BAB 1

PENDAHULUAN

A. Latar Belakang

Sistem kendali dan stabilisasi arah dan sudut merupakan salah satu bidang penting dalam rekayasa dan otomisasi, terutama dalam pengembangan perangkat yang memerlukan presisi arah tertentu, semenjaksistem penjejak satelit, penunjuk arah, turret kamera, hingga alat bantu navigasi. Seiring berkembangnya teknologi, mikrokontroler seperti arduino telah menjadi platform yang populer dan efisien dalam pengembangan sistem kendali karena sifatnya yang fleksibel, open-source, dan mudah digunakan.

Penerapan sistem kendali dan stabilizator sudut berbasis arduino memungkinkan perangkat untuk secara otomatis mengarahkan atau mempertahankan orientasi sudut tertentu terhadap referensi tertentu, walaupun terjadi gangguan dari luar seperti getaran atau perunahan posisi. Hal ini sangat penting dalam berbagai skenario, baik pada perangkat bergerak maupun sistem status yang membutuhkan presisi tinggi

Sebagai studi kasus penerapan, penelitianini menggunakan arah kiblat sebagai contoh praktis. Arah kiblat dipilih karena memiliki nilai penting dalam kontes keagamaan dan membutuhkan presisi arah. Namun, fokus utama penelitian ini bukan pada aspek keagamaannya, melainkan pada bagaimana sistem secara otomatis mengarahkan dan menstabilkan suatu sudut tertentu secara realtime dengan bantuan sensor dan aktuator.

Dalam sistem ini, digunakan sensor magnetometer HMC5883L sebagai pendeteksi arah, stepper motor digunakan sebagai aktuator, dan arduino sebagai pusat pemrosesan data dan kendali. Dengan demikian, penelitian ini menitikberatkan pada aspek perancangan, implementasi dan pengujian performa sistem kendali sudut berbasis arduino secara umum

B. Rumusan Masalah

Berdasarkan masalah diatas dapat dirumuskan masalah sebagai berikut;

1. Bagaimana merancang sistem kendali dan stabilizator sudut berbasis arduino dan dapat mengarahkan otomatis ke arah tertentu?
2. Bagaimana performa sistem dalam menjaga kestabilan arah tertentu terhadap gangguan luar seperti perubahan posisi atau getaran ?
3. Bagaimana sistem dapat diaplikasikan secara fleksibel dalam berbagai konteks penunjuk arah, seperti arah kiblat sebagai studi kasus?

C. Tujuan Penelitian

Berdasarkan rumusan masalah, tujuan dari penelitian ini yaitu :

1. Merancang dan mengembangkan sistem kendali dan stabilisator sudut/arah berbasis arduino
2. Mengimplementasikan dan menguji algoritma kendali untuk mengarahkan dan mempertahankan sudut tertentu dengan akurat dan stabil
3. Menerapkan sistem yang dibangun pada studi kasus penunjuk arah kiblat sebagai bentuk validasi fungsional

D. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah:

1. Penelitian difokuskan pada stabilisasi sudut dalam kondisi statis (tidak diaplikasikan pada perangkat bergerak)
2. Penentuan arah kiblat hanya dijadikan sebagai contoh penerapan, bukan sebagai fokus utama

E. Manfaat penelitian

1. Secara teknis, menjadi referensi pengembangan sistem kendali arah/sudut berbasis mikrokontroler untuk berbagai aplikasi teknik dan teknologi
2. Secara praktis, memberikan solusi kendali sudut yang dapat digunakan pada berbagai perangkat presisi sebagai petunjuk arah, kamera otomatis hingga sistem navigasi mini
3. Secara akademis, memberikan kontribusi dalam pengembangan sistem kendali dan pemrograman mikrokontroler berbasis arduino

BAB II

TINJAUAN PUSTAKA

A. Mikrokontroler Arduino Uno

Mikrokontroller adalah sistem *mikroprosesor* lengkap yang terkandung didalam sebuah chip. *Mikrokontroler* berbeda dari *mikroprosesor* serba guna yang digunakan dalam sebuah *PC*, karena di dalam sebuah *mikrokontroler* umumnya juga telah berisi.

Adapun *mikrokontroler* yang dipakai peneliti adalah *mikrokontroler* jenis Arduino Uno sebagai pengontrol eletronik untuk membaca dan menulis data untuk tersambung ke komputer.

Arduino uno merupakan sebuah *board mikrokontroller* yang mana didalamnya terdapat mikrokontroller ATmega328 yang memiliki 14 pin digital (6 pin dapat digunakan sebagai output PWM),6 input analog sebuah header ICSP,dan sebuah tombol reset yang berfungsi untuk mengulang program. Untuk mendukung mikrokontroller agar dapat digunakan, hanya dengan menghubungkan board arduino ke computer dengan menggunakan kabel USB atau listrik dengan AC ke adaptor DC atau baterai untuk menjalankannya.

Setiap 14 pin digital pada arduino uno dapat di gunakan sebagai input dan output, menggunakan pin *mode*().digital *writer*(), dan *digital read*(). Pada fungsi tersebut beroperasi di tegangan 5V, setiap pin dapat memberikan atau menerima suatu arus maksimum 40 mA dan mempunyai sebuah resistor *pull-up* (terputus secara default) 20-50 Kohm.

Secara umum, arduino dengan sebuah mikrokontroler ini mampu menciptakan suatu program yang dapat mengendalikan berbagai komponen elektronika. Sehingga cukup jelas, bila fungsi yang dimiliki arduino uno adalah untuk memudahkan pengguna dalam melakukan *prototyping*,dan memprogram mikrokontroller serta menciptakan berbagai macam alat canggih berbasis mikrokontroller (Arga, 2020) . Arduino uno dan ekosistemnya punya kelebihan sebagai berikut:

a. Di dukung oleh arduino IDE dan bahasa pemrograman yang sudah cukup lengkap librarynya.
b. Terdapat modul yang siap pakai yang dapat langsung dipasang pada board Arduino.
c. Dukungan dokumentasi yang bagus dan memiliki komunikasi yang solid.



Gambar 2. 1 Mikrokontroler Arduino Uno

Tabel 2. 1 Spesifikasi Arduino Uno

Data Teknis Arduino Uno	
Microkontroler	ATMega 328P
Tegangan operasi	5V
Tegangan input	7-12 V
Pin digital I/O	14 (6 diantaranya PWM output)
Pin Analog input	6 pin
Arus DC per pin I/O	40 Ma
Arus DC untuk pin 3,3 V	50 Ma
Flash memory	32 KB (ATmega328p)
SRAM	2 KB
EEPROM	1 KB
Clock speed	16 MHz
Panjang	68.6 mm
Lebar	53.44 mm
Berat	25 g

Beberapa *power* pin pada arduino adalah sebagai berikut:

a. VIN:Tegangan input pada board arduino uno ketika menggunakan sumber daya (5V dari sambungan USB atau dari sumber regulator lain) anda dapat menyuplai tegangan pada pin ini, dan jika suplai tegangan lewat *power* jack, dapat diakses melalui pin ini atau langsung ke jack *power* 5V.
b. 5V: Catu daya yang digunakan untuk *power* mikrokontroller dan beberapa komponen lainnya telah diatur tegangan sebesar 5 V dari regulator pada *board*.dapat disuplai melalui DC jack power (7-12 Volt).

c. 3V3:Tegangan sebesar 3,3 Volt diperoleh dari FTDI chip yang ada pada *board*. Arus maksimum sebesar 50 mA.
d. Pin *ground*:GND /pin *ground* berfungsi sebagai lintasan *ground* pada Arduino
e. Memory:ATmega328 memiliki 32 KB memori flash Dengan 0,5 digunakan sebagai bootloader, dan juga mempunyai 2 KB SRAM dan memori EEPROM sebesar 1 KB.
f. *Input* dan *output*:setiap pin pada arduino uno dapat digunakan sebagai input maupun output yang menggunakan funsgsi pin mode,digital *write*,dan digital *read*, pada pin tersebut dapat beroperasi dengan tegangan 5V. setiap pin juga mampu memberikan atau menerima arus maksimal hingga 40 mA dan resistor internal pull-up antara 20-50 Kohm.

Selain itu, terdapat beberapa fungsi khusus dari pin tersebut:

a. Serial : 0 (RX) dan 1 (TX) berfungsi untuk menerima dan mengirimkan data serial. Pin ini terhubung pada pin yang sesuai dari USB ke TTL chip
b. Interupsi eksternal : 2 dan 3 pada pin ini difungsikan untuk memicu interupsi pada nilai yang rendah, naik turun, ataupun perubahan nilai.
c. PWM :2,5,6,9,10 dan 11 menyediakan output 8 bit dengan fungsi analog write().
d. SPI : pin 10 (SS), 11 (MISO), dan 13 (SCK).pin ini mendukung komunikasi SPI dengan menggunakan *library* SPI
e. LED : pin 13 terdapat LED internal yang dihubungkan ke pin 13 pada board. Ketika pin berbilai tinggi maka LED akan menyala(ON), dan begitu pula sebaliknya.

f. Pada arduino uno memiliki pin analog dengan 6 input yang berlabel A0 sampai A5, dimana maing-masing menyediakan 10 bit dengan resolusi 1024. Secara default, 5 volt dari *ground.*

1. Motor Stepper



Gambar 2. 2 Motor Stepper

Motor stepper adalah jenis motor listrik yang bergerak dalam langkah-langkah distrik, artinya putaran porosnya dibagi menjadi sejumlah langkah tetap per satuan putaran penuh. Salah satu jenis motor stepper yang banyak digunakan dalam bidang otomasi dan kontrol gerak adalah motor stepper bipolar,

Motor ini bekerja berdasarkan prinsip elektromagnetik, dimana rotor (bagian yang berputar) ditarik oleh medan magnet yang dihasilkan oleh kumparan pada stator (bagian dalam). untuk setiap sinyal yang diberikan ke kumparan secara bergantian, rotor akan melangkah ke posisi berikutnya. Dalam motor bipolar terdapat dua kumparan utama (fasa A dan fasa B), arus harus dibalik-balik dalam arah yang berbeda untuk menghasilkan medan magnet yang bergerak. Karena alasan inilah, motor stepper bipolar tidak bisa dikendalikan langsung dengan arus DC biasa, diperlukan driver khusus

2. HMC5883L



Gambar 2. 3 Sensor HMC5883L

Sensor HMC5883L merupakan salah satu jenis sensor magnetometer tiga sumbul (3-*Axis-magnetometer)* yang dirancang oleh perusahaan Honeywel. Sensor ini memiliki fungsi utama untuk mengukur intensitas medan magnet di sekitarnya, sehingga sering digunakan dalam berbagai aplikasi navigasi digital, seperti kompas elektronik, sistem pemosisian global (GPS), robotika, serta perangkat mobile dan weareble. Sensor ini bekerja berdasarkan prinsip magnetoresistif, yaitu perubahan resistansi material akibat adanya pengaruh medan magnet eksternal. Dengan kemampuan untuk mendeteksi arah utara magnetik berdasarkan medan magnet bumi, sensor ini sangat ideal digunakan dalam sistem orientasi dan navigasi.

Sensor HMC5883L bekerja dengan mengukur medan magnet pada tiga sumbu koordinat, yaitu sumbu X, Y, Z. Hasil pengukuran tersebut kemudian diolah untuk menentukan arah terhadap utara magnetik. Sensor ini menggunakan koneksi langsung dengan mikrokontroler seperti arduino, ESP23, atau Rasberry Pi melalui protokol 12C ini, sensor mengirimkan data dalam bentuk digital yang kemudian dapat diolah untuk menentukan arah atau orientasi perangkat.

Tabel 2. 2 Spesifikasi Sensor HMC5883L

NO	Data Teknis HMC5883L	
1	Tegangan operasi	2.16V – 3.6V
2	Protokol komunikasi	12C (Inte-Integrated Circuit)
3	Pin digital I/O	VCC,GND,SCL,SDA,XDA,XDL,AD0,INT
4	Dimensi	3 x 3 x 0.9 mm
5	Tipe sensor	Magnetometer 3-Axis Digital

3. Driver Motor Stepper

Driver Stepper motor merupakan salah satu jenis motor listrik yang bergerak secara bertahap atau per langkah (step) berdasarkan sinyal kendali tertentu. Setiap pulsa listrik yang dikirimkan ke motor akan menyebabkan poros motor berputar sejauh sudut tertentu. Untuk mengendalikan stepper motor secara efisien dan presisi, dibutuhkan suatu rangkaian pengendali yang disebut dengan driver stepper motor.

Driver stepper motor adalah sebuah rangkaian elektronik yang berfungsi sebagai jembatan antara mikrokontroler dengan stepper motor. Perangkat ini bertugas untuk mengatur arus dan tegangan yang diperlukan oleh motor, serta menentukan arah dan jumlah langkah berdasarkan sinyal logika dari mikrokontroler. Secara umum, driver ini menerima sinyal input berupa pulsa-pulsa digital (biasanya sinyal STEP dan DIR) dari kontroler, kemudian mengubahnya menjadi bentuk arus listrik yang sesuai dengan kebutuhan kumparan motor.

Fungsi utama dari driver stepper motor antara lain adalah membangkitkan urutan pulsa yang tepat agar rotor motor dapat bergerak sesuai arah dan jumlah langkah yang diinginkan, mengatur besar arus ke masing-masing fasa motor untuk menghindari overheat, serta memungkinkan pengaturan kecepatan dan mode operasi motor seperti full-step, half-step, atau microstepping. Beberapa driver juga dilengkapi dengan fitur-fitur pelindung seperti proteksi terhadap overcurrent, overtemperature, dan short-circuit untuk menjaga keamanan sistem.

Secara umum, driver stepper motor terdiri dari beberapa komponen penting, yaitu input logika (biasanya berupa pin STEP dan DIR), driver switching (berfungsi

sebagai saklar elektronik untuk mengatur arus ke kumparan motor), dan sistem kontrol arus (biasanya menggunakan metode chopping atau PWM untuk menjaga arus tetap stabil). Driver juga harus mampu menangani kebutuhan arus dan tegangan sesuai dengan spesifikasi motor, yang dapat bervariasi tergantung jenis dan ukuran motor stepper yang digunakan.

Driver stepper motor dapat diklasifikasikan ke dalam dua kategori utama berdasarkan cara kerjanya, yaitu unipolar dan bipolar. Driver unipolar biasanya lebih sederhana karena hanya mengalirkan arus dalam satu arah pada tiap kumparan, sementara driver bipolar lebih kompleks karena mengalirkan arus bolak-balik (dua arah) sehingga memerlukan pengaturan switching yang lebih rumit, namun menghasilkan torsi yang lebih besar.

Dalam penerapannya, driver stepper motor sangat penting dalam sistem yang memerlukan kontrol posisi yang akurat dan dapat diulang, seperti pada mesin CNC, printer 3D, robotika, kamera tracking system, dan alat ukur otomatis. Kombinasi antara driver, mikrokontroler, dan sensor dapat membentuk sistem kendali tertutup (closed loop) yang memungkinkan operasi yang lebih presisi dan responsif terhadap perubahan beban.



Gambar 2. 4 Driver ULN 2003

BAB III

METODE PENELITIAN

A. Jenis penelitian

Penelitian ini menggunakan metode research dan development. pada penelitian ini pencarian data tentang teori giroskopik dan pengembangan secara mekanis yang menunjukkan tahapan yang harus di ikuti untuk menghasilkan sebuah produk tertentu dan menguji ke efektivan produk tersebut.

B. Lokasi dan waktu

1. Lokasi

Lokasi penelitian ini dilakukan di Laboratorium Teknik Elektro Universitas Muhammadiyah Parepare.

2. Waktu

Untuk waktu penelitian dapat dilihat pada table dibawah ini:

Tabel 3. 1 Waktu Penelitian

No	Kegiatan	Waktu kegiatan											
		Bulan ke-1				Bulan ke-2				Bulan ke-3			
		1	2	3	4	1	2	3	4	1	2	3	4
1	Studi literature	■	■	■									
2	Perancangan perangkat keras				■	■	■						
3	Perancangan perangkat lunak							■	■	■			
4	Pembuatan laporan hasil penelitian										■	■	■

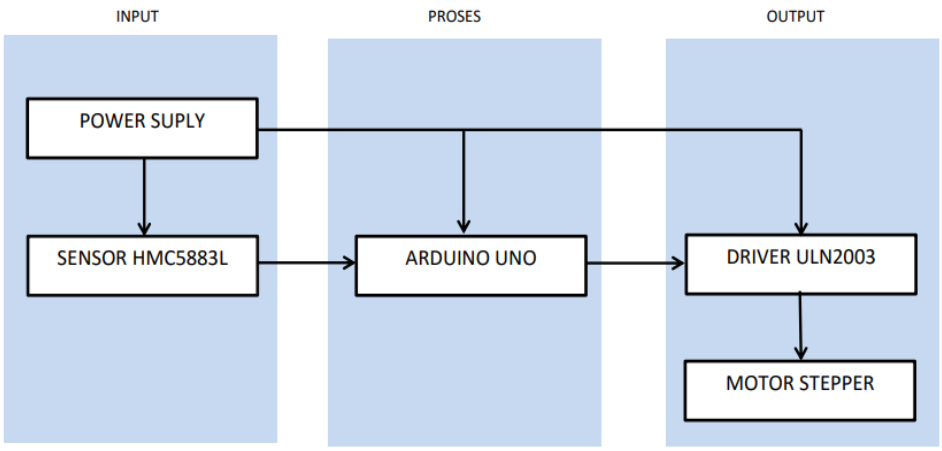
C. Alat Dan Bahan

Dalam pelaksanaan penelitian ini digunakan alat yang terdiri dari perangkat keras (Hardware), arduino uno, Sensor HMC5883L, motor Stepper, driver motor stepper, penelitian ini juga menggunakan perangkat lunak (software) yaitu untuk memprogram mikrokontroler.

D. Rancangan Penelitian

Setelah mengetahui kebutuhan system, dasar-dasar ilmu serta teknologi yang akan digunakan, maka dilakukan beberapa langkah-langkah berikut yaitu Pengujian sensor dan bahan lainnya serta pengambilan data sementara sebagai acuan dalam perakitan alat guna mempermudah dalam pengkalibrasian dan pemrogroman alat

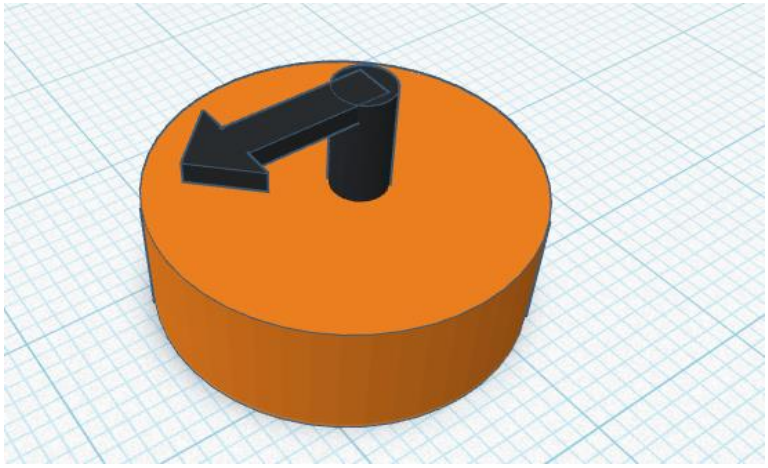
Berikut ini merupakan blok diagram sistem kendali dan stabilizator sudut/arah berbasis arduino.



Gambar 3. 1 Blok Diagram

Pada penelitian ini alat bekerja dimulai dari pembacaan sudut derajat yang diperoleh dari sensor kemudian sensor memberikan data pada mikrokontroler,

mikrokontroler sebagai tempat pengelolahan data akan memproses data kemudian memberikan output ke pada motor driver, lalu motor driver akan memberikan input kepada motor untuk menentukan arah putaran dan menggerakkan aktuator, kemudian sudut kemiringan kembali dibaca oleh sensor



Gambar 3. 2 Desain Konstruksi

E. Teknik pengumpulan data

Teknik pengumpulan data yang digunakan dalam penelitian ini adalah pelaksanaan test dan pengujian. Teknik pengumpulan data dengan melaksanakan beberapa test sebagai pengujian pada alat untuk mengetahui seberapa akurat sensor dan alat lainnya berfungsi untuk di kalibrasikan pada rangkaian yang nantinya akan di buat

Adapun pelaksanaan test yang dilakukan oleh penulis untuk tugas akhir ini meliputi :

1. Pengujian sensor HMC5883L
2. Pengujian Motor Stepper
3. Pengujian Motor driver

4. Pengujian motor DC dan dengan di hubungkan dengan sensor HMC5883L
5. Pengujian alat secara total

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

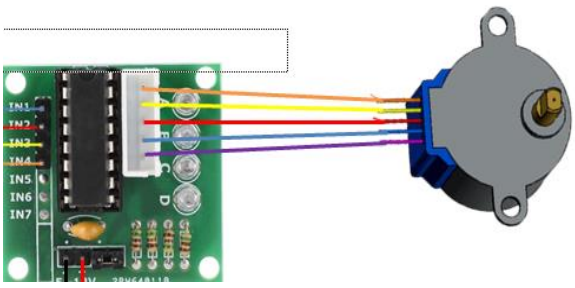
A. Perancangan Perangkat Keras (*hardware)*

Perancangan perangkat keras dilakukan sebagai bagian penting dalam penelitian ini, rancangan ini menjelaskan tentang bentuk dan komponen apa saja yang digunakan serta *wiring* komponen dari beberapa referensi agar dapat memberikan akurasi dalam pembacaan data atau penelitian. Hal ini dimaksudkan agar sistem kendali dan stabilizator arah/sudut berbasis arduino dapat berjalan berdasarkan deskripsi awal seperti yang direncanakan.

Adapun gambaran atau perancangan dari perangkat keras yang digunakan sebagai berikut :

1. Rangkaian Motor Stepper dan Driver motor stepper ULN2003

Rangkaian ini bertujuan untuk mengontrol atau mengendalikan arah dan langkah motor stepper. dengan adanya koneksi antar kedua komponen ini, diharapkan meningkatkan akurasi pergerakan pada motor agar tidak melebihi titik yang telah ditentukan, sebagai contoh dalam penelitian ini yakni arah kiblat. dengan ini diharapkan pergerakan motor dapat selaras dengan pembacaan sudut sensor HMC5883L.



Gambar 4. 1 Rangkaian Motor Sepper Dan Driver

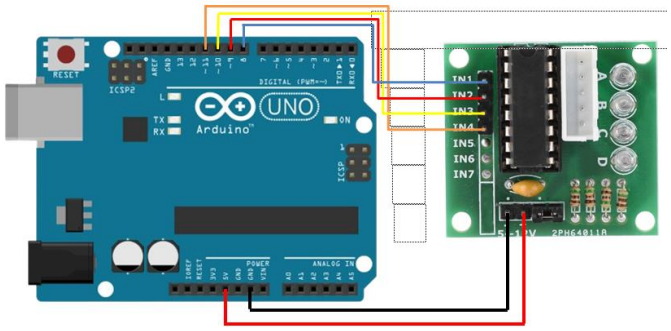
Berikut ini adalah keterangan dari pin yang terhubung antara motor driver dan motor stepper.

Tabel 4. 1 Koneksi Pin Driver Motor dan Motor Stepper

NO	WARNA KABEL	KETERANGAN
1	ORANGE	Sebagai Coil 1, penghubung titik awal kumparan 1
2	PINK	Sebagai Coil 2, penghubung titik akhir kumparan 1
3	KUNING	Sebagai Coil 3, Penghubung titik awal kumparan 2
4	BIRU	Sebagai Coil 4, Penghubung titik awal kumparan 2
5	MERAH	Kabel pusat, penghubung titik tengah dari kumparan

2. Rangkaian Motor driver dan Arduino

Pada Rangkaian ini bertujuan untuk memberikan sinyal perintah ke driver motor untuk menggerakkan motor berdasarkan data yang terbaca oleh sensor.



Gambar 4. 2 Rangkaian Motor driver

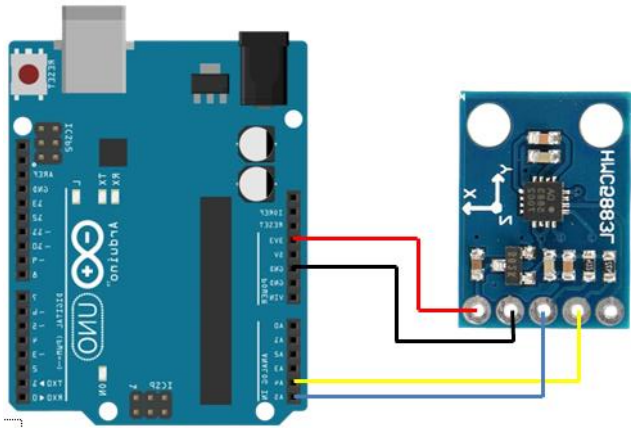
Berikut ini adalah keterangan dari pin yang terhubung pada motor driver

Tabel 4. 2 Koneksi pin motor driver dan arduino

NO	PIN DRIVER	PIN ARDUINO	KETERANGAN
1	IN1	8	Sebagai pin input mode LOW
2	IN2	9	Sebagai pin input mode LOW
3	IN3	10	Sebagai pin input mode HIGH
4	IN4	11	Sebagai pin input mode HIGH
5	POSITIF (+)	5V	Sebagai sumber tegangan driver dan motor
6	NEGATIF(-)	GND	Sebagai grounding driver dan motor

3. Rangkaian sensor HMC58833L dan Arduino

Pada rangkaian ini bertujuan untuk membaca dan menentukan sudut derajat melalui pembacaan medan magnet bumi yang dideteksi oleh sensor dan di konversi ke dalam derajat, yang kemudian diaktualisasikan oleh motor driver dan motor stepper sehingga mendapatkan sudut yang diinginkan, sebagai contoh dalam penelitian ini adalah arah kiblat yakni 292 derajat.



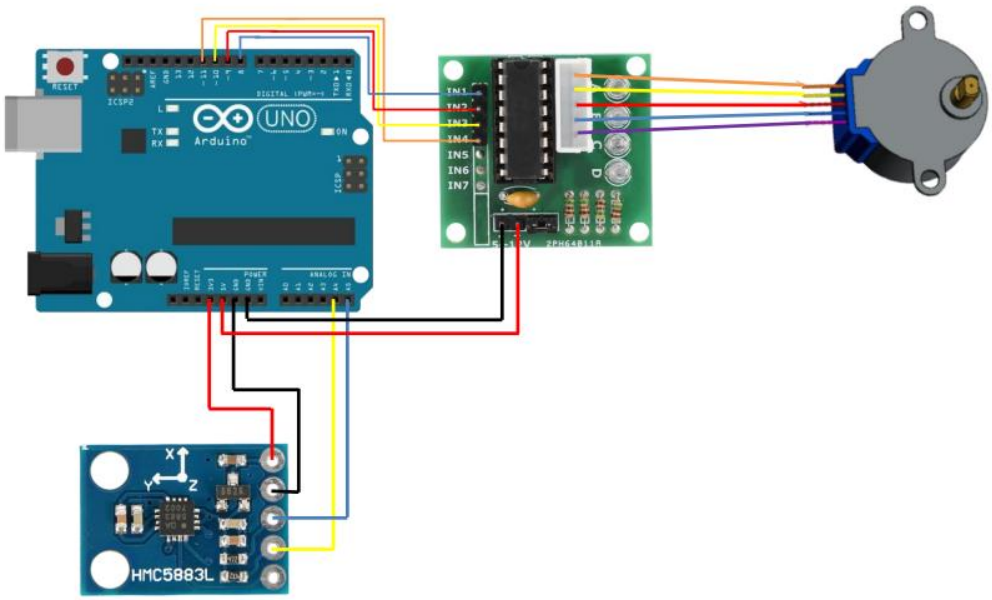
Gambar 4. 3 Rangkaian Sensor HMC5883L dan Arduino

Berikut ini adalah keterangan dari pin yang terhubung pada sensor

Tabel 4. 3 Koneksi pin sensor dan arduno

NO	PIN SENSOR	PIN ARDUINO	KETERANGAN
1	VCC	5V	Sebagai power suply sensor
2	GND	GND	Sebagai grounding sensor
3	SCL	A5	Pengatur singkronasi antara sensor dan arduino
4	SDA	A4	Sebagai pengirim dan penerima data

4. Rangkaian Sistem Keseluruhan



Gambar 4. 4 Rangkaian sistem keseluruhan

Berikut ini adalah rangkaian sistem secara keseluruhan, dengan menyinkronkan semua komponen melalui program dan pengkoneksian pin atau kabel maka terbentuklah sistem kendali dan stabiliizator sudut/arah berbasis arduino. Dimana data dari pembacaan sensor yang olah oleh arduino dan dikirim ke motor driver dan motor stepper untuk mengkoreksi arah secara otomatis ke arah kiblat

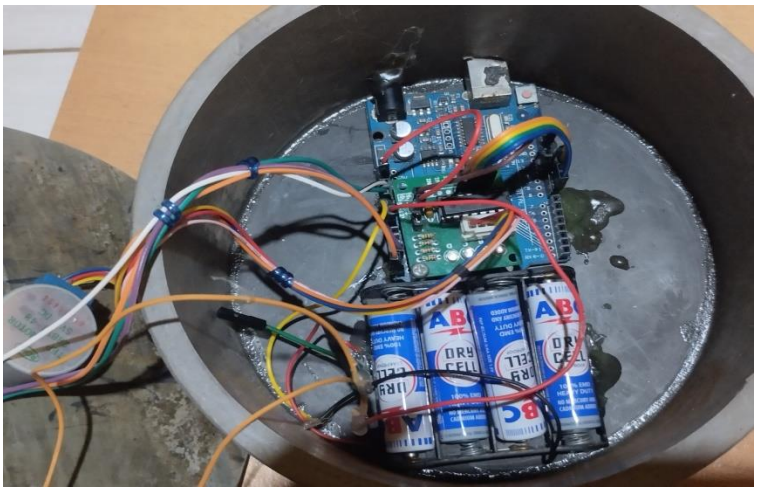
5. Desain Konstruksi

Dalam hal merealisasikan desain alat yang dibuat lebih awal maka dianggap perlu untuk memperhatikan desain konstruksi sebagai bentuk pertanggung jawaban dalam melakukan penelitian ini, dalam hal ini desain konstruksi dibuat semirip mungkin dari bentuk visual desain alat. Desain konstruksi juga sangat penting dalam menunjang keberhasilan dalam penelitian ini, dimana dipilih untuk

menggunakan desain yang berbentuk bulat menyerupai lingkaran dimana pada perhitungan penelitian ini juga merujuk pada hal tersebut dimana rotasi dari motor stepper untuk menentukan sudut adalah 360 derajat, hal itu setara dengan satu lingkaran penuh, penambahan arah panah pada bagian atas alat untuk memperjelas penandaan terhadap pembacaan sudut yang diliat secara kasat mata, hal ini juga dapat mempermudah dalam melihat seberapa besar simpangan pembacaan alat dari sudut yang ditentukan, yakni 292 derajat.



Gambar 4. 5 Desain Konstruksi



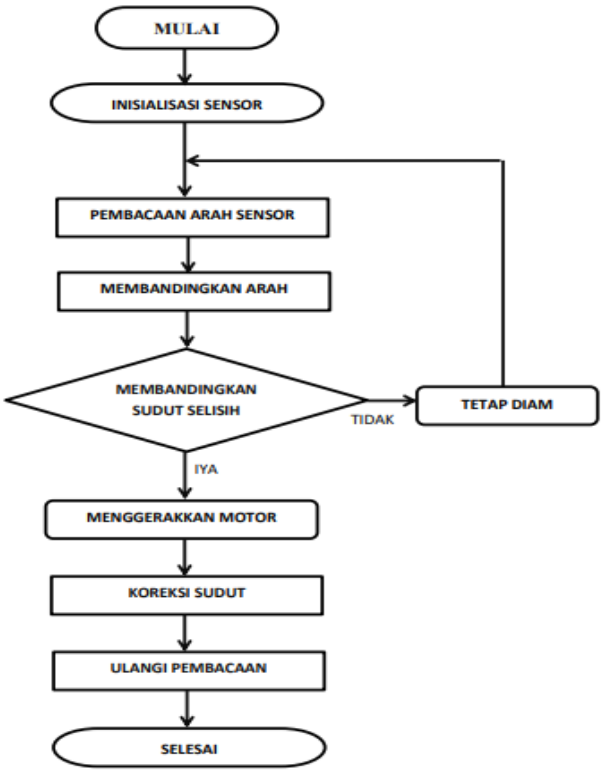
Gambar 4. 6 Rangkaian akhir

B. Perancangan Perangkat Lunak (*software)*

Perancangan perangkat lunak ini berfungsi untuk mengatur kinerja sistem tentang input dan output dari perangkat keras, seperti respon pergerakan atau *feedback* yang dibaca oleh sistem kemudian diolah kembali untuk mendapatkan hasil yang akurat. Dalam hal ini akan menjabarkan dua hal yakni alur pemrograman (*flowchart)* dan pemrograman arduino IDE.

1. Alur pemrograman (*flowchart)*

Alur pemrograman adalah diagram yang menggambarkan alur yang terjadi dalam sistem. Berikut gambaran alur pemrograman (*flowchart)*



Gambar 4. 7 Alur Pemrograman (*flowchart)*

Pada gambar 4.5 Alur Pemrograman (*flowchart)* menunjukan alur pemrograman yang awali dengan sistem, kemudian dilanju inisialisasi sensor. Yang

kemudian dilanjut pembacaan sensor dan perbandingan arah yang dibaca saat ini dan arah yang telah ditentukan, apabila ada nilai error atau kesalahan maka arduino akan memberikan perintah motor untuk bergerak dan mengoreksi kesalahan, apabila tidak ada maka motor tidak akan digerakkan, dan memulai kembali ke pembacaan arah sensor sampai adanya kesalahan yang terdeteksi oleh sistem.

2. Pemrograman

pada pemrograman Sistem Kendali dan Stabilisator sudut/arah Berbasis arduino ini menggunakan bahasa pemrograman *C* yang diprogram melalui aplikasi *Arduino IDE*. Berikut in adalah pemrograman yang digunakan pada penelitian ini

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_HMC5883_U.h>
#include <Stepper.h>
```

Pada awal pemrograman digunakan fungsi pemanggilan *Library* komponen yang digunakan, seperti yang tertera pada contoh diatas yakni, *Wire.h* yang berfungsi sebagai antarmuka komunikasi *I2C (Inter-Integrated Circuit),* kemudian *Adafruit_sensor.h* yang berfungsi sebagai antarmuka abstrak sensor dari adafruit yang menyediakan struktur data dan standar fungsi untuk berbagai sensor, *Adafruit_HMC885L_U.h* sebagai library khusus untuk mengakses sensor berdasarkan data (*Heading), Stepper.h* Digunakan untuk mengontrol motor stepper, yaitu motor bergerak perlahan dalam langkah-langkah kecil (distrik).

Selanjutnya adalah program konfigurasi yang digunakan untuk menentukan nilai atau data yang akan dijadikan acuan dalam penelitian ini, sebagai contoh

dalam penelitian ini adalah arah kiblat dengan nilai derajat yakni 292.

```
const int   STEPS_PER_REV   = 2048;
const float TARGET_AZIMUTH  = 292.0;
const float TOLERANCE       = 0.5;
const int   STEP_SPEED_RPM  = 10;

const int IN1 = 8;
const int IN2 = 9;
const int IN3 = 10;
const int IN4 = 11;
```

Berikut penjelasan dari potongan program diatas, dimana fungsi dari *const int STEP_PER_REV =2048* berfungsi untuk menentukan jumlah langkah (step) yang dibutuhkan motor stepper untuk melakukan satu putaran penuh (360 derajat). Selanjutnya *const float TARGET_AZIMUTH = 292.0* adalah fungsi untuk menentukan arah sudut tujuan (azimuth) dalam derajat, yaitu 292.0. *const float TOLERANCE = 0.5* berfungsi untuk menentukan kesalahan antara sudut aktual dan target. Jika sudut berada dalam 0.5 derajat tidak lebih maka dianggap sudah mencapai target sebagai ambang toleransi. *Const int = STEP_SPEED_RPM = 10* untuk menentukan kecepatan putar motor stepper, yaitu 10 putaran per menit (RPM). Empat *const int IN1,IN2,IN3,IN4* digunakan untuk menentukan pin digital pada arduino yang digunakan mengontrol motor stepper melalui driver. Setiap pin ini akan dikendalikan secara bergantian untuk mengatur arah dan kecepatan motor

```
// ======== Objek ========
Adafruit_HMC5883_Unified mag(12345);
Stepper stepper(STEPS_PER_REV, IN1, IN3, IN2, IN4);

int lastDirection = 0; // 1 atau -1, untuk menyimpan arah koreksi
terakhir
int i = 0;
```

Penjelasan dari potongan program diatas adalah *adafruit_HMC5883L_Unified mag (1234)* berfungsi untuk membuat objek bernama mag dalam library, yang digunakan untuk berkomunikasi dengan sensor magnetometer HMC5883L sedangkan angka (1234) adalah ID sensor yang digunakan oleh library untuk pelacakan. *Stepper stepper(STEP_PER_REV, IN1, IN3, IN2, IN4)* ini bertujuan untuk mengontrol motor stepper. Parameter STEP_PER_REV menentukan jumlah langkah per putaran penuh. *Int lastDirection = 0* berfungsi untuk menyimpan arah koreksi terakhir yang ditentukan oleh sistem. *Int i = 0* adalah variabel umum yang digunakan sebagai indeks pengulangan atau menghitung langkah dalam bagian program selanjutnya

```
void setup()
{
  Serial.begin(9600);
  if (!mag.begin())
    {
      Serial.println("ERROR: HMC5883L tidak terdeteksi!");
      while (true);
    }
  stepper.setSpeed(10);
}
```

```
void loop()
{
  //baca sudut sensor ------------------------------------
  sensors_event_t event;
  mag.getEvent(&event);
  float mx = event.magnetic.x;
  float my = event.magnetic.y;
  float heading = atan2(my, mx);
  if (heading < 0) heading += 2.0 * PI;
  float headingDeg = heading * 180.0 / PI;
```

Penjelasan fungsi dari program diatas yakni, *void setup()* adalah bagian yang akan dijalankan sekali saja saat arduino pertama kali dinyalakan atau direset, berfungsi untuk inisialisasi awal seperti komunikasi setting, sensor maupun perangkat lainnya. *Serial.begin(9600)* berfungsi menginisialisasi komunikasi serial antara arduino di komputer dengan kecepatan 9600 baud. Digunakan untuk mengirim data atau pesan debug ke serial monitor. *If (!mag.begin())* dan turunannya berfungsi untuk mengembalikan *true* jika sensor terdeteksi, jika tidak berhasil *(false)* maka akan mencetak pesan kesalahan ke serial monitor, dan masuk ke *while (true),* yang menghentikan program looping tanpa akhir agar pengguna tahu ada kesalahan pada sensor, *stepper .setSpeed(10)* mengatur kecepatan motor stepper sebesar 10 RPM. Ini memengaruhi seberapa cepat seberapa cepat motor bergerak saat dipanggil fungsi step

Potongan program diatas menjelaskan tentang, *void loop()* digunakan untuk menjalankan program terus-menerus selama arduino atau board dijalankan, ini adalah tempat utama untuk logika yang terus menerus memantau, mengukur atau mengendalikan perangkat. *Sensor_event_t event* adalah struktur data untuk menyimpan hasil pembacaan sensor, *mag_getEvent(&event)* untuk memerintahkan

sensor magnet HMC5883L untuk mengisi data *event* dengan pembacaan medan magnet saat ini (x, y, z), *float mx = event.magnetic.x, float my = event.magnetic.y* berfungsi untuk menyimpan nilai medan magnet sumbu X dan Y dari sensor ke dalam variabel mx dan my, nilainya digunakan untuk menghitung arah kompas (azimuth), *float heading = atan2(my,mx)* berfungsi menghitung sudut (heading) berdasarkan nilai x dan y menggunakan fungsi atan2(), fungsi atan2 (y, x) menghasikan sudut dalam radian, dengan hasil (-)*pi* sampai (+)*pi* yaitu (-180 derajat sampai 180 derajat) yang menunjukkan arah horizontal sensor terhadap medan magnet bumi, *if (heading< 0) heading += 2.0 *PI* berfungsi jika sudut negatif (misalnya -90 derajat) maka ditambahkan 2pi agar nilainya selalu positif(kisaran 0 hingga 2pi radian atau 0-360 derajat) *float headingDeg = heading*180.0 / PI* untuk mengubah nilai sudut radian ke derajat dengan headingDeg sekarang berisi arah kompas aktual dalam derajat (0-360 derajat)

```
//hitung error----- -------------------------------------
  float error = TARGET_AZIMUTH - headingDeg;
  if (error > 180.0) error -= 360.0;
  else if (error < -180.0) error += 360.0;
```

Bagian dari program ini *float error = TARGET_AZIMUTH – headingDeg* bertugas untuk menghitung dari seberapa jauh arah sensor melenceng dari target serta *if (error > 180.0) error -= 360.0, else if (error > 180.0) error += 360.0* menyesuaikan nilai error agar menunjukkan arah dan jarak putaran tercepat, yaitu dengan kisaran -180 gingga 180 derajat. Dengan logika ini program dapat mengatur motor berputar ke arah paling efisien untuk mencapai target azimuth

```
/tampilkan data-------------------------------------
      Serial.print(i);
      Serial.print("  ");
      Serial.print(headingDeg);
      Serial.print("  ");
      Serial.println(error);
    } else
  if (abs(error) > TOLERANCE)
    {
      dir = (error > 0) ? 1 : -1;
      lastDirection = dir;
      stepper.step(dir);
      i = i + 1;                      //hitung step ---------

  //putar motor----- -------------------------------------
  int dir;
  if (abs(error) > 10 * TOLERANCE)
    {
      dir = (error > 0) ? 15 : -15;
      lastDirection = dir;
      stepper.step(dir);
      i = i + 1;
```

Penjelasan dari potongan program ini bertugas untuk memeriksa kesalahan arah (azimuth) apakah cukup besar jika ya, menggerakkan motor stepper ke arah yang benar untuk mengurangi error kemudian dilanjut dengan merekam arah koreksi dan jumlah langkah yang sudah dilakukan

Fungsi dari potongan program ini adalah menampilkan data di serial monitor, seperti *Serial.print(i)* jumlah langkah, *Serial.print (headingDeg)* arah kompas, *Serial.pritln(error)* kesalahan atau error arah. ini memberikan umpan balik visual tentang proses koreksi posisi motor stepper dalam serial monitor

```
//tampilkan data--------------------------------------
      Serial.print(i);
      Serial.print("  ");
      Serial.print(headingDeg);
      Serial.print("  ");
      Serial.println(error);
    }

  // Reset step jika sudah sejajar
  if (abs(error) <= TOLERANCE) {
    i = 0;
    stepper.step(0);
  }
  delay(100);  // bisa dihilangkan
```

Bagian kode ini berfungsi menampilkan data debug seperti, *serial.print(i)* jumlah langkah, *serial.primt (“ “)* memberikan spasi untuk memisahkan data, *serial.print(headingDeg)* arah sensor dan error yang sedang dihitung serta *if (abs(error) <=TOLERANCE)* dan *stepper.step(0)* menghentikan motor jika error sudah sangat kecil (dalam batas toleransi), *i =0* motor berhenti bergerak dan penghitung langkah disetel ulang ke 0, *delay(100)* memberikan sedikit jeda waktu untuk kestabilan eksekusi. Ini merupakan bagian akhir dari proses koreksi, dimana jika ada kesalahan sudah minimal, sistem berhenti dan bersiap untuk perhitungan selanjutnya

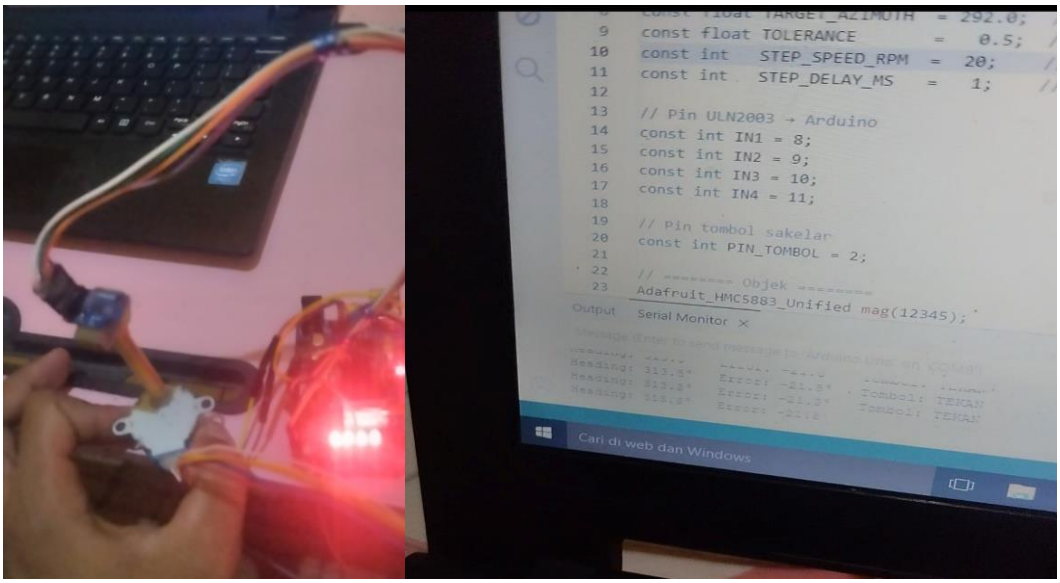
C. PENGUJIAN SISTEM DAN PENGAMBILAN DATA

Pada bagian ini membahas hasil pengujian dan analisis dari sistem kendali dan stabilisator sudut/arah berbasis arduino, guna mengetahui proses kinerja alat dan kendala sistem yang sedang dirancang. Pengujian dan analisis ini dilakukan berdasarkan contoh kasus yang diangkat pada penelitian ini yakni sudut arah kiblat

(azimuth) 292 derajat, berikut hasil pengujian yang dilakukan dalam beberapa kali percobaan

1. Pengujian sistem

Pengujian komponen ini dilakukan dengan cara merangkai komponen dengan rangkaian sementara, dimana ini bersifat hanya sebagai pengujian tingkat keakuratan pembacaan sensor dan gerak komponen.



Gambar 4. 8 Pengujian Komponen

Pada gambar 4.6 teknik pengujian yang dilakukan adalah untuk mengetahui respons sensor dan gerak motor apakah selaras, dengan cara memperhatikan data yang terbaca pada serial monitor. Berdasarkan hasil data yang diperoleh dapat dikatakan pembacaan sensor dan gerak motor selaras dikarenakan semakin jauh perbedaan sudut maka semakin besar kesalahan yang dideteksi motor, begitu pun sebaliknya sampai pada titik yang ditentukan yakni 292 derajat.

2. Pengujian dan pengambilan data

Pada pengujian dan pengambilan data ini dilakukan dengan menguji sistem secara keseluruhan berdasarkan simpangan terjauh dari sudut yang ditentukan dan mencatat hasil dari pembacaan sudut sensor dan langkah motor yang ditampilkan pada serial monitor. Pada pengujian ini menggunakan dua tipe pengujian yakni searah jarum jam *(Clockwise)* dan berlawanan dengan arah jarum jam *(Counter-Clockwise)*. Beriku adalah data yang diambil dari dua tipe pengujian yang disebutkan sebelumnya.

Tabel 4. 4 Data *Clockwise*

STEP	SUDUT	SIMPANGAN
1	120.42	171.58
2	123.44	168.56
3	127.18	164.82
4	131.47	160.53
5	135.00	157.00
6	137.12	154.88
7	139.40	152.60
8	142.80	149.20
9	146.48	145.52
10	149.52	142.48
11	152.54	139.46
12	156.67	135.33
13	160.23	131.77
14	163.17	128.83
15	165.83	126.17
16	169.44	122.56
17	172.35	119.65
18	175.98	116.02
19	178.44	113.56
20	181.34	110.66
21	184.57	107.43
22	187.13	104.87
23	189.83	102.17
24	193.18	98.82
27	200.63	91.37

Bersambung

Lanjutan

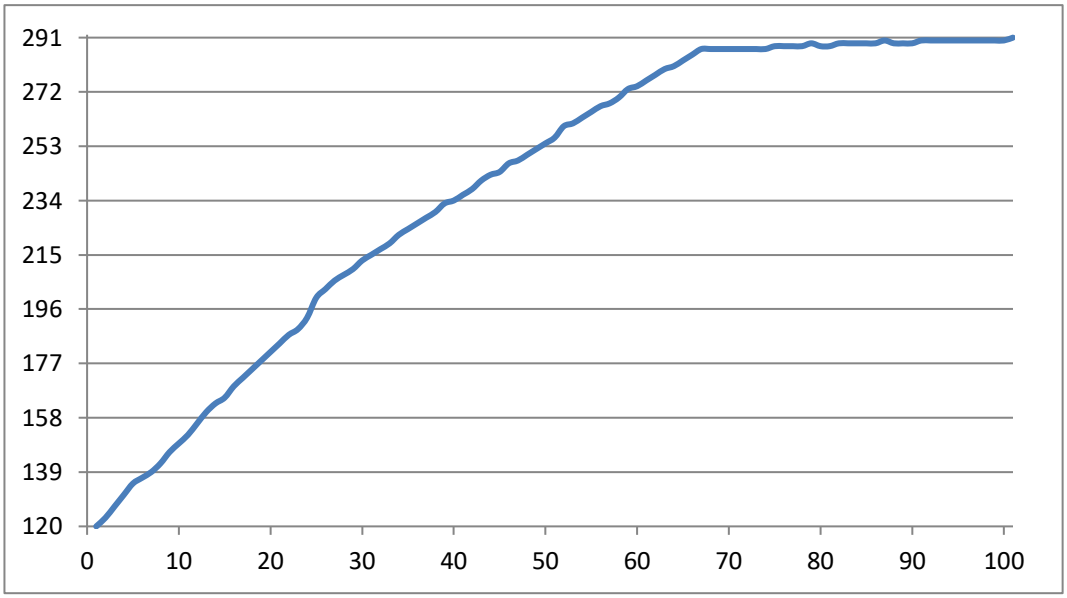
28	203.20	88.80
29	206.06	85.94
30	208.49	83.51
31	210.80	81.20
32	213.17	78.83
33	215.49	76.51
34	217.31	74.69
35	219.46	72.54
36	222.14	69.86
37	224.65	67.35
38	226.22	65.78
39	228.51	63.49
40	230.54	61.46
41	233.11	58.89
42	234.86	57.14
43	236.69	55.31
44	238.82	53.18
45	241.18	50.82
46	243.19	48.81
47	244.72	47.28
48	247.10	44.90
49	248.84	43.16
50	250.92	41.08
51	252.44	39.56
52	254.63	37.37
53	256.95	35.05
54	260.10	31.90
55	261.64	30.36
56	263.87	28.13
57	265.72	26.28
58	267.36	24.64
59	268.79	23.21
60	270.90	21.10
61	273.01	18.99
62	274.62	17.38
63	276.13	15.87
64	278.23	13.77
65	280.31	11.69
66	281.82	10.18
67	283.29	8.71
68	285.56	6.44
69	287.32	4.68

Bersambung

Lanjutan

70	287.26	4.74
71	287.38	4.62
72	287.64	4.36
73	287.71	4.29
74	287.96	4.04
75	287.70	4.30
76	287.76	4.24
77	288.34	3.66
78	288.47	3.53
79	288.28	3.72
80	288.85	3.15
81	289.10	2.90
82	288.66	3.34
83	288.82	3.18
84	289.01	2.99
85	289.36	2.64
86	289.26	2.74
87	289.10	2.90
88	289.48	2.52
89	290.16	1.84
90	289.84	2.16
91	289.99	2.01
92	290.33	1.67
94	290.38	1.62
95	290.19	1.81
96	290.88	1.12
97	290.92	1.08
98	290.96	1.04
99	290.54	1.46
100	290.80	1.20
101	291.53	0.47

Pada tabel 3. data *Clockwise* dapat dilihat bahwa sudut awal adalah 120.42 derajat dan Simpangan yang terbaca adalah 171.58 derajat, juga didapat dilihat untuk medapatkan hasil yang di inginkan motor stepper perlu mengambil langkah sebesar 101 langkah. untuk langkah yang diambil motor akan semakin mengecil apabila mendekati sudut yang ditentukan yakni 292 derajat. seperti pada yang diperlihatkan pada tabel 4.4.



Grafik 1. Grafik *Clockwise*

Berdasarkan Tabel 4. Data *Clockwise* juga dapat dilihat bahwa sudut awal adalah 108.50 dejat dan error yang terbaca adalah 176.50 derajat, dan motor stepper membutuhkan langkah sebanyak 94 kali untuk mencapai sudut 292 derajat, di mana dapat dilihat pada kurva grafik semakin besar simpangan yang terbaca maka semakin besar langkah yang diambil motor stepper dan langkah akan mengecil bila simpangan sudah mendekati sudut yang ditentukan. atau sesuai dengan yang ingin dicapai pada penelitian ini.

Tabel 4. 5 Data *Counter-Clockwise*

STEP	SUDUT	SIMPANGAN
1	108.50	-176.50
2	104.83	-172.83
3	100.44	-168.44
4	95.75	-163.75
5	93.71	-161.71
6	89.57	-157.57
7	86.94	-154.94
8	84.09	-152.09
9	80.96	-148.96

Bersambung

Lanjutan

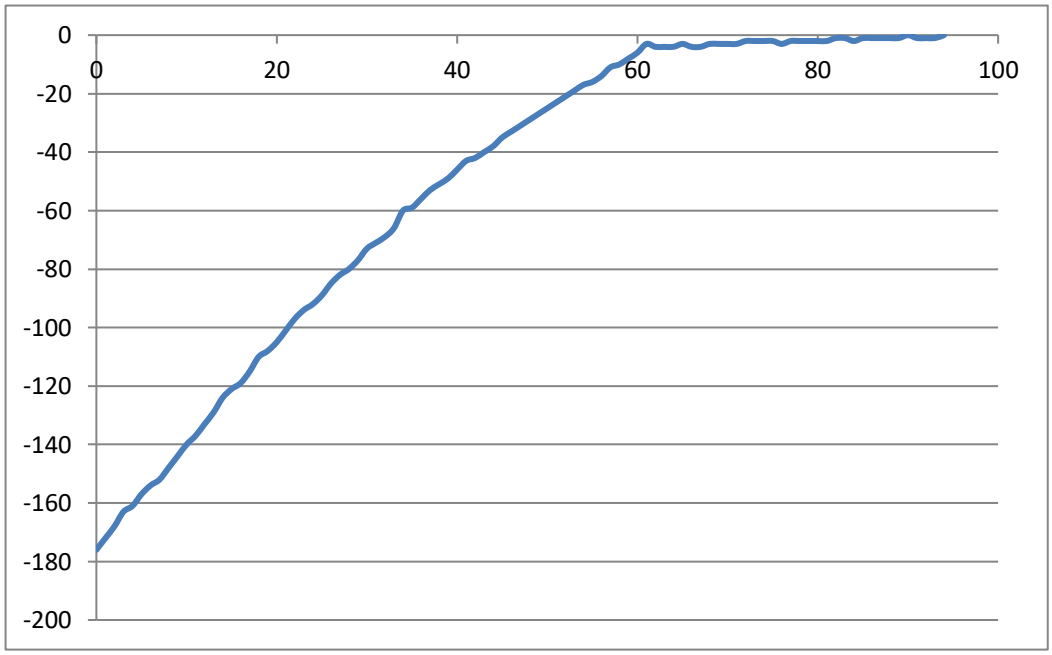
10	76.89	-144.89
11	72.03	-140.03
12	69.02	-137.02
13	65.74	-133.74
14	61.23	-129.23
15	56.76	-124.76
16	53.97	-121.97
17	51.25	-119.25
18	47.45	-115.45
19	42.70	-110.70
20	40.12	-108.12
21	37.21	-105.21
22	33.49	-101.49
23	29.19	-97.19
24	26.64	-94.64
25	24.87	-92.87
26	21.15	-89.15
27	17.14	-85.14
28	14.75	-82.75
29	12.65	-80.65
30	9.18	-77.18
31	5.24	-73.24
32	3.75	-71.75
33	1.63	-69.63
34	358.27	-66.27
35	354.90	-62.90
36	352.98	-60.98
37	351.13	-59.13
38	348.37	-56.37
39	345.16	-53.16
40	343.34	-51.34
41	341.44	-49.44
42	338.97	-46.97
43	335.87	-43.87
44	334.04	-42.04
45	332.56	-40.56
46	330.28	-38.28
47	327.25	-35.25
48	325.54	-33.54
49	323.94	-31.94
50	321.80	-29.80
51	319.19	-27.19

Bersambung

Lanjutan

52	317.12	-25.12
53	315.78	-23.78
54	313.55	-21.55
55	311.08	-19.08
56	309.85	-17.85
57	308.35	-16.35
58	306.04	-14.04
59	303.66	-11.66
60	302.08	-10.08
61	300.61	-8.61
62	298.39	-6.39
63	295.89	-3.89
64	296.39	-4.39
65	296.66	-4.66
66	296.11	-4.11
67	295.58	-3.58
68	296.11	-4.11
69	296.25	-4.25
70	295.62	-3.62
71	295.30	-3.30
72	295.48	-3.48
73	295.48	-3.48
74	294.98	-2.98
75	294.68	-2.68
76	294.77	-2.77
77	295.30	-3.30
78	294.49	-2.49
79	294.00	-2.00
80	294.41	-2.41
81	294.67	-2.67
82	294.03	-2.03
83	293.83	-1.83
84	293.96	-1.96
85	294.28	-2.28
86	293.58	-1.58
87	293.47	-1.47
88	293.51	-1.51
89	293.58	-1.58
90	293.05	-1.05
91	292.62	-0.62
92	293.15	-1.15
93	293.24	-1.24
94	292.62	-0.62

Berdasarkan Tabel 4.5 juga dapat dilihat bahwa sudut awal adalah 108.50 dejat dan error yang terbaca adalah 176.50 derajat, dan motor stepper membutuhkan langkah sebanyak 94 kali untuk mencapai sudut 292 derajat atau sesuai dengan yang ingin dicapai pada penelitian ini.



Grafik 2. Grafik *Counter-Clockwise*

Grafik data diatas dapat dilihat bahwa pergerakan atau langkah motor stepper semakin mengecil apabila akan mendekati titik yang telah ditentukan dimana langkah awal dimulai dari sudut -176 derajat *half circle* dan diakhiri di titik 0 pada langkah ke 94, dimana titik nol adalah arah kiblat 292 derajat

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Setelah melakukan tahap perancangan dan pembuatan sistem kendali dan stabilisator sudut/arah berbasis arduinno kemudian dilanjut dengan tahap pengujian dan analisis maka dapat disimpulkan :

1. Dari dua data yang diambil dapat dilihat bahwa hasil akhir dari pergerakan atau langkah motor berhenti pada titik 292 yang divisualisasikan dengan nilai 0 pada grafik
2. Dari grafik data dapat disimpulkan bahwa performa sistem berusaha untuk tetap menstabilkan pergerakan dengan metode semakin besar sudut kesalahan atau simpangan, maka semakin besar langkah yang diambil motor, Begitu pun sebaliknya semakin kecil sudut kesalahan maka semakin kecil pula langkah yang diambil motor.
3. sistem bisa diaplikasikan dengan sangat fleksibel dalam berbagai konteks penunjuk arah dengan cara mengubah sudut yang diinginkan

B. Saran

Dari hasil laporan akhir ini masih terdapat banyak kekurangan dan juga memiliki potensi untuk pengembangan lebih lanjut. Oleh karena itu penulis merasa perlu untuk memberikan saran sebagai berikut :

1. Untuk pengembangan Hardware selanjutnya alat ini dapat ditambahkan LCD dan tombol setting untuk melihat data pembacaan sensor secara realtime untuk meningkatkan dan mempermudah dalam pengambilan dan penentuan data
2. Untuk pengembangan software, perlu menambahkan beberapa filter dalam pengaplikasian atau pergerakan motor agar lebih halus lagi, guna meningkatkan akurasi pembacaan data

DAFTAR PUSTAKA

Arduino. (n.d.). Arduino Uno Rev3 Technical Specification. Diakses dari https://store.arduino.cc/arduino-uno-rev3

Adafruit. (n.d.). Adafruit HMC5883L Breakout – Triple-Axis Magnetometer Compass Sensor. Diakses dari https://www.adafruit.com/product/1746

Cambou, B. (2018). "Magnetometer Calibration and Compensation Techniques," IEEE Sensors Journal, 18(9), 3605–3614.

Dorf, R. C., & Bishop, R. H. (2011). Modern Control Systems (12th ed.). Prentice Hall

Franklin, G. F., Powell, J. D., & Emami-Naeini, A. (2014). Feedback Control of Dynamic Systems (7th ed.). Pearson.

Galli, J. A., & Schafer, W. (2019). Systems Analysis and Design (11th ed.). Cengage Learning

Honeywell. (2013). HMC5883L 3-Axis Digital Compass IC Datasheet. Honeywell International Inc.

STMicroelectronics. (n.d.). ULN2003A: Seven Darlington Arrays Datasheet. Diakses dari https://www.st.com/en/interfaces-and-transceivers/uln2003.html

StepperWorld. (n.d.). Bipolar Stepper Motor and Control Tutorial. Diakses dari https://www.stepperworld.com/tutorials/pgbipolartutorial.htm

Texas Instruments. (n.d.). ULN2003A: High-Voltage, High-Current Darlington Transistor Arrays Datasheet. Diakses dari https://www.ti.com/product/ULN2003A

Wikipedia. (n.d.). I²C. Diakses dari https://en.wikipedia.org/wiki/I%C2%B2C
Phillips Semiconductor. (1998). The I²C-bus Specification, Version 2.1. NXP Semiconductors.

42 Bots. (2014). Bipolar Stepper Motor Control with Arduino and an H-Bridge. Diakses dari https://42bots.com/tutorials/bipolar-stepper-motor-control-with-arduino-and-an-h-bridge