



Rancang Bangun Sistem Kontrol Fire House Roller Berbasis IoT pada Industri PKT

¹Endrow Seftian, ²Turahyo, ³Arfittariah

^{1,2,3}Teknik Elektro, Sekolah Tinggi Teknologi Bontang

¹endrowseftian27@gmail.com, ²Turahyoahyo@gmail.com, ³fittarhia@gmail.com

ABSTRACT

The fire hose reeling system in the K3 Department of PT. Pupuk Kalimantan Timur is still carried out manually by firemen. This manual process poses ergonomic risks that can lead to fatigue and injuries due to less-than-ideal working positions. Furthermore, the absence of assistive tools makes the work less efficient and slows down the hose reeling process. This research aims to design and create a fire hose reeling device based on the NodeMCU ESP32, controlled via the Blynk application on a smartphone. The device is designed to simplify and speed up the fire hose reeling process. The developed system has several key features, such as virtual buttons for start, stop, rotation direction control (clockwise and counterclockwise), and a jog function for position adjustment. Additionally, there is motor speed control with three levels: low, medium, and fast, as well as a real-time display of rotations per minute (RPM) on the Blynk application. The test results show that the fire hose reeling device functions according to the design and can be operated well via a smartphone. This device can reduce the physical workload of personnel, increase time efficiency, and improve the ergonomic aspects of the fire hose reeling process, thereby supporting the optimal performance of firemen in the field.

Keywords: Automatic Fire Hose Winder, Blynk, NodeMCU ESP 32, Smarthphone

ABSTRAK

Sistem penggulungan *fire hose* di Departemen K3 PT. Pupuk Kalimantan Timur masih dilakukan secara manual oleh petugas *fire man*. Proses manual ini memiliki risiko ergonomis yang dapat menyebabkan kelelahan dan cedera akibat posisi kerja yang kurang ideal. Selain itu, ketiadaan alat bantu membuat pekerjaan menjadi kurang efisien dan memperlambat proses penggulungan selang. Penelitian ini bertujuan merancang dan membuat alat penggulung *fire hose* yang berbasis *Nodemcu ESP32* yang dikendalikan melalui aplikasi *Blynk* di *smartphone*. Alat ini dirancang untuk memudahkan dan mempercepat proses penggulungan *fire hose*. Sistem yang dikembangkan memiliki beberapa fitur utama, seperti tombol virtual untuk *start*, *stop*, pengaturan arah putaran (*clock wise* dan *counter clock wise*), serta fungsi *jog* untuk penyesuaian posisi. Selain itu, terdapat pengaturan kecepatan motor dalam tiga tingkat yaitu *low*, *medium*, dan *fast* serta tampilan nilai putaran per menit (RPM) secara *real-time* di aplikasi *Blynk*. Hasil pengujian menunjukkan bahwa alat penggulung *fire hose* ini berfungsi sesuai dengan rancangan dan dapat dioperasikan dengan baik melalui *smartphone*. Alat ini mampu mengurangi beban kerja fisik petugas, meningkatkan efisiensi waktu, dan memperbaiki aspek ergonomis dalam proses penggulungan *fire hose*, sehingga dapat mendukung kinerja petugas *fire man* secara optimal di lapangan.

Kata Kunci: Blynk, NodeMCU ESP 32, Penggulung Fire Hose Otomatis, Smartphone



I. PENDAHULUAN

Industri petrokimia seperti PT. Pupuk Kalimantan Timur memiliki risiko tinggi terhadap kebakaran dan ledakan. Meskipun teknologi terus berkembang, kejadian kebakaran tetap meningkat dan tidak menunjukkan penurunan yang signifikan. Kebakaran merupakan salah satu ancaman keselamatan yang serius, dengan potensi kerugian besar baik terhadap keselamatan jiwa maupun harta benda. Oleh karena itu, upaya pencegahan sangat diperlukan untuk meminimalkan atau bahkan menghilangkan risiko kebakaran dan mengurangi potensi kerugian tersebut [1].

Fire truck yang digunakan di PT. Pupuk Kalimantan Timur belum dilengkapi dengan *alat fire hose roller*. Proses penggulungan selang masih dilakukan secara manual, melalui sistem mekanik, elektrik, konstruksi bodi, dan sistem perpipaan yang membentuk satu kesatuan unit pemadam kebakaran. Media yang digunakan untuk memadamkan api disesuaikan dengan kebutuhan di lapangan.

Fire hose (selang pemadam) merupakan alat darurat yang digunakan untuk mendistribusikan air dari *hydrant* atau *fire truck* ke *nozzleman* dalam upaya penanggulangan kebakaran di dalam maupun di luar area pabrik PT. Pupuk Kalimantan Timur. Namun, proses penggulungan *fire hose* (selang pemadam) setelah pengujian dilakukan secara manual tanpa bantuan alat khusus. Setiap 2 minggu dilakukan pengujian rutin terhadap *fire hose*, di mana setiap *fireman* bertugas menggulung 11 selang secara manual.

Proses ini sangat membebani petugas karena membutuhkan tenaga besar dan melibatkan gerakan berulang dalam posisi yang tidak ergonomis (membungkuk dan jongkok). Kondisi ini berisiko menyebabkan cedera otot, sementara pengujian rutin tetap harus dilakukan untuk memastikan seluruh alat darurat siap digunakan kapan saja.

Untuk meringankan beban kerja dan meningkatkan efisiensi, diperlukan alat bantu berupa *fire hose roller* berbasis *Internet of Things* (IoT). Alat ini dirancang menggunakan modul mikrokontroler *NodeMCU ESP 32* dan akan diterapkan di divisi *Fire and Rescue* PT. Pupuk Kalimantan Timur. Dengan alat ini, proses penggulungan selang dapat dilakukan secara otomatis, mengurangi beban fisik *fireman*, mempercepat pekerjaan, dan mengurangi risiko cedera.

Alat ini dapat dikendalikan melalui *smartphone* petugas, memungkinkan pengaturan kecepatan dan pemantauan proses penggulungan *fire hose* (selang pemadam) secara *real-time*. Dengan inovasi ini, pekerjaan *fireman* menjadi lebih ringan,

lebih efisien, dan lebih aman dalam mendukung kesiapan sistem pemadam kebakaran di lingkungan PT. Pupuk Kalimantan Timur.

II. TINJAUAN UMUM

A. Definisi *Fire truck*

Mobil pemadam kebakaran atau *fire truck* merupakan kendaraan bermotor baik itu truk atau mobil beroda empat jenis lain yang telah dipasang alat pemadam kebakaran [2]. *Fire truck* atau yang lebih dikenal dengan mobil pemadam kebakaran adalah rangkaian beberapa unit sistem. Secara garis besar unit sistem dalam mobil pemadam kebakaran terdiri dari, 1.) *Engine dan chassis*, 2.) Pompa dan *Power Take Off* (PTO). Kedua hal tersebut disatukan melalui sistem mekanik, elektrik, konstruksi bodi dan sistem perpipaan hingga menjadi satu unit utuh yang nantinya dapat berfungsi alat pemadam kebakaran. Media yang digunakan untuk memadamkan api akan disesuaikan dengan kebutuhannya [2].

B. Jenis-jenis Mobil Pemadam Kebakaran

Dalam hal ini ada beberapa jenis mobil pemadam kebakaran berdasarkan fungsinya, yaitu:

1. Mobil Pemadam Kebakaran Jenis Busa.

Sesuai dengan namanya mobil pemadam kebakaran jenis ini dilengkapi dengan fasilitas yang dapat digunakan untuk menyemprotkan bahan pemadam yang berjenis busa.

2. Mobil Pemadam Kebakaran Jenis Tepung Kimia.

Mobil pemadam kebakaran jenis tepung kimia dalam mobil yang dirancang secara khusus sehingga bisa digunakan untuk menyemprotkan *dry chemical* yang digunakan sebagai bahan pemadam.

3. Mobil Pemadam Kebakaran Bahan Pemadam Ganda

Mobil pemadam kebakaran jenis bahan pemadam ganda 8 merupakan mobil pemadam yang dirancang untuk menyemprotkan beberapa bahan sekaligus seperti busa, air ataupun *dry chemical*.

4. Mobil Pemadam Kebakaran Jenis Pompa

Mobil pemadam kebakaran jenis pompa merupakan mobil pemadam yang dirancang untuk membawa pompa.

5. Mobil Pemadam Kebakaran Jenis Tangga

Mobil pemadam kebakaran jenis tangga memiliki ukuran yang paling besar di antara mobil pemadam kebakaran jenis lain. Mobil pemadam kebakaran jenis ini memiliki tangga, yang secara otomatis dapat naik atau turun.

C. Peralatan Yang Harus Ada Pada Mobil Pemadam Kebakaran

Mobil pemadam kebakaran merupakan sistem



yang terdiri dari berbagai alat penting yang bekerja sama. Tidak semua alat pemadam dibawa, namun ada peralatan utama yang wajib ada agar fungsi mobil maksimal. Alat-alat tersebut meliputi [2]:

1. Selang Pemadam Kebakaran

Untuk meningkatkan jangkauan semprot air, biasanya dapat kita temui pada kotak *hydrant*. Jika tidak ada *hose* pada mobil pemadam kebakaran maka kuantitas air yang disemprotkan tidak terlalu banyak.

2. Fire Nozzle atau pengarah air fire nozzle

Mempunyai tekanan yang berasal dari tangki mobil dan pipa mobil pemadam kebakaran. *Nozzle* berfungsi untuk menutup *Shut off* atau katup penutup yang bermanfaat dalam mengontrol air yang keluar.

3. Kampak Pemadam Kebakaran

Disebut juga *Fire Axe* adalah salah satu perlengkapan yang digunakan oleh petugas pemadam kebakaran. Kampak tersebut berfungsi untuk menjebol pintu atau lainnya saat keadaan darurat.

4. Fire Helmet

Helm khusus yang dirancang untuk petugas pemadam kebakaran. Helm ini dapat melindungi kepala petugas pemadam kebakaran dari cipratan api karena bahan yang digunakan untuk helm adalah bahan tahan akan api.

5. Baju tahan panas Firefly

Dibuat untuk menjaga petugas pemadam kebakaran supaya tetap aman serta meminimalisir terjadinya kecelakaan pada petugas pemadam kebakaran dan juga berfungsi untuk menyelamatkan korban kebakaran.

D. Pengoperasian Mobil Pemadam Kebakaran

Sebelum mobil pemadam kebakaran berangkat, perlu dilakukan pemeriksaan terhadap kondisi kendaraan seperti tekanan ban, bahan bakar, radiator, pelumas, aki, dan bahan pemadam. Petugas juga harus memahami fungsi berbagai komponen seperti tombol pompa busa dan air, transfer valve, handle gas, pressure gauge, discharge valve, dan pressure relief valve untuk mendukung kelancaran tugas di lapangan [2].

E. Internet Of Things (IoT)

Internet of Things (IoT) adalah teknologi yang menghubungkan manusia dan perangkat melalui internet [3][4], memungkinkan otomatisasi dan penggunaan teknologi pintar kapan saja dan di mana saja. IoT bermula dari penggunaan RFID oleh Kevin Ashton pada 1997 dan kini berkembang menjadi sistem besar dengan triliunan perangkat yang saling terhubung. IoT memiliki tiga karakteristik utama: perangkat pengukur pada objek, terminal otonom yang terhubung, dan layanan cerdas. Prosesnya

melibatkan koneksi perangkat ke platform IoT, pengumpulan dan analisis data, serta pemanfaatan informasi untuk pengambilan keputusan yang cerdas. Penerapan IoT mencakup berbagai bidang:

1. Pendidikan: mendukung *m-learning* dan *e-learning* melalui perangkat seperti tablet dan *e-book*.
2. Kesehatan: membantu peningkatan layanan berbasis internet yang efisien, aman, dan berkelanjutan untuk populasi rentan.
3. Keamanan: menghadirkan *IoT Security* untuk memastikan perangkat aman sejak awal pengembangan hingga penggunaan.
4. Transportasi: penggunaan GPS dan *geo-fencing* untuk pelacakan aset secara *real-time*, meningkatkan efisiensi dan transparansi logistik.

F. Perangkat Pendukung

Perangkat pendukung yang dibutuhkan berbagai jenis alat atau jenis perangkat yang membantu fungsi, tugas dan aktivitas tertentu.

1. Blynk IoT

Blynk adalah platform berbasis iOS dan Android yang memudahkan pengguna, termasuk pemula, untuk mengendalikan modul seperti *Arduino*, *Raspberry Pi*, dan *Wemos* melalui internet. Dengan fitur *drag and drop*, proyek dapat dibuat dengan cepat. *Blynk* tidak terbatas pada satu jenis modul dan memungkinkan kontrol perangkat dari jarak jauh selama terhubung ke internet, sejalan dengan konsep IoT (*Internet of Things*) [5].

2. Android

Android adalah sistem operasi berbasis Linux untuk perangkat bergerak seperti *smartphone*, yang mencakup sistem operasi, *middleware*, dan aplikasi inti. Android bersifat open source dan menggunakan bahasa pemrograman Java, sehingga mudah dikembangkan dan memiliki pertumbuhan pesat dibanding OS lain. Pertama kali dikembangkan oleh *Android Inc.* pada 2003, Android kemudian diakuisisi Google pada 2006. SDK Android Beta dirilis pada 12 November 2007, dan versi resmi pertamanya digunakan pada HTC dengan Android 1.0 yang dirilis pada 23 September 2008. Sejak itu, Android terus berkembang dan digunakan secara luas oleh berbagai produsen *smartphone* [6].

3. Arduino IDE (Integrated Development Environment)

Arduino IDE (Integrated Development Environment) adalah perangkat lunak yang digunakan untuk memprogram *board Arduino*. *Software* ini berfungsi sebagai media utama dalam menulis, mengunggah, dan menjalankan kode pada perangkat *Arduino*, dan dapat diunduh secara gratis



melalui situs resmi *Arduino* [7].

4. *Fritzing*

Fritzing adalah perangkat lunak *open source* yang dirancang untuk membantu pengguna, terutama pemula, dalam merancang, memvisualisasikan, dan membuat diagram sirkuit elektronik. Aplikasi ini memungkinkan pembuatan dan pengeditan skematik, *breadboard*, serta *layout PCB* secara virtual, dilengkapi dengan perpustakaan komponen yang luas. *Fritzing* juga memungkinkan pengguna menyesuaikan atau memodifikasi kode sumbernya sesuai kebutuhan [9].

G. *NodeMCU ESP 32*

ESP32 adalah papan pengembangan berbasis chip ESP32 buatan *Espressif*, yang dilengkapi dengan modul WiFi terintegrasi dan cocok untuk aplikasi IoT. Menggunakan mikroprosesor *Xtensa dual-core* 240 MHz, papan ini memiliki berbagai fitur seperti switch antena, filter, RF balun, penguat, dan modul pengatur daya. Dengan pin I/O yang mudah diakses, ESP32 memudahkan pengembangan melalui *breadboard* atau kabel *jumper*. Papan ini merupakan penerus dari ESP8266 dan diproduksi di Shanghai, Cina [10].

H. *Modul Step Down LM2596*

Modul Step Down LM2596 adalah modul penurunan tegangan dengan *output* yang dapat diatur menggunakan potensiometer dan tetap stabil meskipun tegangan *input* berubah. *Trafo step down* juga berfungsi menurunkan tegangan dengan lilitan sekunder lebih sedikit dari primer, umum digunakan dalam adaptor AC-DC. Contohnya, adaptor 12V atau 9V dapat diturunkan menjadi 5V atau 3.3V untuk kebutuhan *mikrokontroler*, *smartphone*, atau perangkat elektronik lainnya [8].

I. *Driver Motor BTS7960*

MOSFET BTS7960 adalah komponen pengendali tegangan dalam rangkaian *power supply* jenis *switching*, yang memiliki terminal sumber, gerbang, dan saluran. Pada skripsi ini, BTS7960 digunakan sebagai *driver motor* untuk mengatur kecepatan putar motor melalui kontrol PWM. *Driver* ini mendukung tegangan 6V–27V dan arus hingga 43A [11],[12],[13].

J. *Sensor Rotary Encoder*

Rotary Encoder adalah perangkat elektronik yang memantau gerakan dan posisi dengan menggunakan sensor optik untuk menghasilkan pulsa serial yang merepresentasikan gerakan, posisi, dan arah. *Encoder* ini terdiri dari piringan tipis dengan lubang-lubang, LED yang mengirimkan cahaya, dan *phototransistor* yang mendeteksi cahaya. Ketika piringan berputar,

cahaya LED mencapai *phototransistor* melalui lubang-lubang, menghasilkan pulsa gelombang persegi. Jumlah lubang pada piringan menentukan akurasi *encoder*, dengan semakin banyak lubang menghasilkan akurasi lebih tinggi [14].

K. *Incremental Encoder*

Incremental Encoder terdiri dari dua *track* dan dua sensor (*channel A* dan *B*). Ketika poros berputar, pulsa muncul di masing-masing *channel* dengan frekuensi yang sebanding dengan kecepatan putar. Hubungan fasa antara *channel A* dan *B* menentukan arah putaran, karena kedua *channel* memiliki perbedaan fasa seperempat putaran (*quadrature signal*). Jumlah pulsa dan resolusi dapat digunakan untuk mengukur putaran dan arah putaran [14].

L. *Selector Switch*

Selector Switch (Saklar Pemilih) adalah saklar yang dioperasikan dengan memutar dan digunakan untuk memilih lebih dari dua posisi. Biasanya diterapkan pada rangkaian seperti pencatu daya, multimeter (untuk memilih fungsi pengujian), dan pemilih suhu pada oven. *Selector Switch* juga dikenal sebagai *Rotary Switch* [15].

M. *Power Supply DC*

Adaptor adalah perangkat yang mengubah tegangan AC (arus bolak-balik) menjadi tegangan DC (arus searah). Fungsi utama adaptor adalah menyediakan catu daya untuk perangkat elektronik yang membutuhkan tegangan DC. Terdapat dua jenis adaptor, yaitu adaptor *switching* dan adaptor trafo *step-down*. Adaptor juga berperan menyambungkan sumber tegangan DC dan menyesuaikan arus listrik sesuai kebutuhan perangkat.

N. *Motor DC*

Motor DC adalah motor listrik yang mengubah energi listrik menjadi energi mekanik berupa gerak rotasi dengan menggunakan suplai tegangan arus searah (DC). Motor ini terdiri dari dua bagian utama: stator (bagian diam) yang berfungsi sebagai kumparan medan dan rotor (bagian bergerak) yang berfungsi sebagai kumparan jangkar. Motor DC bekerja berdasarkan prinsip gaya Lorentz, dimana gaya terbentuk saat arus mengalir dalam medan magnet. Motor DC juga dilengkapi dengan komutator yang berfungsi sebagai saklar kutub ganda untuk mengatur arah arus [16].

III. METODE PENELITIAN

A. Lokasi dan Waktu Penelitian

Penelitian dilakukan pada bulan September 2023



sampai dengan Februari 2025. Kegiatan penelitian terdiri dari tiga bagian, yaitu perencanaan alat, pembuatan alat dan uji coba alat. Perencanaan, pembuatan, dan pengujian alat dilakukan di rumah penulis, Laboratorium Teknik Elektro STITEK Bontang dan PT. Pupuk Kalimantan Timur Departemen K3 di bagian Fire & Rescue.

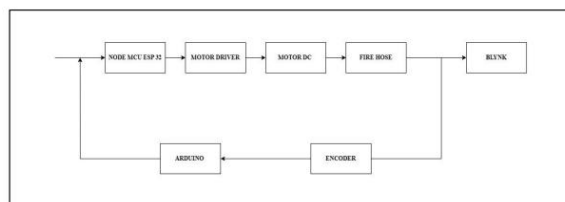
B. Tahap Pengerjaan Penelitian

Tahap pengerjaan penelitian pembuatan sistem *fire hose roller* berbasis IoT dengan mikrokontrol *NodeMCU ESP 32* terdiri dari beberapa langkah:

1. Tahap mengumpulkan data & informasi: Peneliti mengumpulkan data dari berbagai sumber, seperti buku, literatur, artikel, dan karya ilmiah yang relevan.
2. Analisis Kebutuhan Sistem: Mengkaji kebutuhan *hardware* dan *software*, serta membuat skema rangkaian sistem untuk memudahkan perancangan.
3. Perancangan Hardware dan *Software*: Merakit sistem dengan menyambungkan *Blynk* Android ke *NodeMCU ESP 32*, lalu melakukan pengkodean program.
4. Pengujian Alat: Menguji sistem secara menyeluruh untuk memastikan bahwa tidak ada kekurangan dan semua berfungsi sesuai harapan.
5. Implementasi Sistem: Memastikan sistem deteksi kebakaran dapat berfungsi dengan baik tanpa kendala.

C. Implementasi Sistem

Sistem kontrol berbasis IoT menggunakan *NodeMCU ESP 32* bertujuan untuk mengontrol motor DC pada penggulungan *fire hose roller*. Sistem ini dilengkapi dengan sensor *shaft rotary encoder GHS58* untuk memantau kecepatan rpm. Perangkat keras terdiri dari aplikasi Android *Blynk* (sebagai input), motor DC (sebagai output), *NodeMCU ESP 32* (pemroses dan penghubung internet), serta modul *BTS7960* (penghubung sensor analog dengan ESP 32). Perangkat lunak mencakup komunikasi antara ESP 32, modul, dan aplikasi *Blynk*.



GAMBAR 1.
DIAGRAM BLOK UMUM SISTEM FIRE HOSE
ROLLER OTOMATIS

Hasil pembacaan dari sensor-sensor yang dipasang pada sistem dapat dilihat secara *real time* pada aplikasi *blynk* yang dibuat secara sederhana dengan mengkomunikasikan antara *NodeMCU ESP 32* dengan jaringan internet dan data *base* yang dibuat oleh penulis.

D. Analisa Kebutuhan Sistem

Berdasarkan kebutuhan yang akan digunakan dalam pembuatan sistem *fire hose roller* otomatis ini mencakup perangkat keras (*hardware*) dan perangkat lunak (*software*) sehingga perancangan sistem ini dapat terlaksana.

1. Kebutuhan Perangkat keras

Kebutuhan perangkat keras dari alat sistem *fire hose roller* otomatis ini antara lain: obeng, solder listrik, gunting kabel, *NodeMCU ESP 32*, *arduino uno*, *power supply* 24 Volt 10 A, kabel jumper *male to male*, *female to male*, *female to female*, tespen, box PLC sedang, sensor *shaft encoder ghs58 series*, motor driver *BTS 7960*, *selector switch on-off*, motor DC dan adaptor AC-DC

2. Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak dari prototipe sistem deteksi kebakaran ini antara lain: *Arduino IDE*, *fritzing*, dan *microsoft office*.

3. Kebutuhan Input

Kebutuhan input dari prototipe sistem deteksi kebakaran ini antara lain: data *clock wise*, data *counter clock wise* dan data nilai rpm/kecepatan motor.

4. Kebutuhan Output

Kebutuhan output dari prototipe sistem deteksi kebakaran ini antara lain: motor DC berputar *clock wise*/kanan, motor DC berputar *counter clock wise*/kiri, kontrol *joke*, *low*, *medium* dan *fast*, serta kecepatan motor DC

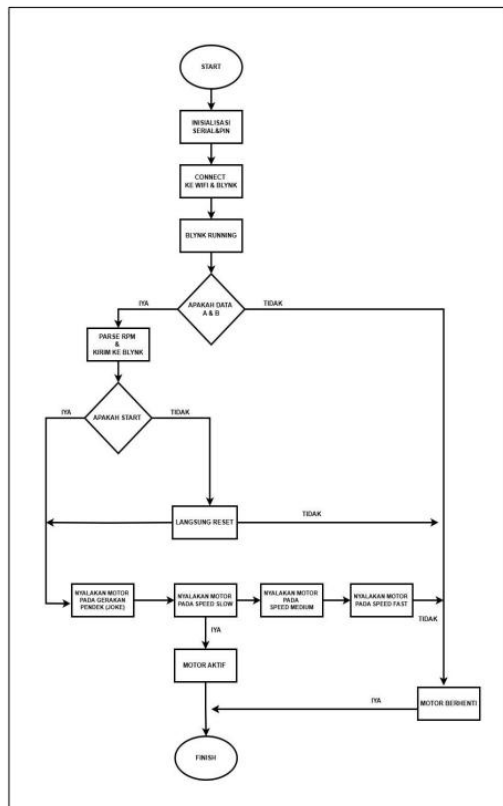
5. Perancangan Sistem

Prototipe Sistem *fire hose roller* ini akan di pasang pada *fire truck* pemadam kebakaran. Sistem ini di rancang agar dapat meringankan pekerjaan *fireman*. Hasil pembacaan yang merupakan implementasi dari IoT akan di tampilkan pada Aplikasi android *Blynk* yang dapat di akses melalui aplikasi android secara *real time*.

a) Perancangan Alur Kerja Sistem

Pada perancangan alur kerja sistem, terdapat Gambar *flowchart* sistem dimana input dari app *blynk*, input tombol cw (*clockwise*) dan input tombol ccw (*counter clockwise*) akan membaca input sesuai dengan input tombol. Pada saat mendeteksi adanya perintah input maka sistem akan mengeluarkan output berupa motor dc akan *running* yang terdapat pada alat *fire hose roller*. Pada saat motor dc *running*

ke arah *clock wise* atau ke arah *counter clock wise* ada juga tombol atau perintah untuk mengatur kecepatan *slow*, *medium*, *fast* berputar motor dc tersebut. Lalu pada alat *fire hose roller* ini ada penambahan sensor kecepatan pada motor yang dapat di liat *real time* pada saat motor *running*, maka dari pembacaan sensor *encoder* setelah itu di baca oleh *arduino* untuk hasil nilai kecepatan pada putaran motor dc tersebut. Habis di baca oleh *arduino* dan hasil nilai kecepatan tersebut di tampilkan oleh *NodeMCU ESP 32*. Selanjutnya untuk alur kerja sistem dapat di lihat pada Gambar 2 *flow chart* sistem berikut.



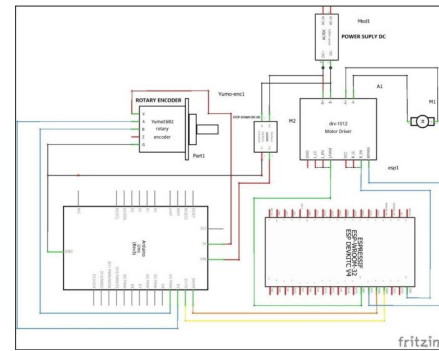
GAMBAR 2.

FLOW CHART ALGORITMA PEMROGRAMAN

b) Perancangan Perangkat Keras

Sistem kontrol ini berbasis *NodeMCU ESP 32* yang terhubung dengan aplikasi *Blynk* di *Android*. *NodeMCU* berperan sebagai mikrokontroler utama untuk memproses sinyal *input/output* dan koneksi internet. *Arduino Uno* digunakan sebagai pembaca sensor *encoder*, sementara hasil pembacaan ditampilkan melalui *NodeMCU*. Sistem mendapat daya dari *power supply* DC 24V yang diturunkan menggunakan modul *step down* untuk mendukung *NodeMCU* dan perangkat lainnya. Skema rangkaian perangkat keras terdapat dalam Gambar 3 Rangkaian

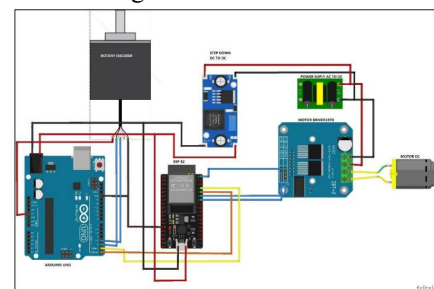
Skematik Keseluruhan berikut.



GAMBAR 3.

RANGKAIAN SKEMATIK KESELURUHAN

Skematik sistem *fire hose roller* berbasis IoT menggunakan *NodeMCU ESP32*, *Arduino Uno*, driver motor *BTS7960*, step down *LM2596*, PCB, switch button, resistor 330 ohm, saklar, LED, dan motor DC. Semua perangkat terhubung ke *ESP32* sebagai pusat kontrol *input-output*. Sistem mendapat daya dari *power supply* 24V yang diturunkan menjadi 7V melalui modul *step down* untuk *Arduino Uno*. Skema rangkaian perangkat keras terdapat dalam Gambar 4 Rancangan Skematik Keseluruhan.



GAMBAR 4.

RANCANGAN SISTEM KESELURUHAN

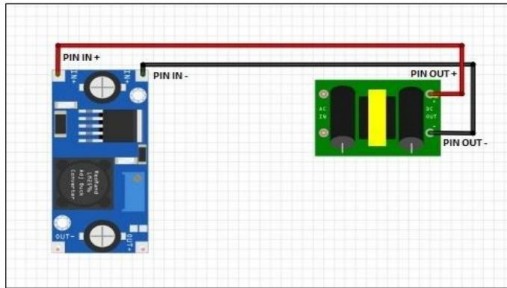
c) Konfigurasi Power Supply ke Modul Step Down DC to DC

Modul *step down* DC *LM2596* berfungsi menyesuaikan perbedaan tegangan antar modul dengan menurunkan tegangan DC ke level yang lebih rendah sesuai kebutuhan rangkaian elektronik atau mikrokontroler.

1. Input voltage : DC 3V - 40V
2. Output voltage: DC 1.5V - 35V (tegangan output harus lebih rendah dengan selisih minimal 1.5 V)
3. Arus max : 3A
4. Ukuran board : 42 mm x 20 mm x 14 mm

Modul regulator ini menggunakan *solid capacitor* dan PCB berkualitas untuk menjaga kestabilan tegangan. Tegangan dapat disesuaikan dengan memutar *potensio* pada board, misalnya dari input

aki 24 volt diturunkan menjadi 7 volt. Dapat di lihat pada Gambar 5 Konfigurasi *Modul Step Down* LM2595 DC TO DC.

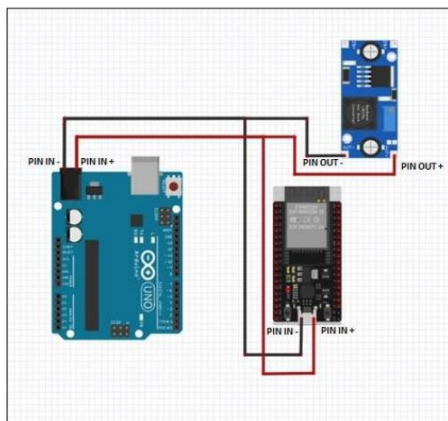


GAMBAR 5.

KONFIGURASI *MODUL STEP DOWN* LM2595 DC TO DC

d) Konfigurasi Mikrokontrol *Arduino Uno* dan Esp 32

Bagian kontrol terdiri dari *Arduino Uno* dan *NodeMCU ESP32*. *Arduino Uno* berfungsi membaca sensor *encoder* dan menghasilkan data RPM motor, sedangkan *NodeMCU ESP32* berfungsi mengontrol motor DC, menerima data RPM dari *Arduino*, dan terhubung ke aplikasi *Blynk*. Dapat dilihat seperti Gambar 6.

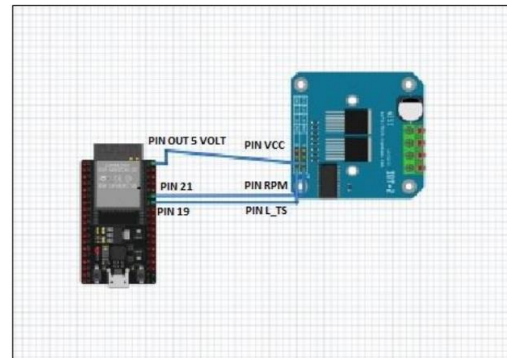


GAMBAR 6.

KONFIGURASI *ARDUINO UNO* DAN *NODEMCU ESP 32*

e) Konfigurasi *Modul Motor Driver* BTS7960

Driver motor BTS7960 adalah modul pengendali motor DC yang umum digunakan untuk mengatur kecepatan dan arah putaran motor. Modul ini menggunakan IC H-bridge dan transistor logik TTL dengan gerbang NAND untuk mempermudah pengendalian arah motor. Modul ini praktis karena pin I/O sudah tersusun rapi, serta unggul dalam presisi kontrol motor. Adapun gambar *pin out* beserta keterangannya dapat diperhatikan pada Gambar 7 di bawah ini.

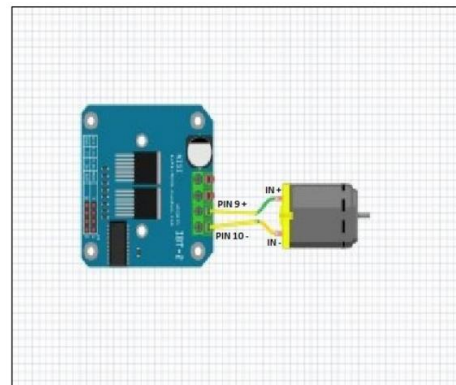


GAMBAR 7.

KONFIGURASI *MOTOR DRIVER* BTS 7960

f) Konfigurasi *Modul Motor Driver* BTS7960

Rangkaian motor DC dikontrol oleh *modul driver* yang berfungsi mengatur kecepatan dan arah putaran motor, serta mampu menangani beban saat motor aktif. Dapat dilihat pada Gambar 8 Konfigurasi *Motor Dc*.



GAMBAR 8.

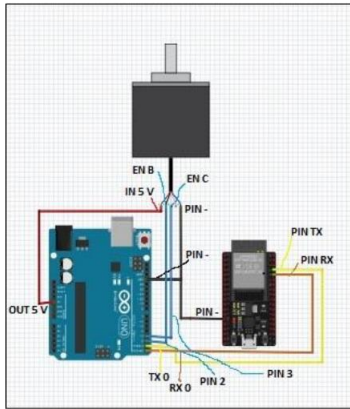
KONFIGURASI *MOTOR DC*

g) Konfigurasi *Modul Motor Driver* BTS7960

Sensor *encoder* digunakan untuk mengubah gerakan mekanik menjadi sinyal digital guna mengukur kecepatan poros motor DC. Nilai kecepatan dibaca oleh *Arduino Uno*, ditampilkan oleh *NodeMCU ESP32*, dan hasilnya muncul di aplikasi Android *Blynk*. Seperti Gambar 9 Konfigurasi *Sensor Rotary Encoder*.

h) Perancangan Sistem Pada *Arduino IDE*

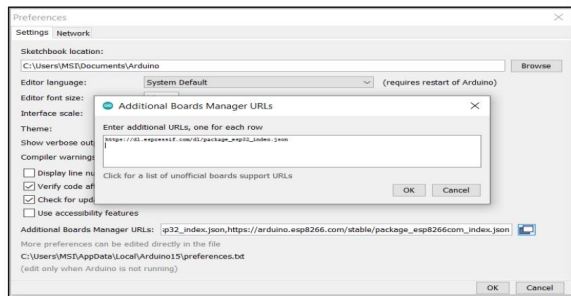
Pemrograman prototipe sistem ini menggunakan *Arduino IDE* dengan bahasa C, dan mikrokontroler ESP32 untuk memproses data *input-output* serta menghubungkan sistem dengan internet.



GAMBAR 9.
KONFIGURASI SENSOR *ROTARY ENCODER*

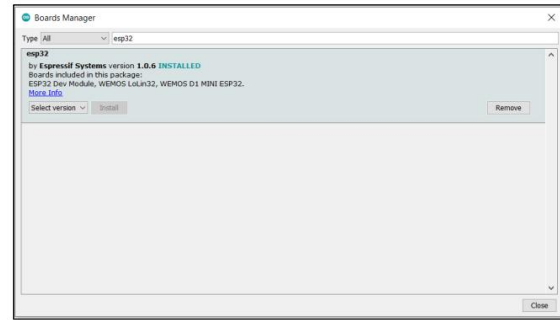
Pembuatan perangkat lunak menggunakan *Arduino IDE* pada laptop untuk memprogram *mikrokontroler Arduino Uno*. Program ini mengakses data dari sensor dan mengolahnya sesuai kebutuhan. Untuk menghubungkan *mikrokontroler* dengan komponen, diperlukan penambahan *library* yang sesuai di *Arduino IDE*, karena *library* berfungsi untuk mengeksekusi perintah tertentu. Beberapa *library* yang dibutuhkan tidak tersedia secara *default*, sehingga harus ditambahkan secara manual.

Untuk menggunakan ESP 32 dengan *Arduino IDE*, perlu menginstal *driver* terlebih dahulu. Caranya adalah dengan membuka "*preferences*" pada menu "*file*" dan memasukkan alamat web "https://dl.ESPRESSIF.COM/dl/package_ESP32_index.json". Seperti yang terlihat di Gambar 10 berikut.



GAMBAR 10.
PREFERENCE PADA *ARDUINO IDE*

Selanjutnya dapat masuk ke "*board manager*" menu "*tool*" dan kemudian mencari "*ESP32 by Espressif System*" untuk dapat di *install* pada *Arduino IDE* sehingga papan pengembangan dapat digunakan seperti yang terlihat di Gambar 11 berikut.



GAMBAR 11.
BOARD MANAGER ESP 32 TERINSTAL RANCANGAN
PENGUJIAN SISTEM

i) Perancangan Sistem Pada *Arduino IDE*

Pengujian sistem *fire hose roller* berbasis IoT menggunakan ESP 32 membuktikan bahwa *modul motor driver BTS 7960*, aplikasi *Blynk*, dan *motor DC* berfungsi sesuai desain. Aplikasi *Blynk* IoT mampu menampilkan data secara *real-time*, dan pengoperasian kontrol dilakukan dengan menekan tombol yang dirancang di aplikasi *Android Blynk*. Hal ini dirancang untuk melihat jika pengontrolan sistem pada motor dc dan bekerja seperti yang diharapkan, hasil pengujian nantinya akan dimasukkan ke dalam Tabel 1 sebagai bukti hasil pengujian sistem dengan parameter keberhasilan sebagai berikut.

TABEL 1.
SKENARIO PENGUJIAN

No.	Perangkat	Pengujian	Hasil yang Diharapkan
1.	Tegangan sumber	Memastikan nilai tegangan sumber dari listrik AC sebelum masuk ke ESP 32 dan tegangan <i>output</i> setelah keluar dari pin 5 volt yang terdapat di ESP 32	- Tegangan sumber 220 volt pada listrik PLN - Tegangan sumber <i>input</i> 5 volt pada ESP 32
2.	ESP 32	Menghubungkan ESP 32 dengan aplikasi android <i>Blynk</i> IoT(<i>internet of things</i>).	- ESP berhasil terhubung ke aplikasi <i>blynk</i> IoT(<i>internet of things</i>). - Waktu terhubung berdasarkan jarak antara

No.	Perangkat	Pengujian	Hasil yang Diharapkan
			modul dan sumber sinyal <i>Hotspot</i> dapat diketahui. - Aplikasi pengontrol berjalan dengan baik.
3.	Motor DC	Responsif memutar ke arah CW(<i>clock wise</i>) dan CCW(<i>countner clock wise</i>) pada motor DC.	- Motor berjalan sesuai perintah pada aplikasi <i>blynk</i> IoT.

j) Perancangan Sistem *NodeMCU* ESP 32 Ke *Blynk*

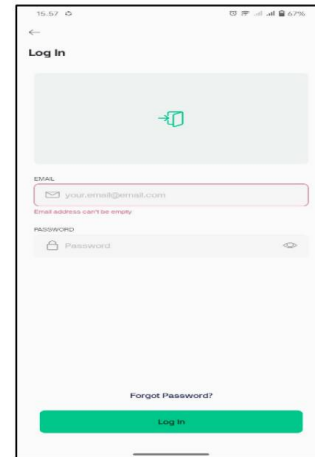
Pada konfigurasi *NodeMCU* ke *Blynk* syarat yang harus dipenuhi adalah sudah melakukan konfigurasi *NodeMCU* ke *Arduino* IDE.



GAMBAR 12.

TAMPILAN AWAL APLIKASI BLYNK

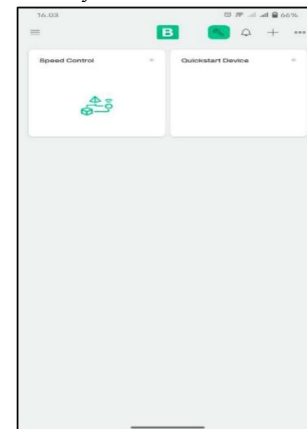
Buka aplikasi *Blynk* di *hand phone* lalu pencet ikon “*Blynk* IoT” maka tampilan yang terlihat seperti Gambar 13



GAMBAR 13.

TAMPILAN LOGIN AKUN BLYNK

Masuk ke akun *Blynk* atau buat akun baru dengan memasukkan alamat email, kemudian ikuti langkah selanjutnya untuk membuat kata sandi dan profil *Blynk*. Seperti Gambar 14 Tampilan Masukkan Akun Atau Buat Akun *Blynk*.



GAMBAR 14.

TAMPILAN AKUN BLYNK

Dalam tampilan *akun Blynk* klik proyek yang sudah di *setting* “*Speed Control*”.

IV. HASIL DAN PEMBAHASAN**A. Hasil Penelitian**

Hasil penelitian ini dari rancang bangun alat pengontrol *fire hose roller* pada alat untuk mempermudah penggulungan *fire hose*. pada perancangan alat rancang bangun motor DC yang di kontrol menggunakan *smartphone* dengan aplikasi *Blynk* android ini terdiri dari berbagai rangkaian yang kemudian digabungkan menjadi satu rangkaian alat secara keseluruhan.

1. Pengujian Perangkat Keras

Pengujian komponen alat pengontrolan motor DC ini bertujuan untuk mengetahui apakah alat ini berfungsi dengan baik atau tidak, maka dari itu perlu dilakukan pengujian agar alat ini bisa beroperasi sebagaimana mestinya, dapat kita lihat pada Gambar 15.



GAMBAR 15.
PENGUJIAN ALAT KONTROL

Berikut ini merupakan Tabel 2 Hasil Pengukuran Komponen pengontrolan motor DC hasil pengukuran keluaran daya pada rancang bangun alat *fire hose roller* berbasis IoT.

TABEL 2.
HASIL PENGUKURAN KOMPONEN FIRE HOSE
ROLLER

No.	Jenis Alat	Tegangan (Volt)	Arus (A)	Daya (W)
1.	Power Supply DC	24	0,71	17,04
2.	Step Down DC-DC	5,1	0,3	1,53
3.	NodeMCU Esp 32	3,3	0	0
4.	Arduino uno	4,9	0	0
5.	Motor Driver BTS7960	24	0,56	13,44
6.	Motor DC	22,1	0,57	12,59

Data keluaran pada alat pengontrolan motor DC pada alat *fire hose roller* berbasis IoT diambil tujuannya untuk mengetahui tegangan, arus dan daya pada saat alat ini dioperasikan.

TABEL 3.
PENGUJIAN KONTROL COUNTER CLOCK WISE

No.	Jenis Kontrol	Rpm Pada Monitor Blynk	Rpm pada Tachometer	Status

1.	Jog	11 Rpm	11,3 Rpm	Berfungsi
2.	Low	11 Rpm	11,2 Rpm	Berfungsi
3.	Medium	24 Rpm	24,4 Rpm	Berfungsi
4.	Fast	29 Rpm	29,2 Rpm	Berfungsi

TABEL 4.
PENGUJIAN KONTROL COUNTER CLOCK WISE

No.	Jenis Kontrol	Rpm Pada Monitor Blynk	Rpm pada Tachometer	Status
1.	Jog	16 Rpm	16 Rpm	Berfungsi
2.	Low	16 Rpm	16,3 Rpm	Berfungsi
3.	Medium	25 Rpm	25,2 Rpm	Berfungsi
3.	Medium	29 Rpm	29,4 Rpm	Berfungsi

Dari hasil pengujian di atas dapat diperoleh bahwa tiap-tiap tombol yang ada di alat *fire hose roller* berbasis IoT berfungsi sesuai dengan apa yang telah di *input* di dalam program *Arduino Uno* dapat kita lihat pada Tabel 3 dan Tabel 4.

2. Pengujian Box Kontrol Fire Hose Roller Berbasis Smartphone

Pada saat alat kontrol *fire hose roller* ini belum di hubungkan ke sumber daya dan belum terhubung ke aplikasi *blynk*, motor DC belum *runing* atau mode ON

suara motor belum terdengar.



GAMBAR 16.
TAMPILAN POWER SUPPLY SUDAH DI
HUBUNGKAN KE SUMBER LISTRIK

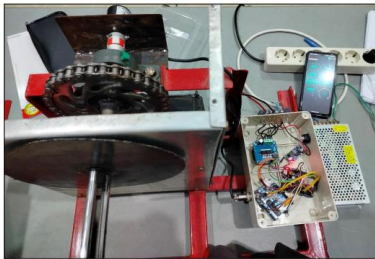


GAMBAR 17.

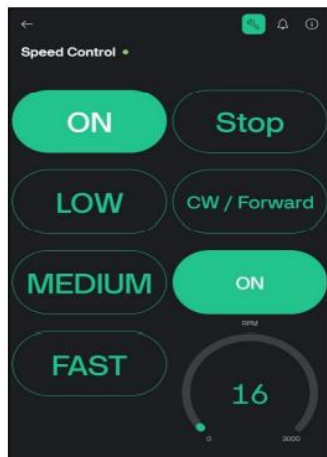
APLIKASI BLYNK BELUM TERKONEKSI KE ESP 32

3. Menghubungkan *Mikrokontroller* ke Aplikasi *Blynk*

Untuk menghubungkan dari aplikasi *Blynk* ke komponen yaitu harus menyalakan *Hotspot* di *hanphone* dan sudah terprogram nama *hotspot* yang akan di hubungkan ke ESP 32 tersebut.



GAMBAR 18.

KOMPONEN *FIRE HOSE ROLLER* TERHUBUNG KE *BLYNK*

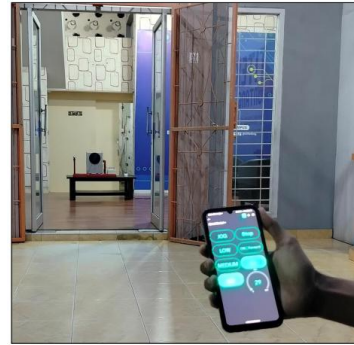
GAMBAR 19.

APLIKASI BLYNK SUDAH TERHUBUNG

Pada Gambar 4.10 Komponen *Fire Hose Roller* terhubung ke *blynk* dan aplikasi *blynk* sudah terhubung merupakan gambar indikator pada aplikasi *blynk* yang menandakan bahwa aplikasi *blynk* dengan komponen *fire hose roller* sudah berhasil terhubung.

4. Pengujian Koneksi Jarak Kontrol

Pengujian koneksi yang dilakukan adalah koneksi ponsel pintar aplikasi *blynk* dengan *Hotspot* pada *smarthphone*. Aplikasi *blynk* ini harus mendapatkan dukungan jaringan internet yang baik. Pengujian dilakukan tanpa sekat (benda) di area pengujian dapat dilihat pada Gambar 20.



GAMBAR 20.

PENGUJIAN JARAK MOTOR DC

TABEL 5.
HASIL UJI COBA

No.	Pengujian Jarak (Meter)	Clock Wise (CW)	Status Sistem	Counter Clock Wise (CCW)	Status Sistem
1.	0	✓	Terhubung	✓	Terhubung
2.	1	✓	Terhubung	✓	Terhubung
3.	2	✓	Terhubung	✓	Terhubung
4.	3	✓	Terhubung	✓	Terhubung
5.	4	✓	Terhubung	✓	Terhubung
6.	5	✓	Terhubung	✓	Terhubung
7.	6	✓	Terhubung	✓	Terhubung
8.	7	✓	Terhubung	✓	Terhubung
9.	8	✓	Terhubung	✓	Terhubung
10.	9	✓	Terhubung	✓	Terhubung
11.	10	✓	Terhubung	✓	Terhubung

Berdasarkan hasil uji coba di atas dapat disimpulkan bahwa koneksi *wifi* atau *Hotspot* pewartu digital dan ponsel pintar dengan sekat (benda) di area pengujian terhubung.

B. Pembahasan



Pembahasan pada penyusunan penelitian tugas akhir ini menjelaskan tentang rancang bangun sistem *fire hose roller* berbasis IoT (*internet of things*) dengan menggunakan *mikrokontrol NodeMCU ESP 32* dengan di dukung *Arduino Uno* dimana mengambil studi kasus petugas *fire man* departemen K3 di PT. Pupuk Kalimantan Timur.

Rancang bangun *fire hose roller* ini menggunakan IoT dengan *NodeMCU ESP32* untuk kontrol tombol via aplikasi *Blynk* dan *Arduino Uno* untuk membaca nilai RPM dari *encoder*. Aplikasi *Blynk* menyediakan tombol kontrol seperti *OFF*, *forward (clockwise)*, *reverse (counter-clockwise)*, *joke*, *low*, *medium*, dan *fast*.

Pada pengujian alat *fire hose roller* dengan pengambilan data yang telah di lakukan menunjukan hasil pengujian semua komponen berfungsi dengan semestinya. Hasil analisis dan rata-rata pada tombol yang telah di rancang menunjukkan hasil yang berbeda-beda karena di setiap tombol memiliki fungsi yang berbeda-beda, seperti:

Pengujian arah *clock wise* tombol *joke* pada saat alat dijalankan menunjukkan hasil kecepatan di monitor *blynk* 11 Rpm dan di alat *tachometer* menunjukkan hasil 11,6 Rpm. Tombol *low* menunjukan hasil kecepatan di monitor *Blynk* 11 dan di alat *tachometer* menunjukan hasil 11,3. Tombol *medium* menunjukkan hasil kecepatan di monitor *Blynk* 24 Rpm dan di alat *tachometer* menunjukan hasil 23,7 Rpm. Tombol *fast* menunjukkan hasil kecepatan di monitor *Blynk* 29 Rpm dan di alat *tachometer* menunjukan hasil 29,3 Rpm.

Pengujian arah *counter clock wise* tombol *joke* pada saat alat dijalankan menunjukkan hasil kecepatan di monitor *blynk* 16 Rpm dan di alat *tachometer* menunjukkan hasil 16,7 Rpm. Tombol *low* menunjukkan hasil kecepatan di monitor *Blynk* 16 dan di alat *tachometer* menunjukan hasil 16,5. Tombol *medium* menunjukkan hasil kecepatan di monitor *Blynk* 25 Rpm dan di alat *tachometer* menunjukan hasil 25,6 Rpm. Tombol *fast* menunjukkan hasil kecepatan di monitor *Blynk* 29 Rpm dan di alat *tachometer* menunjukan hasil 28,9 Rpm.

Pada pengujian sistem kontrol *fire hose roller* otomatis yang di kontrol di *handphone* dengan menggunakan aplikasi *Blynk* IoT dapat di analisa

bahwa jarak dari 0-10 meter alat *fire hose roller* ini bekerja dengan semestinya tanpa ada patah fungsi dari alat *fire hose roller* ini.

Pada pengujian jarak 0 sampai 10 meter *wifi* atau *Hotspot* masih terkoneksi sehingga alat pewaktu masih bekerja dengan baik dan membutuhkan < 5 detik untuk terhubung pada jarak 10 meter. ketika pada jarak 11> meter koneksi *wifi* atau *Hotspot* terkoneksi tetapi untuk respon motor DC agak melambat dan terputus-putus sehingga alat *fire hose roller* otomatis tidak terkoneksi dan alat *fire hose roller* tidak berfungsi. Hasil uji coba koneksi *Hotspot* atau *Wifi*.

Pada implementasi alat *fire hose roller* ini bekerja dengan baik, sesuai dengan pengaturan sistem yang telah dirancang dan disesuaikan dengan kondisi di lapangan pada saat uji coba alat. Terdapat indikator kecepatan (Rpm) yang telah dimonitoring. Pada sistem kontrol *fire hose roller* secara *real-time* berfungsi dengan baik pada saat alat di implementasikan.

V. KESIMPULAN

Berdasarkan hasil dari rancang bangun alat pengontrolan motor DC pada alat *fire hose roller* berbasis *internet of things* (IoT) dengan menggunakan modul *NodeMCU Esp 32*. mulai dari segi teori ataupun pengujian yang telah dilakukan oleh penulis, maka penulis telah menarik beberapa kesimpulan dari hasil perancangan ini, antara lain yaitu:

1. Alat penggulung *fire hose roller* berbasis IoT mampu meningkatkan efisiensi kerja dalam proses penggulangan (*fire hose*) selang dengan mengurangi keterlibatan proses penggulangan manual.
2. Sistem kontrol *fire hose* ini mampu meningkatkan kemudahan dan keamanan dalam memonitoring kondisi alat secara *real time* dan mengontrol kecepatan yang tepat, sehingga meminimalisir risiko kerusakan (*fire hose*) selang.

VI. SARAN

Untuk sistem lebih baik kedepannya perlu Ditambahkan beberapa hal yaitu tidak hanya dapat memonitoring ataupun dikontrol melalui aplikasi *Blynk* Namun dikembangkan dengan alat yang lebih efisien sehingga penggunaan alat lebih baik. Adapun beberapa saran yang akan di sampaikan oleh penulis yaitu :



1. Sebaiknya ditambahkan pengembangan alat yang lebih efisien, tidak hanya sebatas *memonitoring* dan kontrol melalui aplikasi *Blynk*.
2. Menambahkan alur untuk *hose* pada mekanisme alat *fire hose roller* sehingga proses penggulangan *fire hose roller* lebih sempurna
3. Ditambahkan sensor untuk mendeteksi ujung *hose* mendekati ujung alat, agar alat berhenti otomatis

VII. DAFTAR PUSTAKA

- [1] A. T. Salu, "Pembuatan fire hose roller untuk mempercepat proses penggulangan fire hose di bagian fire & rescue Departemen k3," pp. 1-26, oktober 2021.
- [2] Najib, "Pengenaln pemadam kebakaran dinas oemadam kebakaran dan penyelamatan (DPKP)", Kota Banda Aceh, 2013.
- [3] A. N, "Internet of things dan komputasi Edge", Jakarta : Tampuniak Mustika Edukarya, 2020.
- [4] Megawati, "Pengembang Sistem Teknologi Internet of Things," Journal Information Enginerimh and Educational Technology, vol. 5, no. 1, pp. 19- 26, 2021.
- [5] A. Marina, "Aplikasi Smarth Home Menggunakan NodeMCU Esp 32," Jurnal Rekayasa Teknologi Nusa Putra, vol. 1, pp. 1-7, 2020.
- [6] A. D. Bayu, "Perancangan Pengendali Smart Home Menggunakan Konektifitas Bluetooth," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, vol. 1, no. 2, p. 5, 2017.. 1, pp. 16-24, 2023.
- [7] Ihsan, "Pengertian Arduino", 2015
- [8] A. M. Mona, "Rancang Bangun Sistem Monitoring Ketinggian Muka Air Laut Menggunakan Arduino Pro mini dan NodeMCU Esp32," Jurnal Teknologi Riset Terapan (Jatra), vol. 1, no. 1, pp. 25-35, 2023.
- [9] Kryvonos, "M.P Fritzing Is A Program For Creating Visual Electronic Circuits," International Journal Of Automation and Control Engineering, vol. 1, no. 3, pp. 40-44, 2015.
- [10] Espressif, "ESP32-DevKit V4 Getting Started Guide," pp. 25-28, 30 April 2011.
- [11] R. Birdayansah, "Pengendalian Kecepatan Motor DC MEnggunakan Perintah Suara BERbasis Mikrokontroler Arduino," Electrician-Jurnal Rekayasa dan Teknologi Elektro, vol. 9, no. 2, pp. 97-107, 2015.
- [12] Maheriya, "Modelling Of Brushed DC motor and Various type of control methods," journal for research , vol. 01, no. 12, pp. 2395-7549, 2016.
- [13] M. George, "Speed Control of Separately Excited DC Motor," American journal of applied, vol. 5, no. 3, pp. 227-233, 2008.
- [14] K. Dani, "Perencanaan Dan Pembuatan Alatukur Jarak Menggunakan Rotary Encoder dengan Metode Elementri," abstrak UMM Malang, vol. Nomor 1, no. 2, pp. 22-32, 2009.
- [15] S. Gautam, "AC/DC/AC converter based on parrarel AC/AD and cascaded multilevel DC/AC converter," students conference on engineering and system, vol. 1, no. 4, pp. 1-6, 2012.
- [16] M. A. ulum, "Perancangan Sistem monitoring Kecepatan Putar Motor DC Berbasis Internet Of Things Menggunakan Aplikasi Bylink," Jurnal Teknik Elektro, vol. 09, no. 1, pp. 855-862, 2020.