

ROBOT NAVIGATION CONTROL SYSTEM USING HMC5883L

Asti Riani Putri
Department of Information Technology
Education
STKIP PGRI Tulungagung
Tulungagung
East Java, Indonesia
asti@stkipgritulungagung.ac.id

Putri Nurrahayu
Department of Information Technology
Universitas Islam Kediri
Kediri
East Java, Indonesia
Putri_nur_rahayu@yahoo.co.id

Anwar anas
Department of Information Technology
Education
STKIP PGRI Tulungagung
Tulungagung
East Java, Indonesia
anwaranaz0@gmail.com

Abstract— *In the development of technology, the increasingly rapid world of robotics is also very much needed in helping everyday life, for example to move goods, technology that is so rapid requires the world of education to carry out development in various fields of science. Where the nation's next generation enhances science and technology so as not to be left behind by other nations, this study was conducted in order to be able to obtain robotic navigation according to their needs, the purpose of this study is to make robots with autonomous navigation and not affected by temperature or obstacles in front of it because it is equipped with an ultrasonic sensor that functions as a proximity sensor so that it does not affect the obstacles in front while for HMC5883L it is used to determine the angle corresponding to the object, the method used in programming robots still uses logic control programs and combined with the formula used in sensor measurements HMC5883L. The results of the research conducted are that the robot can move according to predetermined data input.*

Keywords—navigation, HMC5883L, Robot

I. INTRODUCTION

Along the development of the times, the increasingly rapid flow of globalization and the rapid advances in science and technology require the world of education to develop in various fields of science. Where the nation's next generation enhances science and technology so as not to be left behind by other nations.

Not to forget, the world of robotics is growing rapidly as the times develop. Many robot innovations have been created such as Special Application Robots, such as underwater robots and military robots. Besides that robots are used for daily activities, to replace the role of humans who are able to do work without feeling tired. Not only that, but also included in robot contests as a place to pour creativity. In addition to the world of robotics, microcontrollers also play a lot in the world of automation.

In the world of robotics there are many innovations that are produced such as legged and wheeled robots that are tailored to their needs, for the development of wheeled robots which were originally found on wheeled line follower robots this time will be changed. the right direction and angle that will be used as a wheeled soccer robot. Compass sensor or HMC5883L is used to adjust the position that is right for wheeled robots.

As a system setting method as stated by [1] that setting the position of the robot using the firefly algorithm to obtain an autonomous robot navigation. For robot navigation settings can also be determined by generalizing computation graphs this method can increase robot movement in the environment [2]. In the testing, later it will use a logic control program that controls the PG 36 motor and HMC5883L sensors to find the coordinate points that are in accordance with the location of the object before using Artificial Intelligence (AI).

II. BACKGROUND

A. Progame logic control(PLC)

PLC is defined as a digital electronic device with memory that can be programmed to store instruction instructions that perform specific function functions such as: logic, sequences, timing, counting and arithmetic, and to control an industrial machine or industrial process as desired. This PLC is designed to replace a sequential relay circuit in a control system. This PLC has a programming language that is easy to understand and can be operated if the program has been made using software that matches the type of PLC used. This tool works based on existing inputs and depends on the situation at a certain time which will then ON or OFF the outputs. Logic 1 shows that conditions are expected to be fulfilled while logic 0 means the conditions expected are not fulfilled. PLC can also be applied to controlling systems that have a lot of output. In the PLC there are parts that consist of a CPU (Central Processing Unit) which is the main part and is the brain of a PLC. This CPU functions to retrieve instructions from memory then encodes and executes those instructions. Memory that serves to store program data or instructions and data entered into the PLC. And Power Suply is used to provide voltage to the PLC of 24 V [3].

Based on its name, the PLC concept is as follows:

1. Programmable, demonstrates the ability in memory to store programs that have been created which can easily be changed function or utility.
2. Logic, shows the ability to process input in arithmetic and logic (ALU), which is to perform

- operations comparing, summing, multiplying, dividing, subtracting, negating, AND, OR, and etc.
3. Controller, shows the ability to control and regulate the process so as to produce the desired output.

This PLC is designed to replace a sequential relay circuit in a control system. Besides being programmable, this tool can also be controlled, and operated by people who do not have knowledge in the field of computer operation specifically. This PLC has a programming language that is easy to understand and can be operated if the program that has been made using software that matches the type of PLC used has been included. This tool works based on existing inputs and depends on the situation at a certain time which will then -ON or OFF outputs. 1 indicates that the expected condition is fulfilled while 0 means the expected condition is not fulfilled. PLC can also be applied to controlling systems that have a lot of output.

B. Navigation Robot

Mobile robots, a term often used to refer to a robot that has exploratory abilities. No matter whether the robot moves using wheels, legs, or fans to swim or even fly, then this kind of robot is included in the mobile robot category. In order not to stray, the robot must be equipped with a navigation system that can provide directional information properly. So that the robot can decide exactly which direction it should move to reach the desired location.

The navigation system is quite good, effective, easy to use and cheap, with a digital compass. Many types of digital compasses are produced specifically for robotics, one of which is very popular is the CMPS03 Magnetic Compass made by Devantech Ltd. The 4 x 4 cm CMPS03 uses a Philips KMZ51 magnetic field sensor that is sensitive enough to detect the earth's magnetic field.

This digital compass only requires a voltage supply of 5 Vdc with a current consumption of 15mA. At CMPS03, the wind direction is divided into degrees, namely: North (00), East (900), South (1800) and West (2700).

Error! Bookmark not defined. There are two ways to get direction information from this digital compass module, namely by reading the PWM signal (Pulse Width Modulation) on pin 4 or by reading the I2C data interface on pins 2 and 3. The PWM signal is a signal modulated pulse width. At CMPS03, the positive pulse width represents the direction angle. The pulse width varies between 1mS (00) to 36.99mS (359.90). In other words the pulse width changes by 100uS every degree. The signal will be low for 65mS between pulses, so the total period is 65mS + positive pulse width (between 66mS to 102mS). The credit is generated by a 16 bit timer inside the processor, which provides a 1uS resolution.

In addition to PWM, CMPS03 is also equipped with an I2C interface that can be used to read direction data in

the form of serial data. In 8 bit mode, the north direction is indicated by data 255 with a resolution of 1,40625 degrees / bit. In 16 bit mode, the north direction is indicated by data 65535 so that the resolution becomes 0.0055 degrees / bit.

Digital data from CMPS03 can be read directly by the robot controller to find out its current position, for example facing east (data 64), then the robot faces north, then the robot must rotate to the left until CMPS03 data becomes 255. Thus, with a very simple algorithm can already be used to form a good robot navigation system [4].

C.HMC5883l

DT-Sense 3 Axis Compass is a magnetic field sensor module that uses IC hmc5883l. IC hmc5883l is a chip designed to read magnetic fields that are suitable for directional applications and magnetometry. Examples of application of this module include sensors on smartphones, netbooks, automatic navigation systems, and other applications that require magnetic field measurements.

The specifications of the HMC5883l IC are as follows:

1. It requires a 3.3 VDC power supply with low current consumption (up to 100uA).
2. It has a magnetic sensor with 3 axis magnetoresistive type.
3. It has a range of magnetic field readings up to ± 8 Gauss with a resolution of 5 milligauss.
4. It has compass accuracy up to 1° to 2° .
5. Maximum output speed of data up to 160 Hz (Single Measurement Mode).
6. Maximum data output speed of 0.75 Hz.d. 75 Hz (Continuous Measurement Mode).
7. Using an I2C interface that can be connected with various types of microcontroller systems.
8. It has a small and compact module dimension so that it is easily placed in various applications.



Figure 2. 1 DT-Sense 3 Axis Compass [5]

D.PG-36



Figure 2.2 Motor PG 36

PG 36 Motor Specifications:

1. Built-in Planetary **Gearbox PG36**.
2. Vs: DC **24V**.
3. Current: 2.5A.
4. Power: **40W**.
5. Speed: 440 rpm.
6. **Torque**: 20 kgfcm.
7. Built-in encoder 2 channel 500 ppr.
8. **Dimensions** of motor body: length 11 cm x diameter 3.6 cm.

E.Arduino 2560

Arduino Mega 2560 is a microcontroller board based on ATmega2560 (datasheet). It has 54 digital input / output pins (of which 14 can be used as PWM output), 16 analog inputs, 4 UART (hardware serial ports), 16 MHz crystal oscillator, USB connection, power jack, ICSP header, and reset button. It contains everything needed to support a microcontroller; Simply connect to the computer with a USB cable or turn it on with the AC-to DC adapter or battery to start. Mega is compatible with most shields designed for Arduino Duemilanove or Diecimila [6].

F.HC-SR04

HC-SR04 ultrasonic sensors use sonar to determine the distance to objects such as bats do when flying. This sensor has excellent detection capabilities with high accuracy and stable readings in an easy-to-use package. From 2cm - 400cm or 1 inch to 4 meters. the operation is not affected by sunlight or black materials such as the Sharp GP infrared distance meter (although the acoustics of soft materials such as cloth can be difficult to detect

III. METHODOLOGY

3.1 Research flow chart

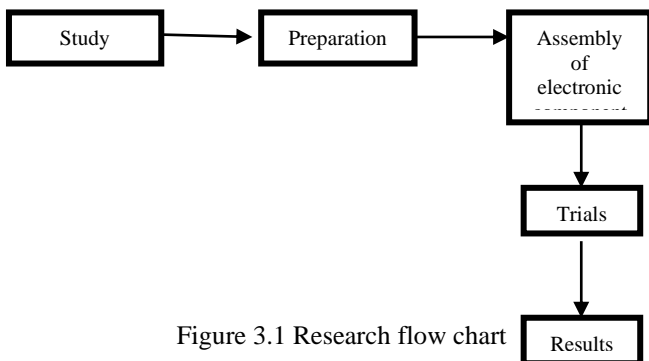


Figure 3.1 Research flow chart

Figure 3.1 The initial stages of literature study are carried out in order to complete the needs in the next stage of research preparation of equipment and electronic components after all have been assembled and assembled and a tool is tested and the final results are obtained.

3.2 Flowchart software

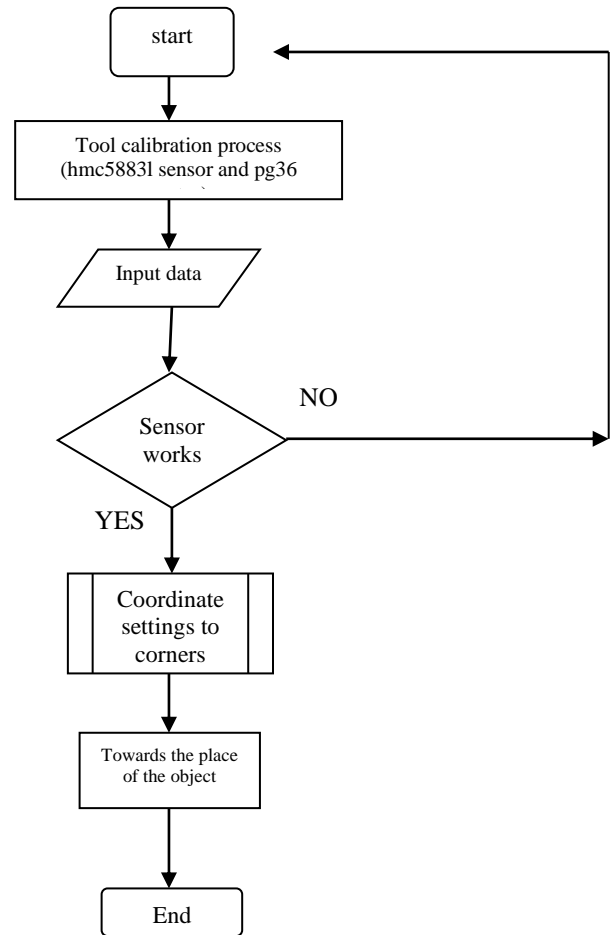


Figure 3.2 Flowchart Software

In Figure 3.2 the process is done by configuring the tool before it is given an input in the form of a logic control program, we test the response of the components installed in the robot after the program is inputted to order the sensor to work if the input fails, it returns to the initial process if the sensor works then the coordinate setting to the corner and towards the object then the mission is complete, in this case it has not used Artificial Intelligence (AI) but still uses the logic control program.

IV. RESULT OF RESEARCH

4.1 Tool Design



Figure 4.1 Tool Desain

In figure 4.1 the design of a tool from a wheeled robot that uses a PG 36 motor as a robot driver in addition there are more supporting components such as HC5883L sensor, HC Sr04, 12 volt battery made parallel 2 so that the number of volts becomes 24 volts and the cable as a connector.

4.2 Robot Design

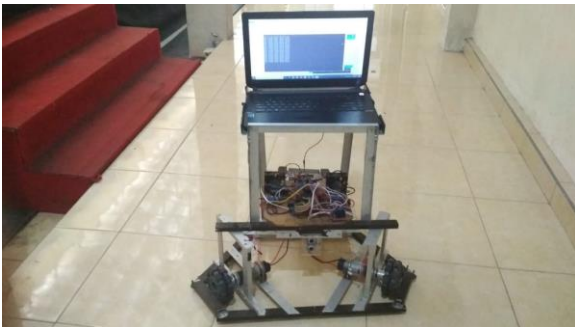


Figure 4.2 Robot Design

Figure 4.2 is a robot design that is ready to run as a data processor used by laptops as one that can send programs to Arduino, then Arduino is the microcontroller of the robot, from the results of testing the robot can move according to the previously programmed command and find the object point corresponding.

4.3 Source Code

```
#include <Wire.h> //i2c library for the Digital Compass
```

```
int led=13;
int cek_point;
int stp=42;
int st=44;
int Datastp=0;
int Datast=0;
int sts=0;
//motor kanan dpn
int DpnKanan=13;
int DpnKiri=12;
int IN1=26;
int IN2=28;
int IN3=30;
int IN4=32;
```

```
//motor kiri & blk
int SPA=11;
int SPB=10;
```

```
int IN5=34;
int IN6=36;
int IN7=38;
int IN8=40;
```

```
//griper
int GR1=52;
int GR2=50;
int GR3=48;
```

```
int GR4=46;
int EGI =9;
int EGA =8;
```

```
const int hmc5883Address = 0x1E; //0011110b, I2C 7bit address for
compass
const byte hmc5883ModeRegister = 0x02;
const byte hmcContinuousMode = 0x00;
const byte hmcDataOutputXMSBAddress = 0x03;
```

```
void setup()
```

```
{
  Wire.begin(); // Start the i2c communication
  Serial.begin(9600);

  //Initialise the Digital Compass
  Wire.beginTransmission(hmc5883Address); //Begin communication with
compass
  Wire.write(hmc5883ModeRegister); //select the mode register
  Wire.write(hmcContinuousMode); //continuous measurement mode
  Wire.endTransmission();

} // end setup
```

```
void baca_kompas(){
```

```
  int x,y,z; //triple axis data.
  //Tell the HMC5883L where to begin reading the data
  Wire.beginTransmission(hmc5883Address);
  Wire.write(hmcDataOutputXMSBAddress); //Select register 3, X MSB
register
  Wire.endTransmission();
```

```
  //Read data from each axis
  Wire.requestFrom(hmc5883Address,6);
  if(6<=Wire.available())
  {
    x = Wire.read()<<8; //X msb
    x |= Wire.read(); //X lsb
    z = Wire.read()<<8; //Z msb
    z |= Wire.read(); //Z lsb
    y = Wire.read()<<8; //Y msb
    y |= Wire.read(); //Y lsb
  }
```

```
  int angle = atan2(-y,x)/M_PI*180;
  if (angle < 0)
  {
    angle = angle + 360;
  }
```

```
  Serial.print("X :");
  Serial.println(x);
  Serial.print("Y :");
  Serial.println(y);
  Serial.print("Z :");
  Serial.println(z);
```

```
  Serial.print("angle :");
  Serial.println(angle);
  delay(250);
```

```
  if(angle>=75 && angle<230){
    putar_kiri();
  }
  if(angle>=231 && angle<=270){
    maju();
    delay(7000);
    mundur();
    delay(7000);
  }
```

```
void putar_kiri(){
  digitalWrite(IN1,HIGH);
  digitalWrite(IN2,LOW);
```

```

analogWrite(DpnKiri,50);
//digitalWrite(IN3,LOW);
//digitalWrite(IN4,LOW);
//kiri blk
digitalWrite(IN5,HIGH);
digitalWrite(IN6,LOW);
digitalWrite(IN7,LOW);
digitalWrite(IN8,HIGH);
analogWrite(SPA,50);
analogWrite(SPB,50);
}

void putar_kanan(){
digitalWrite(IN1,LOW);
digitalWrite(IN2,HIGH);
analogWrite(DpnKiri,50);
//digitalWrite(IN3,LOW);
//digitalWrite(IN4,LOW);
//kiri blk
digitalWrite(IN5,LOW);
digitalWrite(IN6,HIGH);
digitalWrite(IN7,HIGH);
digitalWrite(IN8,LOW);
analogWrite(SPA,50);
analogWrite(SPB,50);
}

void maju(){
//knn
digitalWrite(IN1,HIGH);
digitalWrite(IN2,LOW);
analogWrite(DpnKiri,90);

//kiri blk
digitalWrite(IN5,LOW);
digitalWrite(IN6,HIGH);
digitalWrite(IN7,LOW);
digitalWrite(IN8,LOW);
analogWrite(SPB,100);

//griper();
}

void mundur(){
//knn
digitalWrite(IN1,LOW);
digitalWrite(IN2,HIGH);
analogWrite(DpnKiri,90);

//kiri blk
digitalWrite(IN5,HIGH);
digitalWrite(IN6,LOW);
digitalWrite(IN7,LOW);
digitalWrite(IN8,LOW);
analogWrite(SPB,100);

// griper ka
digitalWrite(GR1,LOW);
digitalWrite(GR1,HIGH);
analogWrite(EGI,50);
//griper ki
digitalWrite(GR1,LOW);
digitalWrite(GR1,HIGH);
}

void kiri(){
//knn
digitalWrite(IN1,HIGH);
digitalWrite(IN2,LOW);
analogWrite(DpnKiri,85);
//digitalWrite(IN3,LOW);
//digitalWrite(IN4,LOW);
//kiri blk
digitalWrite(IN5,LOW);
digitalWrite(IN6,LOW);
digitalWrite(IN7,HIGH);
digitalWrite(IN8,HIGH);
analogWrite(SPA,100); //blk
analogWrite(SPB,95); //dgn kiri
}

void jalan(){
maju();
delay(8000);
kiri();
delay(4000);
mundur();
delay(8000);
kanan();
delay(4000);
}

void griper(){
// griper ka
digitalWrite(GR1,LOW); //kiri
digitalWrite(GR2,HIGH); //kiri
digitalWrite(GR3,LOW); //ka
digitalWrite(GR4,HIGH); //ka
analogWrite(EGA,60);
analogWrite(EGI,60);
//griper ki
}

void mati(){
//knn
digitalWrite(IN1,LOW);
digitalWrite(IN2,LOW);
analogWrite(DpnKiri,95);
//digitalWrite(IN3,LOW);
//digitalWrite(IN4,LOW);
//kiri blk
digitalWrite(IN5,LOW);
digitalWrite(IN6,LOW);
digitalWrite(IN7,LOW);
digitalWrite(IN8,LOW);
analogWrite(SPA,100); //blk
analogWrite(SPB,95); //dgn kiri
}

void loop()
{
/* Datast=digitalRead(st);
Datastp=digitalRead(stp);
if(Datast==HIGH){
sts=1;
if(sts==1){
baca_kompas();
}
}*/
baca_kompas();
}

```

4.4 Result

To find out the results of the trial will be divided into several angles and motor movements, the researcher took a special angle,

Table 4.4

Angle	30°	45°	90°	180°	360°
Sensor HMC58831	active	active	active	active	active
Sensor HC-SR04	Less active	Less active	active	active	active
Motor PG 36	Motor movement is out of sync	Motor movement is less synchronous	Synchronous motor movement	Synchronous motor movement	Synchronous motor movement

From table 4.4 it can be seen that the movement of robots using HMC58831 sensors, HC-SR04 sensors and PG motors is influenced by the predetermined angles that prove that the work of PG motors is taken from data obtained from the sensors installed.

For experiments with robot movements, it can be seen in figure 4.5 below:

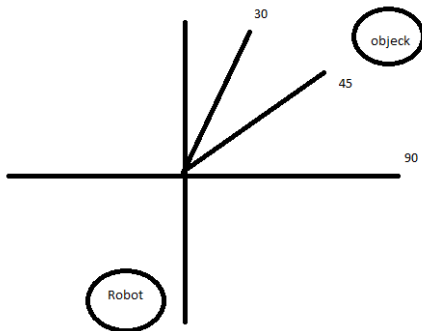


Figure 4.5 robot movements

For Figure 4.5, the robot is tested for rotation of the pg motor that receives data from the HMC58831 sensor or compass sensor, while the HC-SR04 sensor can be seen in Figure 4.6:

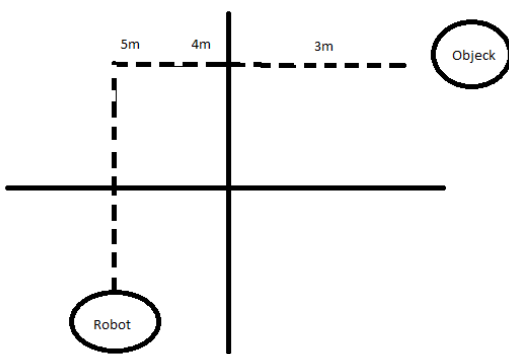


Figure 4.6 motion robot with HC-SR04 sensor

In figure 4.6 the motion picture of the robot that is influenced by the ultrasonic sensor or HC-SR04 is seen from a distance to find an object. For more details, it can be seen in table 4.7 the sensor is tested to detect the object in front of it by distinguishing its distance.

Table 4.7 HC-SR04 sensors and robot movements

Sensor HC-SR04(sensor Ultrasonic)	Distance		
	5m	4m	3m
	Inactive, irregular robot movements	Active, robot movements begin to be organized	Active, regular robot movements

From table 4.7, it can be seen that the sensitivity level of the HC-SR04 sensor affects the movement of the robot.

4.5 Data analysis

From the results of the experiments conducted that the angles that have been set in the program affect the movement of the PG-36 motor, the greater the angle, the motor movement will also be easier, besides that the large angles can facilitate changes in movement towards the object. data obtained from compass sensors and proximity sensors that can affect the movement of the PG-36 motor so that if the microprocessor receives a distance of about 3 m and 900 angle settings then the robot movement will be appropriate and can find an object.

V. CONCLUSION

From the results that have been obtained, this study can be concluded that the determination of the angle on the HMC58831 sensor will affect the movement of the PG 36. In addition, the proximity sensor also determines the speed to find the object and most importantly the voltage stability is needed in the movement of the robot. more organized.

ACKNOWLEDGMENT (Heading 5)

For the next researcher, it is better to use Artificial Intelligence (AI) in its logic control program so that the robot does not need to go back and forth between settings manually.

REFERENCES

- [1] sujito, "IMPLEMENTASI PROGRAMMABLE LOGIC CONTROLLER (PLC) PADA PENGENDALIAN ROBOT PEMINDAH BOTOL MINUMAN".
- [2] DKK B.K. Patle, "Path planning in uncertain environment by using firefly algorithm," *defence technologi*, vol. 691-701, p. 14, juni 2018.
- [3] Gregory Kahn DKK, "Self-supervised Deep Reinforcement Learning with Generalized Computation Graphs for Robot Navigation," *Berkeley AI Research (BAIR)*, vol. 3, p. 14, may 2018.
- [4] hendrawan, *Kompas Digital Untuk Navigasi Mobile Robot.*, 2007.
- [5] universitas sumatera utara. (2018)
- [6] labelelektronika.com. (2019) <http://eprints.polsri.ac.id/4613/8/FILE%20VIII.pdf>.