

**RANCANG BANGUN SISTEM PENGENDALIAN MOTOR DC
BERBASIS GRAPHICAL USER INTERFACE (GUI)
MENGGUNAKAN PROCESSING**

Prima Alfiatin Jannah

Jurusan Teknik Elektro Fakultas Teknik Universitas Muhammadiyah Yogyakarta
Jalan Lingkar Selatan Kecamatan Kasihan Bantul Daerah Istimewa Yogyakarta
55183 Indonesia
Email: primaalfiatinj@gmail.com

ABSTRACT

In the era of technological advances that are so rapid, it requires humans to think critically in creating new ideas to support the ease of doing things. No exception with the existence of new innovations related to the Human-Robot Interface. This Human-Robot Interaction allows humans to interact with computer devices. In this design built a DC Motor controller system based on Graphical User Interface (GUI), this system was built using the Processing programming language. The Graphical User Interface (GUI) system can make it easier for users to digitally operate on a DC Motor. This system is operated using hardware in the form of personal computers. The object that is controlled on this GUI system is a DC motor circuit. This GUI system is used to control the speed and direction of the rotation of the DC Motor. In the design of the GUI display there are several buttons that are used to control the DC Motor, namely in the form of sliders and knobs as speed control, CW (clockwise) and CCW (current clockwise) as the control direction of the rotation, and the STOP button to stop the rotation on the DC Motor. In designing the GUI program, GUI Version 3 has the best response from respondents because the GUI has the best color combination in appearance, and is easy to operate.

Keywords: Human-Robot Interface, Graphical User Interface (GUI), Processing, DC Motor.

BAB I PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi dan pengetahuan sekarang ini sudah sangat jauh berbeda dengan dahulu. Banyak sekali hal-hal baru yang ditemukan dan dikembangkan. Pada era serba digital sekarang ini, manusia mengharapkan adanya inovasi baru dalam segala hal, tidak terkecuali dibidang teknologi. Inovasi pada bidang teknologi ini bertujuan agar memudahkan setiap individu dalam melakukan suatu hal dengan kemas yang praktis. Pada bidang teknologi, manusia menciptakan beberapa sistem, dimana sistem itu nantinya akan menggantikan pengaturan manual atau analog menjadi digital.

Pengaturan analog menjadi digital dapat dilakukan dengan cara merancang sebuah program, dimana program tersebut memungkinkan adanya pertukaran informasi antara pengguna dengan berbagai jenis mesin, komputer dan perangkat alat elektronik lainnya. Interaksi dengan perangkat modern ini sering kali melalui media visual, audio dan taktil antar muka pengguna. Interaksi antara pengguna (manusia) dengan komputer kali ini dimodelkan dalam bentuk *Graphical User Interface* (GUI) atau tampilan pada komputer untuk melakukan pengendalian Motor DC.

Maka selanjutnya perlu dirancang sebuah sistem pengendali Motor DC berbasis *Graphical User Interface* (GUI) menggunakan *Processing*. Dengan menggunakan

bahasa pemrograman *Processing* pengguna yang tidak berlatar belakang ilmu komputer dapat dengan mudah merancang sebuah program GUI, dikarenakan *Processing* hanya menggunakan sedikit kode. Selain itu, dengan menggunakan sistem yang berbasis GUI ini, pengguna tidak perlu lagi merubah hardware saat melakukan pengembangan, pengguna hanya perlu melakukan upgrade pada *software* GUI.

1.2 Rumusan Masalah

Rumusan masalah yang akan dibahas tugas akhir ini adalah sebagai berikut :

- 1) Bagaimana perancang sistem pengendalian motor DC berbasis *Graphical User Interface* (GUI) menggunakan *Processing* ?
- 2) Bagaimana implementasi *Human-Robot Interface* interaksi dengan model GUI?

1.3 Batasan Masalah

Batasan masalah pada tugas akhir ini adalah sebagai berikut :

- 1) Perancangan ini berbasis *Graphical User Interface* (GUI) menggunakan *Processing*.
- 2) Menggunakan Arduino Uno sebagai pengontrol utama.
- 3) Menggunakan sensor L298 sebagai driver motor.
- 4) Menggunakan Motor DC sebagai output dari program.

BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Beberapa penelitian yang berhubungan dengan Interaksi Manusia-Robot

(*Human – Robot Interaction*) dan *Graphical User Interface* (GUI) diantaranya:

Interaksi manusia-robot memungkinkan pertukaran informasi antara pengguna manusia dengan berbagai jenis mesin, komputer dan perangkat elektronik lainnya. Interaksi dengan perangkat modern ini paling sering dilakukan melalui visual, auditori, dan taktil antar muka pengguna (Jaka Sodnik dan Saso Tomazic, 2015)

Pengujian *software Graphical User Interface* (GUI) merupakan salah satu tantangan utama dalam siklus hidup dari setiap *software*. Pengujian GUI secara inheren lebih sulit daripada pengujian antarmuka tradisional (*remote control*) dan *command line*. Beberapa faktor yang membuat pengujian GUI berbeda dengan pengujian *software* tradisional yaitu: jumlah besar objek, tampilan dan nuansa objek yang berbeda, disetiap objek banyak parameter terkait, pengungkapan progresif, input yang kompleks dari berbagai sumber, dan *output* grafis. Teknik pengujian yang sudah ada perlu adanya penyesuaian dan peningkatan untuk GUI, dan Teknik pengujian yang baru diharapkan dapat menciptakan sesuatu yang lebih mudah dan efektif (Zafar, Singhera, Ellis Horowitz, dan Abad Shah, 2009).

Graphical User Interface (GUI) adalah bagian penting dari setiap aplikasi perangkat lunak bagi pengguna pada saat ini dan GUI juga harus memenuhi kriteria berikut: *significant design*, *development*, dan *testing*

activities (Zafar Singhera, Ellis Horowitz dan Abad Shah, 2009).

Graphical User Interface (GUI) menyajikan sesuatu hal yang lebih mudah dalam penggunaan fungsi aplikasi, yaitu dengan cara mengorganisir kedalam bentuk pilihan dan hanya menyajikan sesuatu hal yang mudah diterima akal. Dalam implementasinya, GUI memiliki beberapa sifat yaitu: menggunakan modus *asynchronous* dalam pengoperasiannya, menggunakan *input* dan *output* tradisional, mempunyai struktur yang mudah untuk berinteraksi dengan pengguna sehingga menghasilkan pengujian yang berbeda secara signifikan dan lebih sulit dari pengujian *software* tradisional (Zafar Singhera, Ellis Horowitz, dan Abad Shah, 2009).

Mustar, Muhammad, Dkk (2014) yang berjudul “Perancangan Model Interaksi Manusia dan Robot Dalam Bentuk Tampilan Visual Pada Komputer”, dalam penelitian tersebut dijelaskan model interaksi manusia dengan robot dengan metode tampilan visual pada komputer, robot yang dikendalikan yaitu robot lengan (*manipulator*) dengan 6 derajat kebebasan (*degree of freedom*). Setiap gerakan oleh robot menggunakan motor servo sebagai actuator. Tampilan visual pada perangkat komputer menggunakan *Graphical User Interface* (GUI) yang dibuat dengan menggunakan *software* LabVIEW sebagai pemrosesan data

sekaligus sebagai pengontrolan gerak robot.

2.2 Landasan Teori

2.2.1 *Graphical User Interface* (GUI)

Metode antar muka antara manusia dengan computer (*user interface*) merupakan metode yang memungkinkan pengguna dapat berinteraksi dengan sebuah perangkat. Antar muka (*interface*) merupakan tempat dimana pengguna dapat berinteraksi dengan komputer.

Metode antar muka dapat digolongkan menjadi dua metode, yaitu:

1. *Command Line Interface* (CLI), yaitu metode antar muka yang memungkinkan pengguna berinteraksi dengan sistem melalui text-terminal. Pengguna memberikan perintah dengan cara mengetikkan baris-baris tertentu.
2. *Graphical User Interface* (GUI), yaitu sebuah sistem yang dirancang agar pengguna (*user*) dapat berinteraksi dengan sebuah perangkat komputer yang digunakan oleh pengguna tersebut. *Graphical User Interface* (GUI) dapat dikontrol dengan berbagai macam inputan seperti *mouse (pointer)*, *touchscreen*, *keyboard*, dan lainnya. Pada sistem operasi *Graphical User Interface* (GUI) terdapat menu tombol, jendela, dan berbagai icon yang dirancang agar pengguna dapat mengoperasikan sistem tersebut dengan mudah. Pengguna hanya cukup memilih ikon, tombol, gambar, atau grafik yang sudah ada

pada sistem tersebut dikarenakan *Graphical User Interface* (GUI) telah dirancang untuk merubah perintah menjadi ikon dalam layer computer yang dapat dipilih untuk menjalankan perintah.

Pada awalnya, saat lahirnya komputer, komputer dapat dioperasikan dengan memberikan perintah-perintah yang diinputkan menggunakan *keyboard* ke layar monitor. Metode dengan cara mengetikkan perintah-perintah teks tertentu pada komputer ini disebut *Command Line Interface* (CLI). Metode *Command Line Interface* ini memiliki banyak kekurangan dikarenakan metode ini dirasa kurang efisien dan kurang canggih. Seiring perkembangan teknologi dan pengetahuan *interface*, maka terciptalah media interaksi yang mudah, praktis, efektif, dan dapat digunakan setiap orang, yaitu *Graphical User Interface* (GUI). *Graphical User Interface* (GUI) memudahkan pengguna dalam mengoperasikan perangkat tanpa harus mengingat setiap perintah, pengguna hanya perlu memilih icon, gambar, grafik, atau tombol yang sudah disediakan pada sistem. *Graphical User Interface* (GUI) menggunakan sifat alamiah manusia yang lebih mudah mengingat gambar daripada tulisan.

2.2.2 Sistem Kendali Robot

Saat ini dengan adanya kemajuan teknologi dan ilmu pengetahuan dimana kegiatan dalam kehidupan sehari-hari banyak terbantu dengan adanya sistem kontrol. Sistem kendali dapat mengontrol berbagai peralatan dimana

dapat meringankan pekerjaan manusia, dari hal yang sangat sederhana hingga yang sangat kompleks.

Sistem kendali dapat dikatakan sebagai hubungan antara komponen yang membentuk sebuah konfigurasi sistem, yang akan menghasilkan tanggapan sistem yang diharapkan. Sehingga harus ada yang dikendalikan, yang merupakan suatu sistem fisis, yang biasa disebut dengan kendalian (*plant*) (Rahmat Fauzi Siregar, 2017).

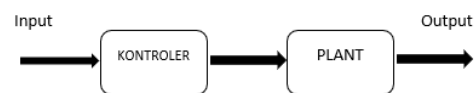
Pada dasarnya sistem kendali terdiri atas beberapa komponen dasar, yaitu berupa masukan, proses pengendalian, dan keluaran. Masukan dan keluaran merupakan *variable* atau besaran *fisis*. Keluaran merupakan hal yang dihasilkan oleh kendalian, artinya yang dikendalikan. Masukan adalah yang mempengaruhi kendalian, yang mengatur keluaran. Kedua dimensi masukan dan keluaran tidak harus sama (Rinda Hedwig, 2004).

Rinda Hedwig (2004) mengatakan bahwa secara umum, tujuan sistem kendali adalah mengendalikan keluaran dari berbagai masukan tertentu melalui unsur-unsur sistem kendali. Secara garis besar, sistem kendali dapat dibagi menjadi dua kategori yaitu sistem kendali terbuka (*open loop system*) dan sistem kendali tertutup (*close loop system*).

Sistem kendali dibagi menjadi dua, yaitu:

1. Sistem kendali terbuka (*Open Loop System*), sistem kendali terbuka yaitu sistem kendali yang tidak memperhatikan keluarannya

(*output*). Sistem kendali terbuka disebut juga sebagai sistem bukan umpan balik, dikarenakan output tidak mempengaruhi atau efek tindakan *control* sinyal *input*. Sehingga sistem kendali terbuka dapat mengikuti perintah *input* atau *set point* tanpa memperhatikan hasil akhirnya.



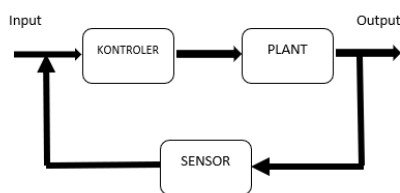
Gambar 2.1 Diagram sistem kendali terbuka (*Open Loop System*)

a. Karakteristik sistem kendali terbuka (*Open Loop System*)

- 1) Tidak terdapat perbandingan nilai actual dengan nilai yang diinginkan.
- 2) Sistem kendali terbuka tidak memiliki pengaturan tersendiri atau aksi kendali terhadap nilai output.
- 3) Pada pengaturan input menentukan posisi operasi agar tetap sebagai pengendali. Perubahan atau gangguan pada kondisi eksternal tidak akan mengakibatkan perubahan output secara langsung.

2. Sistem kendali tertutup (*Close Loop System*), sistem kendali tertutup yaitu sistem kendali yang sinyal keluarannya mempunyai tidak diperhatikan sehingga tidak ada pengaruh langsung pada aksi pengendalian, sistem kendali tertutup juga merupakan sistem kendali umpan balik. Sinyal kesalahan penggerak, merupakan selisih antara sinyal *input* dan sinyal umpan balik

yang berupa sinyal *output* atau suatu fungsi sinyal *output*, diumpangkan ke kontroler untuk memperkecil kesalahan dan menghasilkan *output* sistem mendekati nilai yang diinginkan. *Close Loop System* dapat diartikan menggunakan aksi umpan balik untuk memperkecil kesalahan sistem.



Gambar 2.2 Diagram Sistem kendali tertutup (*Close Loop System*)

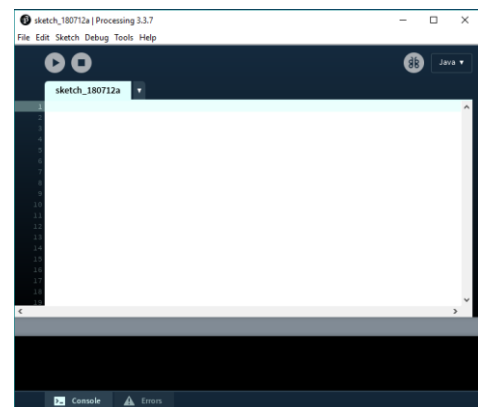
2.2.3 Processing

Processing adalah nama Bahasa pemrograman yang ditujukan untuk komputer. Bahasa pemrograman ini dirancang oleh Casey Reas dan Ben Fry di Massachusetts Institute of Technology (MIT) pada tahun 2001. Tujuannya adalah untuk memudahkan siapa saja yang tidak berlatar belakang ilmu computer dapat menangani hal-hal seperti:

- a. Gambar;
- b. Animasi;
- c. Suara;
- d. Video;
- e. Perangkat keras.

Dengan hanya menggunakan sedikit kode, hasil secara visual dapat diperoleh. Hal ini berbeda dengan bahasa-bahasa lain yang lebih cenderung mengutamakan pemrosesan teks saja. Pada *Processing* dilengkapi

juga dengan IDE sehingga dapat dibuat program (yang secara khusus dinamakan sketsa) melalui *editor* teks yang tersedia pada IDE (Abdul Kadir, 2017). *Processing* merupakan *software* yang dapat digunakan untuk merepresentasikan gambar, animasi, ataupun *interface* peralatan elektronika (Arduino) dengan komputer. Gambar 2.3 memperlihatkan tampilan dari *Processing* IDE. Tipe *Processing* yang digunakan pada penelitian ini adalah *Processing 3.3.7* merupakan tipe terbaru dari *Processing*.



Gambar 2.3 Processing IDE

Pada mulanya, Bahasa pemrograman *processing* dibuat menjadi *software* sketsa untuk mengajarkan dasar-dasar pemrograman pada komputer dalam kategori visual, *processing* menghadirkan tampilan yang berbeda dan sangat mudah untuk digunakan sehingga banyak mahasiswa, seniman, desainer hingga peneliti menggunakan *Processing* untuk belajar membuat visual pada komputer. *Processing* dibuat juga dengan tujuan mengajarkan dasar-dasar pemrograman

komputer untuk memudahkan semua orang belajar desain komputer.

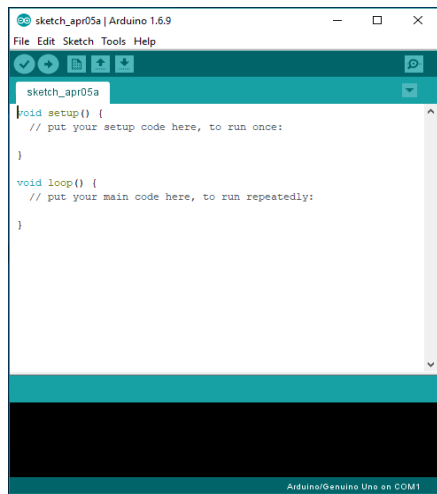
Bahasa *processing* pada implementasinya tidak hanya dibuat dengan bahasa Java, tetapi juga menggunakan sintaks-sintaks yang lebih sederhana sehingga banyak yang lebih memilih menggunakan *software processing* untuk membuat proyek-proyek yang berhubungan dengan sesuatu hal yang sifatnya visual, gerak, dan interaksi. *Processing* dibuat khusus agar mudah digunakan untuk menghasilkan dan memodifikasi gambar yang sedemikian rupa sehingga untuk para pemula tidak mengalami kesulitan untuk menulis program sendiri. Pada *software processing* terdapat banyak pilihan fitur yang mendukung untuk belajar komputer grafik dan interaksi, termasuk juga vector atau raster gambar, model warna, pengolahan citra, jaringan komunikasi, *keyboard*, serta pemrograman berorientasi obyek. *Library* dengan fungsi tambahan dapat di-*download* dan dipelajari dengan mudah. *Library* dengan mudah memperpanjang *Processing* untuk menghasilkan suara, mengirim/menerima data dalam format yang beragam, dan untuk *import/export* dari format file 2D dan 3D (Sulistyo, 2013).

Pemrograman *Processing* terdiri dari teks editor terintegrasi dan jendela tampilan untuk menampilkan program. Apabila tombol “*Run*” ditekan, program akan mengkompilasi dan berjalan di jendela grafis (jendela tampilan). Pada

bagian jendela utama, dapat menerapkan, mengakhiri, menyimpan, membuka, dan mengekspor file. *Processing* menggunakan sistem koordinat kartesian dengan titik asal terletak pada sudut kiri bagian atas. Contohnya, jika program tersebut berukuran lebar 320 piksel dan 240 piksel maka koordinat [0,0] terletak pada kiri bagian atas dan koordinat [320, 240] terletak di bagian bawah.

2.2.4 Arduino Uno ATmega 328

Arduino merupakan platform *computing* fisik atau *embedded*, yang berarti sistem interaktif yang dapat berinteraksi dengan lingkungannya melalui penggunaan perangkat keras dan perangkat lunak. Arduino dapat merasakan lingkungan dengan menerima masukan (*input*) dari berbagai sensor, serta dapat mempengaruhi sekitarnya dan mampu mengendalikan sebuah lampu, actuator, motor, dan lainnya. Mikrokontroler pada papan (*board*) rangkaiannya dapat diprogram menggunakan Bahasa pemrograman Arduino (berdasarkan *Wiring*) dan lingkungan pengembangan Arduino (berdasarkan *Processing*). Arduino dapat berdiri sendiri atau dapat berkomunikasi dengan perangkat lunak yang berjalan pada komputer. Ketika akan meng-*compile*, menulis, dan men-*download* program Arduino dibutuhkan *software* yang bernama Arduino IDE atau lebih dikenal sketch. Gambar 2.5 memperlihatkan tampilan dari Arduino IDE.



Gambar 2.5 Arduino IDE

Tipe Arduino yang digunakan pada penelitian ini adalah Arduino Uno ATmega 328, Arduino Uno ATmega 328 yaitu chip mikrokontroler 8-bit berbasis AVR-RISC buatan Atmel. Chip ini memiliki 32 kb memori ISP flash dengan kemampuan baca-tulis (*read write*) 1kb EEPROM, dan 2 kb SRAM. Kapasitas memori flash sebesar 32 kb pada chip, menjadi alasan chip ini diberi nama ATmega 328. Chip lain memiliki memori 8 kb dan diberi nama ATmega 8, begitu juga dengan ATmega 16 untuk chip yang memiliki memori 16 kb.

Chip tersebut memiliki 23 jalur general purpose I/O (*input/output*), 32 buah register, 3 buah *timer/counter* dengan mode perbandingan, *interrupt* internal dan eksternal, serial *programmable* USART, 2-wire *interface* serial, serial port SPI, 6 buah channel 10-bit A/D converter, *programmable watchdog timer* dengan *oscillator* internal, dan lima *power saving mode*. Chip bekerja pada tegangan antara 1.8V ~ 5.5V. *Output* komputasi bisa mencapai 1 MIPS per Mhz. Maksimum *operating frequency*

adalah 20 Mhz. ATmega328 menjadi cukup populer setelah chip ini dipergunakan dalam board Arduino. Dengan adanya Arduino yang didukung oleh *software* Arduino IDE, pemrograman chip ATmega328 menjadi jauh lebih sederhana dan mudah.

2.2.5 Catu Daya

Arduino dapat diberikan catu daya melalui koneksi USB atau catu daya dari luar non-USB seperti Adaptor to DC dan baterai. Jangkauan tegangan yang dapat disuplai ke Arduino sebesar 4-20 Volt. Namun tegangan yang direkomendasikan yaitu dari 7-12 Volt.

2.2.6 Komunikasi Serial

Komunikasi serial merupakan metode pengiriman data berurut bit demi bit melalui sebuah saluran transmisi. Komunikasi serial memiliki kelebihan seperti, transmisi data yang dilakukan hanya dengan melalui sebuah saluran sehingga mengurangi *cost* yang dikeluarkan untuk kebutuhan saluran transmisi.

2.2.7 Motor DC (*Direct Current*)

1. Motor DC (*Arus searah*)

(C. T. Leondes, 1972) Motor DC adalah pranti elektronik yang mengubah energi listrik menjadi energi mekanik berupa gerak rotasi. Pada Motor DC terdapat jangkar dengan satu atau lebih kumparan terpisah. Tiap kumparan berujung pada cincin belah (*komutator*). Dengan adanya *insulator* antara *komutator*, cincin belah dapat berperan

sebagai saklar kutub ganda (*double pole, double throw switch*). Motor DC bekerja berdasarkan prinsip gaya *Lorentz*, yang menyatakan ketika sebuah konduktor beraliran arus diletakkan dalam medan magnet, maka sebuah gaya (dikenal dengan gaya *Lorentz*) akan tercipta secara *orthogonal* diantara arah medan magnet dan arah aliran arus.

a. Prinsip kerja Motor DC

Pada Motor DC terdapat dua bagian yaitu *Stator* dan *Rotor*. *Stator* merupakan bagian motor yang diam (*statis*) atau tidak berputar. Pada bagian *statis* ini terdiri dari rangka dan kumparan medan. *Rotor* merupakan bagian yang berputar. Pada bagian *rotor* terdiri dari kumparan jangkar. Kedua bagian utama terbagi lagi menjadi beberapa komponen penting, yaitu *Yoke* (kerangka magnet), *Poles* (kutub motor), *Field winding* (kumparan medan magnet), *Armature Winding* (kumparan jangkar), *Commutator* (komutator), dan *Brushes* (kuas/sikat)

Motor DC pada prinsipnya menggunakan reaksi *elektromagnet* untuk bergerak, ketika kumparan diberi arus listrik, maka permukaan kumparan yang bersifat utara akan bergerak menghadap ke magnet yang berkutub selatan dan kumparan yang bersifat selatan akan bergerak menghadap ke arah utara. Ketika kumparan kutub utara bertemu dengan kutub selatan atau juga sebaliknya, maka terjadi tarik menarik yang menyebabkan pergerakan kumparan berhenti. Saat kumparan berhenti dan ingin digerakkan kembali,

saat kutub kumparan berhadapan dengan kutub magnet, arah arus pada kumparan akan dibalik. Maka, kutub utara kumparan akan berubah menjadi kutub selatan, dan kutub selatan akan berubah menjadi kutub utara. Ketika terjadi perubahan kutub, kutub selatan kumparan akan berhadapan dengan kutub selatan magnet, begitu juga kutub utara kumparan akan berhadapan dengan kutub selatan magnet. Saat kutub yang sama berhadapan-hadapan maka terjadi reaksi tolak menolak sehingga kumparan bergerak memutar hingga kutub utara kumparan berhadapan dengan kutub selatan magnet, dan kutub selatan kumparan berhadapan dengan kutub utara magnet. Pada saat ini, arus yang mengalir ke kumparan dibalik kembali dan kumparan akan berputar kembali karena terjadi perubahan kutub. Hal ini akan terjadi berulang-ulang hingga arus listrik pada kumparan diputus atau dihilangkan.

b. Kecepatan putaran Motor DC

Kecepatan putar motor dc dapat dirumuskan dengan persamaan :

$$N = \frac{V_{TM} - I_A R_A}{K_{\phi}}$$

Keterangan:

- N = kecepatan putar motor dc
- V_{tm} = Tegangan Terminal
- I = Arus jangkar motor
- R = Hambatan jangkar motor
- K = Konstanta motor
- Ø = Fluks magnet

Pengendalian kecepatan putar motor dc dapat dilakukan dengan mengatur besar tegangan terminal motor V_{tm} . Metode lain yang bisa digunakan untuk mengendalikan kecepatan motor dc yaitu dengan Teknik modulasi lebar pulsa atau *Pulse Width Modulation* (PWM).

c. Pulse Width Modulation (PWM)

Pulse Width Modulation (PWM) merupakan salah satu cara untuk memanipulasi lebar sinyal yang dinyatakan dengan pulsa dalam suatu periode, untuk mendapatkan tegangan rata-rata yang berbeda. Contoh implementasi PWM yaitu pemodulasi data untuk telekomunikasi, pengontrolan daya atau tegangan yang masuk ke beban, regulator tegangan, audio *effect*, penguatan, dan lainnya. Implementasi PWM berbasis mikrokontroler biasanya digunakan untuk pengendalian kecepatan Motor DC, pengendalian Motor Servo, pengaturan nyala LED, dan lainnya.

Sinyal PWM biasanya memiliki amplitudo frekuensi dasar yang stabil, tetapi lebar pulsa dapat bervariasi. Lebar pulsa PWM berbanding lurus dengan amplitudo sinyal asli yang belum termodulasi. Maka disimpulkan bahwa sinyal PWM memiliki frekuensi gelombang yang tetap namun *duty cycle* bervariasi (antara 0% hingga 100%).

2.2.8 Motor Driver L298

L298 merupakan jenis IC driver motor yang dapat mengendalikan arah putaran dan kecepatan Motor DC

ataupun Motor Stepper. Mampu mengeluarkan output tegangan untuk Motor DC dan Motor Stepper sebesar 50 volt.

L298 yaitu sebuah IC tipe H-Bridge yang dapat mengendalikan beban-beban induktif seperti *relay*, *solenoid*, Motor DC, dan Motor Stepper. IC L298 terdiri dari *transistor-transistor logic* (TTL) dengan gerbang nand yang berfungsi untuk memudahkan dalam menentukan arah putaran sebuah motor dc maupun motor stepper. Modul driver Motor L298 memiliki kelebihan yaitu dalam hal ketelitian dalam mengontrol motor sehingga motor mudah untuk dikontrol.

BAB III METODOLOGI PERANCANGAN

3.1 Alat dan Bahan

3.1.1 Software

Software yang digunakan dalam perancangan sistem pengendali Motor DC berbasis *Graphical User Interface* (GUI) adalah:

- 1) Processing 3.3.7
- 2) Arduino IDE 1.6.9

3.1.2 Hardware

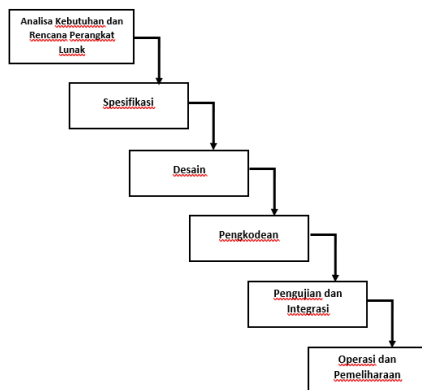
Hardware yang digunakan sebagai sarana penunjang dalam perancangan sistem adalah:

- 1) Arduino Uno ATmega328
- 2) Driver Motor L298
- 3) Motor DC 1000 Rpm 16mm
- 4) Adaptor
- 5) Kabel Jumper
- 6) Kabel Ties
- 7) Double Foam
- 8) Lem Tembak

- 9) Multimeter
- 10) Acrylic
- 11) Specer
- 12) Mur & Baut
- 13) Toolset
- 14) Solder dan tenol
- 15) Atraktor
- 16) Bor
- 17) Box
- 18) *Personal Computer*

3.2 Metode Perancangan Tampilan GUI

Perancangan sistem ini menggunakan metode *waterfall* (model air terjun) yaitu sebuah metode yang paling banyak digunakan dalam *Software Engineering*. Model ini menggunakan pendekatan secara sistematis dan urut pada pengembangan Perangkat Lunak (*Software*).



Gambar 3.2 Diagram Model Waterfall

1) Analisa Kebutuhan User

Pada metode *waterfall*, tahap pertama yang perlu dilakukan yaitu melakukan komunikasi yang bertujuan untuk memahami kebutuhan *software* yang diharapkan oleh *user*. Informasi

dapat diperoleh dengan berbagai cara seperti wawancara dan diskusi. Tahap ini juga merupakan tahap dari mempelajari sistem yang telah ada sebelumnya. Apabila data kebutuhan pengguna sudah didapatkan, maka perlu dilakukan analisa sistem, yaitu:

a. Analisa Identifikasi masalah

Pada tahap ini dilakukan analisa masalah-masalah yang mungkin akan timbul, sehingga nantinya dapat diatasi dengan dibuatnya sebuah sistem yang menjadi solusi dalam permasalahan yang ada.

b. Analisa Kebutuhan

Pada tahap ini dilakukan sebuah analisa tentang kebutuhan user dan keinginan yang mungkin dapat dikembangkan.

c. Analisa Kelayakan Sistem

Pada tahap ini, analisa kelayakan sistem dilakukan agar solusi yang nantinya akan dibuat dapat benar-benar mengatasi masalah yang ada.

2) Spesifikasi

Pada tahap kedua yang perlu dilakukan yaitu menentukan spesifikasi dari *software* yang dibutuhkan oleh *user*. Spesifikasi ditentukan setelah melakukan analisis kebutuhan *user* pada tahap awal.

3) Desain Sistem

Pada tahap ketiga, akan dimulai merancang desain sebuah sistem. Desain sistem yang akan dibuat berdasarkan spesifikasi yang sudah di dapat pada tahap sebelumnya. Pada tahap ini juga dilakukan perancangan arsitektur tampilan GUI, dan menentukan perangkat keras

(*hardware*). Pada tahap ini akan dilakukan dalam beberapa tahapan yaitu:

a. Perancangan Proses

Perancangan proses adalah rancangan sistem yang akan dibuat dalam sebuah perangkat lunak dan dapat di tampilkan dengan diagram alir. Pada diagram alir akan ditampilkan sebuah gambaran yang lebih mendetail tentang fungsi-fungsi dan logika yang berkerja.

b. Perancangan Basis Data

Perancangan basis data adalah rancangan yang menunjukkan hubungan antara realita dan ekspetasi.

c. Perancangan Antarmuka

Perancangan antar muka merupakan tahapan perancangan untuk mendesain sistem. Desain sistem perlu dilakukan sebelum melakukan pembuatan sistem.

4) Pengkodean

Pada tahap pengkodean ini dilakukan merealisasikan perangkat lunak (*software*) yang telah di desain sebelumnya. Pengkodean ini akan menghasilkan suatu tampilan sistem, dimana nantinya dapat digunakan oleh *user* sesuai dengan kebutuhan yang telah dianalisa sebelumnya.

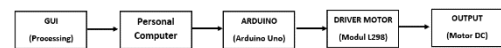
5) Pengujian dan Integrasi

Pada tahap ini, sistem akan diterapkan dan di lakukan pengujian pada sebuah rangkaian alat. Pada tahap ini juga akan dilakukan evaluasi pada *hardware*, pengembangan dan modifikasi *software*. Pengujian ini bertujuan untuk mengecek setiap kegagalan maupun kesalahan yang terjadi pada sistem dan rangkaian alat.

6) Operasi dan Pemeliharaan

Pada tahap ini merupakan tahapan terakhir dimana *software* maupun *hardware* sudah jadi, dan dilakukan pengoperasian dan pemeliharaan. Pemeliharaan meliputi perbaikan kesalahan-kesalahan yang tidak dijumpai pada tahapan sebelumnya.

3.3 Blok Diagram Pembangunan GUI



Gambar 3.3 Diagram Pembangunan GUI dan Rangkaian Motor DC

Gambar 3.3 diatas merupakan diagram blok pembangunan GUI dan Rangkaian Motor DC. Dalam Perancangan ini GUI digunakan sebagai media interaksi antarmuka. Tampilan GUI dibuat menggunakan bahasa pemrograman *Processing*. *Software Processing* digunakan untuk membangun tampilan GUI dengan beberapa *library* pendukung, seperti *library* ControlP5 dan *library Processing* serial. Dengan *library* ControlP5 dapat digunakan untuk pembuatan tombol dan berbagai komponen (atau control) yang menyusun antar muka seperti slider, knob, dan dapat juga untuk mempermudah dalam pemilihan warna tampilan. Tombol-tombol yang ada nantinya akan digunakan untuk mengendalikan putaran pada Motor DC. Pada *library Processing* serial ini memungkinkan perangkat computer dan

Arduino dapat berkomunikasi melalui serial.

Tampilan GUI ini dibangun untuk mengendalikan Motor DC. Terdapat 2 hal yang akan dikendalikan GUI pada Motor DC, yaitu kecepatan dan arah putrannya. Pada tampilan GUI terdapat knob, dan tobo-tombol yang mempunyai fungsi masing-masing. Knob pada tampilan GUI berfungsi sebagai pengendali kecepatan putaran pada Motor DC, range nilainya yaitu 0-255. Tombol CW dan CCW berfungsi sebagai pengendali arah putaran Motor DC, sedangkan untuk tombol STOP berfungsi untuk menghentikan putaran pada motor DC. Pada pengoperasian knob dan tombol ini dikombinasikan melalui algoritma tertentu sehingga ketika kecepatan pada knob diatur dan tombol ditekan maka GUI akan mengirimkan perintah data kecepatan dan arah putaran pada motor DC.

BAB IV HASIL DAN

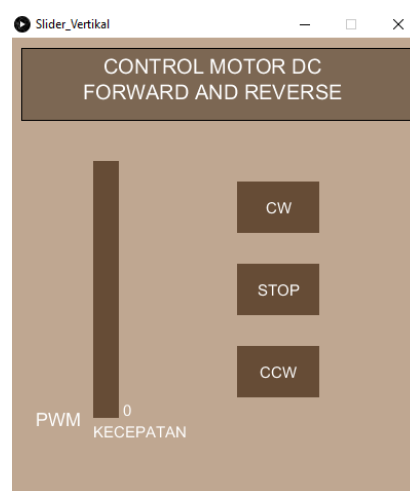
PEMBAHASAN

4.1 Hasil Perancangan

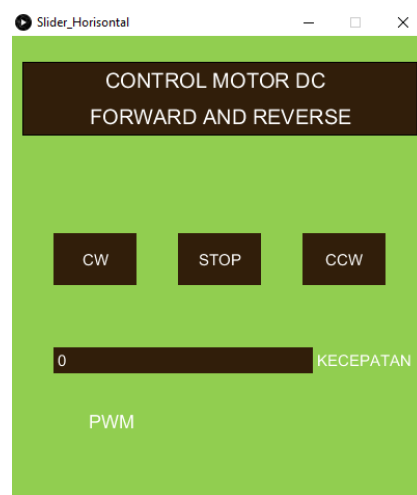
4.1.1 Tampilan Program GUI

Berdasarkan pada metode perancangan Bab III, perancangan ini membentuk 3 (tiga) versi desain tampilan program *Graphical User Interface* (GUI). Pada ketiga versi desain ini telah dilakukan pengujian dengan cara membagikan kuesioner ke beberapa responden yaitu mahasiswa yang berkuliah di Fakultas Teknik Universitas Muhammadiyah

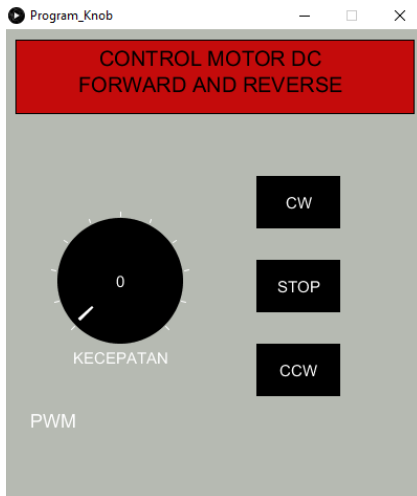
Yogyakarta, dimana memang sasaran utama pengguna (*user*) program GUI ini adalah mereka yang berkuliah di jurusan keteknikan, khususnya Teknik Elektro. Desain tampilan program *Graphical User Interface* (GUI) ini akan ditunjukkan seperti pada gambar 4.1, gambar 4.2, dan gambar 4.3.



Gambar 4.1 GUI Versi 1



Gambar 4.2 GUI Versi 2



Gambar 4.3 GUI Versi 3

Pada perancangan ketiga desain program *Graphical User Interface* (GUI) di atas (gambar 4.1, gambar 4.2, gambar 4.3) dirancang dengan mempertimbangkan metode kombinasi warna terbaik dan kombinasi warna terburuk. Pemilihan kombinasi warna pada desain program GUI ini dipilih agar terlihat lebih menarik dan juga pengguna (*user*) akan merasa nyaman, mata tidak merasa cepat kelelahan saat menggunakan program ini.

4.1.2 Hasil Kuesioner

Dari hasil pengujian dengan cara membagikan kuesioner kepada beberapa responden, dapat dilihat hasilnya dalam *chart* berikut:

1. Penilaian untuk Pemilihan Kombinasi Warna

Penilaian untuk pemilihan kombinasi warna pada program GUI, dapat dilihat pada diagram bahwa:

- a. Pada GUI versi 1, sebanyak 3 (tiga) responden memberikan nilai 2 (Buruk), 13 (tiga belas) responden memberikan nilai 3 (Cukup), 6

(enam) responden memberikan nilai 4 (Baik), dan 3 (tiga) responden memberikan nilai 5 (Sangat Baik). Pada lembar kuesioner, beberapa responden mengatakan bahwa pada GUI versi satu, kombinasi warna yang dipilih terkesan sangat monoton dan membosankan

- b. Pada GUI versi 2, sebanyak 13 (tiga belas) responden memberikan nilai 3 (Cukup), 8 (delapan) responden memberikan nilai 4 (Baik), dan 4 (empat) responden memberikan nilai 5 (Sangat Baik). Pada lembar kuesioner, responden mengatakan bahwa pemilihan warna pada program GUI versi 2 sudah baik, namun untuk pemilihan warna backgroundnya terlalu terang (*eye catching*), sehingga menyebabkan mata tegang saat melihat tampilannya.
- c. Pada GUI versi 3, sebanyak 2 (dua) responden memberikan nilai 2 (Buruk), 7 (tujuh) responden memberikan nilai 3 (Cukup), 8 (delapan) responden memberikan nilai 4 (Baik), dan 8 (delapan) responden nilai 5 (Sangat Baik). Pada lembar kuesioner, beberapa responden mengatakan bahwa kombinasi warna pada program GUI versi 3 sudah sangat baik.
- d. Berdasarkan diagram diatas, sebanyak 12 (dua belas) responden memberikan nilai 3 (Cukup) pada program GUI versi 1 dan 2, sedangkan untuk program GUI Versi 3 responden banyak memberikan nilai 4 (Baik) dan nilai 5 (Sangat

Baik) yaitu sebanyak 8 (delapan) responden pada masing masing nilai. Dilihat dari penilaian responden maka didapat disimpulkan bahwa mayoritas responden lebih menyukai kombinasi warna pada program GUI versi 3. Hanya saja untuk warna tombol CW, CCW, dan stop lebih baik dibedakan.

2. Penilaian untuk Kemudahan Pengoperasian

Penilaian untuk kemudahan Pengoperasian pada program GUI, dapat dilihat pada diagram bahwa:

- a. Pada GUI versi 1, sebanyak 1 (satu) responden memberikan nilai 1 (Sangat Buruk), 5 (lima) responden memberikan nilai 3 (Cukup), 13 (tiga belas) responden memberikan nilai 4 (Baik), dan 6 (tiga) responden memberikan nilai 5 (Sangat Baik).
- b. Pada GUI versi 2, sebanyak 1 (satu) responden memberikan nilai 1 (Sangat Buruk), 5 (lima) responden memberikan nilai 3 (Cukup), 15 (lima belas) responden memberikan nilai 4 (Baik), dan 4 (empat) responden memberikan nilai 5 (Sangat Baik).
- c. Pada GUI versi 3, sebanyak 1 (satu) responden memberikan nilai 1 (Sangat Buruk), 3 (tujuh) responden memberikan nilai 3 (Cukup), 13 (tiga belas) responden memberikan nilai 4 (Baik), dan 8 (delapan) responden nilai 5 (Sangat Baik).
- d. Berdasarkan diagram diatas, banyak responden memberikan nilai 4 (baik) dan nilai 5 (Sangat Baik) pada program GUI versi 3. Pada lembar

kuesioner, responden mengatakan bahwa pengoperasian GUI sudah baik, dan mudah, dan kebanyakan responden lebih menyukai pemilihan indicator kecepatan dalam bentuk knob, karena terlihat lebih familiar seperti speedometer pada kendaraan serta mudah dalam pengaturannya, lalu untuk tombol CW, CCW dan STOP lebih baik menggunakan simbol saja daripada tulisan (cw, ccw, stop), agar pengoperasiannya lebih mudah. Maka didapat disimpulkan bahwa mayoritas responden merasa program GUI versi 3 lebih mudah dalam pengoperasiannya.

3. Penilaian untuk Tampilan Posisi Tombol

Penilaian untuk Tampilan Posisi Tombol pada program GUI, dapat dilihat pada diagram bahwa:

- a. Pada GUI versi 1, sebanyak 12 (dua belas) responden memberikan nilai 3 (Cukup), 11 (sebelas) responden memberikan nilai 4 (Baik), dan 2 (dua) responden memberikan nilai 5 (Sangat Baik).
- b. Pada GUI versi 2, sebanyak 1 (satu) responden memberikan nilai 2 (Buruk), 9 (sembilan) responden memberikan nilai 3 (Cukup), 10 (sepuluh) responden memberikan nilai 4 (Baik), dan 5 (lima) responden memberikan nilai 5 (Sangat Baik).
- c. Pada GUI versi 3, sebanyak 3 (tiga) responden memberikan nilai 2 (Buruk), 4 (empat) responden memberikan nilai 3 (Cukup), 11 (sebelas) responden memberikan

nilai 4 (Baik), dan 7 (tujuh) responden nilai 5 (Sangat Baik).

- e. Berdasarkan diagram diatas, banyak responden memberikan nilai yang kurang baik dalam penempatan posisi tombol pada program GUI versi satu, (dua belas) responden memberikan nilai 3 (Cukup) pada program GUI versi 1 dan 2, sedangkan untuk program GUI Versi 3 responden banyak memberikan nilai 4 (Baik) dan nilai 5 (Sangat Baik) yaitu sebanyak 8 (delapan) responden pada masing masing nilai. Pada lembar kuesioner, responden menyebutkan bahwa penempatan tombol masih kurang rapi, terlebih pada GUI versi 2. Responden menjelaskan bahwa untuk indicator kecepatan lebih baik di letakkan di atas tombol CW, CCW, dan STOP. Maka didapat disimpulkan bahwa mayoritas responden menilai GUI versi 3.

4. Penilaian GUI secara Keseluruhan

Penilaian untuk Tampilan keseluruhan pada program GUI, dapat dilihat pada diagram bahwa:

- a. Pada GUI versi 1, sebanyak 1 (satu) responden memberikan nilai 2 (Buruk), 12 (dua belas) responden memberikan nilai 3 (Cukup), 9 (sembilan) responden memberikan nilai 4 (Baik), dan 3 (tiga) responden memberikan nilai 5 (Sangat Baik).
- b. Pada GUI versi 2, sebanyak 1 (satu) responden memberikan nilai 2 (Buruk), 8 (delapan) responden memberikan nilai 3 (Cukup), 12 (dua

belas) responden memberikan nilai 4 (Baik), dan 4 (empat) responden memberikan nilai 5 (Sangat Baik).

- c. Pada GUI versi 3, sebanyak 1 (satu) responden memeberikan nilai 2 (Buruk), 6 (enam) responden memberikan nilai 3 (Cukup), 8 (delapan) responden memberikan nilai 4 (Baik), dan 10 (sepuluh) responden nilai 5 (Sangat Baik).

5. Penilaian GUI yang paling disukai Responden

Penilaian untuk program GUI yang paling disukai oleh responden, dapat dilihat pada diagram bahwa:

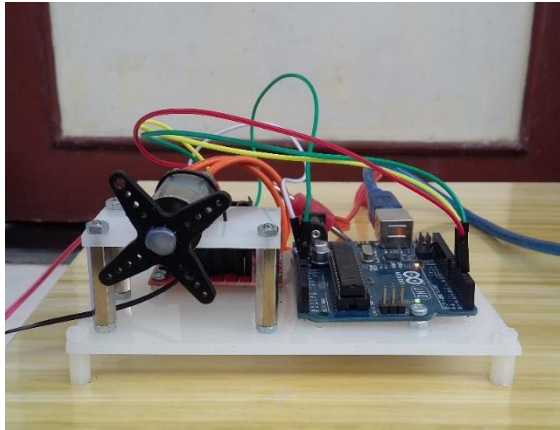
- a. Sebanyak 1 (satu) responden memilih program GUI versi 1.
- b. Sebanyak 5 (lima) responden memilih program GUI versi 2.
- c. Sebanyak 19 (Sembilan belas) responden memilih program GUI versi 3.

4.1.3 Tampilan Rangkaian Motor DC

Rangkaian Motor DC ini dirancang dengan beberapa komponen yaitu:

- 1) Arduino Uno ATmega328
- 2) Driver Motor L298
- 3) 1 buah Motor DC 16mm 1000 RPM
- 4) Adaptor 12V
- 5) Kabel Jumper
- 6) Kabel Ties
- 7) Double Foam
- 8) Acrylic

- 9) Mur & Baut
- 10) Specer



Gambar 4.9 Rangkaian Motor DC

4.2 Pembahasan

4.2.1 Pengujian Tombol

1. Skrip 1 (Pengujian Tombol CW (*clockwise*))

Pada perancangan program *Graphical User Interface* ini terdapat tombol CW (*clockwise*), yaitu merupakan tombol yang digunakan untuk memberi perintah pada Motor DC untuk berputar searah jarum jam. Tombol ini akan mulai bekerja setelah *user* (pengguna) mengatur kecepatan putaran Motor DC pada slider ataupun knob, lalu setelah mengatur kecepatan, *user* akan memilih dan menekan tombol CW tersebut. Pengujian akan dilakukan dengan cara mengatur kecepatan putaran mulai dari minimum kecepatan hingga maksimum kecepatan (Range 0-255).

2. Skrip 2 (Pengujian Tombol CCW (*Counter Clockwise*))

Pada perancangan program *Graphical User Interface* (GUI) ini terdapat pula tombol CCW (*Counter Clockwise*), yaitu merupakan tombol yang digunakan untuk memberi

perintah pada Motor DC untuk berputar berlawanan arah jarum jam. Tombol ini akan mulai bekerja setelah *user* (pengguna) mengatur kecepatan putaran Motor DC pada slider ataupun knob, lalu setelah mengatur kecepatan *user* akan memilih dan menekan tombol CCW tersebut. Pengujian akan dilakukan dengan cara mengatur kecepatan putaran mulai dari minimum kecepatan hingga maksimum kecepatan (Range 0-255).

3. Skrip 3 (Pengujian Tombol STOP)

Pada perancangan program *Graphical User Interface* (GUI) ini terdapat tombol STOP (berhenti), yaitu merupakan tombol yang digunakan untuk memberi perintah pada motor DC untuk berhenti berputar. Tombol ini akan bekerja setelah *user* (pengguna) memilih dan menekan tombol STOP tersebut.

Pada tabel 4.1 merupakan hasil dari pengujian tombol CW pada program GUI ke rangkaian Motor DC

Tabel 4.1 Pengujian Tombol CW (*clockwise*)

No	Nilai PWM	Kondisi Motor DC	Keterangan	Duty Cycle
1	0-10	Berhenti	Tidak Berputar	0-3.92%
2	11-24	Berhenti	Tidak Berputar	4.31% - 9.41%
3	25-30	CW (<i>clockwise</i>)	Berputar Sangat Pelan	9.80% - 11.76%
4	31-40	CW (<i>clockwise</i>)	Berputar Pelan	12.15% - 15.68%
5	41-50	CW (<i>clockwise</i>)	Berputar Cukup Cepat	10.07% - 19.6%
6	51-60	CW (<i>clockwise</i>)	Berputar Cepat	20% - 23.52%
7	61-90	CW (<i>clockwise</i>)	Berputar Cepat	23.92% - 35.29%
8	91-120	CW (<i>clockwise</i>)	Berputar Lebih Cepat	35.68% - 47.05%
9	121-150	CW (<i>clockwise</i>)	Berputar Lebih Cepat	47.45% - 58.82%
10	151-180	CW (<i>clockwise</i>)	Berputar Sangat Cepat	59.21% - 70.58%
11	181-255	CW (<i>clockwise</i>)	Berputar Sangat Cepat	70.98% - 100%

Dari hasil data pengujian pada table 4.1 dapat dilihat bahwa nilai PWM sangat berpengaruh terhadap putaran motor, ketika nilai PWM kecil maka motor akan berputar lambat bahkan jika dilihat dari table 4.1, saat nilai PWM 0-24 motor dalam kondisi berhenti atau tidak berputar, begitu juga dengan nilai PWM yang semakin besar maka putaran yang ada pada motor akan semakin cepat. Hal ini dapat disimpulkan bahwa nilai PWM berbanding lurus dengan kecepatan putaran pada motor.

Pada tabel 4.2 merupakan hasil dari pengujian tombol CCW pada program GUI ke rangkaian Motor DC

Tabel 4.2 Pengujian Tombol CCW (*counter clockwise*)

No	Nilai PWM	Kondisi Motor DC	Keterangan	Duty Cycle
1	0-10	Berhenti	Tidak Berputar	0-3.92%
2	11-24	Berhenti	Tidak Berputar	4.31% - 9.41%
3	25-30	CCW (<i>counter clockwise</i>)	Berputar Sangat Pelan	9.80% - 11.76%
4	31-40	CCW (<i>counter clockwise</i>)	Berputar Pelan	12.15% - 15.68%
5	41-50	CCW (<i>counter clockwise</i>)	Berputar Cukup Cepat	10.07% - 19.6%
6	51-60	CCW (<i>counter clockwise</i>)	Berputar Cepat	20% - 23.52%
7	61-90	CCW (<i>counter clockwise</i>)	Berputar Cepat	23.92% - 35.29%
8	91-120	CCW (<i>counter clockwise</i>)	Berputar Lebih Cepat	35.68% - 47.05%
9	121-150	CCW (<i>counter clockwise</i>)	Berputar Lebih Cepat	47.45% - 58.82%
10	151-180	CCW (<i>counter clockwise</i>)	Berputar Sangat Cepat	59.21% - 70.58%
11	181-255	CCW (<i>counter clockwise</i>)	Berputar Sangat Cepat	70.98% - 100%

Dari hasil data pengujian pada table 4.1 dapat dilihat bahwa nilai PWM sangat berpengaruh terhadap putaran motor, ketika nilai PWM kecil maka motor akan berputar lambat bahkan jika dilihat dari table 4.2, saat nilai PWM 0-24 motor dalam kondisi berhenti atau tidak berputar, begitu juga dengan nilai PWM yang semakin besar maka putaran

yang ada pada motor akan semakin cepat. Hal ini dapat disimpulkan bahwa nilai PWM berbanding lurus dengan kecepatan putaran pada motor.

Pada tabel 4.3 merupakan hasil dari pengujian tombol STOP pada program GUI ke rangkaian Motor DC

Tabel 4.3 Pengujian Tombol STOP (berhenti)

No	Nilai PWM	Kondisi Motor DC	Keterangan
1	0-10	Berhenti	Tidak Berputar
2	11-24	Berhenti	Tidak Berputar
3	25-30	Berhenti	Tidak Berputar
4	31-40	Berhenti	Tidak Berputar
5	41-50	Berhenti	Tidak Berputar
6	51-60	Berhenti	Tidak Berputar
7	61-90	Berhenti	Tidak Berputar
8	91-120	Berhenti	Tidak Berputar
9	121-150	Berhenti	Tidak Berputar
10	151-180	Berhenti	Tidak Berputar
11	181-255	Berhenti	Tidak Berputar

Dari hasil data pengujian pada tabel 4.3 dapat dilihat bahwa saat tombol STOP ditekan maka putaran motor akan berhenti. Hal ini dikarenakan saat tombol STOP ditekan, GUI mengirimkan perintah berupa data karakter "S" dan nilai PWM 0, sehingga berapapun kecepatan yang di atur, nilai PWM yang akan diterima oleh Arduino tetap 0.

4.2.2 Hasil Pengujian Sinyal PWM menggunakan Osiloskop

- 1) Telah dilakukan pengujian sinyal PWM dengan osiloskop sebenarnya. Dari pengujian yang telah dilakukan tampak bahwa mula-mula diberikan

- posisi awal motor tidak bergerak (Nilai PWM 0) tampak bahwa tampilan sinyal di osiloskop tampak datar (horisontal).
- 2) Seiring penambahan nilai sinyal PWM, maka akan tampak muncul sinyal gelombang kotak di osiloskop. Dengan jumlah kotak vertical sebanyak 5 kolom dengan setting 1 V/div. Hal ini membuktikan bahwa sinyal PWM berupa digital sudah mulai bekerja.
 - 3) Seiring bertambahnya nilai PWM, dapat terlihat juga bahwa duty cycle dari gelombang tersebut tampak semakin melebar, hal ini dilihat dari pembacaan periode pada osiloskop.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan perancangan sistem pengendali Motor DC berbasis *Graphical User Interface* (GUI) menggunakan *Processing*, maka dapat diambil beberapa kesimpulan sebagai berikut:

- 1) *Library* yang digunakan pada bahasa Pemrograman *Processing* yaitu *ControlP5*. *Library* ini mempermudah dalam pembuatan tombol dan berbagai komponen yang menyusun antar muka seperti slider, knob, pemilihan warna, dan sebagainya.
- 2) *Graphical User Interface* (GUI) dapat diterapkan sebagai representasi dari interaksi manusia dengan robot seperti pada pengendalian Motor DC.

- 3) *Graphical User Interface* (GUI) dapat digunakan untuk memudahkan mengontrol arah putaran dan kecepatan putaran Motor DC.
- 4) Berdasarkan hasil pengujian, tampilan sistem sudah sesuai dengan yang dibutuhkan oleh user.
- 5) Berdasarkan hasil penilaian untuk program GUI yang paling disukai oleh *user* yaitu GUI Versi 3.

5.2 Saran

Dalam perancangan ini masih terdapat banyak kekurangan yang memungkinkan untuk dikembangkan lebih lanjut. Diantaranya adalah:

- 1) Dalam perancangan program GUI ini, pengendalian kecepatan pada Motor DC masih belum bisa otomatis dilakukan, harus melewati tombol STOP terlebih dahulu untuk merubah kecepatan putarannya. Diharapkan kedepannya dapat dikembangkan lagi pengendalian kecepatan secara otomatis dengan cara merubah struktur program pada GUI.
- 2) Perancangan GUI sebagai pengendalian Motor DC ini masih sebatas Prototipe, diharapkan kedepannya perancangan ini dapat dikembangkan lagi sehingga tidak hanya sebagai pengendali kecepatan dan arah putarannya saja.

DAFTAR PUSTAKA

- [1] Sodnik, Jaka., dan Tomazic, Saso. "Spatial Auditory Human-Computer Interface". Springer. 1-2, 2015.
- [2] Singhera, Zafar., Horowitz, Ellis., dan Shah, Abad."A Graphical User Interface (GUI) Testing Methodology". Chapter: 7.24, 2009.
- [3] Mustar, M.Y., Santosa, P.I., dan Hartanto, Rudy. "Perancangan Model Interaksi Manusia dan Robot Dalam Bentuk Tampilan Visual Pada Komputer". Yogyakarta: STMIK AMIKOM, 2014.
- [4] Kadir, Abdul. 2017. *Pemrograman ARDUINO dan Processing*. Jakarta: PT Elex Komputindo.
- [5] Siregar, Rahmat Fauzi.2017. *Interaksi Manusia Dan Robot Berbasis GUI (Graphical User Interface)*. Yogyakarta: Universitas Muhammadiyah Yogyakarta.
- [6] Hedwig, Rinda. "Teori Sistem". Jakarta: Universitas Bina Nusantara, 1-12, 2004.
- [7] DATASHEET ARDUINO UNO ATMEGA328
http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf
Diakses tanggal 31 Maret 2018.
- [8] <http://labdegaragem.com/forum/topics/ponte-h-l298n-nao-funciona>
Diakses pada tanggal 31 Maret 2018
- [9] <https://teknikelektronika.com/pengertian-n-motor-dc-prinsip-kerja-dc-motor/>
Diakses pada tanggal 31 Maret 2018
- [10] <https://www.nyebarilmu.com/tutorial-arduino-mengakses-driver-motor-l298n/>
Diakses pada tanggal 31 Maret 2018
- [11] <http://opensource.telkomspeedy.com/repo/abba/v06/Kuliah/SistemOperasi/BUKU/SistemOperasi-4.X-1/ch06s03.html>
Diakses pada tanggal 1 Juli 2018
- [12] <http://www.teorikomputer.com/2016/10/pengertian-kelebihan-dan-kelemahan.html> Diakses pada tanggal 1 Juli 2018
- [13] <http://www.tespenku.com/2018/02/sistem-kontrol-loop-terbuka.html>
Diakses pada tanggal 1 Juli 2018
- [14] [http://achamad.staff.ipb.ac.id/wp-content/plugins/as-pdf/andri_mz-Pulse%20Width%20Modulation%20\(PWM\).pdf](http://achamad.staff.ipb.ac.id/wp-content/plugins/as-pdf/andri_mz-Pulse%20Width%20Modulation%20(PWM).pdf)
Diakses pada tanggal 1 Juli 2018
- [8] <http://beritanet.com/Literature/Tutorial/psikologi-warna-design-interface.html>
Diakses pada tanggal 9 Agustus 2018