

RANCANGAN SIMULASI SISTEM MONITORING LEVEL KETINGGIAN OLI PADA GEARBOX RADAR VIA HANDPHONE BERBASIS ESP32 DI PERUM LPPNPI CABANG MEDAN

Yosie Ht Simanjuntak¹, Susi Diriyanti N², Said Hafas³

yosiesimanjuntak15@gmail.com¹, susidiriyantinovalina@gmail.com², saidhafas75@gmail.com³

Politeknik Penerbangan Medan

ABSTRAK

Indonesia, sebagai negara kepulauan, menghadapi tantangan besar dalam mengelola dan mengawasi transportasi udara. Badan Pengembangan Sumber Daya Manusia Perhubungan (BPSDMP) melalui Pusat Pengembangan Sumber Daya Manusia Perhubungan Udara (PPSDMPU), bertanggung jawab dalam pengembangan kompetensi sumber daya manusia di sektor penerbangan, termasuk melalui pendidikan di Politeknik Penerbangan Medan (Poltekbang Medan). Poltekbang Medan menawarkan program studi Teknik Telekomunikasi dan Navigasi Udara (TNU) yang bertujuan menghasilkan tenaga kerja yang kompeten di bidang perawatan dan operasi peralatan navigasi udara. Penelitian ini bertujuan untuk merancang sistem monitoring level ketinggian oli pada gearbox RADAR menggunakan ESP32 yang dapat diakses melalui handphone. Hal ini penting mengingat pernah terjadi masalah pada komputer SRG di Perum LPPNPI Cabang Medan yang mengakibatkan kesulitan dalam pemantauan tinggi oli pada gearbox RADAR. Dengan sistem yang dirancang, teknisi dapat memonitor tinggi oli dari jarak jauh melalui aplikasi Blynk dan menerima notifikasi melalui Telegram saat oli berada di bawah batas minimum. Metode penelitian yang digunakan adalah Research and Development, yang meliputi penelitian dan pengumpulan data, perencanaan, pengembangan produk, uji coba awal, penyesuaian produk, dan uji coba akhir. Hasil penelitian menunjukkan bahwa sistem monitoring yang dirancang dapat berfungsi dengan baik dan memberikan notifikasi yang akurat terkait tinggi oli. Penelitian ini diharapkan dapat menjadi solusi alternatif dalam monitoring tinggi oli pada gearbox RADAR, memudahkan teknisi dalam pemeliharaan, dan meningkatkan efisiensi operasional di Perum LPPNPI Cabang Medan. Adapun saran untuk penelitian lanjutan adalah penggunaan gearbox sesungguhnya dalam uji coba serta pengembangan sistem pompa otomatis untuk menambah oli secara otomatis saat tinggi oli kurang dari batas minimumnya..

Kata Kunci: Sistem monitoring, Gearbox RADAR, ESP32, Aplikasi Blynk, Notifikasi Telegram, Airtaxi Cabang Medan, Pemeliharaan peralatan navigasi udara.

ABSTRACT

Indonesia, as an archipelagic country, faces significant challenges in managing and supervising air transportation. The Human Resource Development Agency for Transportation (BPSDMP), through the Air Transportation Human Resource Development Center (PPSDMPU), is responsible for developing human resource competencies in the aviation sector, including through education at the Medan Aviation Polytechnic (Poltekbang Medan). Poltekbang Medan offers a study program in Air Navigation and Telecommunications Engineering (TNU) aimed at producing competent professionals in the maintenance and operation of air navigation equipment. This research aims to design an oil level monitoring system on the RADAR gearbox using ESP32 that can be accessed via a mobile phone. This is crucial given previous issues with the SRG computer at Perum LPPNPI Medan Branch that caused difficulties in monitoring the oil level in the RADAR gearbox. With the designed system, technicians can remotely monitor the oil level through the Blynk app and receive notifications via Telegram when the oil falls below the minimum threshold. The research methodology used is Research and Development, which includes research and data collection, planning, product development, initial testing, product adjustment, and final testing. The results of

the research show that the designed monitoring system functions effectively and provides accurate notifications regarding the oil level. This research is expected to be an alternative solution for monitoring oil levels in RADAR gearboxes, facilitating maintenance for technicians, and improving operational efficiency at Perum LPPNPI Medan Branch. Suggestions for further research include testing with an actual gearbox and developing an automatic pump system to add oil automatically when the level falls below the minimum threshold.

Keywords: *Monitoring system, RADAR gearbox, ESP32, Blynk application, Telegram notifications, Airnav Medan Branch, Air navigation equipment maintenance.*

PENDAHULUAN

Indonesia sebagai negara kepulauan memiliki tantangan besar dalam mengelola dan mengawasi transportasi udara. Kementerian Perhubungan Republik Indonesia, melalui berbagai lembaganya, berkomitmen untuk meningkatkan keselamatan, keamanan, dan efisiensi penerbangan. Salah satu lembaga yang berperan penting dalam hal ini adalah Badan Pengembangan Sumber Daya Manusia Perhubungan (BPSDMP) yang memiliki tanggung jawab dalam mengembangkan sumber daya manusia yang kompeten di bidang perhubungan, termasuk di sektor penerbangan (Dwi 2023).

Pusat Pengembangan Sumber Daya Manusia Perhubungan Udara (PPSDM Perhubungan Udara atau PPSDMPU) yang berlokasi di curug adalah salah satu unit di bawah Badan Pengembangan Sumber Daya Manusia Perhubungan (BPSDMP), yang merupakan bagian dari Kementerian Perhubungan Republik Indonesia. PPSDMPU bertanggung jawab untuk mengembangkan dan meningkatkan kompetensi sumber daya manusia di sektor perhubungan udara baik melalui pelatihan ataupun melalui sekolah yang ada dibawah naungannya seperti Politeknik Penerbangan Medan yang sering disebut Poltekbang Medan. Poltekbang Medan adalah sebuah institusi pendidikan tinggi vokasi yang berada di bawah naungan Kementerian Perhubungan Republik Indonesia yang berfokus pada pendidikan dan pelatihan di bidang penerbangan, dengan tujuan utama menghasilkan tenaga kerja yang terampil dan kompeten di sektor penerbangan.

Poltekbang Medan memiliki beberapa program studi, salah satunya ialah program studi Teknik Telekomunikasi dan Navigasi Udara (TNU). TNU merupakan sebuah program studi yang mempelajari tentang bagaimana cara memelihara, memperbaiki dan mengoperasikan peralatan-peralatan telekomunikasi dan navigasi udara serta elektronika bandara. Dalam menciptakan insan perhubungan yang terampil dan kompeten, Poltekbang Medan mewajibkan seluruh Taruna nya mengikuti pelatihan yang sering disebut On the Job Training (OJT) di industri penerbangan seperti Perusahaan Umum Lembaga Penyelenggara Pelayanan Navigasi Penerbangan Indonesia (Perum LPPNPI) yang sering disebut sebagai Airnav Indonesia.

AirNav Indonesia, sebagai perusahaan yang bertanggung jawab atas penyelenggaraan layanan navigasi udara di Indonesia, memiliki peran penting dalam memastikan keselamatan dan efisiensi penerbangan. Perum LPPNPI Cabang Medan yang sering disebut Airnav Cabang Medan, salah satu cabang AirNav Indonesia yang bertanggung jawab untuk mengawasi dan memelihara peralatan navigasi udara di wilayah Medan dan sekitarnya.

Peralatan atau fasilitas pendukung sangat dibutuhkan dalam menunjang pelayanan navigasi. Salah satu fasilitas pendukung pelayanan navigasi adalah RADAR (Radio Detection and Ranging) yang berfungsi untuk mendeteksi keberadaan sebuah pesawat yang sedang mengudara. RADAR digunakan oleh ATC (Air Traffic Controller) untuk memantau posisi pesawat. Informasi dari RADAR memungkinkan ATC untuk mengelola pergerakan pesawat, mencegah tabrakan, dan memastikan alur lalu lintas yang aman dan efisien.

RADAR beroperasi dengan antenna yang berputar terus-menerus. Antena RADAR

diputar oleh gear yang terdapat pada gearbox RADAR. Gear yang terus berputar membutuhkan oli sebagai pelumas. Gear yang terus berputar dapat menghasilkan getaran yang menyebabkan mur pada gearbox menjadi longgar dan oli bocor melalui celah tersebut. Gearbox RADAR yang kekurangan oli dapat mengganggu kinerja sebuah RADAR karena dapat menyebabkan gear aus bahkan overheat pada gearbox yang mengakibatkan RADAR tidak berfungsi secara optimal

Airnav cabang Medan memiliki komputer Site Local Generation (SLG) yang berada di shelter RADAR dan komputer Site Remote Generation (SRG) yang berada di ruangan teknisi CNSA (Communication, Navigation, Surveillance, Automation) lantai 1 Airnav Cabang Medan untuk memantau kinerja RADAR termasuk oli pada gearbox RADAR. Komputer SRG Airnav cabang Medan pernah mengalami masalah yang mengakibatkan Teknisi tidak dapat memonitor level oli pada gearbox RADAR dari jarak jauh dan membuat Teknisi harus melakukan pengecekan atau monitoring level oli secara langsung ke shelter RADAR. Hal tersebut mengakibatkan pemeliharaan RADAR menjadi kurang optimal.

Kehidupan sebagian besar manusia saat ini tidak terlepas dengan Handphone. Ketergantungan yang tinggi terhadap handphone telah membuka peluang besar untuk memanfaatkannya sebagai monitor jarak jauh dan alarm berupa notifikasi. Oleh sebab itu, penulis membuat sebuah penelitian berupa rancangan simulasi yang diharapkan dapat diimplementasikan di Airnav cabang Medan apabila masalah pada komputer Site Remote Generation terjadi kembali. Dengan judul penelitian “RANCANGAN SIMULASI SISTEM MONITORING LEVEL KETINGGIAN OLI PADA GEARBOX RADAR VIA HANDPHONE BERBASIS ESP32 DI PERUM LPPNPI CABANG MEDAN”.

METODOLOGI

Dalam proses menyelesaikan penelitian ini, penulis membuat sebuah rancangan alat untuk mengembangkan komputer SRG untuk memonitor level atau tinggi oli pada gearbox RADAR sehingga metode penelitian yang digunakan adalah metode Research and Development. Metode Research and Development merupakan suatu pendekatan atau proses yang digunakan dalam melakukan penelitian ilmiah, mengembangkan produk, proses, atau inovasi baru yang dapat memberikan kontribusi bagi kemajuan pengetahuan dan teknologi. Metode ini bertujuan untuk mencari solusi atas masalah yang ada atau meningkatkan produk dan proses yang sudah ada sebelumnya. Langkah-langkah yang dilakukan adalah sebagai berikut :

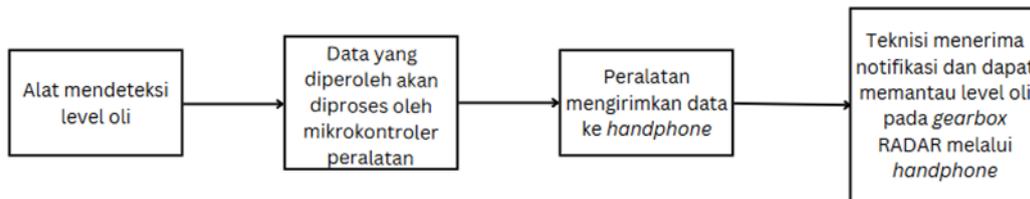
- a. Penelitian dan pengumpulan data (Research and information collecting) Dalam tahapan ini dilakukan studi literatur terhadap konsep atau landasan teoritis mengenai peralatan, bahan yang akan digunakan pada penelitian, langkah-langkah yang tepat dalam pengembangan produk, serta memahami gambaran hasil penelitian yang telah dilakukan sebelumnya sebagai objek perbandingan dalam pelaksanaan pengembangan. Pada tahap penelitian dan pengumpulan data ini dilakukan juga observasi (studi lapangan) dalam mengukur kebutuhan dan penelitian dalam skala kecil. Pengukuran kebutuhan pada peralatan menjadi dasar dalam pengembangan produk
- b. Perencanaan (Planning) Dalam melakukan perencanaan perlu dilakukan identifikasi permasalahan serta dilakukan identifikasi tujuan dari penelitian.
- c. Pengembangan produk (Develop preliminary form of product) Pengembangan produk pada tahap awal merupakan kegiatan dalam membuat rancangan dalam bentuk rangkaian kasar (draft) yang belum dikemas ke dalam bentuk prototipe yang lebih konkret.
- d. Uji coba awal (Preliminary field testing) Pada pengujian tahap awal, dilakukan untuk mengkaji apakah rangkaian rancangan dapat berfungsi dengan baik dalam mendeteksi

- level oli pada gearbox dan mengkaji kekurangan serta penyesuaian yang harus diterapkan pada produk tersebut.
- e. Penyesuaian pada produk (Main product revision) Setelah dilakukannya pengamatan pada uji coba tahap awal, produk yang masih dalam bentuk rancangan kasar (draft) diperbaiki lagi dan disesuaikan agar dapat beroperasi menjadi prototipe yang berfungsi sebagai monitoring level oli pada gearbox RADAR sesuai dengan konsep rancangan yang telah dibuat.
 - f. Uji coba akhir (Main field testing) Pada tahap ini dilakukan uji coba akhir pada produk yang sudah disempurnakan, pengujian dilakukan pada simulasi menggunakan wadah berisi oli sebagai pengganti gearbox RADAR.

Kondisi yang terjadi pada saat ini adalah belum adanya alat yang dapat digunakan untuk melakukan monitoring level oli pada RADAR Airnav cabang Medan apabila komputer SRG mengalami masalah yang mengakibatkan teknisi kesulitan untuk mengetahui level oli pada RADAR. Permasalahan yang mungkin bisa terjadi adalah apabila gearbox RADAR mengalami kebocoran teknisi baru bisa mengetahui oli gearbox yang berkurang dengan cara melihat langsung indikator oli di control box atau komputer SLG yang terdapat jauh dari ruangan teknisi. RADAR yang menyala setiap waktu membutuhkan kondisi gearbox motor dalam keadaan baik, gearbox dengan oli yang cukup dapat mengurangi beberapa permasalahan yang dapat terjadi. Dikarenakan fungsi oli adalah sebagai pelumas dan pendingin bagi gearbox, ketika oli tidak berada pada level yang cukup maka gear pada gearbox bisa rusak serta dapat mengalami overheat, maka dari itu pelaksanaan monitoring level oli harus dikembangkan untuk mendapatkan informasi terkait level oli RADAR serta efektifitas perhitungan waktu dalam melakukan penambahan oli pada gearbox RADAR.

Dilihat dari kondisi seperti ini dapat dinyatakan bahwa teknisi harus memiliki alat yang berfungsi sebagai monitoring level oli gearbox RADAR yang dapat memberikan informasi terkait level oli RADAR secara langsung kepada teknisi. Alat ini dapat mempermudah teknisi dikarenakan monitor dilakukan menggunakan handphone yang telah terhubung dengan peralatan menggunakan aplikasi blynk.

Kondisi yang diinginkan pada saat ini yaitu alat dapat melakukan monitoring level oli RADAR dari jarak jauh pada saat komputer SRG mengalami masalah. Berdasarkan kondisi yang telah dijelaskan, berikut alur keadaan yang diharapkan setelah adanya alat monitoring level oli pada gearbox RADAR :



Gambar 1 Alur keadaan yang diinginkan
(Sumber: Karya Penulis, 2024)

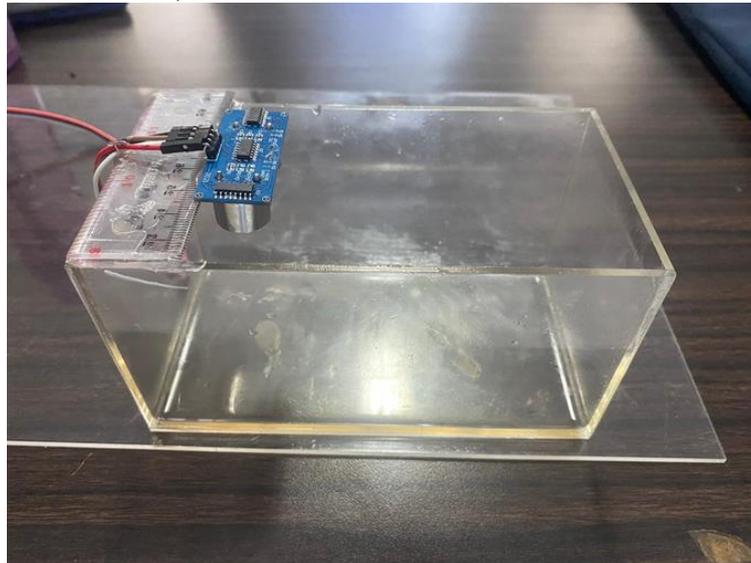
HASIL DAN PEMBAHASAN

1. Hasil Penelitian

1. Perangkat Keras

Pada penelitian ini, dibutuhkan beberapa perangkat keras yang saling terkoneksi dengan ESP32. Komponen tersebut ialah adaptor, Ultrasonic HC-SR04 dan I2C LCD dan Laptop atau PC untuk menulis pemrograman.

Sensor Ultrasonic HC-SR04 diletakkan diatas sebuah wadah yang bersungsi sebagai pengganti gearbox RADAR. Wadah yang digunakan adalah sebuah wadah akrilik berbentuk balok sebagai tempat oli dengan panjang 14,5 cm, lebar 7,9 cm dan tinggi 8 cm. Dikarenakan rancangan hanya mendeteksi level ketinggian oli, maka kita hanya membutuhkan berapa tinggi atau jarak dari sensor ultrasonic HC-SR04 dengan dasar wadah karena jarak yang dideteksi oleh oleh sensor akan dikonversi menjadi tinggi oli sesuai dengan program yang kita terapkan ke ESP32. Pada rancangan ini jarak atau tinggi sensor ultrasonic HC-SR04 dengan dasar wadah adalah 6,92 cm.

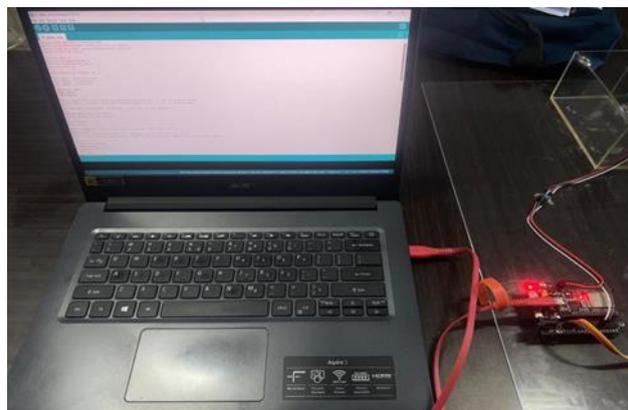


Gambar 1 Wadah oli

(Sumber: Dokumentasi Penulis, 2024)

a. Personal Computer (PC)

PC, singkatan dari Personal Computer, adalah sebuah perangkat elektronik yang dirancang untuk penggunaan individu. PC biasanya digunakan untuk menjalankan berbagai tugas seperti pengolahan kata, browsing internet, bermain game, membuat konten, pemrograman, dan banyak lagi. PC yang digunakan dalam perancangan peneitian ini adalah Laptop yang berfungsi untuk menulis Bahasa pemograman melalui aplikasi Arduino IDE. Pemograman yang telah berhasil kemudian akan di upload dari laptop ke ESP32 melalui kabel data USB.



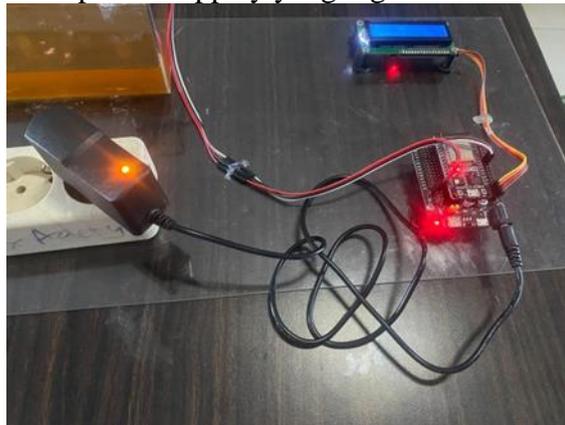
Gambar 2 Upploud Program dari laptop ke ESP32

(Sumber: Dokumentasi Penulis, 2024)

b. Rangkaian Adaptor sebagai Catu Daya

Adaptor dalam perancangan system monitoring level oli ini adalah adaptor 220V AC

to 9V AC. Berikut gambar dari power supply yang digunakan dalam rangkaian ini.



Gambar 3 Rangkaian Power Supplay
(Sumber: Dokumentasi Penulis, 2024)

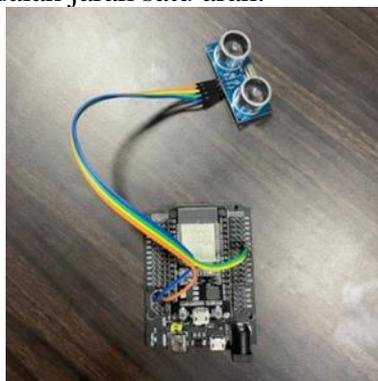
Dari Gambar 3 dapat dilihat plug adaptor dihubungkan ke power input ESP32 untuk mendapatkan tegangan yang dibutuhkan oleh mikrokontroler tersebut dalam melakukan operasinya.

c. Rangkaian Sensor Ultrasonik HC-SR04

Rangkaian sensor ultrasonic HCSR-04 pada monitoring level oli RADAR akan terhubung langsung ke mikrokontroler ESP32. Sensor Ultrasonic HCSR-04 digunakan bersama ESP32 untuk mengukur jarak, dan hasil pengukuran jarak tersebut akan digunakan untuk mengukur level oli pada gearbox. Ketika sensor diaktifkan, modul HCSR-04 mengirimkan pulsa ultrasonik pada frekuensi 40 kHz melalui transduser pemancar (trig pin) berupa serangkaian gelombang suara yang tidak terdengar oleh telinga manusia.

Pulsa ultrasonik yang telah dikirimkan atau dipancarkan, akan bergerak di udara dengan kecepatan suara. Ketika pulsa mengenai suatu objek, pulsa akan memantul kembali ke sensor. Transduser penerima (echo pin) pada sensor akan menerima gelombang suara yang dipantulkan oleh objek. Waktu yang diperlukan bagi pulsa untuk melakukan perjalanan dari sensor ke objek dan kembali lagi disebut waktu pulang-pergi (round-trip time).

Mikrokontroler yang terhubung ke sensor HCSR-04 mengukur waktu pulang-pergi pulsa ultrasonik tersebut. Sensor HCSR-04 mengirimkan sinyal tinggi (HIGH) pada pin echo selama waktu yang sama dengan waktu pulang-pergi pulsa ultrasonik. Di mana kecepatan suara di udara kira-kira 343 meter per detik (atau 0.0343 cm per mikrodetik). Pembagian dengan 2 dilakukan karena waktu yang diukur adalah waktu pulang-pergi, sedangkan yang dibutuhkan adalah jarak satu arah.



Gambar 4 Rangkaian sensor Ultrasonik HC-SR04
(Sumber: Dokumentasi Penulis, 2024)

Berikut koneksi antara pin Sensor Ultrasonik HC-SR04 dengan ESP32 :

Tabel 1 Koneksi pin HC-SR04 dengan pin ESP32

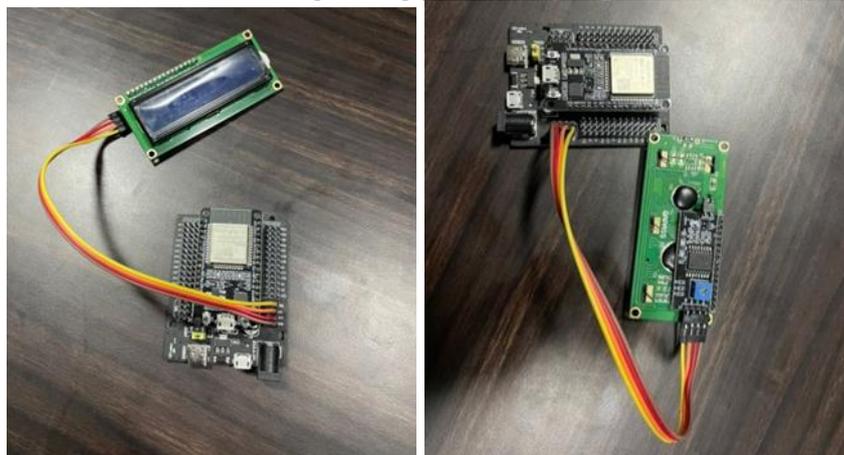
Pin Sensor Ultrasonic HC-SR04	Pin Arduino ESP32
VCC	5V
Trigg	D18
Echo	D19
GND	GND

Sumber: Karya Penulis, 2024.

d. Rangkaian I2C LCD

LCD (Liquid Crystal Display) digunakan sebagai penampil inputan yang diperoleh oleh sensor dan diproses oleh Arduino ESP32. Tampilan yang akan terlihat pada LCD adalah level oli yang terdapat pada wadah yang diibaratkan sebagai gearbox RADAR. LCD dengan interface I2C menggunakan komunikasi serial I2C untuk mengirimkan data dan perintah dari mikrokontroler Arduino ESP32 ke modul LCD. Mikrokontroler dan LCD dihubungkan melalui dua jalur I2C yaitu SDA (Serial Data) dan SCL (Serial Clock). Mikrokontroler akan menginisialisasi komunikasi I2C dengan menetapkan alamat perangkat LCD yang ditentukan oleh modul I2C yang terpasang pada LCD (misalnya 0x27 atau 0x3F).

ESP32 akan mengirimkan perintah ke LCD untuk mengatur mode operasi, seperti mengatur kursor, membersihkan layar, mengatur mode penulisan. Perintah-perintah akan dikirim dalam bentuk byte melalui jalur SDA dengan sinkronisasi melalui jalur SCL. Setelah LCD diinisialisasi, mikrokontroler mengirimkan data karakter yang akan ditampilkan pada layar. Data karakter dikirim dalam bentuk kode ASCII. Setiap karakter yang dikirim mewakili satu posisi pada layar LCD. LCD mengubah data yang diterima ke posisi yang sesuai pada layar berdasarkan perintah yang diterima sebelumnya. LCD memiliki kontroler yang mengatur tampilan karakter pada posisi yang sesuai di layar. Kontroler LCD mengaktifkan piksel tertentu pada layar kristal cair untuk membentuk karakter yang sesuai dengan data yang dikirimkan. Karakter yang ditampilkan dapat berupa angka, huruf, atau simbol khusus tergantung pada data yang diterima.



(i)

(ii)

Gambar 5 Rangkaian Sensor I2C LCD

(i) I2C LCD tampak depan

(ii) I2C LCD tampak belakang

(sumber, dokumentasi penulis 2024)

Tabel 2 Koneksi pin Pin I2C LCD dengan pin ESP32

Pin I2C LCD	Pin Arduino ESP32
GND	GND
VCC	VCC
SDA	D21
SCL	D22

Sumber: Karya Penulis, 2024.

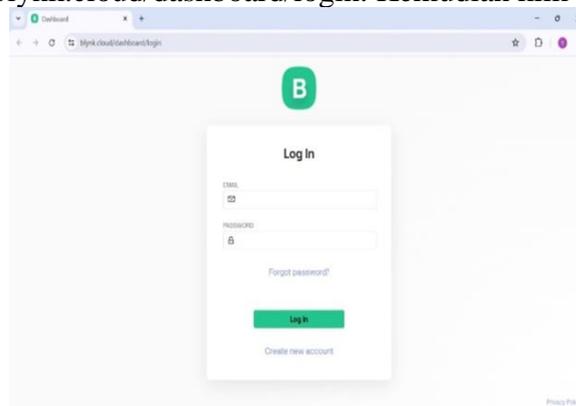
2. Perangkat Lunak dan Aplikasi

Perancangan simulasi monitoring level oli pada gearbox RADAR membutuhkan perangkat lunak supaya sistem dapat berjalan dan sesuai dengan yang diharapkan. Perangkat lunak tersebut ialah aplikasi Arduino IDE dan Blynk.

a. Program Perangkat Lunak Blynk

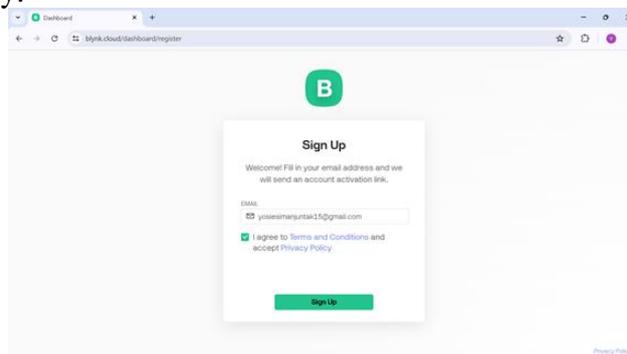
Simulasi sistem monitoring level oli menggunakan handphone sangat membutuhkan aplikasi blynk supaya dapat terkoneksi dengan handphone pengguna yang ingin mengetahui level oli dari jarak jauh menggunakan handphone. Aplikasi blynk akan terkoneksi dengan alat ketika kita memasukkan token dari blynk yang telah kita buat kedalam program Arduino IDE. Berikut tahapan yang dapat dilakukan dalam konfigurasi blynk :

1. Sebelum download aplikasi blynk, kita harus membuat akun blynk terlebih dahulu melalui <https://blynk.cloud/dashboard/login>. Kemudian klik Creat new account.



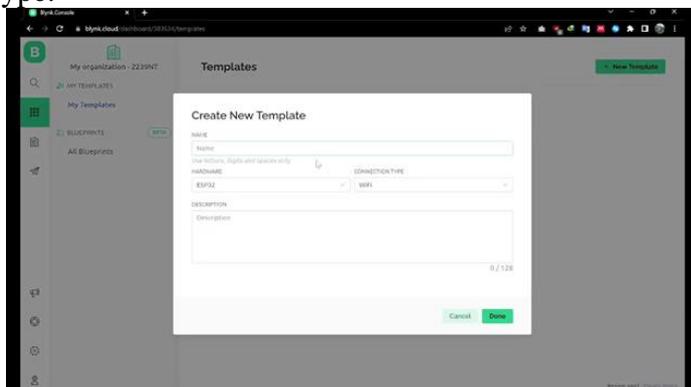
Gambar 6 Halaman daftar akun blynk
(Sumber: Dokumentasi Penulis, 2024)

Masukkan alamat email dan centang kolom I agree to Terms and Conditions and accept Privacy Policy.



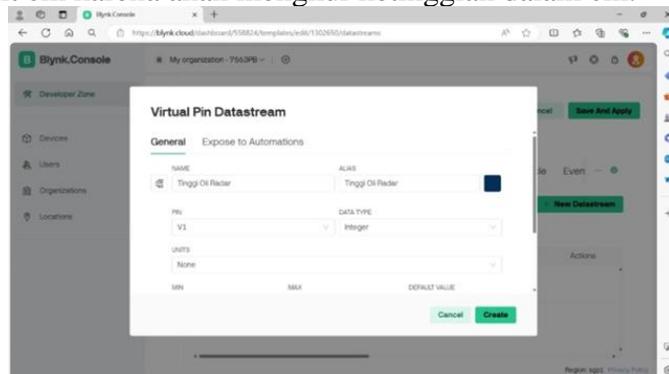
Gambar 7 Halaman verifikasi email
(Sumber: Dokumentasi Penulis, 2024)

2. Verifikasi melalui link yang dikirim ke email dan buat password untuk akun blynk. Setelah akun blynk berhasil dibuat, klik create new device. Kemudian isi kolom name untuk nama device yang akan kita buat. kemudian hardware yaitu ESP32, connection type.



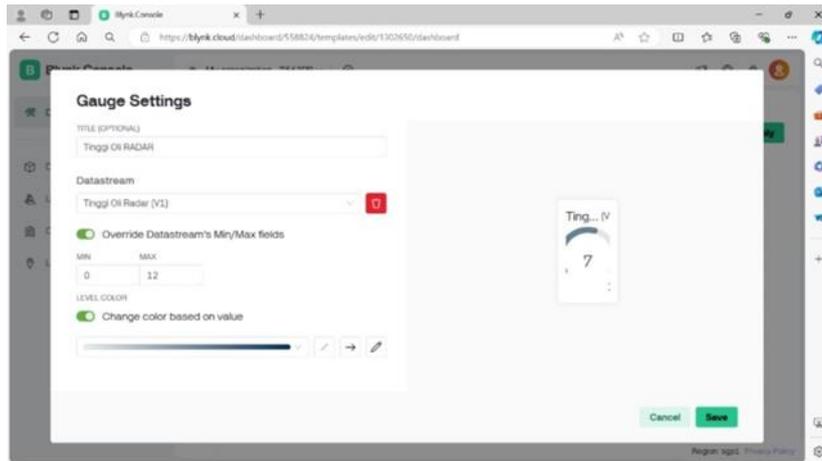
Gambar 8 Halaman create new template
(Sumber: Dokumentasi Penulis, 2024)

3. Setelah device dibuat, klik new datastream untuk membuat datastream baru. Buat dua datastream yaitu datastream untuk menampilkan volume oli dan datastream untuk menampilkan tinggi oli pada gearbox RADAR. Selanjutnya tambahkan virtual dan masukkan nama datastream yang ingin kita buat, lalu tambahkan pin misalnya V1, kemudian unit cm karena akan mengukur ketinggian dalam cm.



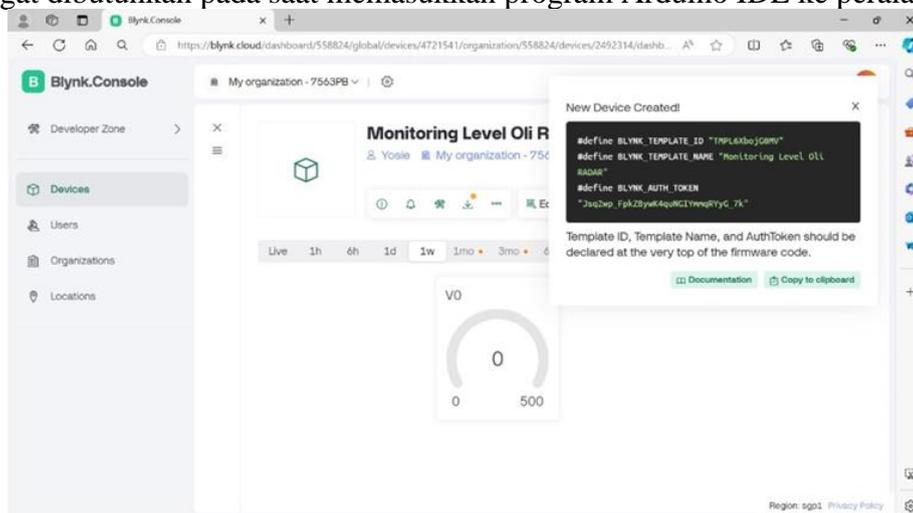
Gambar 9 Halaman Virtual Pin Datastream
(Sumber: Dokumentasi Penulis, 2024)

Pada bagian web dashboard, buat 2 label gauge untuk menampilkan volume oli dan tinggi oli lalu setting.



Gambar 4. 10 Halaman gauge settings
(Sumber: Dokumentasi Penulis, 2024)

4. Klik ke menu home, lalu pilih add first device untuk mendapatkan Blynk_Auth_Token; Blynk_Template_Id; Blynk_Template_Name. Kode tersebut sangat dibutuhkan pada saat memasukkan program Arduino IDE ke peralatan.



Gambar 11 Token dari aplikasi blynk
(Sumber: Dokumentasi Penulis, 2024)

b. Program Perangkat Lunak Telegram

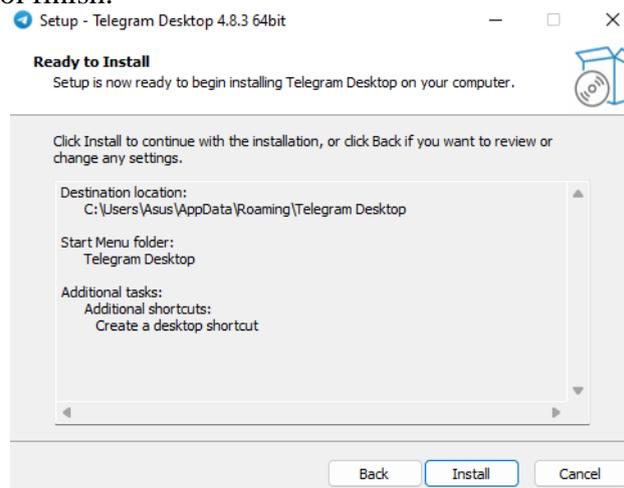
Dalam melakukan perancangan monitoring level oli pada Gearbox, dibutuhkan aplikasi Telegram untuk melakukan pengoperasian peralatan, sebelum melakukan pemrograman pada Arduino Ide terlebih dahulu kita harus melakukan konfigurasi pada aplikasi Telegram, hal ini bertujuan untuk membuat Bot Telegram dan mendapatkan ID pengguna yang akan dimasukkan kedalam pemrograman, Bot Telegram dan ID ini nanti dimasukkan kedalam pemrograman mikrokontroler agar akun Telegram pengguna dapat menerima informasi yang dikirimkan oleh peralatan dengan baik, dalam sistem perancangan monitoring level oli pada gearbox RADAR ini akan menggunakan satu bot Telegram namun dapat diakses oleh beberapa akun Telegram lainnya dengan cara memasukkan beberapa ID pengguna tersebut kedalam pemrograman mikrokontroler, sebelum membuat Bot Telegram dan mendapatkan ID user maka dilakukan beberapa Langkah dimulai dari instal aplikasi sampai dengan didapatkannya token Bot Telegram dan user ID Telegram, Berikut tahapan yang dapat dilakukan dalam konfigurasi Telegram :

1. Download aplikasi pada link website resmi yang telah disediakan oleh Telegram:
<https://desktop.telegram.org/>



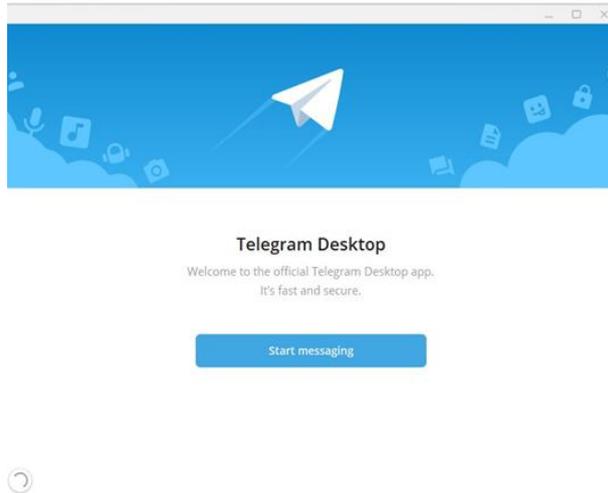
Gambar 12 Halaman download Telegram
(Sumber: Dokumentasi Penulis, 2024)

Setelah aplikasi terdownload, langkah selanjutnya yang harus dilakukan adalah menginstal aplikasi Telegram pada Device yang akan digunakan dengan cara memilih tombol install pada laman Setup Telegram Destop. Kemudian Centang Launch Telegram, kemudian klik tombol finish.

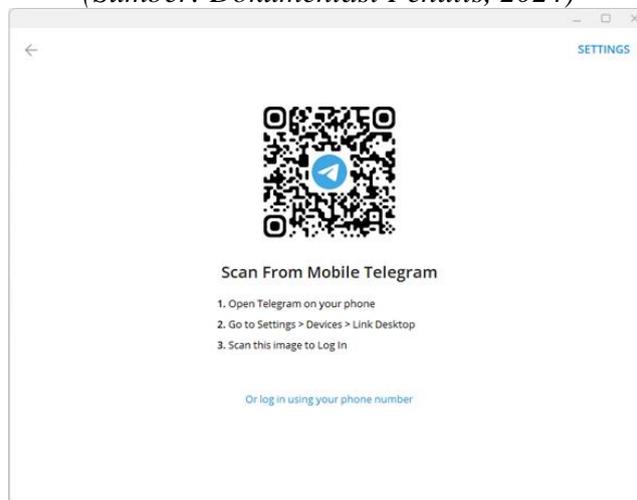


Gambar 13 Set up install aplikasi
(Sumber: Dokumentasi Penulis, 2024)

2. Buka aplikasi Telegram, mulai Telegram dengan cara menekan tombol “Start messaging”, kemudian log in akun dengan cara menggunakan scan QR atau dengan menggunakan nomor handphone yang telah didaftarkan sebelumnya, jika belum memiliki akun, maka dapat mendaftarkan nomor handphone dengan menggunakan Telegram pada handphone terlebih dahulu.

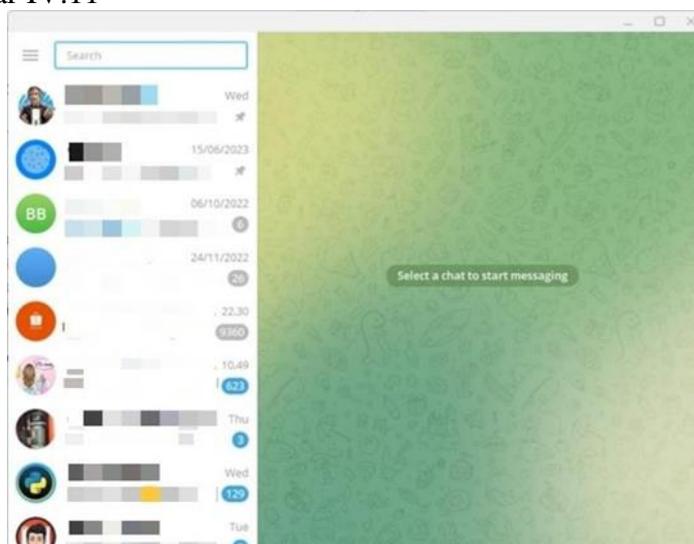


Gambar 14 Halaman untuk memulai aplikasi
(Sumber: Dokumentasi Penulis, 2024)



Gambar 15 QR code untuk login akun
(Sumber: Dokumentasi Penulis, 2024)

3. Setelah log in pada aplikasi Telegram, maka akan muncul tampilan Telegram seperti pada Gambar IV.11



Gambar 16 Tampilan chat Telegram
(Sumber: Dokumentasi Penulis, 2024)

4. Kemudian buatlah Bot Telegram, Langkah pertama yang harus dilakukan terlebih dahulu adalah melakukan pencarian “BotFather” pada kotak pencarian dan masuk ke halaman chat bot tersebut, BotFather merupakan bot resmi yang telah disediakan oleh Telegram yang berfungsi untuk mengelola bot Telegram pengguna.



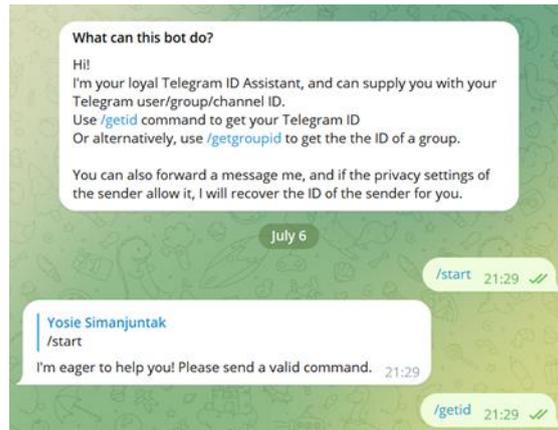
Gambar 17 Pencarian chat BotFather
(Sumber: Dokumentasi Penulis, 2024)

Selanjutnya akan muncul tombol “START” pada halaman chat, tekanlah tombol Start tersebut untuk memulai percakapan dengan BotFather, setelah itu BotFather akan mengirimkan chat berupa daftar informasi perintah yang dapat digunakan untuk mengendalikan atau mengelola berbagai pengaturan terkait Bot Telegram.



Gambar 18 Membuat Bot dan username untuk Bot
(Sumber: Dokumentasi Penulis, 2024)

5. Langkah selanjutnya yang harus dilakukan adalah mendapatkan ID Telegram pengguna, ID ini berfungsi sebagai identifikasi telegram pengguna yang memiliki ID berbeda setiap akun, dan ID digunakan untuk mengirimkan pesan notifikasi ke pengguna secara langsung melalui bot Telegram. Cara menemukan ID pada Telegram adalah dengan mengetikkan “IDBot” pada kotak pencarian, kemudian buka laman chat dan mulailah percakapan dengan cara mengklik tombol “START”, untuk mendapatkan user ID buat perintah chat berupa “/getid”, selanjutnya bot akan mengirimkan ID yang dapat digunakan pada pemrograman Arduino IDE.



Gambar 19 Perintah mendapatkan user ID
(Sumber: Dokumentasi Penulis, 2024)

c. Sinkronisasi Perangkat Keras dan Aplikasi

Monitor menggunakan handphone tentunya perangkat harus sinkron dengan blynk dan Telegram pada handphone. Untuk menyinkronkannya dibutuhkan sebuah laptop atau PC yang memiliki aplikasi Arduino IDE untuk menuliskan Bahasa pemrograman yang kita butuhkan untuk memberikan perintah kepada mikrokontroler. Berikut cara yang dilakukan untuk menyinkronkan atau mengintegrasikan perangkat ke handphone.

1. Sinkronisasi dengan blynk

Pengguna atau teknisi dapat memantau level oli melalui aplikasi blynk dikarenakan blynk telah tersinkronisasi dengan perangkat. Untuk menyinkronkan antara perangkat dan aplikasi blynk, kita harus menambahkan token blynk yang kita buat pada blynk dashboard kedalam program Arduino IDE.

```
#define BLYNK_TEMPLATE_ID "ID TEMPLATE"
#define BLYNK_TEMPLATE_NAME "NAMA TEMPLATE"
#define BLYNK_AUTH_TOKEN "Token Blynk"
```

Gambar 20 Token blynk device

(Sumber: Dokumentasi Penulis, 2024)

Untuk menampilkan nilai tinggi oli yang diukur, maka kita harus menambahkan pemrograman seperti pada Gambar 4.20 berikut.

```
// Mengirimkan data ke Blynk
Blynk.virtualWrite(V1, tinggiOli);
```

Gambar 21 Program untuk menampilkan nilai pada blynk

(Sumber: Dokumentasi Penulis, 2024)

2. Sinkronisasi dengan Telegram

Untuk menyinkronkan perangkat dengan telegram, kita harus menambahkan token bot yang telah kita buat pada telegram dan ID telegram kita serta ID telegram pengguna.

```
CTBot myBot;
String token = "Token telegram"; // Token bot Telegram
int chat_id = ID telegram; // Chat ID untuk Telegram
```

Gambar 22 Sinkronisasi dengan bot telegram

(Sumber: Dokumentasi Penulis, 2024)

Teknisi atau pengguna tidak akan mengetahui apabila oli RADAR telah mengalami penurunan hingga batas minimumnya jika tidak memiliki notifikasi. Oleh karena itu, supaya pengguna dapat menerima notifikasi melalui telegram, perangkat harus di program menggunakan Arduino ide dengan kode pernyataan “if”. Pernyataan if digunakan untuk membuat keputusan dalam kode berdasarkan kondisi tertentu. Pernyataan if memeriksa kondisi dan menjalankan blok kode tertentu jika kondisi tersebut benar (true).

```
// Mengirimkan notifikasi ke Telegram jika tinggi oli < 2 cm, dengan interval 1 menit
if (tinggiOli < 2.0) {
  unsigned long currentMillis = millis();
  if (currentMillis - lastNotificationTime >= notificationInterval) {
    String message = "▲□ Tinggi oli rendah: " + String(tinggiOli) + " cm. Segera tambahkan oli.";
    myBot.sendMessage(chat_id, message);
    lastNotificationTime = currentMillis; // Perbarui waktu notifikasi terakhir
  }
}
```

Gambar 23 Program untuk notifikasi Telegram
(Sumber: Dokumentasi Penulis, 2024)

d. Program Perangkat Lunak Arduino IDE

Simulasi sistem monitoring level oli pada gearbox RADAR sangat membutuhkan Arduino IDE (Integrated Development Environment). Arduino IDE merupakan aplikasi yang bersifat open-source yang dapat digunakan dalam menulis dan mengedit kode program yang akan diunggah ke mikrokontroler. bahasa pemrograman yang dapat digunakan pada aplikasi Arduino IDE ini adalah bahasa pemrograman C atau C++, dengan adanya aplikasi Arduino IDE ini bahasa pemrograman tersebut dapat dikompilasi menjadi bahasa mesin yang dapat dimengerti oleh board Arduino ESP32. Pada rancangan dilakukan pemrograman Arduino IDE untuk membaca sensor, kemudian informasi yang didapatkan akan dikirim oleh mikrokontroler untuk ditampilkan pada LCD dan di handphone melalui aplikasi blynk.

Berikut tahapan-tahapan dalam penginstalan Arduino IDE :

1. Download aplikasi Arduino IDE pada website resmi Arduino, berikut link halaman website resmi dari Arduino IDE:

<https://www.arduino.cc/en/Main/SoftwareDownloads>

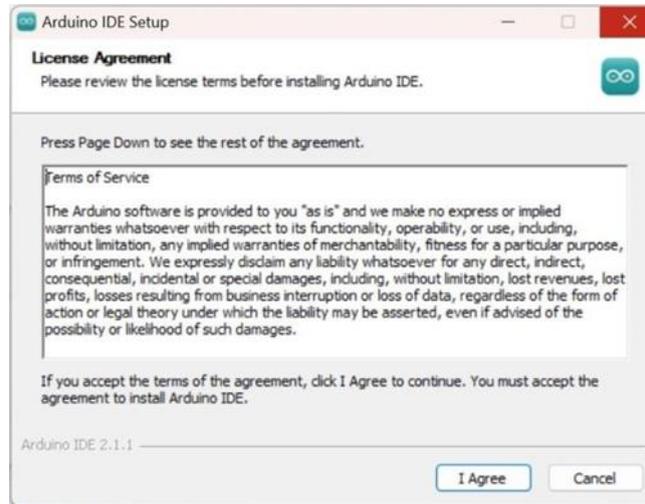


Gambar 24 Halaman download Arduino IDE
(Sumber: Dokumentasi Penulis, 2024)

Setelah halaman muncul, kemudian pilih Arduino sesuai dengan sistem operasi yang akan digunakan pada komputer untuk dilakukan penginstalan. Kemudian akan muncul halaman berikutnya yang dapat digunakan untuk mendownload aplikasi, klik “Just

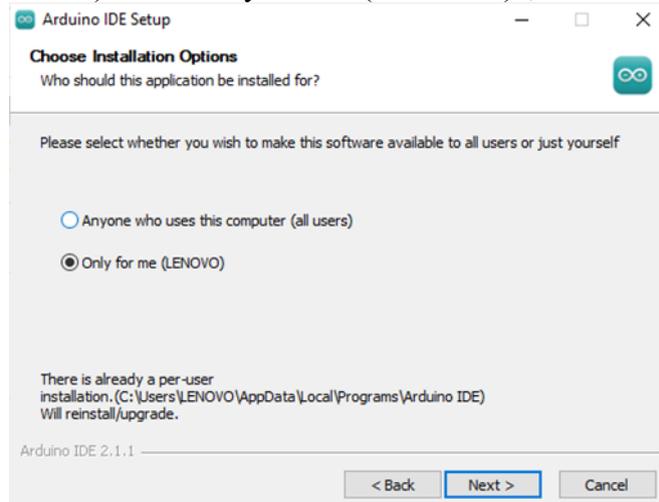
Download” atau bisa juga memilih “Contribute & Download” untuk memberikan donasi.

2. Setelah aplikasi terunduh pada komputer maka akan muncul license agreement atau Persetujuan Instalasi aplikasi, klik tombol I Agree untuk memulai instal aplikasi Arduino IDE.



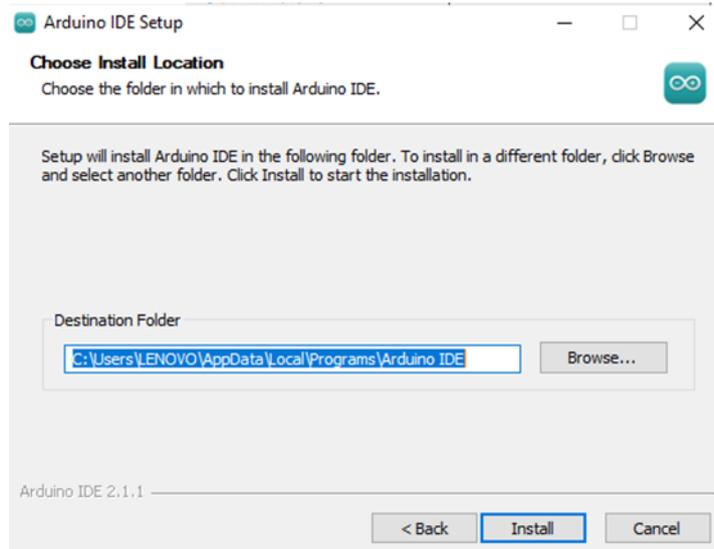
Gambar 25 License agreement untuk aplikasi
(Sumber: Dokumentasi Penulis, 2024)

3. Untuk Installation Option penggunaan aplikasi silahkan pilih “Anyone who uses this computer (All Users)” atau “only for me (LENOVO)”, kemudian klik tombol next.



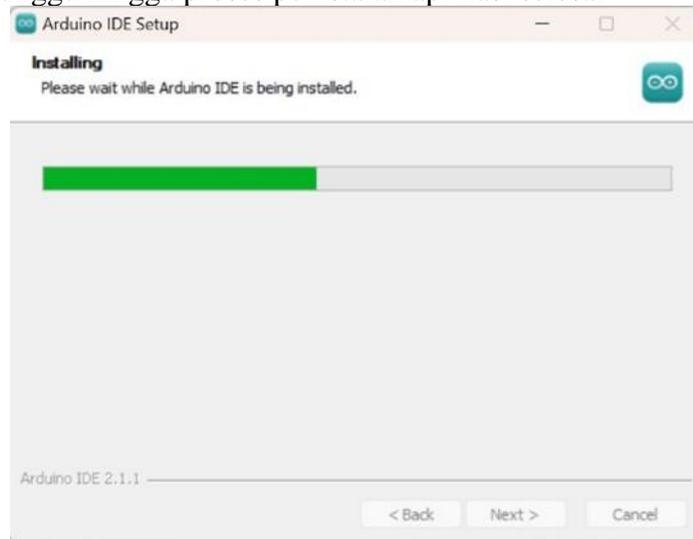
Gambar 26 Pilihan pengguna aplikasi
(Sumber: Dokumentasi Penulis, 2024)

4. Halaman selanjutnya yang akan tampil adalah halaman yang berfungsi untuk memilih folder tempat penyimpanan software Arduino IDE, klik tombol install untuk memulai proses instalasi aplikasi.



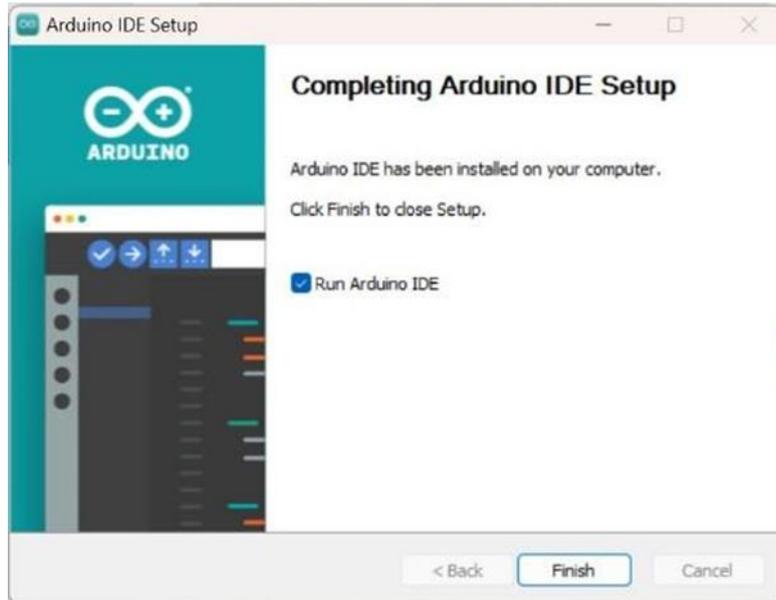
Gambar 27 Pemilihan lokasi instal Arduino IDE
(Sumber: Dokumentasi Penulis, 2024)

5. Kemudian tunggu hingga proses penistalan aplikasi selesai



Gambar 28 Proses instal Arduino IDE
(Sumber: Dokumentasi Penulis, 2024)

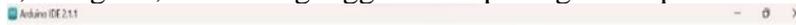
6. Setelah aplikasi selesai terinstal kemudian pilih tombol finish



Gambar 29 Proses instal selesai

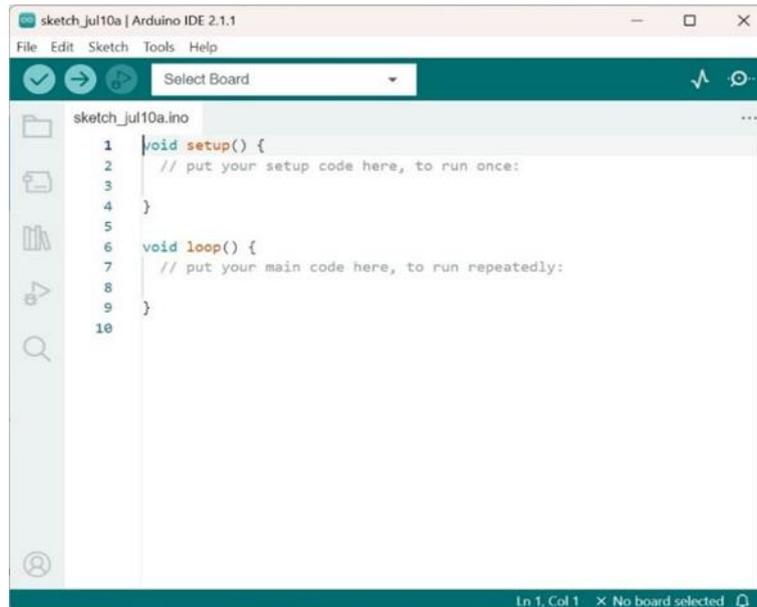
(Sumber: Dokumentasi Penulis, 2024)

7. Buka aplikasi untuk menjalankan software Arduino IDE. Aplikasi telah siap untuk membuat, mengedit, serta mengunggah kode pemrograman pada mikrokontroler.



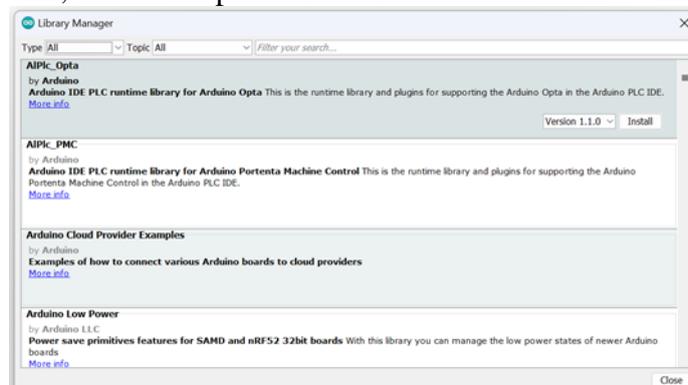
Gambar 30 Proses open aplikasi

(Sumber: Dokumentasi Penulis, 2024)



Gambar 31 Tampilan awal Arduino IDE
(Sumber: Dokumentasi Penulis, 2024)

8. Untuk dapat menjalankan pemrograman yang telah dibuat, maka dibutuhkan beberapa libraries sesuai dengan modul yang digunakan pada rancangan, cara pertama dalam mendownload libraries adalah dengan memilih tombol Tools > Manage libraries, atau dapat juga melakukan kombinasi tombol CTRL+Shift+I, setelah itu akan muncul kotak pencarian libraries, kemudian ketikkan nama libraries yang dibutuhkan pada kolom pencarian, setelah itu pilih libraries dan instal.



Gambar 32 Library Manager
(Sumber: Dokumentasi Penulis, 2024)

9. Sebelum melakukan upload pemrograman ke mikrokontroler, maka terlebih dahulu merubah board sesuai dengan mikrokontroler yang digunakan, rancangan dalam penelitian ini menggunakan board “ESP32 Dev Module”, Langkah-langkahnya adalah dengan mengklik menu Tools > Board > Arduino ESP32 > ESP32 Dev Module. Pemilihan port juga dibutuhkan agar pemrograman bisa diupload ke mikrokontroler, caranya adalah dengan mengklik menu Tools > Port, kemudian pilih sesuai dengan port yang terdeteksi pada komputer.
10. Level atau tinggi oli dapat dihitung oleh sensor ultrasonic HCSR04 dikarenakan adanya program untuk menghitung berapa tinggi oli pada wadah dimana nilai tinggi oli akan berubah apabila oli pada wadah ditambahkan atau dikurangi.

```

void sendSensor() {
  // Mengukur jarak oli dengan sensor ultrasonik
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  jarak = duration * 0.034 / 2; // konversi ke jarak sebenarnya (cm)

  // Menghitung tinggi oli
  tinggiOli = tinggiWadah - jarak;
  if (tinggiOli < 0) {
    tinggiOli = 0;
  }
}

```

Gambar 33 Program menghitung level oli
(Sumber: Dokumentasi Penulis, 2024)

Fungsi `sendSensor()` bertujuan untuk mengukur tinggi oli dalam wadah menggunakan sensor ultrasonik HC-SR04. Prosesnya melibatkan pengiriman pulsa ultrasonik dimana kecepatan ultrasonik adalah sekitar 234 m/s atau setara dengan 0,0343 cm/ μ s dan dibagi 2 karena waktu yang dibutuhkan oleh pulsa untuk memancar dan memantul kembali dari permukaan oli sehingga dapat menghitung jarak berdasarkan durasi tersebut. Kemudian, tinggi oli dihitung dengan mengurangi jarak yang diukur dari tinggi wadah. Jika hasil perhitungan tinggi oli kurang dari nol, nilai tersebut diatur menjadi nol untuk menghindari nilai negatif.

11. Upload pemrograman sudah bisa dilakukan dengan cara mengklik tombol upload, kemudian tunggu hingga penguploadan kodingan selesai.

```
TugasAkhirYosie | Arduino 1.8.19
File Edit Sketch Tools Help

TugasAkhirYosie §
}
}

void sendSensor() {
  // Mengukur jarak oli dengan sensor ultrasonik
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  jarak = duration * 0.034 / 2; // konversi ke jarak sebenarnya (cm)

  // Menghitung tinggi oli
  tinggiOli = tinggiWadah - jarak;
  if (tinggiOli < 0) {
    tinggiOli = 0;
  }

  // Menampilkan hasil di LCD
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Tinggi Oli:");
  lcd.print(tinggiOli);
  lcd.setCursor(0, 1);
  lcd.print(" cm ");

  // Mengirimkan data ke Blynk
  Blynk.virtualWrite(V1, tinggiOli);

  // Mengirimkan notifikasi ke Telegram jika tinggi oli < 2 cm, dengan interval 1 menit
  if (tinggiOli < 2.0) {
    unsigned long currentMillis = millis();
    if (currentMillis - lastNotificationTime >= notificationInterval) {
      String message = "AD Tinggi oli rendah: " + String(tinggiOli) + " cm. Segera tambahkan oli.";
      myBot.sendMessage(chat_id, message);
      lastNotificationTime = currentMillis; // Perbarui waktu notifikasi terakhir
    }
  }
}

void loop() {
  Blynk.run();
  timer.run();
}

Done uploading.
Hard resetting via RTS pin...
```

Gambar 34 Upload pemograman selesai
(Sumber: Dokumentasi Penulis, 2024)

Pembahasan Hasil Penelitian

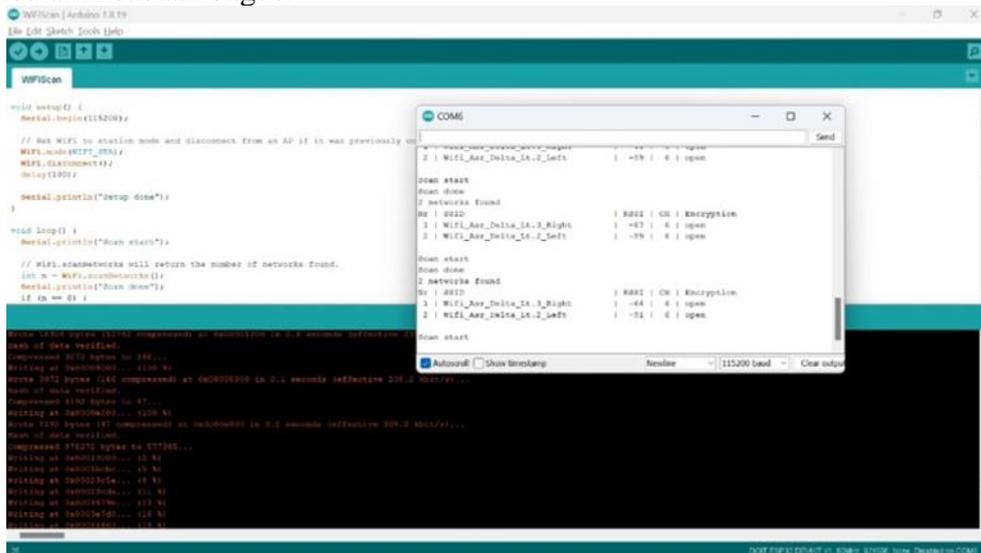
Rancangan simulasi sitem monitoring level ketinggian oli pada gearbox RADAR ini dirancang untuk merancang alat alternatif dalam melakukan pemantauan ketinggian dan volume oli yang terdapat pada gearbox motor RADAR dari jarak jauh melalui handphone apabila komputer SRG Airnav Cabang Medan mengalami masalah kembali. Dalam memastikan sistem bekerja dengan baik dan sesuai dengan kebutuhan yang telah ditentukan, maka dilakukan uji coba pada rancangan, hasil dari uji coba dan analisis yang diperoleh digunakan untuk melakukan penyesuaian, perbaikan atau peningkatan sebelum melanjutkan implementasi dan evaluasi lanjutan

1. Hasil Pengujian

Rancangan ini diuji pada dua bagian, yaitu pengujian terhadap modul dan pengujian pada sistem. Hasil pengujian bertujuan untuk memastikan apakah rancangan sudah berfungsi dengan baik atau tidak. Berikut pengujian yang dilakukan dalam penelitian ini.

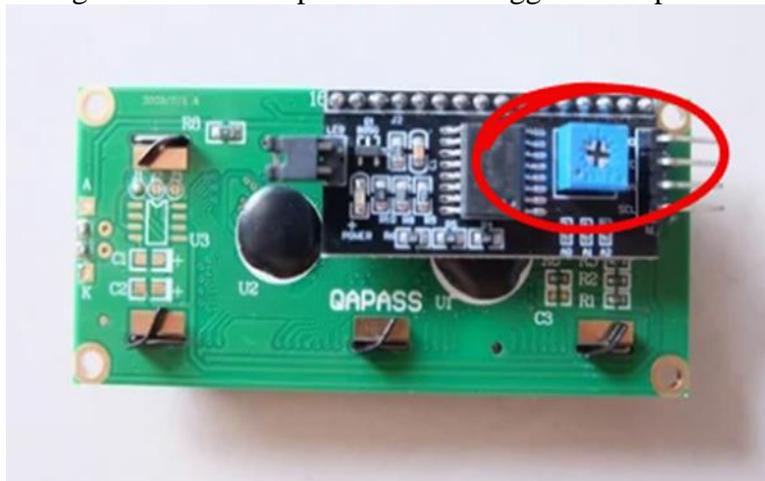
a. Pengujian modul

Pengujian modul pada rancangan ini dimulai dari memastikan bahwa modul ESP32 dapat dihidupkan oleh adaptor AC to DC. ESP32 hidup diindikasikan oleh nyala lampu led yang berwarna merah pada board ESP32. Selanjutnya, pastikan bahwa komponen ESP32 bisa terkoneksi dengan internet karena alat tidak dapat mengirimkan informasi level oli ke handphone apabila ESP32 tidak dapat terhubung ke internet melalui jaringan Wi-Fi. Untuk memastikannya, dimasukkan contoh program yang tersedia pada aplikasi Arduino IDE yaitu Wi-Fi Scan. Setelah diupload, ESP32 bisa terhubung ke Wi-Fi dan disimpulkan bahwa ESP32 dalam keadaan bagus.



Gambar 35 Scan Wi-Fi
(Sumber: Dokumentasi Penulis, 2024)

Selanjutnya dilakukan pengujian modul I2C LCD, dengan memasukkan contoh program Hello_Word yang telah tersedia pada Arduino IDE. I2C LCD terhubung ke baseboard ESP32. Pada pengujian awal, LCD tidak dapat menampilkan sesuatu. Untuk mengatasi hal tersebut, dilakukan pemutaran potensiometer. Potensiometer pada I2C berfungsi untuk mengatur kontras tampilan LCD sehingga teks dapat terlihat dengan jelas.



Gambar 36 Potensiometer I2C
(Sumber: Dokumentasi Penulis, 2024)

Pontensiometer diputar hingga LCD dapat menampilkan angka maupun huruf pada layar LCD. LCD yang dapat menampilkan angka ataupun huruf menandakan bahwa LCD

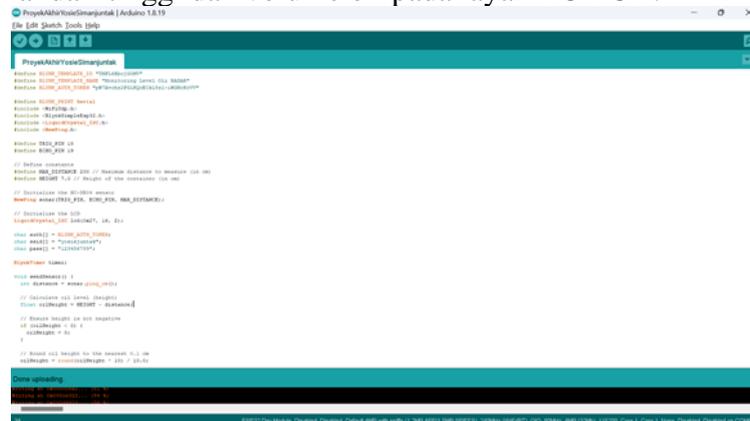
dalam keadaan normal.



Gambar 37 LCD normal

(Sumber: Dokumentasi Penulis, 2024)

Setelah seluruh modul sudah dipastikan berfungsi dengan baik, pemrograman yang telah dibuat di aplikasi Arduino IDE untuk mengukur tinggi dan volume oli dimasukkan ke alat melalui ESP32 menggunakan kabel USB. Alat yang telah terprogram dapat menampilkan nilai dari tinggi dan volume oli pada layar I2C LCD.

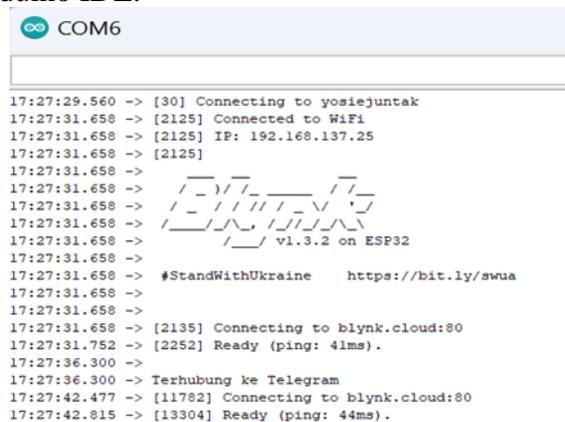


Gambar 38 Upload program berhasil

(Sumber: Dokumentasi Penulis, 2024)

b. Pengujian Sistem

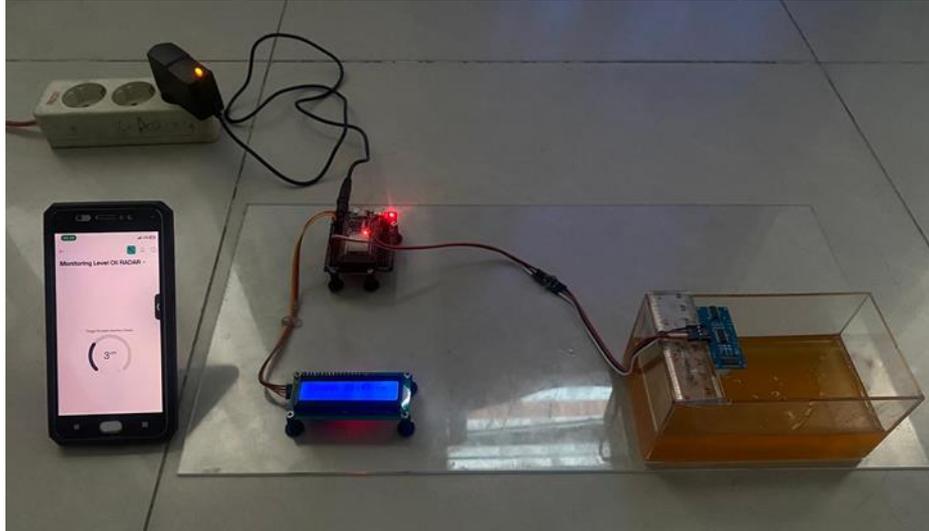
Pengujian yang dilakukan untuk memastikan bahwa sistem dapat berjalan dengan baik ialah dengan adanya nilai pada layar LCD dan notifikasi telegram serta nilai level oli di aplikasi blynk pada handphone pengguna. Langkah pertama yang dilakukan untuk memastikan bahwa alat dapat berkomunikasi dengan baik ialah dengan memperhatikan serial monitor pada Arduino IDE.



Gambar 39 Tampilan serial monitor Arduino IDE

(Sumber: Dokumentasi Penulis, 2024)

Pada Gambar 4.37 menunjukkan bahwa alat dapat terhubung dengan aplikasi blynk dan telegram. Untuk lebih memastikan bahwa sistem bekerja dengan baik maka dilakukan pengecekan kesamaan nilai yang ditampilkan pada layar LCD dan nilai yang ditampilkan pada aplikasi blynk serta memastikan bahwa telegram dapat menerima notifikasi apabila level oli pada wadah kurang dari batas minimum yang ditentukan yaitu 2 cm.



Gambar 40 Dokumentasi Alat
(Sumber: Dokumentasi Penulis, 2024)

Berikut perbandingan tampilan pada LCD, Blynk dan Telegram pada saat oli pada wadah dikurangi.

Tabel 3 Perbandingan tampilan pada LCD, Blynk dan Telegram

Tampilan pada LCD	Tampilan Pada Blynk	Tampilan Pada Telegram
		Tidak ada notifikasi

	<p>Tinggi Oli pada Gearbox Radar</p> 	<p>⚠️⚠️⚠️ Tinggi oli rendah: 1 cm. Segera tambahkan oli.</p>
	<p>Tinggi Oli pada Gearbox Radar</p> 	<p>⚠️⚠️⚠️ Tinggi oli rendah: 0 cm. Segera tambahkan oli.</p>

Sumber: Dokumentasi penulis, 2024.

Pada Tabel 3 menunjukkan bahwa sistem telah bekerja dengan baik karena dapat mengirimkan informasi terkait level oli dari perangkat yang ditampilkan pada layar LCD dan Blynk serta mengirimkan notifikasi berupa peringatan ketika tinggi oli melewati batas minimum yaitu 2 cm.

KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan mengenai rancangan simulasi sistem monitoring level ketinggian oli pada gearbox RADAR via handphone berbasis ESP32 di Perum LPPNPI cabang Medan, maka dapat ditarik beberapa kesimpulan:

- Rancangan telah bekerja dengan baik dalam memberikan informasi terkait level oli ke handphone pengguna melalui aplikasi blynk karena nilai pada layar LCD sama dengan nilai yang ditampilkan pada aplikasi blynk.
- Rancangan dapat memberikan notifikasi ke handphone pengguna melalui telegram ketika tinggi oli kurang dari batas minimal yang ditetapkan yaitu 2cm.
- Sistem ini dirancang menggunakan komponen yang murah dan mudah didapatkan, menjadikannya solusi yang efektif untuk monitoring level oli pada gearbox RADAR karena dapat menekan biaya instalasi nya.
- Dengan menggunakan ESP32 yang terhubung ke handphone melalui jaringan internet, sistem ini memungkinkan monitoring level oli secara real-time yang dapat membantu teknisi dalam deteksi dini masalah dan pemeliharaan preventif guna mengurangi risiko kerusakan pada gearbox.
- Sistem tidak dapat mengirimkan level ketinggian oli pada aplikasi blynk dan notifikasi telegram apabila handphone pengguna dan ESP 32 tidak terkoneksi dengan internet.

Saran

Berikut beberapa saran dari penelitian ini :

- Penelitian ini menggunakan wadah sebagai tempat oli dan untuk penelitian selanjutnya dapat menggunakan gearbox sebelum diimplementasikan ke gearbox RADAR.
- Penelitian selanjutnya diharapkan menambah pompa otomatis apabila gearbox RADAR mengalami penurunan supaya teknisi tidak harus naik ke gearbox yang berada dibawah antenna RADAR untuk melakukan penambahan oli.

- c. Diharapkan dapat melakukan penelitian lebih lanjut supaya alat memberikan informasi terkait level oli tanpa dipengaruhi oleh jaringan internet.

DAFTAR PUSTAKA

- Aliakbarian, Hadi, Mohammad Khak, Morteza Shahpari, Ali Nooraei Yeganeh, Farhad Mazlumi, Seyed Hesamoddin Najmolhoda, Reza Baghlani, and Rasoul Salimi Nejad. 2019. "An Efficient Multi-Beam Array Architecture for L-Band Secondary Surveillance Radars." *Radioengineering* 27(1):84–91. doi: 10.13164/RE.2019.0084.
- Ardiliansyah, Aldiaz Rasyid, Mariana Diah Puspitasari, and Teguh Arifianto. 2021. "Rancang Bangun Prototipe Pompa Otomatis Dengan Fitur Monitoring Berbasis IoT Menggunakan Sensor Flow Meter Dan Ultrasonik." 5(36):59–67.
- Arief, Ulfah Mediaty. 2011. "Pengujian Sensor Ultrasonik PING Untuk Pengukuran Level Ketinggian Dan Volume Air." 09(02).
- Batool, Sidra, Fabrizio Frezza, Fabio Mangini, and Patrizio Simeoni. 2020. "Introduction to Radar Scattering Application in Remote Sensing and Diagnostics: Review." *Atmosphere* 11(5):1–17. doi: 10.3390/atmos11050517.
- Bhatta, Niraj Prasad. 2020. "RADAR and Its Applications - MOAM.INFO.Pdf." (March).
- Dwi, Afriyanto. 2023. "Peran Personel Keamanan Penerbangan Yang Kompeten Dalam Meningkatkan Keamanan Dan Keselamatan Penerbangan." *Jurnal Multidisiplin Indonesia* 2(6):1243–52. doi: 10.58344/jmi.v2i6.275.
- Faramarzi, Sajad, Hossein Heidari Tabrizi, and Azizeh Chalak. 2019. "Telegram: An Instant Messaging Application to Assist Distance Language Learning." *Teaching English with Technology* 19(1):132–47.
- Kaviyarasu, A., A. Saravanakumar, and K. Pradeesha. 2018. "Air Traffic Control Secondary Radar 1." 119(12):2935–44.
- Liu, Lan, Kun Kang, Yingjie Xi, Zhengxi Hu, Jingyi Gong, and Geng Liu. 2022. "Optimal Design and Experimental Verification of Low Radiation Noise of Gearbox." *Chinese Journal of Mechanical Engineering (English Edition)* 35(1). doi: 10.1186/s10033-022-00801-5.
- Pereira, Gilroy P., Mohamed Z. Chaari, and Fawwad Daroge. 2023. "IoT-Enabled Smart Drip Irrigation System Using ESP32." *Internet of Things* 4(3):221–43. doi: 10.3390/iot4030012.
- Poetra, Ade Arya, Reza Nandika, Toni Kusuma Wijaya, Program Studi, Teknik Elektro, Fakultas Teknik, and Universitas Riau Kepulauan. 2023. "PROTOTIPE SISTEM MONITORING KETINGGIAN AIR PADA TANGKI BERBASIS INTERNET OF THINGS Program Studi Teknik Elektronika , Politeknik Negeri Padang." 6(1):97–108.
- Suryantoro, Hery. 2019. "Prototype Sistem Monitoring Level Air Berbasis Labview Dan Arduino Sebagai Sarana Pendukung Praktikum Instrumentasi Sistem Kendali." *Indonesian Journal of Laboratory* 1(3):20. doi: 10.22146/ijl.v1i3.48718.
- Sutikno, Tole, Lina Handayani, Deris Stiawan, Munawar Agus Riyadi, and Imam Much Ibnu Subroto. 2016. "WhatsApp, Viber and Telegram: Which Is the Best for Instant Messaging?" *International Journal of Electrical and Computer Engineering* 6(3):909–14. doi: 10.11591/ijece.v6i3.10271.