesia, Teknik Infomatika, 2021.
- [3] N. KHOIRUNNISSA, *INVESTIGASI MODEL DAN KESEIMBANGAN KEADAAN BATERAI PADA SISTEM MANAJEMEN BATERAI BERBASIS DIGITAL TWIN - WRAP Entrepreneurship (Capstone)*, vol. 007, Bandung: Universitas Telkom, S1 Teknik Fisika, 2023, pp. 1-10.
- [4] A. S. Ritha Sandra Veronika Simbar, "Prototype Sistem Monitoring Temperatur Menggunakan Arduino Uno R3 Dengan Komunikasi Wireless," *Jurnal Teknologi Elektro, Universitas Mercu Buana*, vol. 2, no. 1, pp. 1-7, 2017.
- [5] A. W. Aditya and Ihsan Ihsan, "Rancang Bangun Battery Monitoring System (BMS) berbasis LabVIEW," *JTT (Jurnal Teknologi Terpadu)*, vol. 9, no. 1, pp. 44-49, October 2021.
- [6] U. Nursusanto, K. and H. , "Real Time Battery Monitoring Control in Mini Generating System," *Jurnal Edukasi Elektro*, vol. 6, no. 1, pp. 96-104, Agustus 2022.
- [7] M. N. H. Lubudi, *RANCANG BANGUN BATTERY MANAGEMENT SYSTEM ACTIVE BALANCING PADA BATERAI LI-ION 12V 2,5Ah*, vol. 1, Yogyakarta, 2020, pp. 155-162.
- [8] A. Khumaidi, "Sistem Monitoring dan Kontrol Berbasis Internet of Things untuk Penghematan Listrik pada Food and Beverage," *JURNAL ILMIAH MERPATI*, vol. 8, no. 1, p. 3, June 2020.
- [9] R. Hariri, M. A. Novianta and S. Kristiyana, "PERANCANGAN APLIKASI BLYNK UNTUK MONITORING DAN KENDALI PENYIRAMAAN TANAMAN," *Jurnal Elektrikal*, vol. 6, no. 10, pp. 1-10, October 2019.

- [10] S. N. Fataha, "Perancangan alat pengukur suhu air laut dengan sensor LM35," *Jurnal Ilmiah Teknik Elektro*, vol. 6, no. 1, p. 1, June 2019.
- [11] A. Salamah, R. Kusumanto and E. , "SISTEM MONITORING VOLUME DAN BERAT SAMPAH PADA ALAT PEMILAH SAMPAH ORGANIK DAN ANORGANIK BERBASIS INTERNET OF THINGS MENGGUNAKAN APLIKASI BLYNK," *Junal.polsri*, vol. 16, no. 2, pp. 1-6, 2023.
- [12] R. Aditya, V. H. Pranatawijaya and P. B. A. A. Putra, "Penggunaan Aplikasi Blynk untuk Sistem Monitoring dan Kontrol Jarak Jauh pada Sistem Kompos Pintar berbasis IoT," *JOINTECOMS (Journal of Information Technology and Computer Science)*, vol. 1, no. 1, p. 48, October 2021.
- [13] R. Aditya, V. H. Pranatawijaya and P. B. A. A. Putra, "Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode Prototype," *JOINTECOMS (Journal of Information Technology and Computer Science)*, vol. 1, no. 1, p. 48, June 2021.
- [14] N. I. WIDIASTUTI and R. SUSANTO, "KAJIAN SISTEM MONITORING DOKUMEN AKREDITASI TEKNIK INFORMATIKA UNIKOM," *Majalah Ilmiah UNIKOM*, vol. 12, no. 2, p. 196, 2014.
- [15] F. S. Permana, M. N. Suyatno Putro and R. Suwartika, "PEMANFAATAN TEKNOLOGI CLOUD BLYNK DALAM SISTEM KONTROLLING STOP KONTAK LAMPU RUMAH BERBASIS APLIKASI ANDROID," *JUTEKIN*, vol. 9, no. 10, p. 2, April 2021.