

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem

Implementasi sistem merupakan tahap realisasi dari rancangan perangkat keras dan perangkat lunak yang telah dirancang pada bab sebelumnya. Sistem ini bertujuan untuk memantau posisi hewan ternak secara jarak jauh menggunakan teknologi GPS dan pengiriman informasi melalui SMS. Perangkat utama yang digunakan terdiri dari Arduino Nano sebagai pengendali, modul GPS Neo-6M untuk mendeteksi koordinat lokasi, dan modul GSM SIM800L V2 untuk mengirimkan koordinat tersebut ke pengguna.

Sumber daya sistem menggunakan satu buah baterai Li-Po berkapasitas 2500 mAh yang disuplai melalui modul charger dan step-up converter all-in-one, sehingga mampu memberikan tegangan stabil untuk seluruh rangkaian. Seluruh komponen diintegrasikan dalam satu rangkaian *compact* yang dirakit dalam *casing* portabel agar mudah digunakan di lapangan seperti terlihat pada Gambar berikut ini.



Gambar 4. 1 GPS Tracker

Implementasi sistem dibagi menjadi dua bagian utama, yaitu implementasi perangkat keras dan implementasi perangkat lunak yang dijelaskan secara rinci pada subbab berikut.

4.1.1 Implementasi Perangkat Keras

Perangkat keras dari sistem GPS Tracker ini mencakup beberapa komponen utama yang dihubungkan dan dirangkai secara sistematis agar dapat berfungsi secara optimal. Berikut ini adalah uraian tiap komponen:

1. Arduino Nano

Digunakan sebagai mikrokontroler utama untuk memproses data dari GPS dan mengendalikan komunikasi dengan modul GSM. Dipilih karena ukurannya yang kecil dan konsumsi daya rendah.

2. Modul GPS Neo-6M

Berfungsi untuk menerima sinyal satelit dan mengirimkan data koordinat (*latitude* dan *longitude*) ke Arduino. Modul ini dihubungkan ke pin RX/TX Arduino melalui komunikasi serial.

3. Modul GSM SIM800L V2

Digunakan untuk menerima dan mengirim SMS. Modul ini terhubung ke Arduino dan akan mengirimkan koordinat lokasi dalam bentuk tautan Google Maps ke nomor pengguna.

4. Baterai Li-Po 2500 mAh

Menyediakan sumber daya utama untuk seluruh sistem. Baterai ini dipilih karena ringan, tahan lama, dan dapat diisi ulang.

5. Modul Charger + Step-Up All-in-One

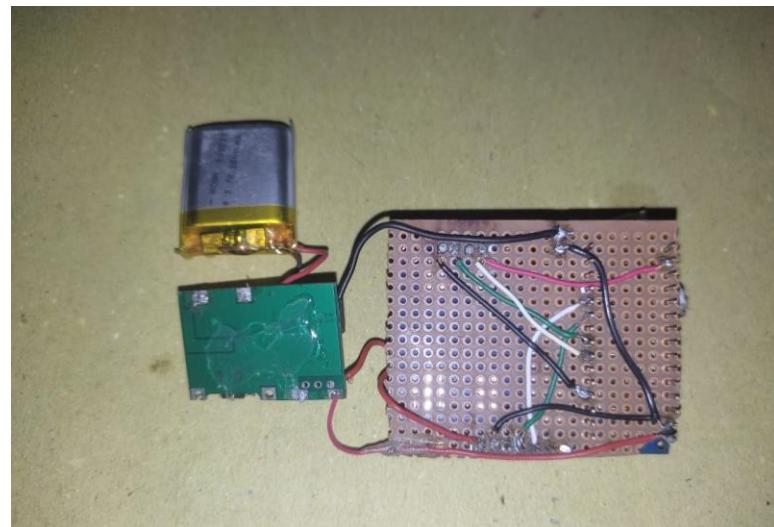
Modul ini menggabungkan fungsi pengisian dan penguat tegangan.

TP4056 digunakan untuk *charging*, sedangkan step-up digunakan untuk menaikkan tegangan dari 3.7V menjadi 5V agar sesuai dengan kebutuhan Arduino dan modul SIM800L V2.

Seluruh rangkaian komponen utama disusun di atas papan PCB lubang dan dipasang di dalam casing agar lebih kokoh dan ringkas. Gambar 4.2 berikut menunjukkan kondisi akhir perangkat keras setelah proses perakitan dan integrasi semua modul.



Gambar 4. 2 GPS Dirakit Pada PCB Lubang



Gambar 4. 3 Wiring Perangkat GPS pada PCB

Pada Gambar 4.3 diatas semua modul yang digunakan untuk membuat perangkat GPS Tracker disatukan dalam sebuah PCB lubang berukuran 5x4,5cm dan berikut adalah rincian koneksi pin atau wiring antar komponen utama yang digunakan dalam sistem GPS Tracker ini:

Tabel 4. 1 *Wiring* Keseluruhan Perangkat

No	Komponen	Pin Modul	Pin Arduino Nano	Keterangan
1	Modul GPS Neo-6M	TX	D4 (RX SoftwareSerial)	Mengirim data koordinat ke Arduino
		RX	D5 (TX SoftwareSerial)	Menerima perintah (jarang digunakan)
		VCC	3.3V	Sumber daya (perlu 3.3V stabil)
		GND	GND	Ground umum
2	Modul GSM SIM800L V2	TX	D7 (RX SoftwareSerial)	Mengirim SMS ke pengguna
		RX	D8 (TX SoftwareSerial)	Menerima SMS dari pengguna
		VCC	5V dari Step-Up	Dihubungkan dari <i>output</i> charger
		GND	GND	Ground
3	Baterai Li-Po 3.7V	+	BAT+ (modul charger)	Sumber daya sistem
		-	BAT- (modul charger)	
4	Modul Charger + Step-Up	OUT+	VIN Arduino / VCC modul GSM	Tegangan dinaikkan (5V atau 4.2V)
		OUT-	GND	Ground sistem
		BAT+ BAT-	Baterai Li-Po	Terminal <i>input</i> pengisian baterai
		USB IN	-	Untuk charging via USB / jack

4.1.2 Implementasi Perangkat Lunak

Perangkat lunak dalam sistem GPS Tracker ini dirancang untuk menjalankan seluruh logika utama dari proses monitoring lokasi, mulai dari

komunikasi serial dengan modul GPS, penerimaan SMS perintah dari pengguna, hingga pengiriman balasan berisi koordinat lokasi dalam bentuk tautan Google Maps. Implementasi program ini ditulis menggunakan bahasa pemrograman C/C++ pada platform Arduino IDE.

Struktur program memanfaatkan dua buah komunikasi serial software (*SoftwareSerial*) yang mengatur komunikasi antara Arduino Nano dengan dua modul eksternal, yaitu GPS Neo-6M dan GSM SIM800L. Komunikasi ini dilakukan secara bergantian agar tidak terjadi konflik pin, mengingat Arduino Nano memiliki jumlah pin terbatas.

Logika kerja program dibagi menjadi beberapa bagian utama sebagai berikut:

1. Inisialisasi Komponen

Pada tahap awal, sistem melakukan inisialisasi komunikasi serial antara Arduino dengan modul GPS dan GSM menggunakan library SoftwareSerial. Modul SIM800L juga dikonfigurasi ke mode teks (SMS) dan diatur agar dapat menampilkan SMS secara otomatis saat diterima.

```
GPS_Tracker.ino
1 #include <SoftwareSerial.h>
2 #include <TinyGPSPlus.h>
3
4 TinyGPSPlus gps;
5 SoftwareSerial gpsSerial(5, 4);      // GPS: TX ke 5, RX ke 4
6 SoftwareSerial sim800l(7, 8);        // GSM: TX ke 7, RX ke 8
7
8 String currentline = "";
9 String senderNumber = "";
10 bool waitingForMessage = false;
11
12 void setup() {
13   Serial.begin(9600);
14   gpsSerial.begin(9600);
15   sim800l.begin(9600);
16
17   Serial.println("Memulai GPS Tracker...");
18
19   delay(1000);
20   sim800l.listen(); // Fokus dulu ke GSM
21   sim800l.println("AT");
22   delay(500);
23   sim800l.println("AT+CMGF=1"); // Mode teks
24   delay(500);
25   sim800l.println("AT+CNMI=1,2,0,0,0"); // Tampilkan otomatis SMS masuk
26   delay(500);
27
28   Serial.println("GSM dan GPS siap.");
29 }
```

Gambar 4. 4 Sketch inisialisasi Awal Semua Komponen

2. Pembacaan Data GPS

Data GPS dibaca secara terus-menerus melalui gpsSerial, lalu di-decode menggunakan library TinyGPSPlus. Sistem akan menunggu data lokasi valid dari satelit sebelum digunakan.



```

GPS_Tracker.ino

70     gpsSerial.listen(); // Fokus ke GPS
71     unsigned long timeout = millis() + 5000;
72     while (millis() < timeout) {
73         while (gpsSerial.available()) {
74             gps.encode(gpsSerial.read());
75         }
76         if (gps.location.isUpdated()) break;
77     }
78
79     if (gps.location.isValid()) {
80         double lat = gps.location.lat();
81         double lng = gps.location.lng();
82     }

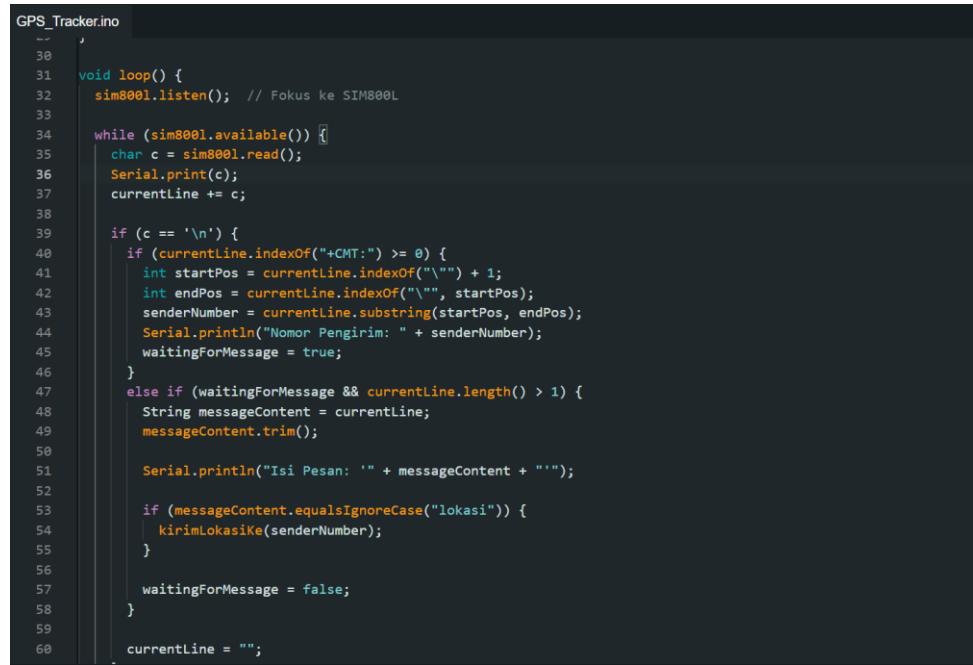
```

Gambar 4. 5 Sketch Pembacaan Data GPS

Program menunggu hingga data koordinat terbaru berhasil diperoleh dari GPS dalam waktu maksimum 5 detik. Bila valid, data latitude dan longitude siap digunakan.

3. Penerimaan SMS

Arduino terus memantau modul SIM800L untuk mendeteksi apakah ada pesan SMS masuk. Ketika pesan diterima, sistem akan mengambil nomor pengirim dari baris metadata +CMT: dan menunggu isi pesan pada baris berikutnya. Jika isi pesan mengandung kata kunci "lokasi", maka sistem akan memanggil fungsi kirimLokasiKe().



```

GPS_Tracker.ino
-- 
30
31 void loop() {
32     sim800l.listen(); // Fokus ke SIM800L
33
34     while (sim800l.available()) {
35         char c = sim800l.read();
36         Serial.print(c);
37         currentLine += c;
38
39         if (c == '\n') {
40             if (currentLine.indexOf("+CMT:") >= 0) {
41                 int startPos = currentLine.indexOf("") + 1;
42                 int endPos = currentLine.indexOf("}", startPos);
43                 senderNumber = currentLine.substring(startPos, endPos);
44                 Serial.println("Nomor Pengirim: " + senderNumber);
45                 waitingForMessage = true;
46             }
47             else if (waitingForMessage && currentLine.length() > 1) {
48                 String messageContent = currentLine;
49                 messageContent.trim();
50
51                 Serial.println("Isi Pesan: " + messageContent + "");
52
53                 if (messageContent.equalsIgnoreCase("lokasi")) {
54                     kirimLokasiKe(senderNumber);
55                 }
56
57                 waitingForMessage = false;
58             }
59
60             currentLine = "";
61         }
62     }
63 }

```

Gambar 4. 6 *Sketch* Penerimaan SMS

Setelah menerima SMS, Arduino menyimpan nomor pengirim dan menunggu isi pesannya. Jika isi pesan adalah "lokasi", maka sistem akan menjalankan proses pengambilan data koordinat.

4. Pengiriman Balasan Koordinat

Jika data GPS valid, sistem akan membuat tautan Google Maps dari koordinat latitude dan longitude, lalu mengirimkan link tersebut ke nomor pengirim melalui SMS. Jika tidak valid, sistem akan mengirimkan pesan notifikasi bahwa lokasi belum tersedia.

```

GPS_Tracker.ino
67 void kirimLokasiKe(String nomorTujuan) {
68     Serial.println("Menyiapkan lokasi GPS...");
69
70     gpsSerial.listen(); // Fokus ke GPS
71     unsigned long timeout = millis() + 5000;
72     while (millis() < timeout) {
73         while (gpsSerial.available()) {
74             gps.encode(gpsSerial.read());
75         }
76         if (gps.location.isUpdated()) break;
77     }
78
79     if (gps.location.isValid()) {
80         double lat = gps.location.lat();
81         double lng = gps.location.lng();
82
83         // ✅ Perbaikan hanya bagian link ini saja
84         String mapsLink = "https://www.google.com/maps/search/?api=1&query=" + String(lat, 6) + "," + String(lng, 6);
85
86         Serial.println("Koordinat ditemukan!");
87         kirimSMSKe(nomorTujuan, mapsLink);
88     } else {
89         Serial.println("Gagal mendapatkan koordinat.");
90         kirimSMSKe(nomorTujuan, "Lokasi belum tersedia. Coba lagi sebentar lagi.");
91     }
92 }

```

Gambar 4. 7 Sketch Pengiriman Balasan Koordinat

Fungsi ini mengambil koordinat GPS dalam waktu 5 detik. Jika valid, sistem akan membuat link Google Maps dan mengirimkannya melalui SMS. Jika tidak, dikirimkan balasan berupa pesan teks “Gagal mendapatkan koordinat”.

5. Kembali ke Mode Siaga

Setelah mengirim balasan lokasi atau notifikasi, sistem kembali ke mode siaga. Fungsi kirimSMSKe() menangani pengiriman SMS dan tidak perlu reset ulang Arduino setelah satu kali respon.

```

94 void kirimSMSKe(String nomor, String text) {
95     sim800l.listen(); // Fokus ke SIM800L
96     delay(500);
97     sim800l.println("AT+CMGF=1");
98     delay(500);
99     sim800l.print("AT+CMGS=\\"");
100    sim800l.print(nomor);
101    sim800l.println("\\"");
102    delay(500);
103    sim800l.print(text);
104    delay(500);
105    sim800l.write(26); // CTRL+Z untuk kirim
106    Serial.println("SMS terkirim ke " + nomor + ": " + text);
107
108 }

```

Gambar 4. 8 Sketch Mode Siaga

Sistem menggunakan perintah AT untuk mengirim pesan ke nomor yang dituju. Setelah SMS dikirim, program langsung kembali ke loop utama dan siap menerima perintah baru.

6. Fitur Pengiriman Lokasi Otomatis

Untuk meningkatkan kepraktisan sistem, ditambahkan fitur pengiriman lokasi secara otomatis setiap satu jam sekali ke nomor pengguna yang telah ditentukan. Hal ini berguna saat pengguna tidak dapat mengirim SMS secara manual, tetapi tetap membutuhkan pemantauan lokasi secara berkala.

Fitur ini dikembangkan menggunakan fungsi millis() dari Arduino yang membaca waktu berjalan sejak sistem diaktifkan. Sistem akan mengirimkan lokasi tanpa harus ada perintah dari pengguna.

```
// Variabel waktu kirim otomatis
unsigned long previousMillis = 0;
const long interval = 3600000; // 1 jam (3.600.000 ms)
String nomorOtomatis = "+6285129682874"; // Ubah sesuai nomor tujuan

// Auto-kirim lokasi tiap 1 jam
unsigned long currentMillis = millis();
if (currentMillis - previousMillis >= interval) {
    previousMillis = currentMillis;
    kirimLokasiKe(nomorOtomatis);
}
```

Gambar 4. 9 Sketch Kirim Lokasi Otomatis

7. Fitur Mode Darurat

Sistem ini juga dilengkapi dengan fitur *mode darurat*, yang memungkinkan alat mengirimkan koordinat lokasi secara otomatis setiap 30 detik selama mode ini aktif. Fitur ini berguna pada kondisi pelacakan intensif seperti saat objek yang dipantau dalam kondisi bahaya atau tersesat.

Mode darurat dapat diaktifkan dengan mengirimkan pesan “darurat” ke alat. Sistem akan menyimpan nomor pengirim sebagai tujuan utama pengiriman lokasi, dan mulai mengirimkan tautan lokasi ke nomor tersebut setiap 30 detik. Untuk menghentikan mode ini, pengguna cukup mengirimkan pesan “off darurat”, dan alat akan menghentikan pengiriman otomatis serta kembali ke mode siaga.

Fitur ini dibangun menggunakan variabel daruratAktif sebagai indikator status, previousMillisDarurat untuk menghitung waktu menggunakan fungsi millis(), serta nomorDarurat untuk menyimpan nomor penerima. Dengan logika ini, sistem mampu mengirim lokasi secara berkala tanpa mengganggu fitur-fitur lainnya seperti permintaan lokasi manual ataupun pengiriman otomatis satu jam sekali.

```

17 // Fitur darurat
18 bool daruratAktif = false;
19 unsigned long previousMillisDarurat = 0;
20 const long intervalDarurat = 30000; // 30 detik
21 String nomorDarurat = "";
22
23 | | | | | } else if (messageContent == "darurat") {
24 | | | | | | daruratAktif = true;
25 | | | | | | nomorDarurat = senderNumber;
26 | | | | | | previousMillisDarurat = millis();
27 | | | | | | kirimSMSKe(senderNumber, "Mode darurat AKTIF. Lokasi akan dikirim tiap 30 detik.");
28 | | | | | } else if (messageContent == "off darurat") {
29 | | | | | | daruratAktif = false;
30 | | | | | | kirimSMSKe(senderNumber, "Mode darurat DINONAKTIFKAN.");
31 | | | | | }
32
33 | | | | waitingForMessage = false;
34
35 | | | |
36 | | | | // Kirim darurat tiap 2 menit
37 | | | | if (daruratAktif && (currentMillis - previousMillisDarurat >= intervalDarurat)) {
38 | | | | | previousMillisDarurat = currentMillis;
39 | | | | | if (nomorDarurat != "") {
40 | | | | | | kirimLokasiKe(nomorDarurat);
41 | | | | | }
42 | | | | }
43 | | | | delay(100);
44 | | |
45 | | | }
46
47 | | |
48 | | | |
49 | | | | |
50 | | | | |
51 | | | | |
52 | | | | |
53 | | | | |
54 | | | | |
55 | | | | |
56 | | | | |
57 | | | | |
58 | | | | |
59 | | | | |
60 | | | | |
61 | | | | |
62 | | | | |
63 | | | | |
64 | | | | |
65 | | | | |
66 | | | | |
67 | | | | |
68 | | | | |
69 | | | | |
70 | | | | |
71 | | | | |
72 | | | | |
73 | | | | |
74 | | | | |
75 | | | | |
76 | | | | |
77 | | | | |
78 | | | | |
79 | | | | |
80 | | | | |
81 | | | | |
82 | | | | |
83 | | | | |
84 | | | | |
85 | | | | |
86 | | | | |
87 | | | | |
88 | | | | |
89 | | | | |
90 | | | | |
91 | | | | |
92 | | | | |
93 | | | | |
94 | | | | |
95 | | | | |
96 | | | | |
97 | | | | |
98 | | | | |
99 | | | | |
100 | | | | |

```

Gambar 4. 10 Sketch Fitur Darurat

4.2 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan bahwa perangkat GPS Tracker dapat bekerja sesuai dengan tujuan perancangan, yaitu menerima perintah dari pengguna melalui SMS, memperoleh lokasi secara akurat menggunakan modul GPS, dan mengirimkan tautan lokasi kembali kepada pengguna secara otomatis. Selain itu, pengujian juga melibatkan daya tahan baterai serta durasi pengisian daya untuk menilai efisiensi energi sistem. Berikut ini merupakan hasil pengujian sistem berdasarkan fungsionalitas utama.

4.2.1 Pengujian Akurasi Koordinat GPS

Pengujian ini dilakukan di beberapa lokasi berbeda untuk mengetahui seberapa akurat sistem dalam memperoleh lokasi. Berikut hasilnya:

Tabel 4. 2 Hasil Pengujian Lokasi tertentu

No	Lokasi Uji	Koordinat GPS Diterima	Waktu Akuisisi	Akurasi (\pm m)	Balasan SMS	Keterangan
1	Belakang kandang	2.087870, 99.895790	20 detik	± 6 m	Ya	Respon lancar
2	Lapangan terbuka	2.0891966, 99.8923739	10 detik	± 4 m	Ya	Respons sangat cepat
3	Dalam Kandang	2.088855, 99.895294	36 detik	± 12 m	Ya	Respon agak lambat
4	Dekat pepohonan	2.0969463, 99.8935733	30 detik	± 9 m	Ya	Sinyal agak terhalang pepohonan

Dari pengujian lokasi yang sudah ditentukan, GPS Tracker mampu memperoleh koordinat dengan akurasi antara 4–12 meter. Respon sistem bergantung pada keterbukaan lokasi terhadap sinyal satelit.

Selain melakukan pengujian pada daerah yang sudah ditentukan pada tabel diatas, pengujian juga di lakukan untuk melacak objek yang bergerak secara acak, alat

dipasang pada seekor sapi dan dipantau selama kurang lebih 5 jam. Data lokasi dikumpulkan secara manual melalui perintah SMS “lokasi” setiap 30 sampai 60 menit. Hasil menunjukkan bahwa sistem mampu memberikan respons lokasi yang valid, dengan setiap link Google Maps menampilkan posisi terkini sapi secara akurat dan sesuai dengan pola pergerakan yang diamati di lapangan. Jarak perpindahan bervariasi dari ±100 hingga 600 meter, menunjukkan bahwa sistem sangat cocok digunakan dalam aplikasi pelacakan ternak di area semi-terbuka.

Tabel 4. 3 Pengujian Pada Sapi yang Bergerak

No	Waktu Pemantauan	Koordinat Diterima	Perkiraan Lokasi (Deskripsi Umum)	Perpindahan (Estimasi)	Link Google Maps
1	08.30 WIB	2.088855, 99.895294	Area peternakan, titik awal	-	https://www.google.com/maps/search/?api=1&query=2.088855,99.895294
2	09.00 WIB	2.087870, 99.895790	Pindah ke sisi kiri kandang	±100 meter	https://www.google.com/maps/search/?api=1&query=2.087870,99.895790
3	10.30 WIB	2.089215, 99.8948583	Dekat pagar barat	±120 meter	https://www.google.com/maps/search/?api=1&query=2.089215,99.8948583
4	11.00 WIB	2.0891966, 99.8923739	Pinggir semak, dekat kebun kecil	±270 meter	https://www.google.com/maps/search/?api=1&query=2.0891966,99.8923739
5	12.30 WIB	2.0907283, 99.8936018	Kembali mendekat ke tengah area	±200 meter	https://www.google.com/maps/search/?api=1&query=2.0907283,99.8936018

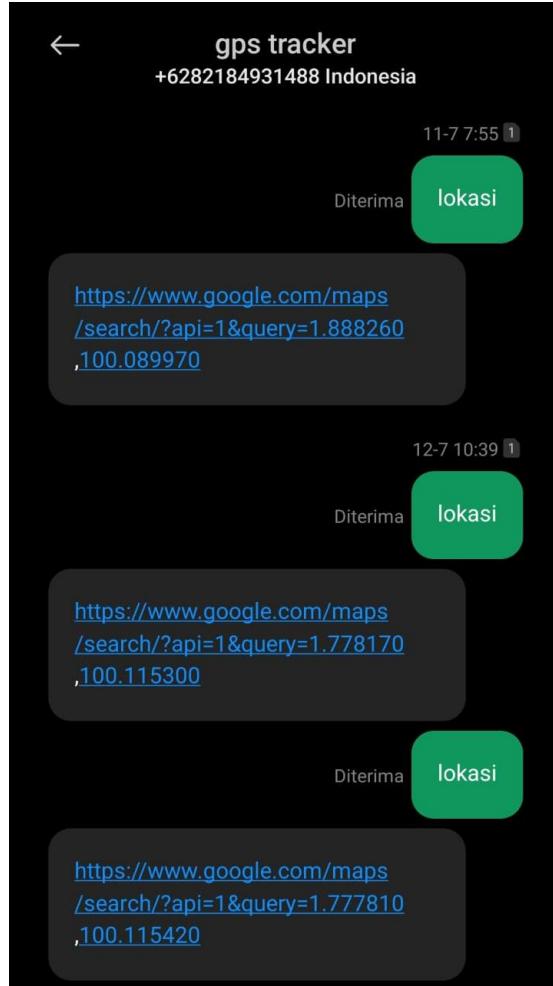
			peternakan		99.8936018
6	13.00 WIB	2.0969463, 99.8935733	Menyusuri batas luar area peternakan	±600 meter	https://www.google.com/maps/search/?api=1&query=2.0969463,99.8935733
7	14.30 WIB	2.0978293, 99.8930466	Area paling jauh dari titik awal	±100 meter	https://www.google.com/maps/search/?api=1&query=2.0978293,99.8930466

Berdasarkan seluruh rangkaian pengujian yang telah dilakukan, sistem GPS tracker berbasis Arduino Nano dan modul SIM800L terbukti dapat berfungsi dengan baik dalam melacak lokasi objek melalui perintah SMS. Respon sistem dalam berbagai kondisi lingkungan menunjukkan ketahanan dan keandalan yang memadai, termasuk saat digunakan dalam pengujian langsung pada hewan ternak. Seluruh balasan lokasi yang dikirimkan dalam bentuk tautan Google Maps dapat digunakan dengan akurasi yang cukup tinggi. Dengan demikian, sistem ini cukup layak diterapkan sebagai solusi pelacakan berbasis SMS yang efisien untuk kebutuhan *monitoring* objek bergerak seperti ternak maupun aset lapangan lainnya.

4.2.2 Pengujian Fungsi SMS

1. Pengujian SMS Manual Dengan Perintah "lokasi" Dikirim Dari *User*

Pengujian ini bertujuan untuk mengetahui apakah perangkat dapat merespons perintah dari pengguna melalui SMS dengan tepat. Perintah yang digunakan adalah "lokasi". Ketika perintah ini dikirim ke nomor SIM pada modul GSM SIM800L V2, sistem akan membaca koordinat GPS terbaru dan mengirimkan balasan ke nomor pengirim dalam bentuk tautan Google Maps. Hasilnya Perangkat berhasil membaca SMS perintah "lokasi" dan mengirim balasan otomatis berupa link Google Maps



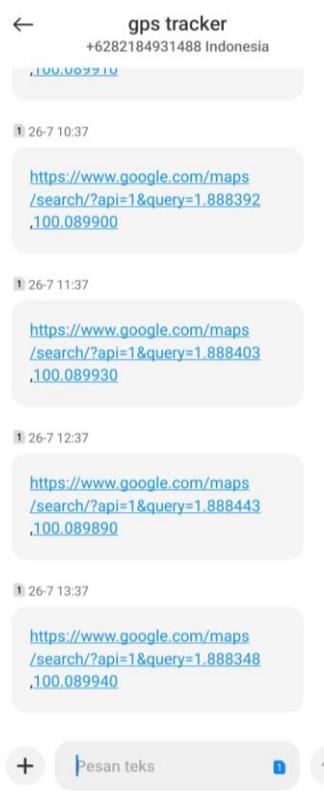
Gambar 4. 11 Pengujian Hasil SMS

Balasan dapat langsung diklik oleh pengguna dan diarahkan ke titik lokasi di Google Maps.

2. Pengujian Pengiriman Lokasi Otomatis Setiap 1 Jam

Pengujian ini bertujuan untuk memastikan bahwa sistem dapat secara otomatis mengirimkan lokasi terkini tanpa adanya perintah manual dari pengguna. Dalam pengujian ini, alat diaktifkan dan dibiarkan menyala secara terus-menerus selama kurang lebih 10 jam. Nomor tujuan pengiriman SMS otomatis telah ditentukan secara langsung di dalam kode program. Selama periode tersebut, sistem berhasil mengirimkan SMS berisi tautan lokasi Google Maps setiap satu jam sekali secara konsisten.

Pengiriman lokasi otomatis ini dilakukan dengan memanfaatkan fungsi millis() pada Arduino yang digunakan untuk menghitung durasi sistem berjalan sejak pertama kali dinyalakan. Fitur ini sangat membantu dalam kondisi pengguna tidak dapat mengirimkan pesan secara aktif untuk meminta lokasi, namun tetap ingin mendapatkan pemantauan lokasi secara berkala.

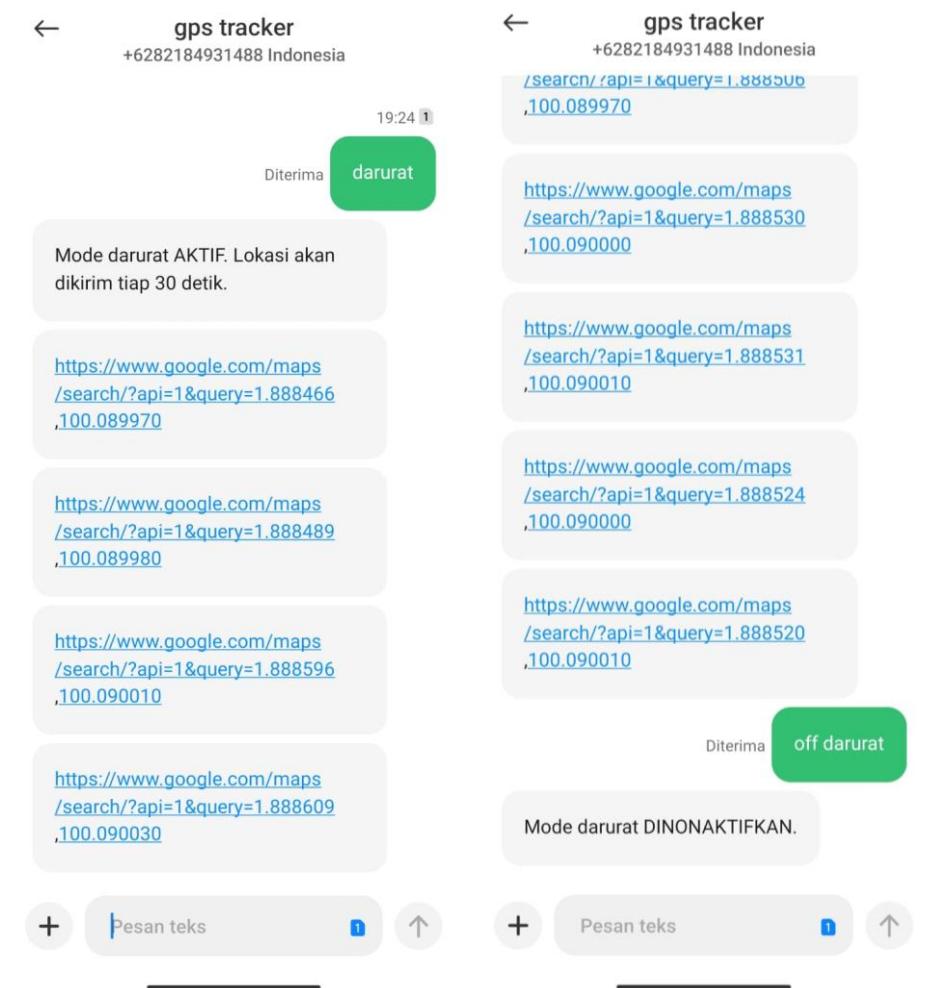


Gambar 4. 12 Pengujian Fitur Kirim Lokasi Otomatis 1 Jam Sekali

Dari hasil pengujian, didapatkan total sebanyak 10 pesan lokasi otomatis yang berhasil dikirim dan diterima dengan format tautan yang valid serta dapat diklik. Hal ini menunjukkan bahwa fitur berjalan sesuai ekspektasi dan tidak mengganggu fungsi manual lainnya seperti pengiriman lokasi berdasarkan perintah SMS "lokasi".

3. Pengujian Fitur Mode Darurat

Selain pengujian untuk pengiriman lokasi melalui perintah “lokasi”, sistem juga diuji dengan fitur mode darurat, yaitu pengiriman otomatis lokasi setiap 30 detik. Fitur ini aktif ketika sistem menerima SMS dengan isi “darurat”, dan akan mengirimkan tautan Google Maps ke pengirim secara berkala. Pada pengujian selama 10 menit, sistem berhasil mengirimkan lokasi sebanyak 20 kali. Selanjutnya, pengujian dilanjutkan dengan mengirimkan pesan “off darurat” untuk menghentikan mode, dan alat kembali ke kondisi siaga tanpa mengirim SMS lagi.



Gambar 4. 13 Pengujian Mode Darurat

Hasil pengujian menunjukkan bahwa fitur darurat ini berfungsi secara stabil dan tidak mengganggu fitur SMS lainnya, termasuk pengiriman manual dan mode otomatis per jam.

4.2.3 Pengujian Daya Tahan Baterai

Pengujian dilakukan dengan menggunakan satu buah baterai Li-Po 3.7V 2500mAh. Sistem diaktifkan dalam durasi panjang dan diuji melakukan proses penerimaan dan pengiriman SMS secara berkala yang dapat dilihat pada tabel berikut ini

Tabel 4. 4 Pengujian Daya Operasional

No	Kapasitas Baterai	Durasi Operasi	Jumlah SMS Dikirim	Status	Keterangan
1	2500 mAh	± 10 jam	10 kali dengan fitur otomatis kirim lokasi	OK	Sistem stabil dan efisien
2	2500 mAH	9 jam 20 menit	9 kali dengan fitur otomatis kirim dan 20 kali dengan mode darurat	Oke	Sistem agak boros karena banyak SMS yang dikirim.

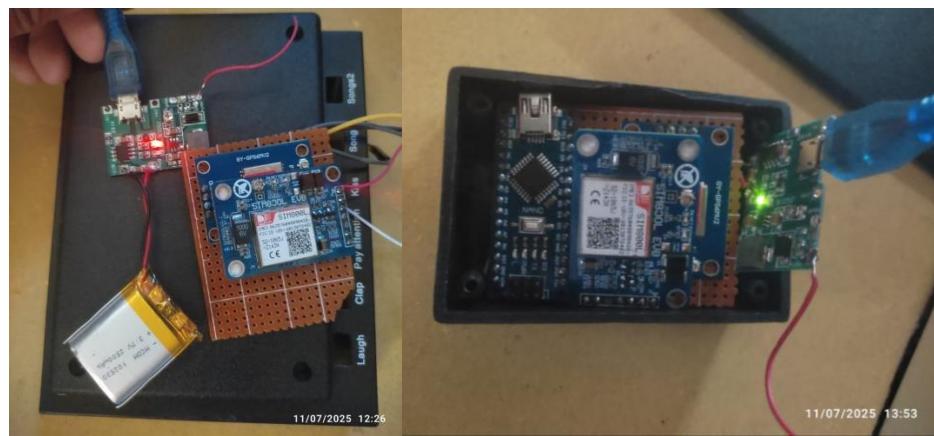
Berdasarkan hasil pengujian, sistem mampu beroperasi selama ± 10 jam saat hanya menggunakan fitur pengiriman lokasi otomatis setiap 1 jam sekali. Ketika fitur tambahan mode darurat diaktifkan selama 10 menit di mana sistem mengirimkan pesan lokasi setiap 30 detik terjadi sedikit peningkatan konsumsi daya. Namun, sistem tetap dapat beroperasi dengan estimasi durasi kerja sekitar 9 hingga 9,2 jam. Hal ini menunjukkan bahwa penggunaan mode darurat dalam jangka waktu pendek tidak berdampak signifikan terhadap performa baterai secara keseluruhan. Dengan demikian, sistem dinilai efisien dan masih layak digunakan di lapangan untuk pemantauan objek dalam waktu yang cukup panjang, meskipun terdapat beban tambahan sesaat dari fitur darurat.



Gambar 4. 14 Pengujian Daya Tahan Baterai

4.2.4 Pengujian Waktu Pengisian Daya

Pengujian ini bertujuan untuk mengetahui lama waktu yang dibutuhkan untuk mengisi ulang baterai dari kondisi habis hingga penuh menggunakan modul charger bawaan. Pada Gambar berikut ini dapat dilihat LED Merah akan hidup saat proses pengisian daya berlangsung dan LED hijau akan hidup apabila Baterai sudah terisi penuh



Gambar 4. 15 Pengujian Durasi Pengisian Baterai

Hasilnya:

- Waktu pengisian dari 12.26 WIB hingga 13.53 WIB
- Total durasi pengisian adalah 1 jam 27 menit
- Indikator charger menunjukkan baterai penuh LED berwarna hijau akan hidup dan LED merah akan tetap hidup saat proses pengisian baterai.

4.2.5 Pengujian Ketahanan Terhadap Air (*Water Resistance*)

Pengujian ketahanan air dilakukan untuk mengetahui seberapa baik casing alat dalam melindungi komponen elektronik dari percikan atau semprotan air. Proses pengujian dilakukan dengan menyemprotkan air menggunakan semprotan bertekanan ringan ke seluruh bagian luar perangkat selama kurang lebih 10 menit secara terus-menerus. Pada pengujian ini, perangkat dalam keadaan aktif untuk memastikan tidak terjadi gangguan fungsional akibat kontak dengan air.

Casing alat telah dirancang tertutup rapat dan diperkuat dengan lapisan lem unibon di bagian tutup dan sekeliling baut, sehingga tidak terdapat celah masuknya air. Hasil dari pengujian ini menunjukkan bahwa selama durasi penyemprotan, air tidak masuk ke dalam casing dan seluruh fungsi alat tetap berjalan dengan normal. Hal ini menunjukkan bahwa alat memiliki ketahanan terhadap percikan air ringan atau simulasi gerimis. Berikut merupakan dokumentasi proses pengujian ketahanan air yang dilakukan pada perangkat:



Gambar 4. 16 Simulasi Ketahanan Air

Namun, dari hasil pengamatan lebih lanjut, diketahui bahwa perlindungan ini bersifat terbatas. Jika terkena tekanan air yang lebih tinggi atau terpapar hujan

deras dalam waktu yang lama, kemungkinan besar air akan merembes ke bagian dalam alat. Meski demikian, posisi pemasangan alat yang dikalungkan di leher sapi secara tidak langsung memberikan perlindungan tambahan karena bagian tubuh hewan menutupi alat dari paparan air secara langsung. Dengan demikian, saat terjadi hujan, pemilik ternak masih memiliki waktu untuk mengirim perintah SMS “lokasi” sebelum kondisi alat terganggu oleh air, sehingga fitur pelacakan tetap dapat digunakan secara efektif.

4.2.6 Dokumentasi Pengujian Lapangan

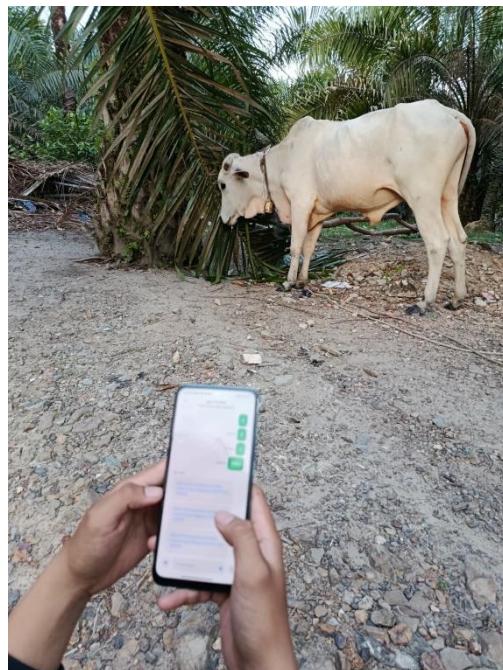
Untuk memastikan bahwa sistem berfungsi dengan baik dalam kondisi nyata, alat dipasang sapi dengan cara dikalungkan menggunakan tali. Tujuannya adalah untuk menguji ketahanan perangkat ketika digunakan dalam situasi sebenarnya di lingkungan terbuka. Pengujian ini menunjukkan bahwa sistem tetap dapat mengirim koordinat lokasi dengan baik, meskipun alat bergerak bersama objek.



Gambar 4. 17 Pemasangan GPS Oleh Pemilik Sapi



Gambar 4. 18 Perangkat GPS Dikalungkan Pada Leher Sapi



Gambar 4. 19 Percobaan Pelacakan Lewat SMS

Gambar-gambar diatas menunjukkan proses pengujian alat GPS tracker yang dikalungkan pada leher sapi. Terlihat bahwa alat dipasang secara aman menggunakan tali pengikat agar tidak mengganggu gerak hewan. Posisi alat yang tergantung di bawah leher

juga membantu melindunginya dari paparan langsung air hujan dan debu. Selain itu, dokumentasi ini memperlihatkan lingkungan pengujian yang meliputi area kandang dan lapangan terbuka, yang menjadi medan nyata untuk menguji kemampuan sistem dalam mengirimkan data lokasi secara akurat. Dengan dokumentasi ini, dapat dilihat bahwa alat digunakan dengan stabil dalam kondisi lapangan yang dinamis dan tidak selalu ideal.