HO CHI MINH CITY NATIONAL UNIVERSITY

BACH KHOA UNIVERSITY

FACULTY OF MECHANICAL ENGINEERING

MECHATRONIC SYSTEM DESIGN

LINE FOLLOWING ROBOT

Instructor: PhD. Doan The Thao

Group members:

No	Full name	Student's ID
1	Nguyễn Quốc Thịnh	1752514
2	Lư Bát Thành	1752496
3	Nguyễn Thành Trung	1652643
4	Nguyễn Văn Tới	1552378

HCMC, 24th December 2020

HO CHI MINH CITY NATIONAL UNIVERSITY

BACH KHOA UNIVERSITY

FACULTY OF MECHANICAL ENGINEERING

MECHATRONIC SYSTEM DESIGN

LINE FOLLOWING ROBOT

Instructor: PhD. Doan The Thao

Group members:

No	Full name	Student's ID
1	Nguyễn Quốc Thịnh	1752514
2	Lư Bát Thành	1752496
3	Nguyễn Thành Trung	1652643
4	Nguyễn Văn Tới	1552378

HCMC, 24th December 2020

GROUP MEMBER:

NGUYEN QUOC THINH	**: 1752514**	
LU BAT THANH	**: 1752496**	
NGUYEN THANH TRUNG	**: 1652643**	
NGUYEN VAN TOI	**: 1552378**	

TOPIC:
"DESIGN, CREATE AND CONTROL LINE FOLLOWING ROBOT"

INSTRUCTOR'S EVALUATION

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

…………………………………………………………………………………………………

ACKNOWLEDGMENT

First of all, we want to say thank you to Faculty of Mechanical Engineering and all tutors of Bach Khoa University, who have taught and created conditions for us to approach valuable knowledge.

Further, we want to say thank you to our instructor PhD. Doan The Thao, who has conveyed knowledge as well as experience to us and always supports us to complete this project. Beside specialized knowledge, he also taught us a lot of practical experiences that are very useful and meaningful in our life.

Nevertheless, we want to thank you to all tutors, who have spent valuable time for review and evaluation our report. This will be useful advice for us in our future career.

Finally, we want to say thank you to our family for always supporting us during the time we studied at Bach Khoa University.

BEST REGARDS

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1. INTRODUCTION ABOUT LINE FOLLOWING ROBOT

Line following robot is a special case of mobile robot, in which the robot detects the relative position of the robot and follows the existing line (word line, color line). Currently, line detection robots are widely used in industrial manufacturing processes, carrying objects to special locations.

In order to design and operate a line following robot, all the technical components of a robot need to be considered: principle diagram, sensor type, motor, control structure and controllable algorithm.

The range of advantages of Line Follower Robot is quite widely. This robot movement is automatic so that it is used not only in home, industrial automations but also for long distance applications as well. The main applications of Line Follower Robot is carrying packages in daily life.

1.2. RESEARCH SITUATION IN THE WORLD

1.2.1. Arduino Line following robot by Anusha

Sensors: using IR sensor module as the line detecting sensor for the project. Consists of an IR LED and a Photo diode and some other components like comparator etc.

Controller: Arduino UNO is the main controller. Data from the sensors (IR sensors) will be given to Arduino and it gives corresponding signals to the Motor Driver IC.

Motor Driver: L239D Motor Driver IC is used in this project to drive the motors of the robot. It receives signals from Arduino based on the information from the IR Sensors.

- ❖ **Advantages**:
    - + Simple to assembly.
    - + Speed remains constantly.
- ❖ **Disadvantages**:
    - + Can not stop at any desired location.
    - + There are just 2 sensors so black line detection accuracy is not high. The robot also cannot perform sharp 90 degree turns or intersection.
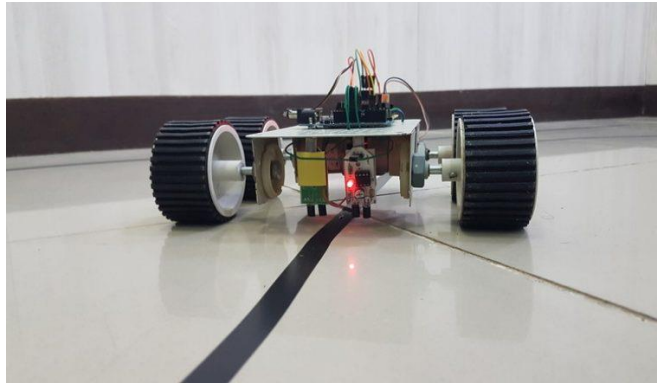
Figure 1.1 Arduino Line follower robot by Anusha

(Source: electronicshub.org, 2017)

1.2.2. Line Follower by midhun_s

Same principle with Anusha, as shown in Figure 1.1, but this time he uses 9 sensors, 8 of them arranged in a line and the other in front of the line at the center. Choosing the number 8 has the advantage that you can represent the status of all eight sensors using a single byte with each bit representing the state of one sensor.

- ❖ **Advantages**:
    - + Can run in course that contains sharp 90 degree turns, intersections and acute angle turns.
- ❖ **Disadvantages**:
    - + Components are hard to find since it's a "scratch-build".
    - + Can't stop at any desired location.
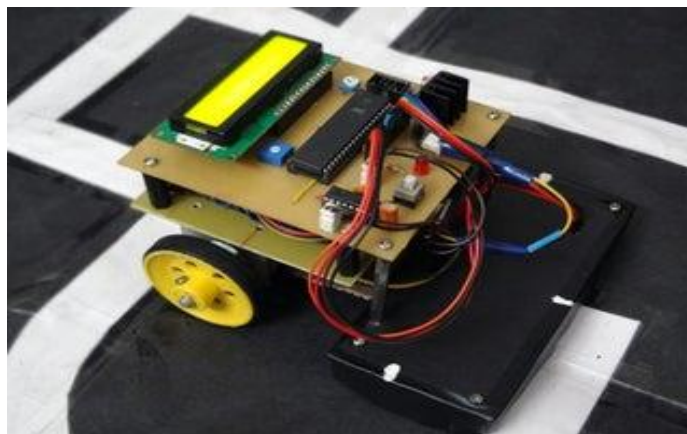
Figure 1.2 Line Follower by midhun_s

(Source: instructables.com, 2017)

1.2.3. Line follower by pantechsolution.net

Input: Read the white/black on the floor and condition the input signal(s) for transmission into the microcontroller in a way that questions can be asked and decisions made.

Process: Based on the inputs received, microcontroller decide what change (if any) needs to be adjusted the robot's speed and direction. Convert the results of any decisions made into something that can be sent to motor speed control and/or steering.

Output: Send the old or the newly adjusted control signals to speed and/or steering devices.

- ❖ **Advantages**:
  - + Cost effective.
  - + Simplicity of building.
- ❖ **Disadvantages**:
  - + Thin course: only about 1 or 2 inches in width on a white surface.
  - + Slow speed and instability on different line thickness or hard angles.
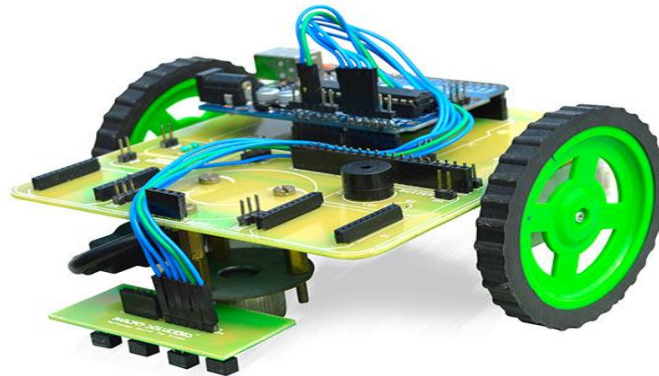
Figure 1.3 Line follower robot

(Source: pantechsolution.net)

1.2.4. Fireball robot

FireBall's first contest was the 2010 Bot Brawl in Peoria. Practice was black 3M electrical tape on the white kitchen floor

Fireball, as shown in Figure 1.4 participated in many of competition in the World. Its construction is quite simple with including 4 wheels and all of them are driving car transmitted by 4 specific motors.

❖ **Advantages**:

+ Simplicity of building, high solidification structure, small curve radius

❖ **Disadvantages**:

+ Controller is quite complicated because of synchronizing adjusting in 4 specific motor in order to avoid robot sliding
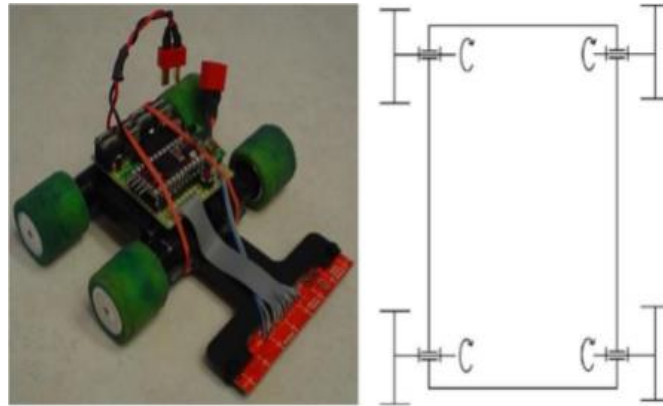
Figure 1.4 Fireball robot

(Source:brooksbots.com)

1.2.5. FH Westküste

This robot, as shown in **Figure 1.5** is champion in Competition of MCU Car Rally, took in Nuremberg 2015.

Parameter	Value	Unit
Length x Width x Height	550 x 170 x 140	mm
Average Velocity	1.04	m/s
Mass	1100	g

It was constructed quite similar to racing car in reality with low chassis and long body toward the front. This robot applied Ackerman driving structure. with 2 front wheels is fixed to frame of line follower sensor, at the same time a servo motor controls the direction of robot. The back motor transmits the motion to push the car forward through differential structure.

❖ **Advantages**:

+ Ackerman driving structure allows robot to operate in stable, high performance of line following, avoid sliding.

❖ **Disadvantages**:

+ Controller is quite complicated because of synchronizing adjusting in 4 specific motor in order to avoid robot sliding.
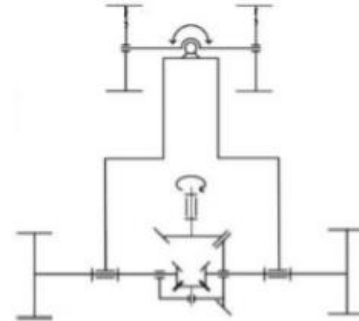
Figure 1.5 FH Westküste

(Source: fh-westkueste.de)

1.3. RESEARCH SITUATION IN VIETNAM

1.3.1. Mr.zero – Trinh Nguyen Trong Huu

As shown in Figure 1.1, including 4 wheels, with 2 rear wheels driving, 2 front wheels.

❖ Velocity: 1.6 m/s

❖ Deviation: 15 mm

❖ **Advantages**:

+ Guarantee balancing capability.

❖ **Disadvantages**:

+ Coplanar is not guaranteed, hence a suspension system is required to maintain contact between the wheels and road
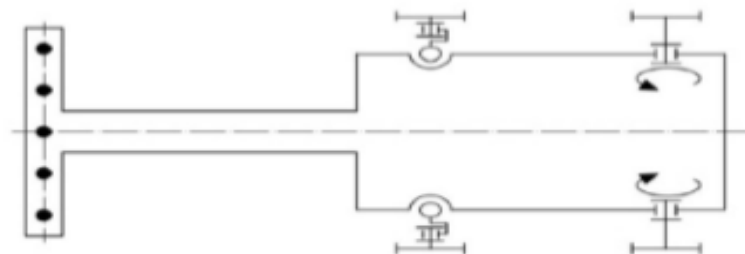
Figure 1.6 Mr. Zero - *2nd in competition in Vietnam 2015*

(Source:saigontech.edu.vn)

1.3.2. UIT-Mon – Nguyen Tien Dinh

As shown in Figure 1.1, including 3 wheels, with 2 rear wheels driving, a front wheel.

❖ Velocity: 1.5 m/s

❖ Deviation: 200 mm

❖ **Advantages**:

   + Both of 3 wheels tangent to move surface, respond time short since distance between sensor part and driving wheels is long.

❖ **Disadvantages**:

   + Risk of rollover if there's eccentric loading, hard to turn due to its length.
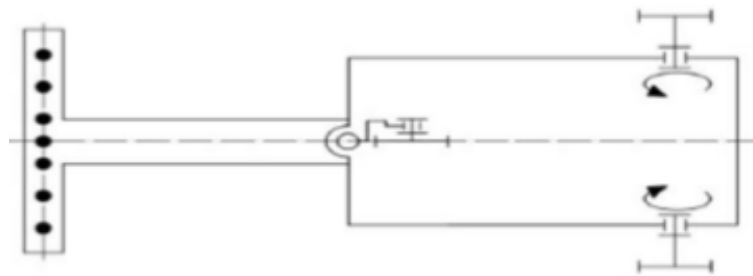
Figure 1.7 UIT-Mon – 1st in competition in Vietnam 2013

(Source:saigontech.edu.vn)

1.4. PROBLEMS

1.4.1. Content descriptions

❖ Design the mechanical structure, construct technical drawing.

   + Choose the most appropriate design for the body of the robot.

   + Calculate the dimension of mechanical links and test if they are suitable or not.

   + Choose motor that meet the required load.

❖ Design the electrical diagram, connect the components.

   + Determine the amount of input/output that the robot need, then choose the appropriate control unit.

   + Check the system for electrical safety requirement.

❖ Construct an algorithm to control the motor.

   + Solve the mathematical model of the motor.

   + Write a program to control the motor.

1.4.2. Design contraints

Design and manufacture the line following robot that moves at high speed on the bench according to the following characteristics and design constraints.

+   Movement Line System

+   Minimum radius of the line: 500mm

+   Line color: Black

+   Error: $e_{max} = \pm 20$mm

+   Background color: White

+   Maximum speed: $v_{max} = 1$ m/s

+   Line's width: 26mm.

+   Minimum speed: $v_{min} = 0.2$ m/s

+   Movement surface: flat

+   Wheel diameter: $d \leq 200$ mm

+   Maximum size of robot (length×width×height): 350mm×250mm×350mm

Robot is placed at START (point A), then the robot runs in the order that it passes through the specified nodes one after another:

Robot's route as shown in Figure 1.
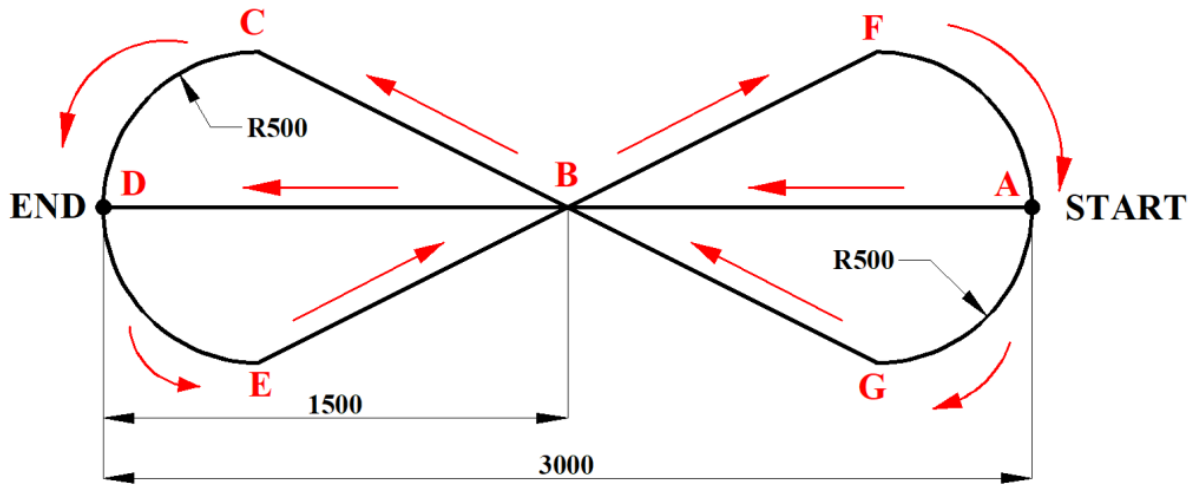
(START) A → B → C → D → E → B → F → A → G → B → D (END)

Figure 1.8 Robot's route

CHAPTER 2: DESIGN METHODOLOGY

2.1. MECHANICAL
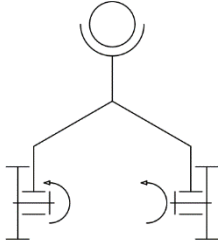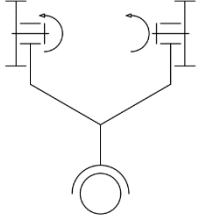
2.1.1 Principle diagram
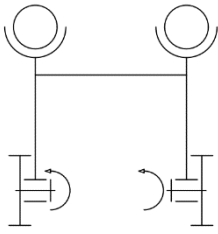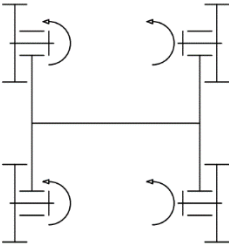
❖ Requirement for driving system

+ Flat and co-planar system is needed to increase the accuracy of the movement control

+ Good stability on the curve path

+ Achieve as least 0.2 "m"

2.1.1.1. Types of principle diagram

Table 2.1 Comparison of principle diagram

Mechanism	Model	Advantage	Disadvantage
Three-wheel model, backward driving wheels		+ Effective co-planar system + Simple design of mechanical system + Low torque requirement on the turning operation	+ Tendency of instability on the curve path
Three-wheel model, frontal driving wheels		+ Effective co-planar system + Simple design of mechanical system + Reliability on the curve path due to heavy-headed design	+ Slow respond + Slow speed

Four-wheel model, backward driving wheels and two frontal passive wheels		+ Good stability as two frontal wheels tend to balance the system + Simple design	+ Co-planar system is more difficult to be achieved
Four-wheel model, independent driving wheels		+ Extreme balanced and stabilized system + High-load capacity	+ More complex mechanical design + Control system is more difficult because of the synchronous requirement of the system of four motors

2.1.1.2 Group proposal

Because the robot only needs to follow the curve of a large radius (R = 500mm), the ability to change direction abruptly at the broken line positions and the vehicle's structure must be simple, with appropriate manufacturing costs. Therefore, the principle diagram of two-wheeled type using omnidirectional wheels is proposed (**Figure 2.1**).
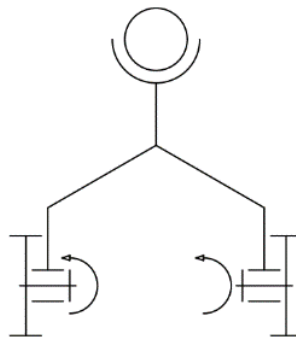
Figure 2.1 Principle diagram

2.1.2. Types of motor

Table 2.2 Comparison types of motor

Type	Advantages	Disadvantages
DC motor	+ Lower cost + No wearing components + Simple control with PWM	+ Lower power density & efficiency
Brush Motors	+ Medium power density + Lower cost + Simpler motor control	+ Includes wearing components (i.e., brushes)
Step motor	+ Simple control by pulse + Good positioning control	+ Bigger size than others + Lost step under high-torque operation

2.1.2.1. Group proposal

The DC motor is chosen as it can be simple in control with acceptable accuracy and lower cost. Also, another reason is that this type of motor is more common in the market

2.2. ELECTRICAL

2.2.1. Sensor

In terms of sensors, most of the current detection robots use optical sensors to detect the relative position of the line relative to the vehicle, thereby processing it to give control signals. There are two commonly used methods for line detection robots:

2.2.1.1. Cameras

The track image is taken from the camera, through processing and giving control signals.

> ❖ Advantages:

> > +   High accuracy, less interference.

❖ Disadvantages:

+ Requires a lot of handling, so it requires fast processing speed, otherwise it
will slow down the car

2.2.1.2. IR sensor

IR sensor is an electronic device, that emits the light in order to sense some object of the
surroundings. An IR sensor can measure the heat of an object as well as detects the motion.
Usually, in the infrared spectrum, all the objects radiate some form of thermal radiation
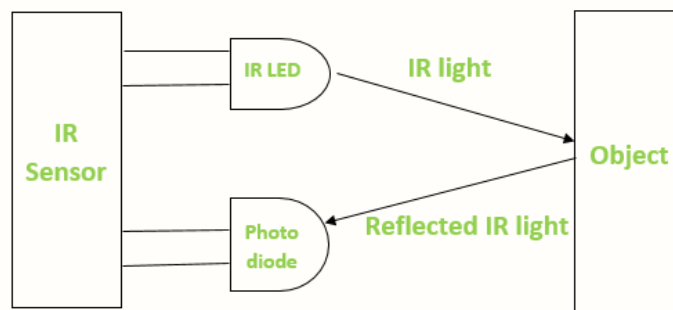
Figure 2.2 IR sensor working principle.

❖ **Advantages**

+ Their low power requirements make them suitable for most electronic
devices
+ Strong noise immunity.
+ It is not require contact with object to for detection

❖ **Disadvantages**

+ Required Line of sight.
+ Limited range

2.2.1.3. Group proposal

❖ *Types of sensor*

In order to satisfy the requirement of responding to abrupt fracture lines of the sand table,
the sensor variant must have appropriate sensitivity. Based on the high sensitivity
characteristics of the phototransistor compared to other types of optical sensors, two options
for using this type of sensor are proposed.

+   Phototransistor combined with LED usually

+   Phototransistor combined with infrared LED

Because the vehicle's running conditions are not specified indoors or outdoors, the ambient light intensity can be great, causing sensor interference. Currently on the market, optical and photoelectric diodes work best in visible light (380-700 nm wavelength), so they do not work correctly in outdoor conditions. Sensors that follow these two principles operating in the infrared are currently very scarce in the Vietnamese market.

For high contrast color lines, infrared LEDs offer higher sensitivity but need to be shielded against interference. On the other hand, the phototransistor consists of two signal transceivers in the infrared region (wavelength 950 nm) which are quite far away from the visible region (380-700 nm), thus limiting interference due to visible light from the environment. We decided to use TCRT5000, infrared sensor with average accuracy and depends a lot with outside light
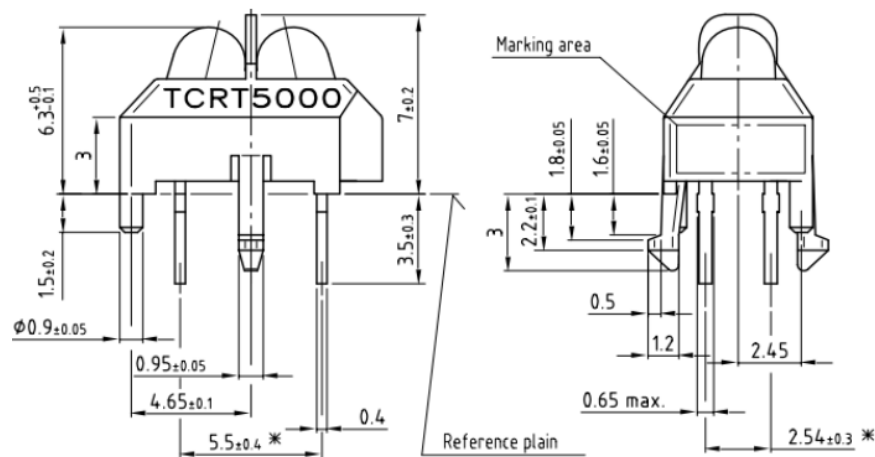
Figure 2.3 Sensor size TCRT5000

Table 2.3 Technical specifications of TCRT5000

Technical specifications	Value
Maximum operating current I_C	100mA
Maximum operating current I_F	60mA
Voltage	5V

Power	200mW
Size L x W x H	10.2 x 5.8 x 7 mm^3
Operating distance	0.2 – 15mm

❖ *Number of sensor*

Table 2.4 Comparison of the number of sensors

Quantity	Advantages	Disadvantages
Less than 4	Less work for calibration for the robot to run Fairly cheap	Causing difficulty in recognition when to turn, making the function complex and not effective
From 4 to 6	The function and work for calibration of the robot is at average. Adequate condition to pass all of the precision requirement in receiving signal.	The sensor can sometimes pass the line, making the robot drive less proficiency. More cost compare to the first one.
7	Detect much better and follow curve easier. Decrease the noises better.	More time to do the calibration because more part are added in Highly cost

Therefore, we decided to use 7 sensors, we can apply better control algorithm to handle the disadvantages.

2.3. SIGNAL PROCESSING ALGORITHM

2.3.1. The on/off control method

Using a comparator to determine the on/off state of the sensors, then infer the vehicle position according to a predetermined status table (**Figure 2.4**). With this method, the line detection error will depend on the parameters of the sensor used, or the distance between

the sensors. This method has the characteristics of depending mainly on the comparative threshold level of the sensors, so the processing speed is very fast
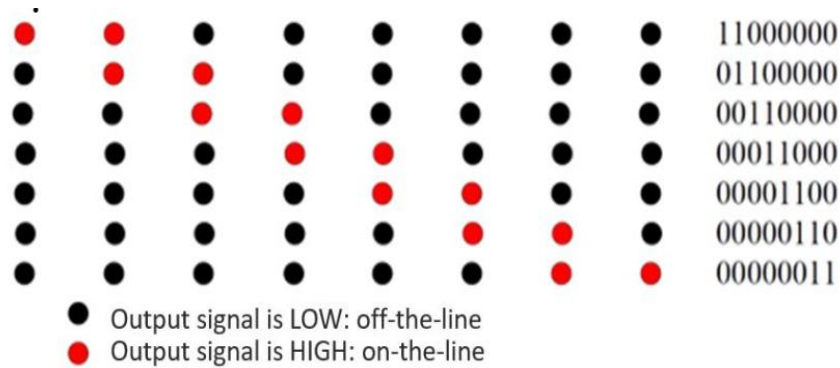
Figure 2.4 The on/off state method

❖ **Characteristics**

+ The output state of the signal is on / off

+ Depends on comparative threshold value and distance between sensors.

+ Processing speed is very fast.

+ System error = $\frac{1}{2}$ distance between 2 sensors

❖ **Advantage**:

+ Simple model and comprehension

+ Fast response

❖ **Disadvantage**

+ Low accuracy

+ Unsmooth trajectory of the robot as the smoothness is based on system resolution (distance and placing method of the sensor)

2.3.2. The quadratic approximation algorithms

This method is based on the analog signal. The position of the robot is estimated by the coordinate of the sensors. In this method, the three highest values of the sensor are used to estimate the position based on interpolation processing with three corresponding sensor coordinates. The mathematic model for the position is that.

The relationship between coordinates and output of the sensors

$$y_1 = ax_i^2 + bx_i + c$$
$$y_2 = a(x_i + 1)^2 + b(x_i + 1) + c$$
$$y_3 = a(x_i + 2)^2 + b(x_i + 2) + c$$

We can calculate the position and value of a and b as:

$$x = -\frac{b}{2a} \ (mm)$$

Where

$$a = \frac{y_1 + y_3 - 2y_2}{2} \quad \text{and} \quad b = y_2 - y_1 - 2ax_i - a$$

in which 1, 2, and 3 is three relative consecutive sensor

❖ **Characteristics**
+ The row distance of two sensors is 1
+ As the output of the sensor is higher when it is closer to the black line, so always identify 3 consecutive sensors with higher output than the others.

❖ **Advantages**:
+ Highest accuracy of 3 algorithms
+ Smooth trajectory of the robot

❖ **Disadvantages**:
+ Slowest response of 3 algorithms
+ Affected by calibration process
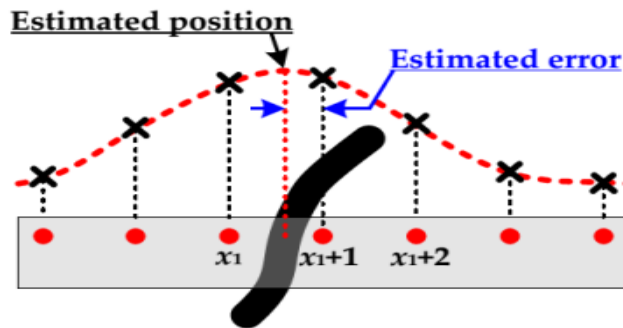
Figure 2.5 The quadratic approximation algorithms

2.3.3. The weighted average approximation algorithms

This method is based on the analog signal of the robot. The position is estimated by how high the analog signal of each sensors with the weight of the sensor itself

The position is estimated by the formula:

$$x = \frac{\Sigma x_i y_i}{\Sigma y_i} \ (mm)$$
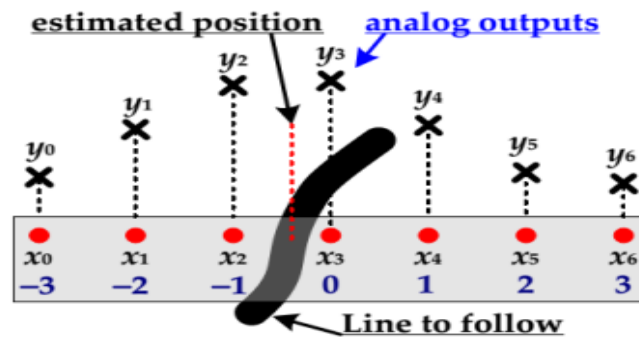
Figure 2.6 The weighted average approximation algorithms

❖ **Advantage**

+ Higher accuracy than ON-OFF control

+ Smooth trajectory of the robot

❖ **Disadvantage**

+ Slower response than ON-OFF control

+ Affected by calibration process

2.3.4. Group proposal

Our group decided the weighted average approximation method (**Figure 2.6**). Because the error depends on the number of sensors and how their altitude relative to the ground is chosen. Moreover, the resolution of this method is significantly higher than that of the comparison method, making it possible for the sensor system to achieve better error. Therefore, the response time of this method will be longer than the one above because the microcontroller needs to perform ADC conversion for all sensors.

2.4. CONTROL STRUCTURE

In terms of control structure, line detection robot has main modules including sensor module, control module and motor control module. In which there are two main methods to connect these modules together: centralized control and decentralized control method

2.4.1. Centralized control

In the centralized control method (**Figure 2.7**), an MCU receives the signal from the sensor, processes the data and then transmits the control signal to the actuator. This is a structure used quite a lot in cars. The centralized control structure has simple hardware characteristics, but the MCU must process all information before updating new information.

❖ **Advantages**
+   Simple implementation
+   Low cost hardware
❖ **Disadvantages**
+   Multi-task for one MCU tends to reduce the speed of processing and efficiency of the MCU
+   Lower the system maintenance ability
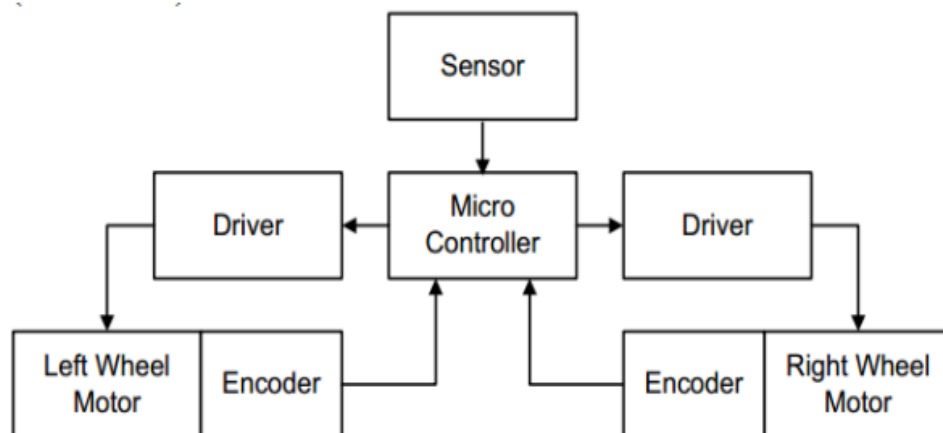
Figure 2.7 Centralized control

2.4.2. Distributed control

In the distributed control method (**Figure 2.8**) more than one MCU will be used in the system. In addition to the MCU master responsible for the overall calculation, some robots

also have an additional Slave MCU specialized in processing encoder signals or a slave MCU to process signals from sensors (RobotALF).
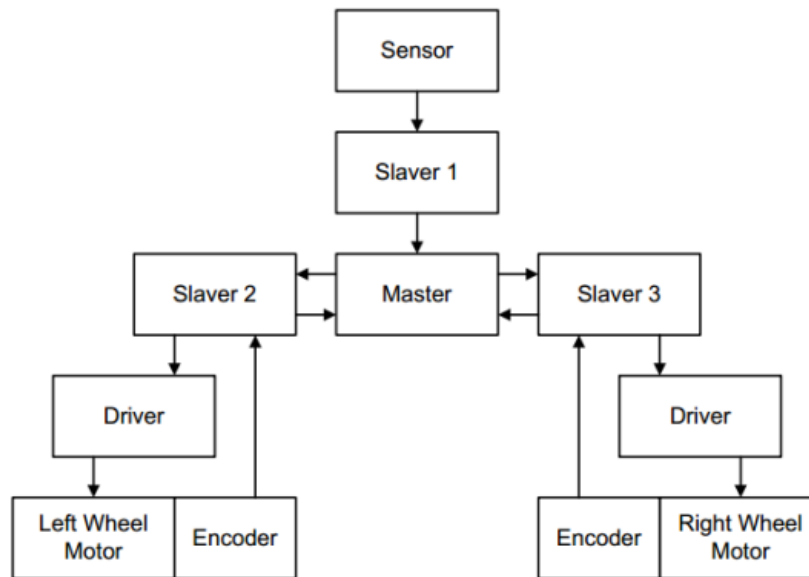
Figure 2.8 Distributed control

2.4.3. Group proposal

We decided to use the distributed control. Because, this structure helps to reduce the amount of computation for the master and allows the robot to perform multiple tasks at the same time. The hierarchical control structure has more complex hardware characteristics, paying attention to the communication problem between the MCU, but is capable of handling many tasks at the same time, making the system sampling time more faster than using the concentrated structure. There are some advantages of distributed method such as:

+ Lower workload for every MCU in the system
+ Faster response
+ Easy to debug the software problem
+ High cost-effective
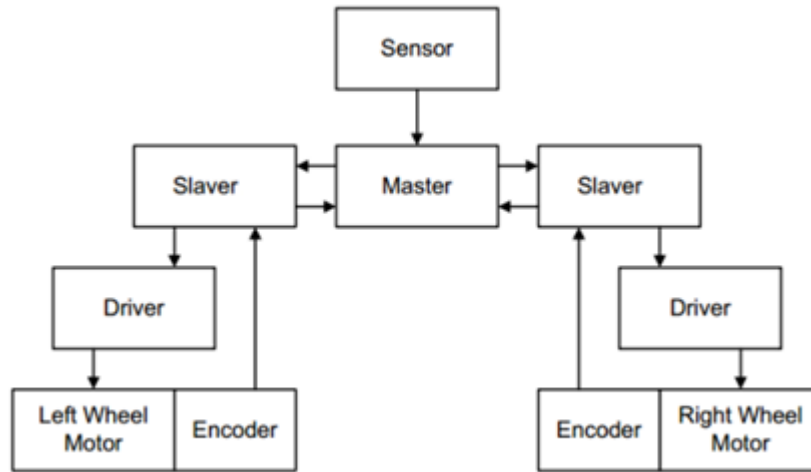
Figure 2.9 Distributed control

2.5. CONTROL ALGORITHM

Based on the requirement that the robot can grip on straight, curved lines and the error of the position of the line being meandering (500mm), two options for the controller are proposed:

- ❖ PD controller
- ❖ Tracking controller

2.5.1. PD controller

This option has the feature to help the robot improve its ability to grip the line after each run. However, the self-learning algorithm is complicated and need to incorporate a gyro sensor so that the controller can remember the acceleration state of the robot during the race.

2.5.2. Tracking controller

This is a common controller in research on the trajectory of a given mobile robot. Theoretically and experiment has proven that the controller is capable of moving the robot to given coordinates at the desired speed. In addition, the parameters of the K_x, K_y, K_θ controller can be customized to increase the robot's ability to grip lines for different road sections.

2.5.3. Group proposal

We choose tracking controller with Lyapunov stability because Lyapunov is our choice because it is a common controller in research on ability to follow a given trajectory of mobile robot. Theory and experiment have proven that the controller is capable of moving the robot to given coordinates at the desired speed

2.6. CONCLUSION

From the above proposals, we proceed to choose the appropriate option:

- ❖ **Principle diagram**: Differential 2-wheeled robot with passive omnidirectional
- ❖ **Sensor**: Infrared-Phototransistor LED set
- ❖ **Signal processing algorithm**: Use the weighted average algorithm to find the robot's position relative to the line
- ❖ **Motor**: DC servo motor
- ❖ **Control structure**: The distributed control
- ❖ **Control algorithm**: Tracking controller with lyapunov stability

CHAPTER 3: MECHANICAL DESIGN

3.1. WHEEL SELECTION

3.1.1. Driving wheel

❖ **Requirement:**

+ Light, durable, good cycling ability.

+ The diameter of wheel d ≤ 200mm, moving on flat surface.

❖ **Conclusion**:

According to the requirements, our group choose V2 wheel 80mm, as shown in figure below

Figure 3.1 V2 wheel

3.1.2. Passive wheel

❖ **Requirement**:

+ Small and flexible, suitable with student's budget.

+ Quick handle when turning.

❖ **Conclusion**:

According to the requirements, our group choose ball transfer unit, as shown in Figure

Figure3.2 Ball transfer

3.2. MOTOR SELECTION

❖ **Requirement:**

+ The velocity of robot v ≥ 0.2m/s.

+ Suitable with our budget.

❖ **Conclusion**:

+ According to the requirements, our group choose DC motor with encoder.

In order for the robot to move, the engine's role is to provide moment force to the wheels. This rotary movement is significantly affected by the robot's weight and friction between wheels and road. A mathematic model for a wheel is showed in **Figure 3.3**
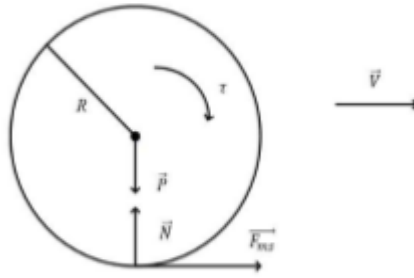
Figure 3.3 Mathematic model for the wheels

Moment of the wheel can be calculated:

$$I = \frac{1}{2}mR^2$$

Equalizing moment around wheel's center we get:

$$\tau - F_{ms}R = I\gamma$$

$$=> \tau = I\gamma + F_{ms}R$$

In order for the wheel not to slide when engine is turning, moment τ has to satisfy:

$$\tau \leq \frac{1}{2}mR^2\gamma + \mu\left(\frac{1}{2}M + m\right)gR$$

Newton's second law:

$$2Fms = (2m + M)a$$

$$\Rightarrow Fms = \frac{(2m + M)a}{2}$$

The necessary moment provided by motor can be calculate:

$$\tau = \frac{mR^2}{2}\gamma + \frac{(2m+M)aR}{2}$$

Power required in each engine:

$$P = \tau\omega$$

In which:

+ I (kg.m^2): Inertia moment of the wheel

+ R (m): radius of the wheel

+ τ (N.m): moment.

+ P (W): power of each engine

+ m (kg): weight of the wheel

+ F_{ms} (N): frictional force

+ a (m/s^2): desired acceleration.

+ ω (rad/s): angular velocity

+ M (kg): Total load of the robot

+ γ (rad/s^2): angular acceleration

+ g (m/s^2): gravitational acceleration.

We base on the input information as well as estimation to calculate engine parameter in order to satisfy given conditions:

Table 3.1 Input parameter and Required parameter of the robot

Highest velocity (v_{max})	1 (m/s)
Response time (t)	1 (s)
Acceleration (a)	1 m/s^2
Wheel radius (r)	0.04 (m)
Wheel's weight (m)	0,025 (kg)
Mass of car and load (M)	5 (kg)
Gravitational acceleration (g)	9.81 m/s^2
Coefficient of friction	0,8
Coefficient of safety	2

Perimeter of wheel: $C = 2\pi r = 251.327$ mm

RPM of wheel: $n = \dfrac{v}{C}.60 = \dfrac{1}{251.327}.60 = 238.73 \approx 239$ rpm

Angular velocity of wheel: $\omega = \dfrac{2\pi}{\dfrac{1}{\dfrac{239}{60}}} = 25.028$ rad / s

Moment equation around center of wheel: $\tau = \dfrac{mR^2}{2}\gamma + \dfrac{(2m+M)aR}{2} = 0.1010008$ Nm

Rated Power:

$$P = \tau.\omega = \left(\dfrac{mR^2}{2}\gamma + \dfrac{(2m+M)aR}{2} \right).\omega = 0.1010008 * 25.028 = 2.528 \text{ W}$$

With safety efficiency is 2, we got the power needed for each motor:

$$P = 2.528 * 2 = 5.056 \text{ W}$$

With the above parameter of the motor and safety coefficient 2, we choose JGA25 – 370 – CE DC 12V, ($\omega = 281$ rpm, $\tau = 184$ Nmm)

Re-calculate the velocity:

$$V = \dfrac{280}{60}.0,08.\pi = 1{,}17 \ (m/s)$$

With the chosen engine we can still satisfy desired velocity

3.3. CHASSIS DESIGN

3.3.1. Material simulation

We simulate the floor plate (which is the most easily damaged part) with the following conditions:

+ The floor plate size is the car's largest x length: 300x220mm
+ Material: mica (PMMA) 5 mm thick
+ The plate is rectangular, fixed to 4 corners
+ Load distribution 50N (5kg is equal to the maximum weight of the vehicle according to the set criteria)

Simulation content: Simulate deformation and safety factor on the floor plate

Criteria:

+ Distortion should not exceed 3 mm

+ Safety factor is greater than 2.0

3.3.2. Chassis size constraints

3.3.2.1. Horizontal anti-roll condition
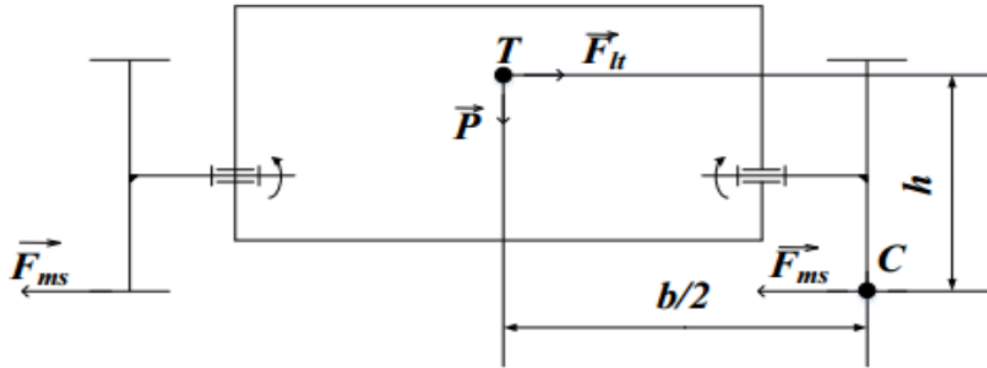
Figure 3.4 Force diagram when cornering

+ F_C: Centrifugal force   + C: The mind spins instantly

+ P: Gravity   + F_r: Friction force acting on the wheel.

+ G:Center of gravity   + b: The distance between two active wheels

+ h: Height of center of gravity

In order to avoid overturning, the moment due to gravity generated for center C must be greater than the moment caused by the centrifugal force for center C. We have:

$$P\frac{b}{2} \geq F_C h \qquad\qquad => b \geq \frac{2v^2 h}{Rg}$$

$$=> mg\frac{b}{2} - \frac{mv^2}{R}h \geq 0 \qquad => h \leq \frac{Rgb}{2v^2}$$

Submit all of the variables,

+ V = 1 m/s

+ R = 500 mm

+ b = 180 mm

=> h ≤ 441 mm

3.3.2.2. Vertical anti-roll condition
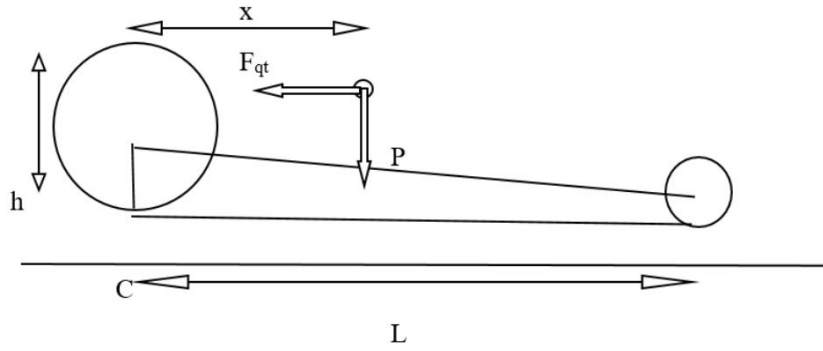
Figure 3.5 Diagram of impact on vehicle center of gravity

When the vehicle accelerates, consider torque balance at C:

$$Px - F_{qt}h = 0$$

$$Px - F_{qt}h = mgx - mah \geq 0$$

Thus,

$$\Rightarrow x \geq \frac{a}{g}h_{max} = \frac{a}{g}.441 = \frac{1}{9.81}.455$$

$$\Rightarrow x \geq 44.95 \ mm$$

3.3.2.3. Traction conditions of 2 active wheels
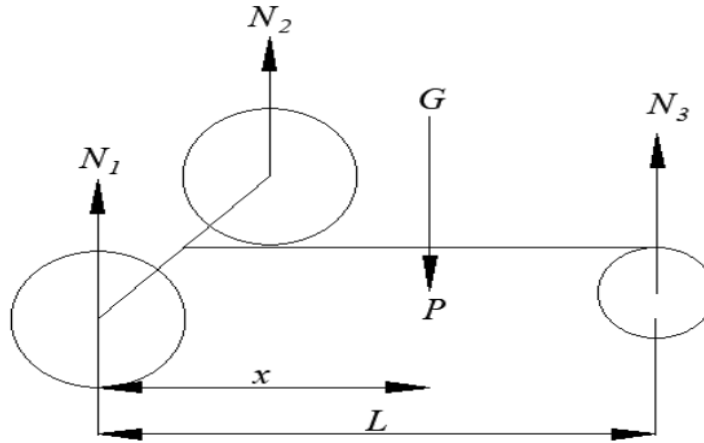
Figure 3.6 Analyze the force acting on the vehicle

Applying Newton's second law and considering the moments of the forces around the center of gravity G we have a system of equations:

$$\left\{ \begin{array}{c} N_1 = N_2 \\ (N_1 + N_2)x = N_3(L-x) \\ N_1 + N_2 + N_3 = P \end{array} \right\} => \left\{ \begin{array}{c} N_1 = N_2 \\ 2N_1 x = N_3(L-x) \\ 2N_1 + N_3 = P \end{array} \right\}$$

$$=> \left\{ \begin{array}{c} N_1 = N_2 = \dfrac{Mg(L-x)}{2L} \\ N_3 = \dfrac{Mgx}{L} \end{array} \right\}$$

Consider the force acting on the active wheel when rotating:
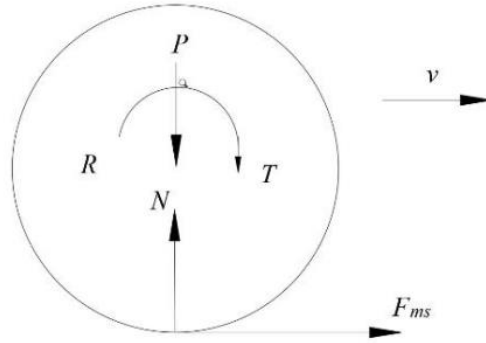
Figure 3.7 Analysis of the force acting on the active wheel

when the vehicle does not slip when turning:

$$F_r = \frac{T}{r} \le \mu N_1$$
$$=> T \le \mu N_1 r$$
$$=> T \le \frac{\mu Mg}{2}(1 - \frac{x}{L})$$
$$=> L \ge \frac{x}{1 - \dfrac{2T}{\mu Mgr}}$$

In which:

+   x is the distance from the center of the vehicle to the two active wheels

+ T_{max} is the maximum torque of the engine

+ μ is slip friction coefficient

+   r is the radius of the active wheel

3.4. SIMULATION

From mat lab simulation results, we choose vehicle width b = 180mm

We arrange the preliminary equipment on the chassis as follows:
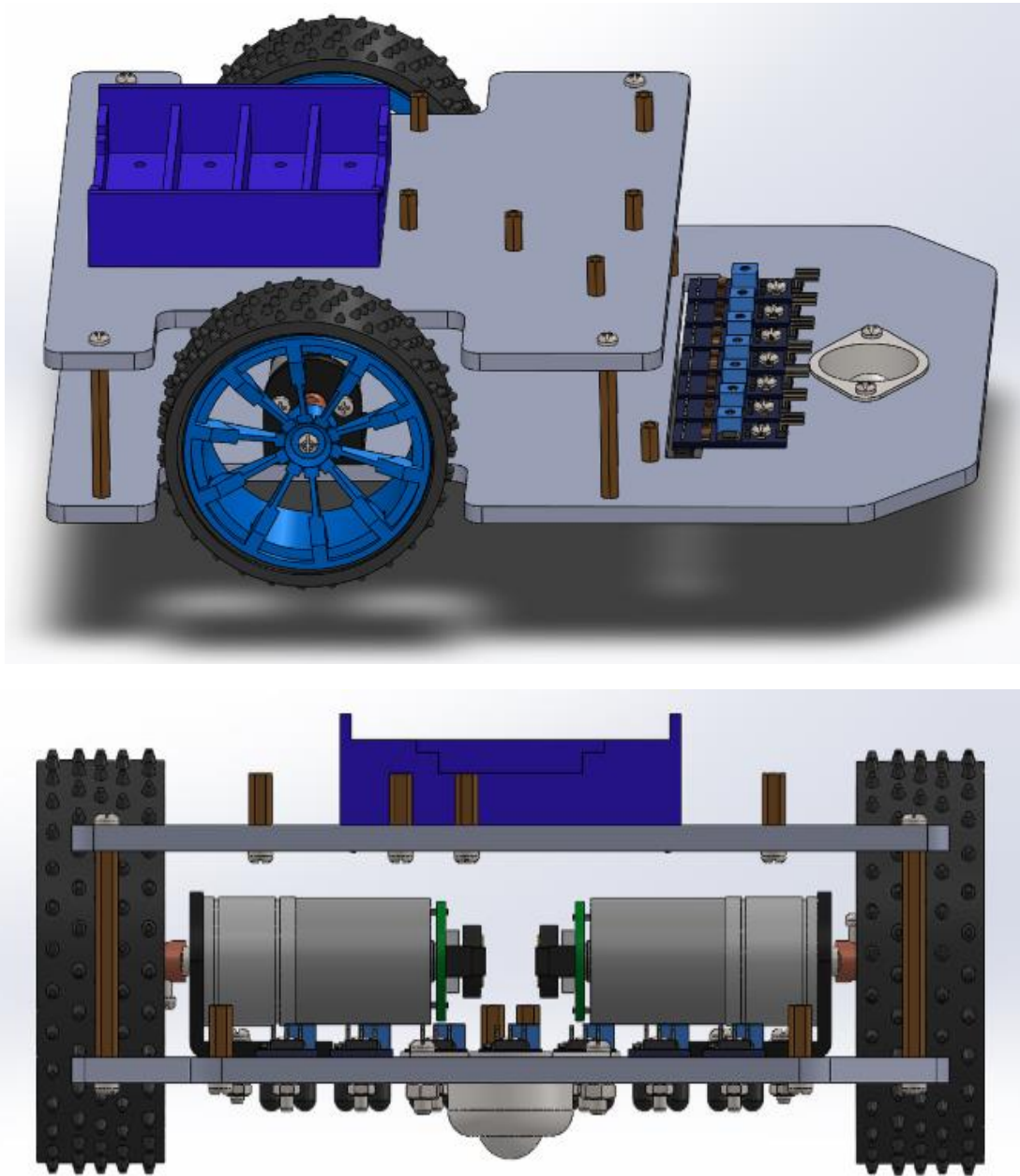
Figure 3.8 Model of robot

After setting up materials for the vehicle's components, we have:

Weight M = 5kg (including load)

Center of gravity h = 30mm

The distance of center of gravity compared to the drive wheel shaft x = 70 mm

Check for the binding conditions:

❖ Anti-overturn condition:

$$h = 30 \text{ mm} \leq 441 \text{ mm}$$

❖ Vertical anti-roll condition

$$x = 70 \text{ mm} \geq 44.95 \text{ mm}$$

❖ Traction conditions of 2 active wheels

$$L = 150 \text{ mm} <=> L \geq \frac{x}{1 - \dfrac{2T}{\mu M g r}} \geq 119 \ mm$$

.

CHAPTER 4: SYSTEM MODELING

In this chapter, our group consider two models that can provide a solution for topic are: kinematics and dynamics model

Kinematics is a subfield of physics, developed in classical mechanics, that describes the motion of points, bodies (objects), and systems of bodies (groups of objects) without considering the forces that cause them to move.

In Dynamics model, we must give full consideration of time varying phenomena in the interaction between motions, forces and material properties. Typically there is an time-integration process where results from one time frame effect the results on the next time frame. This model could lead to more precise control to the system. However, it will also lead to high difficulty and deeper intervention to the system.

For example the kinematics of a rigid body in space describes its possible coordinate positions and orientations and the range of velocities and angular velocities etc. The dynamics describes how these would change under the influence of a given system of forces.

In this project, the requirement is only to control trajectory and velocity of the system. Therefore, the kinematics model will be used for such consideration in order to lower the difficulty.

4.1. KINEMATIC MODEL

Analyze a differential drive robot model moving in a 2D plane with a linear velocity v and angular velocity ω. Its kinematics at point M, which is the point between two active wheels, can be described as:

$$\begin{bmatrix} \dot{x}_M \\ \dot{y}_M \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} cos\varphi & 0 \\ sin\varphi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \qquad (4.1)$$
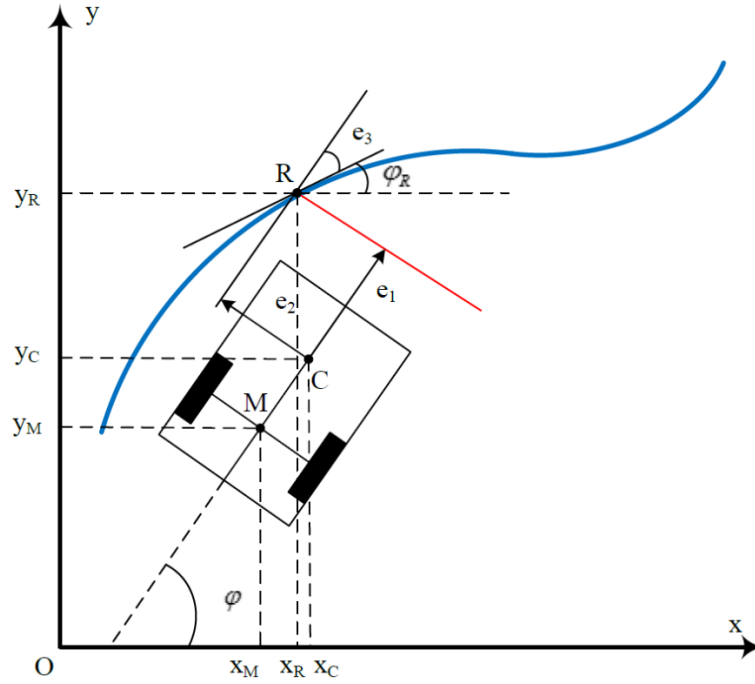
Figure 4.1 Kinematic model of a differential drive robot

Where:

+ $\{X_m, Y_m\}$: Coordinate system of Robot   + R: reference point for robot

+ $\{X, Y\}$: Global Coordinate system   + M: midpoint between two active wheels

+ v(t): velocity of the wheels   + C: tracking point of the robot

+ ω(t): angular velocity of the wheels   + L: the distance between 2 wheels.

However, in this case we are choosing point C, which is the center-point of the sensor array, to be our robot's tracking point. The kinematics at point C can be described as:

$$\begin{bmatrix} \dot{x}_C \\ \dot{y}_C \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -d.sin\varphi \\ 0 & 1 & d.cos\varphi \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x}_M \\ \dot{y}_M \\ \dot{\varphi} \end{bmatrix} \tag{4.2}$$

4.2. ERROR DYNAMICS

Establish a reference point R moving with a linear velocity v_R and angular velocity ω_R for the robot to follow. The kinematics at point R can be described as:

$$\begin{bmatrix} \dot{x}_R \\ \dot{y}_R \\ \dot{\varphi}_R \end{bmatrix} = \begin{bmatrix} cos\varphi_R & 0 \\ sin\varphi_R & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_R \\ \omega_R \end{bmatrix} \tag{4.3}$$

The error e_1, e_2, e_3 (**Figure 4.1**) can be determined as follow:

$$\begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} = \begin{bmatrix} cos\varphi & sin\varphi & 0 \\ -sin\varphi & cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_R - x_C \\ y_R - y_C \\ \varphi_R - \varphi \end{bmatrix} \tag{4.4}$$

The error dynamics:

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} = \begin{bmatrix} v_R.cos\ e_3 \\ v_R.sin\ e_3 \\ \omega_R \end{bmatrix} + \begin{bmatrix} -1 & e_2 \\ 0 & -e_1-d \\ 0 & -1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{4.5}$$

Where d is the distance between point M and point C.

4.3. METHOD FOR DETERMINING ERRORS IN REALITY

The ability to measure all three errors e_1, e_2, e_3 is very important in a good line-following robot system, but in reality the phototransistor array can only measure the error e_2, which is perpendicular to the robot heading (**Figure 4.2**). Therefore, we have to obtain e_1, e_3 based on e_2.
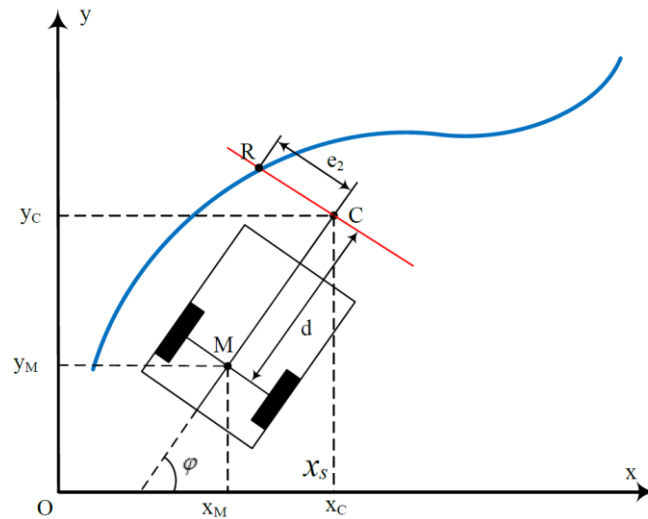
Figure 4.2 The error e_2 as measured by the sensor array

To find e_3, we make the robot move for a short distance ds small enough so that RR' become tangential with the desired path (**Figure 4.3**). Then e_3 can be determined by the following formula:

$$e_3 = \arctan \frac{e_2 - e_2{}'}{ds} \tag{4.6}$$
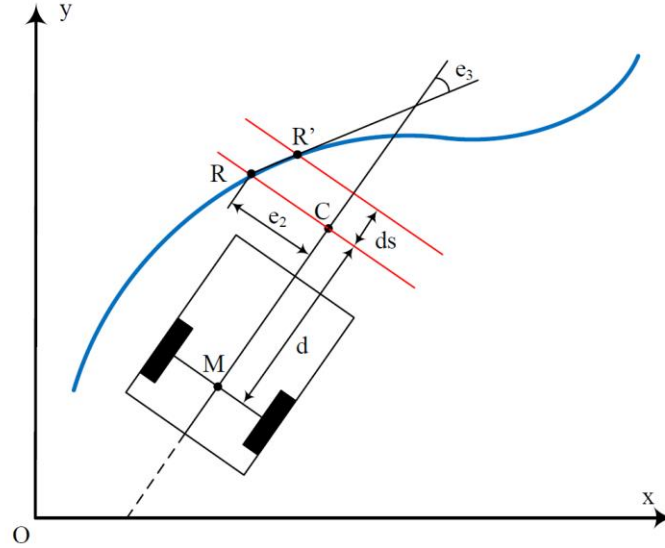
Figure 4.3 The error e_3 obtained by e_2

The error e_1 can be calculated using the formula:

$$e_1 = t_s \cdot \left(v_R \cdot \cos e_3 - v + e_2 \cdot \omega \right)$$

Where t_s is the sampling time. $\tag{4.7}$

4.4. CONTROLLER DESIGN

Determine the control law as follow:

$$\begin{cases} v = v_R \cdot \cos e_3 + k_1 \cdot e_1 \\ \omega = k_2 \cdot v_R \cdot e_2 + \omega_R + k_3 \cdot \sin e_3 \end{cases} \tag{4.8}$$

where k_1, k_2, k_3 are positive numbers.

This tracking controller (4.8) stabilizes the error dynamics (4.5) of the system. We have the Lyapunov's function:

$$V = \frac{1}{2} e_1^2 + \frac{1}{2} e_2^2 + \frac{1 - \cos e_3}{k_2} \geq 0 \tag{4.9}$$

Its derivative:

$$\dot{V} = e_1\left(v_R.\cos e_3 - v\right) + \frac{\sin e_3}{k_2}\left(k_2.v_R.e_2 + \omega_R - \omega\right) \qquad (4.10)$$

Substitute (4.8) into (4.10), we have:

$$\dot{V} = -k_1.e_1^2 - \frac{k_3}{k_2}\sin^2 e_3 < 0 \qquad \forall e_1, e_2, e_3$$

Therefore, the errors e_1, e_2, e_3 will converge to zero when time approaches infinity. We tested the controller (4.8) in MATLAB with the following parameters:

Table 4.1 Parameters for error convergence simulation

Parameter	Value	Unit
½ track width, b	*90*	*mm*
Distance from M to C, d	*100*	*mm*
Robot's velocity, v_{tb}	*0.5*	*m/s*
k_1	*50*	*N/A*
k_2	*5.10^{-4}*	*N/A*
k_3	*0*	*N/A*
Sampling time, t_s	*0.01*	*second*
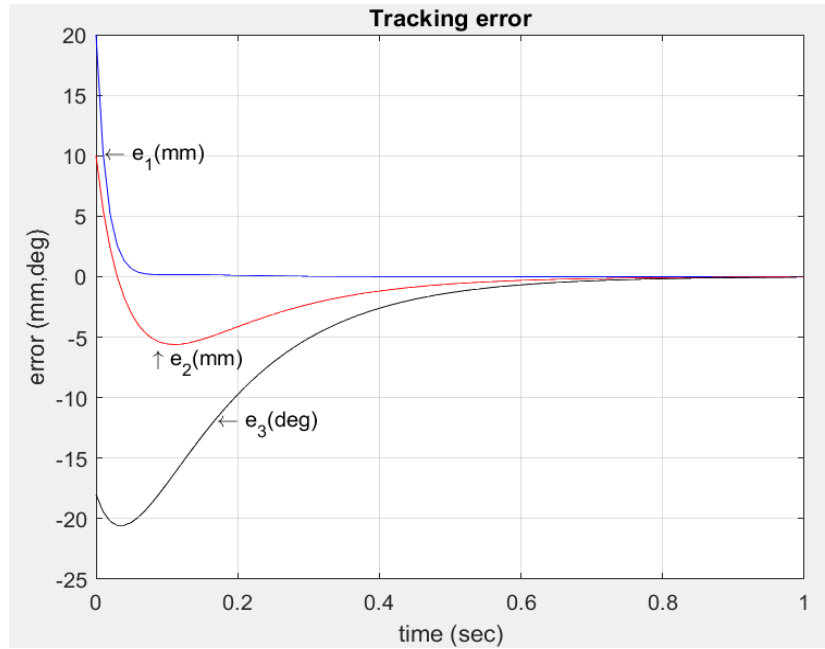
Figure 4.4 Tracking error convergence
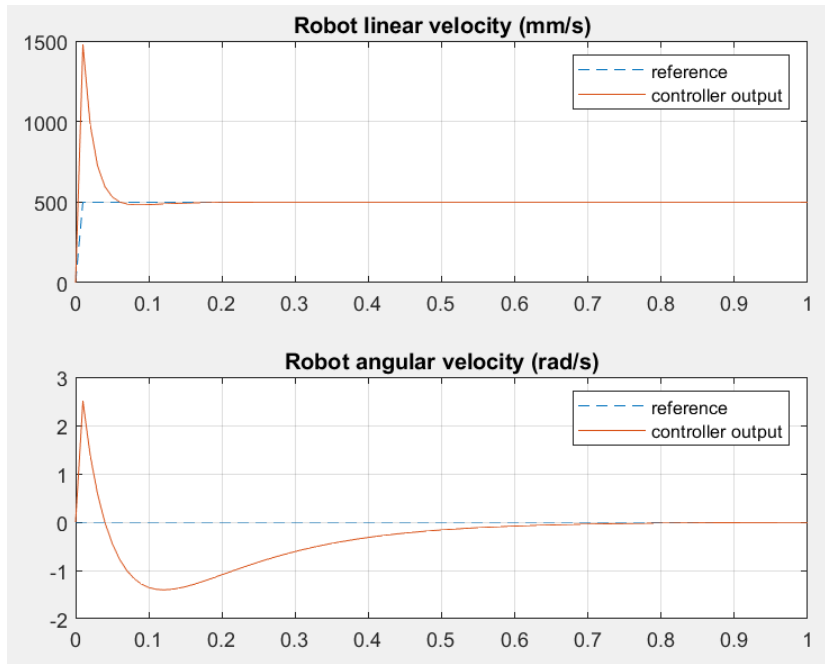
Figure 4.5 Linear and angular velocity of the robot

4.5. SIMULATION ON THE LINE MAP

We simulate the line-following behavior of the robot with d=50 ÷ 200 mm and the parameters in **Table 4.1** to find the value of d that produces the lowest error. The error e_2

at the turning sections A-B-C and G-B-D will be neglected since the robot will use a different algorithm to turn
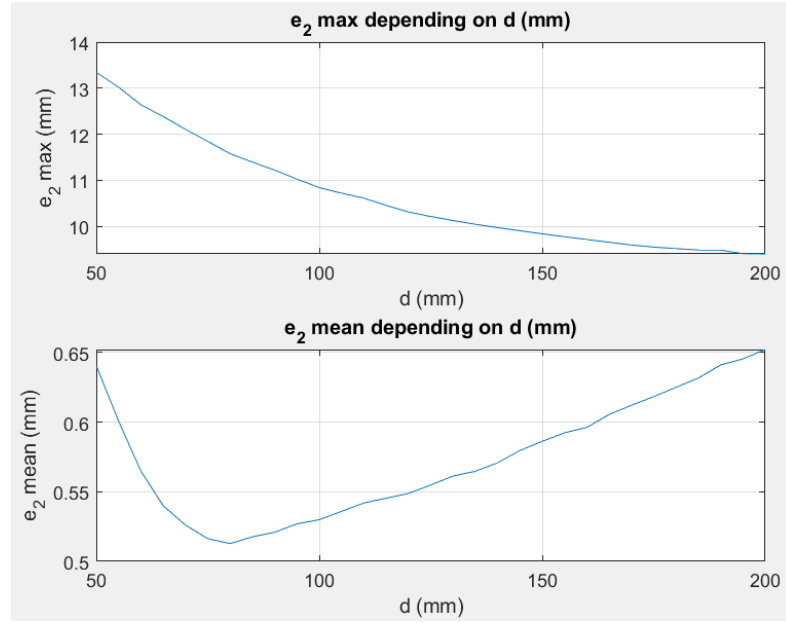
Figure 4.6 Relationship between d and the maximum and average error e_2

As shown in **Figure 4.6**, a larger *d* will result in a decreasing maximum of e_2. However, for values of *d* larger than *80mm*, the average of e_2 gradually increases. Based on **Figure 4.6** as well as taking into account the mechanical factors (space, other components' positions, strength of the frame), we choose *d = 100mm*. The maximum error is less than *11mm*, which satisfies the required value of *20mm*. However, this value is provisional only and does not perfectly represent the real error of the robot operating in the real environment.

CHAPTER 5: ELECTRICAL AND ELECTRONIC DESIGN

5.1. SYSTEM DIAGRAM
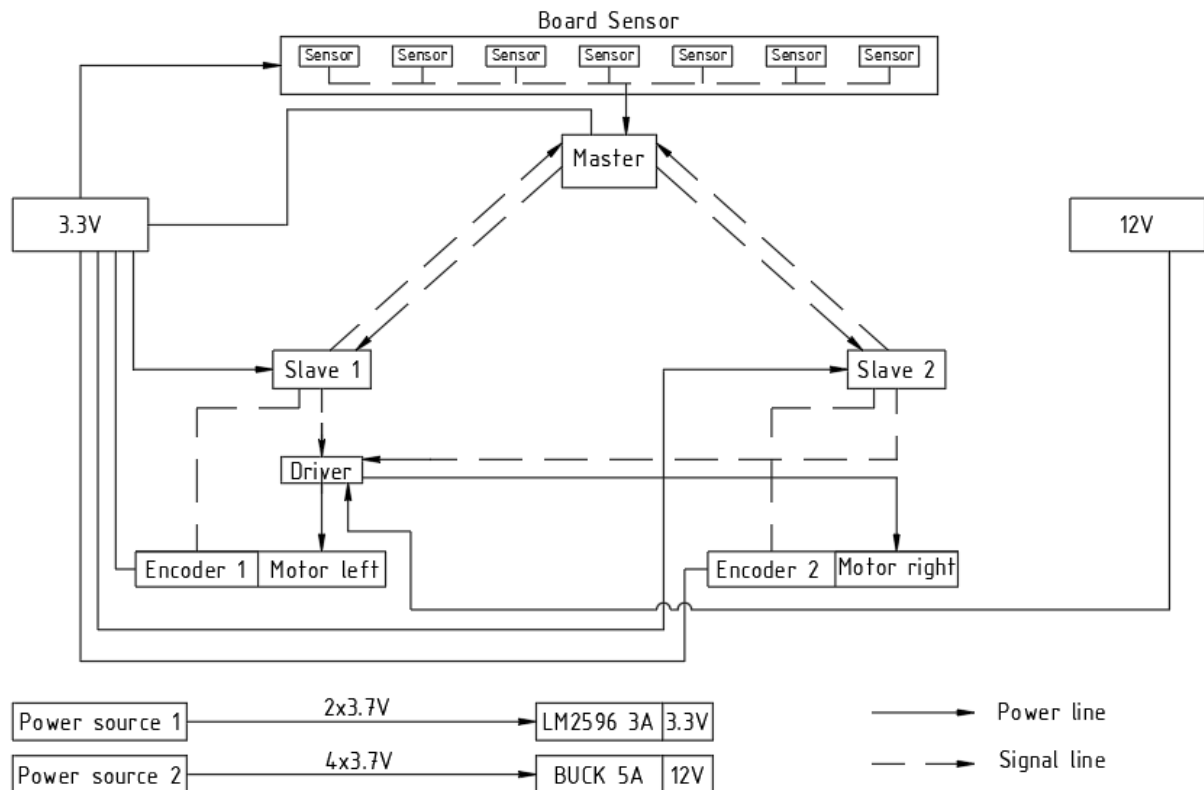
Figure 5.1 Working principle of line following robot

5.2. PHOTOTRANSITOR SENSORS

5.2.1. Sensor requirements

+ Fast Response and accuracy
+ Sensitivity and Stability
+ Suitable price

5.2.2. Choosing the sensor

Our group uses 7 sensors corresponding to the sensor bar length of 130 mm to ensure the better responsiveness.

The sensors we use in the project are TCRT5000

+ Sensor: Phototransistor TCRT5000

+ The maximum I_C operating current is 100mA, I_F is 20mA

+ Operating voltage: 5V

+ Power consumption: 200mW

5.2.3. Height of sensor

Relation between height and black-white difference:
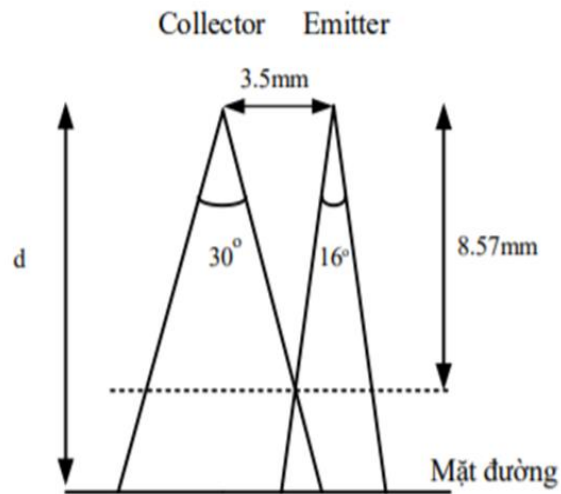
Figure 5.2 Sensor Testing Information

For sensors to function properly, the distance from the surface d should be ensured the communication area between the transmitter and collector. In this case, the height of sensor h > 8.57mm.
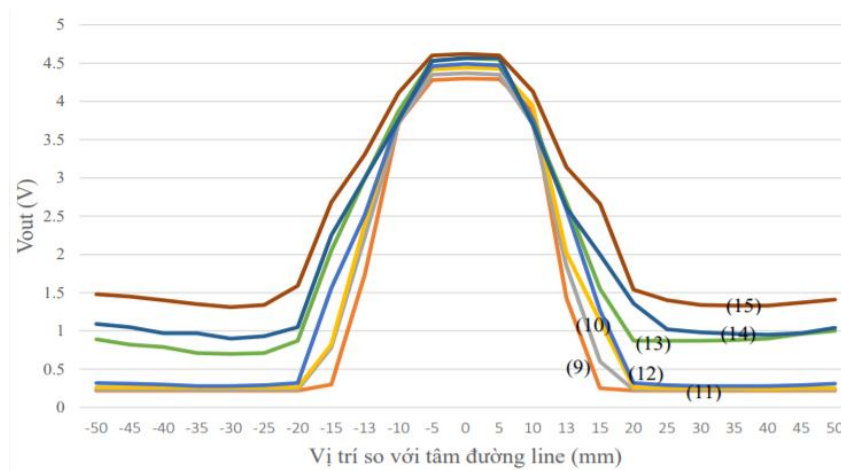
Figure 5.3 Voltage value returned from the sensor signal

❖ **Results**:

+ With a large distance (from 12 - 15mm), the difference in voltage between the position at the center of the line and on the white background is narrowed compared to the low distance (9 - 11 mm), the measured voltage value on the white background is unstable.

+ With low distance (from 9 - 11mm), the voltage measured at the white background is more stable (the values are almost the same).

+ Voltage values at white background corresponding to 9mm distance are more stable than 10mm and 11mm.

Thus, the sensor height compared to the surface h = 10mm.

5.2.4. Distance between sensors

The distance between two consecutive sensors should be ensured that the active areas of the transmitter and receiver LEDs on the same sensor do not coincide with the interference area of the adjacent set.

Corresponding to the selected sensor height of 10 mm, the interference area of the receiver and transmitter is calculated as follows:
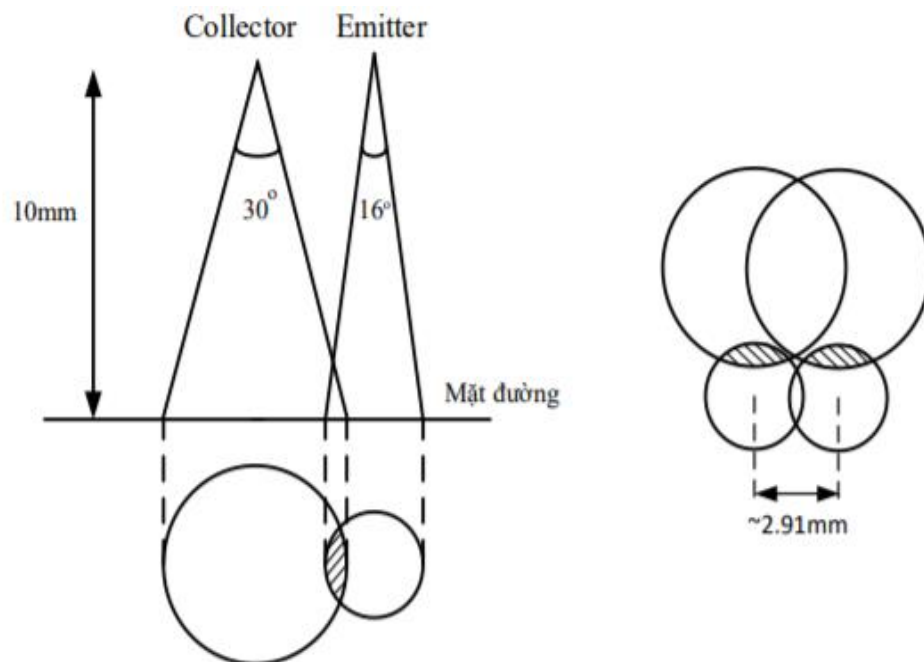
Figure 5.4 The interference between the collector and the emitter

The working zone of the sensor can be shown in **Figure 5.5**
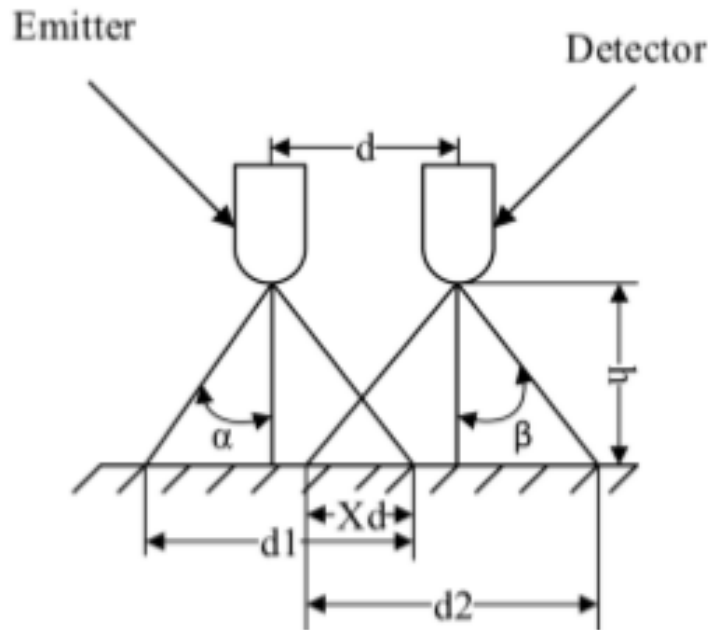
Figure 5.5 Operation principle of TCRT5000 infrared sensor

To eliminate disturbance, the distance has to satisfy equation:

$$l \geq h \tan \alpha + \tan \beta$$

With h = 10 mm, $\alpha = 16°$, $\beta = 30°$, we have: $l \geq 8.64$ mm

The gap between 2 LEDs in 1 sensor is 5 mm so the minimum distance between 2 sensors will be:

$$d = l + 5 \geq 13.64mm$$

Moreover, when operating, there will be cases where the sensor is in an erratic region, the returned analog value of the sensor will be the same. Therefore, it is inconceivable to determine the specific position of the sensor relative to the center of the line. The indeterminate region of the sensor is described as follows:
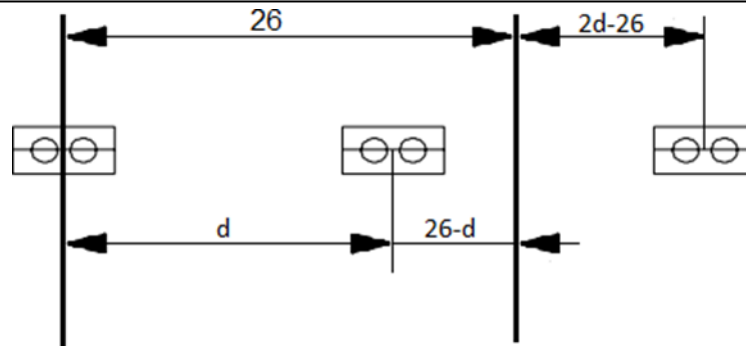
Figure 5.6 Erratic region of sensors

In **Figure 5.6**, it can be seen that when the sensor is moved to the right by a 26-d segment, there are always 2 LEDs in the line; therefore, the measured analog signal will be the same. Similarly, when the sensor moves to the left of the 2d-26 segment then only 1 led in the line. We must choose the value of d so that these distances are the smallest.

$$e = \min(\max(2d - 26, 26 - d))$$

$$\Rightarrow d = 17mm$$

5.2.5. Calibrate sensor

The analog signal from each sensor is different when they work in same condition. So we have to calibrate sensor with follow formula:

$$y_{j0} = y_{min} + \frac{y_{max} - y_{min}}{x_{max,i} - x_{min,i}}(x_{j,i} - x_{min,i})$$

With:

+ $X_{min,i}$, $X_{max,i}$ are minimum and maximum value from sensor i.
+ y_{max}, y_{min} are minimum and maximum value we want.
+ $x_{j,i}$ is value read from sensor i.
+ y_{j0} is value after calibrated.

We choose $x_{min}=0$, $x_{max} = 255$

Table 5.1 Value of sensors after calibrated

Sensor	X_{min}	X_{max}	Y
1	50	562	Y=0 + 0.498*(x - 48)
2	48	552	Y=0 + 0.506*(x - 48)
3	47	469	Y=0 + 0.6043*(x - 52)
4	48	559	Y=0 + 0.499*(x - 47)
5	52	636	Y=0 + 0.4366*(x - 49)
6	49	583	Y=0 + 0.4775*(x - 49)
7	48	557	Y=0 + 0.5009*(x - 48)

The deviation of the above method depends on the ambient light during the experiment. Due to the variety of conditions and the absence of the necessary equipment to limit the effects of light the method will have some tolerances. Therefore, we decided to use an automated caliber method approach to be able to accommodate a wide variety of conditions.

5.3. MICROCONTROLLERS

5.3.1. Master controller

For this system, the criteria to choose the master controller are:

+   Have more than 7 ADC channel input
+   Digital Communication Peripherals: UART, SPI
+   Fast time response
+   Reasonable price

After research on the market, PIC 18f46k20 is the most suitable for this project because it has all the feature robot need:

Program Memory Type	Flash
Program Memory Size (KB)	64
CPU Speed (MIPS/DMIPS)	16
SRAM (B)	3,936
Data EEPROM/HEF (bytes)	1024
Digital Communication Peripherals	1-UART, 1-SPI, 1-I2C1-MSSP(SPI/I2C)
Capture/Compare/PWM Peripherals	1 CCP, 1 ECCP,
Timers	1 x 8-bit, 3 x 16-bit
ADC Input	13 ch, 10-bit
Number of Comparators	2
Temperature Range (°C)	-40 to 125
Operating Voltage Range (V)	1.8 to 3.6
Pin Count	40
Low Power	Yes

Figure 5.7 PIC18f46k20 specs

5.3.2. Slave controllers

For the slave controllers, the criteria are quite simple:

+ Digital Communication Peripherals: SPI
+ Fast time response
+ PWM input
+ Reasonable price

After considering, our group choose the 2 slave controllers are PIC16f877a because it meets our requirements:

Program Memory Type	Flash
Program Memory Size (KB)	14
CPU Speed (MIPS/DMIPS)	5
SRAM (B)	368
Data EEPROM/HEF (bytes)	256
Digital Communication Peripherals	1-UART, 1-SPI, 1-I2C1-MSSP(SPI/I2C)
Capture/Compare/PWM Peripherals	2 Input Capture, 2 CCP,
Timers	2 x 8-bit, 1 x 16-bit
ADC Input	8 ch, 10-bit
Number of Comparators	2
Temperature Range (°C)	-40 to 125
Operating Voltage Range (V)	2 to 5.5
Pin Count	40

Figure 5.8 PIC16f877a specs

5.4. POWER SUPPLY

The map of the contest includes the following segments: a 3000mm straight line, 2 curves with a 500mm radius and 4 intersecting sections with 1118mm length. The total length of the ramp is S = 10.613 m. Average vehicle speed is 0.5m/s, so the time it takes to finish the road is t = 10.613/0.5 = 21.226s. To ensure that during testing, the battery capacity is required to be large enough for the vehicle to be repeatedly tested.

❖ **Technical requirements:**
+ The battery voltage must be greater than or equal to the maximum voltage of the device in the system.
+ Ability to provide current for the system to operate for about 2 hours.

The 18650 battery which has parameter 3.7V, 2500mAh, 5A rechargeable battery is the most used type for backup because of its high capacity (actual test for capacity above 2500mAh) and large discharge current 5A. The 18650 battery 3.7V, 2500mAh, 5A

rechargeable battery is durable and affordable. The battery 18650 shown in **Figure 5.9** has parameters:

+ Battery model: 18650

+ Average voltage 3.7V, full charge 4.2V.

+ Capacity: 2500mAh.

+ Discharge current: 5A

+ Size: 18x65mm.

+ Weight: 46g.

Figure 5.9 Battery 18650

Table 5.1 Main electronic devices list

Devices	Number	Max Current	Voltage	Total
Sensors	7	0.04A	3.3V	0.28A – 3.3V
PIC 18f46k20	1	0.2A	3.3V	0.2A – 3.3V
PIC 16f877a	2	0.2A	3.3V	0.4A – 3.3V
Motors	2	1A	12V	2A – 12V
Total				0.88A – 3.3V 2A – 12V

According to the table above, robot system should have a power source for control circuit and power source for driving motors:

$$P_{controlcircuit} = 0.88 \times 3.3 = 2.9 \ watt$$

$$P_{motors} = 2 \times 12 = 24 \ watt$$

Choose safety coefficient equals 2, we have:

$$P_{controllercircuit} = 2.9 \times 2 = 5.8 \ watt$$

$$P_{motors} = 2 \times 2 \times 12 = 48 \ watt$$

With the setting up of 4 18560 batteries (14.8V), to provide enough current for motors, the minimum discharge current is:

$$I_{battery \ for \ motors} = \frac{48}{14.8} = 3.24 \ A$$

The capacity of 4 batteries are 10Ah. With the minimum discharge current 3.24A, we could run the motors continuously in 3 hours (meet the requirements). So, the power supply for the motor is 4 18560 batteries.

With the setting up of 2 18560 batteries (7.4V), to provide enough current for the control circuit, the minimum discharge current is:

$$I_{battery\ for\ control\ circuit} = \frac{5.8}{7.4} = 0.78\ A$$

The capacity of 2 batteries are 5Ah. With the minimum discharge current 0.78A, we could use the control circuit continuously in about 6 hours (meet the requirements). So, the power supply for the motor is 218560 batteries.

5.5. ACTUATORS

5.5.1. Determining sampling time

To design a controller for the DC motors, we first need to obtain the mathematical model of the motors. Some technical specifications of the JGA-370-CE DC motors:

+ Rated voltage: 12 V

+ Gear reduction ratio: 21.3

+ No-load speed: 280 RPM

+ No-load current: 0.07 A

+ Rated speed: 215 RPM

+ Rated current: 0.3 A

+ Encoder: 2 channels, 11 pulses per revolution

There are two main ways to sample a DC motor's speed: pulse counting and pulse timing. A comparison chart between the two is shown below:

Table 5.3 Sample a DC motor's speed

	Pulse counting	**Pulse timing**
Principle	Pulse counting uses a sampling period (t) and the number of pulses (n) that are counted over the sampling period to determine the average time for one pulse (t/n). Knowing the number of pulses per revolution (N) for the encoder, the speed can be calculated: $$\omega = \frac{2\pi n}{Nt}$$ Where: ω = angular speed (rad/s) n = number of pulses t = sampling period (s) N = pulses per rotation	With the pulse timing method, a high-frequency clock signal is counted during one encoder period (the pitch, or interval between two adjacent lines or windows). The number of cycles of the clock signal (m), divided by the clock frequency (f), gives the time for the encoder period (the time for the encoder to rotate through one pitch). If the encoder PPR is denoted by N, the angular speed of the encoder is given by: $$\omega = \frac{2\pi f}{Nm}$$ Where: ω = angular speed (rad/s) f = clock frequency (Hz) m = number of clock cycles N = pulses per rotation
Application	At low speeds, the resolution of pulse counting is poor, so this method is best applied in high speed applications.	At high speeds, there may be too little time between pulses for pulse timing (also referred to as pulse frequency) to accurately measure clock cycles, so this method is best for low speed applications.

For our motor, the number of pulses that can be obtained in one revolution is $11 \times 21.3 \times 4 = 936$ pulses which is quite high; therefore, we choose the pulse counting method with n = 25 pulses per sampling period for best accuracy. The maximum pulse frequency can be calculated:

$$f_{max} = \frac{280 \times 936}{60} = 4368 \quad \left(pulses / sec \right)$$

With the slave MCU PIC16F877A operating with a 20MHz crystal, we can calculate its instruction cycle:

$$T_{CY} = \frac{4}{20000000} = 0.2 \quad \left(\mu s \right)$$

and its instruction frequency: $F_{CY} = 5$ (MHz)

Since $F_{CY} \gg 2f_{max}$, the MCU is fast enough to measure motor speed without missing encoder counts. Next, we determine the speed sampling time using pulse counting method:

$$f_{sampling} = \frac{280 \times 936}{60 \times 25} = 175 \quad \left(Hz \right)$$

Choose $f_{sampling} = 200 \, Hz$, which gives the sampling time $t_{sampling} = 0.005 \, sec$

5.5.2. Choosing motor driver

Two popular choices for motor driver are L298N and TB6612FNG. Comparison between the two's technical specifications is shown below:

Table 5.4 L298N and TB6612FNG specifications

Parameter	L298N	TB6612FNG
Maximum voltage *(V)*	*30*	*15*
Maximum current per channel *(A)*	*2*	*3.2*
Number of channels	*2*	*2*
Maximum PWM frequency *(kHz)*	*N/A*	*100*

We also tested both motors with both drivers, input PWM duty cycle ranging from 10% to 100% with 5% resolution. Below are the results:
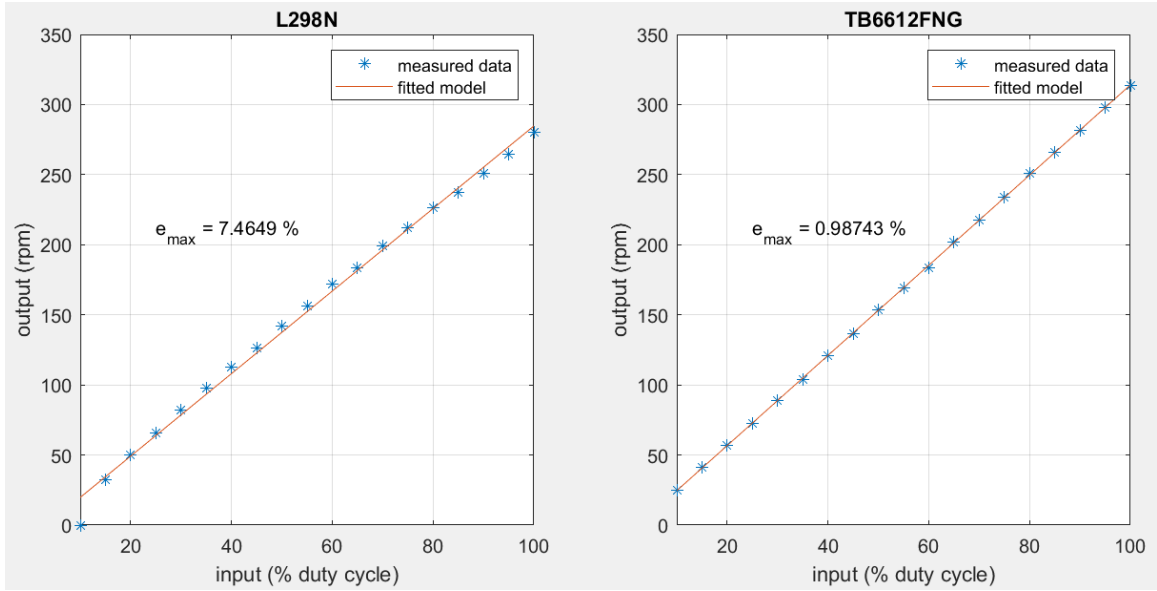
Figure 5.10 Left motor's speed with different inputs
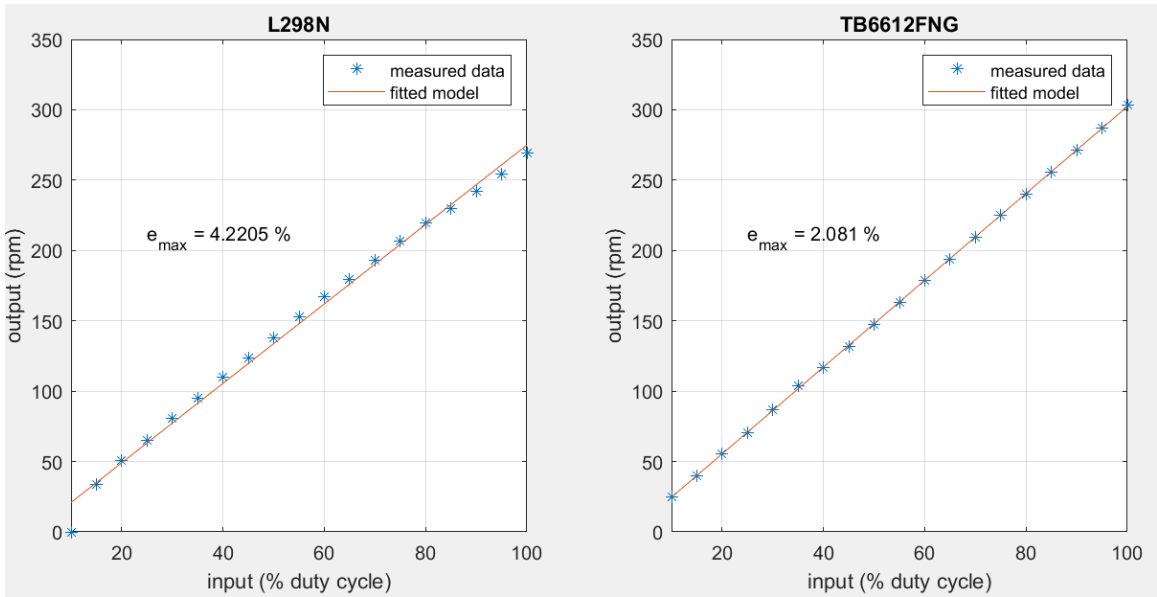
Figure 5.11 Right motor's speed with different inputs

As shown in **Figure 5.10** and **Figure 5.11**, for both motors, the motor and TB6612FNG system exhibits a more linear output-input behavior than the other one (lower error with respect to the least-square fitted model). Notably, for the case of TB6612FNG the motors can still rotate with an input of 10% duty cycle, unlike with the L298N where the motors cannot rotate at all at this input. Therefore, we choose the TB6612FNG as our motor driver.
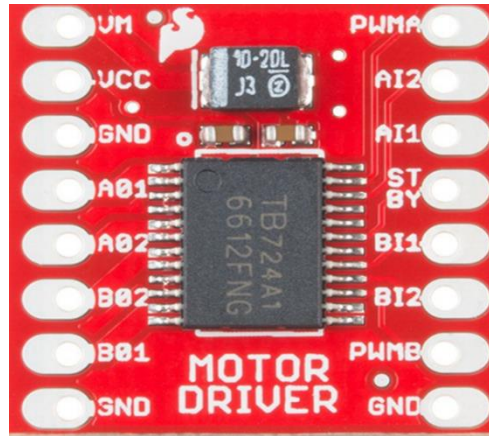
Figure 5.12 Driver TB6612

5.5.3. Transfer function identification

For the motors, we choose the PWM's percentage duty cycle as the input and the motors' speed in rpm as the output. Motors' speed will be measured using the built-in Hall-effect quadrature encoders and two MCUs PIC16f877A. The driver used is TB6612FNG operating at 12V. PWM frequency is chosen as 5 kHz which is within the recommended range.

When identifying the transfer function of motor – driver system, we need to choose sampling time long enough for the motor to reach steady state. Through experiments, we concluded that 0.2 second is fair. The rate at which the speed is measured is 200 Hz as calculated above, and the experiment's period is 1 second.

Firstly, the number of samples is calculated:

$$N.\delta t = m.T => N = \frac{m.T}{\delta t} = \frac{m.5}{0.005} = 1000m$$

Choose m=1, we have N=1000 samples. The left motor is sampled with input values ranging from 20% to 100% duty cycle with a resolution of 20% (**Figure 5.13**)

$$PWM = \begin{cases} 20\%, & 0 \leq t < 0.2 \\ 40\%, & 0.2 \leq t < 0.4 \\ 60\%, & 0.4 \leq t < 0.6, \ T = 1s \\ 80\%, & 0.6 \leq t < 0.8 \\ 100\%, & 0.8 \leq t < 1.0 \end{cases}$$

Figure 5.13 Diagram of duty cycle input value supplied to motor

The response of the left motor was recorded in the following table and figure:

Figure 5.14 Response of the left motor

Table 5.5 Dataset of the left motor

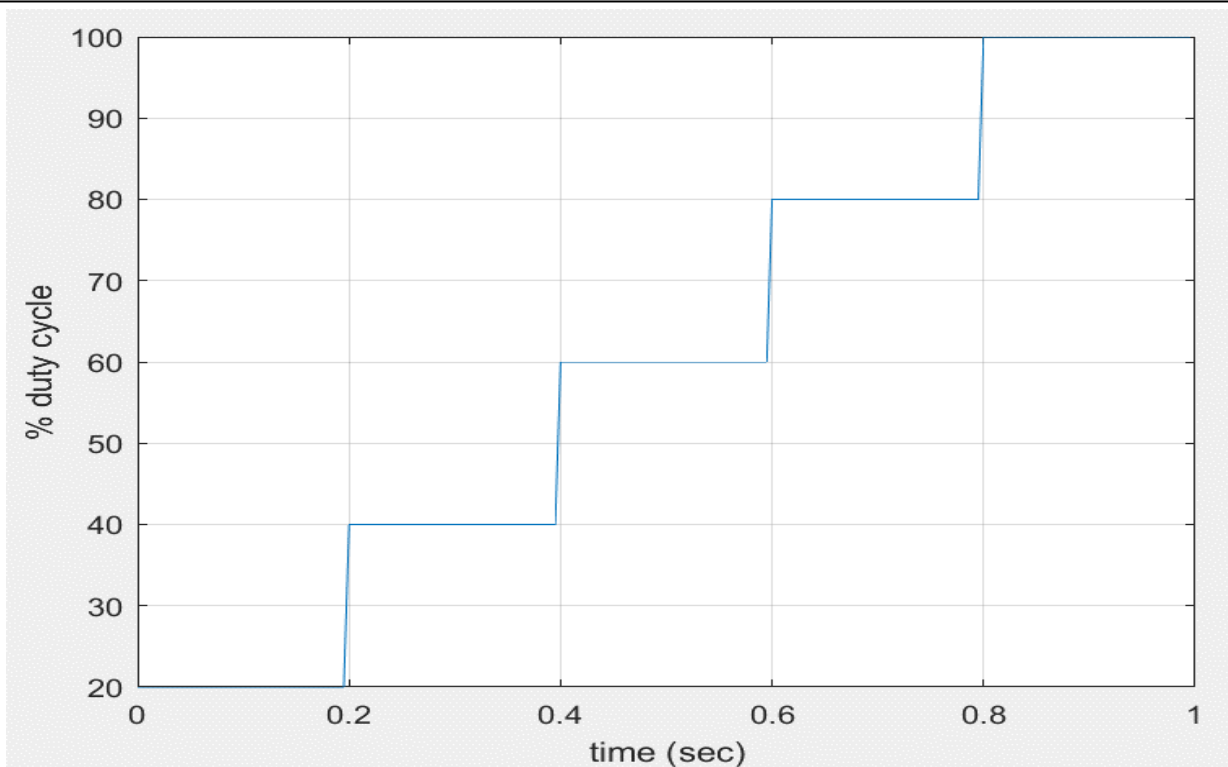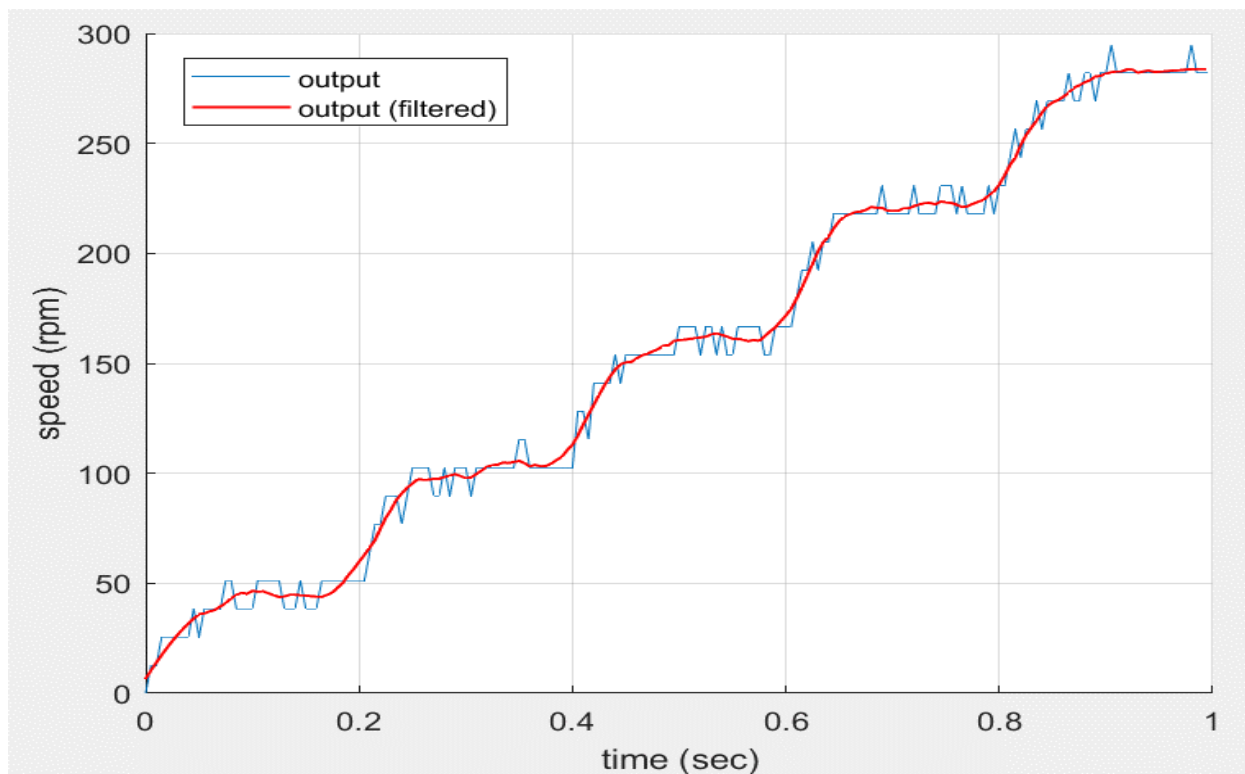| Time	RPM	Time	RPM	Time	RPM	Time	RPM	Time	RPM
0.000	0.00	0.200	51.28	0.400	102.56	0.600	166.67	0.800	230.77
0.005	12.82	0.205	51.28	0.405	128.21	0.605	166.67	0.805	230.77
0.010	12.82	0.210	64.10	0.410	128.21	0.610	179.49	0.810	243.59
0.015	25.64	0.215	76.92	0.415	115.38	0.615	192.31	0.815	256.41
0.020	25.64	0.220	76.92	0.420	141.03	0.620	192.31	0.820	243.59
0.025	25.64	0.225	89.74	0.425	141.03	0.625	205.13	0.825	256.41
0.030	25.64	0.230	89.74	0.430	141.03	0.630	192.31	0.830	256.41
0.035	25.64	0.235	89.74	0.435	141.03	0.635	205.13	0.835	269.23
0.040	25.64	0.240	76.92	0.440	153.85	0.640	205.13	0.840	256.41
0.045	38.46	0.245	89.74	0.445	141.03	0.645	217.95	0.845	269.23
0.050	25.64	0.250	102.56	0.450	153.85	0.650	217.95	0.850	269.23
0.055	38.46	0.255	102.56	0.455	153.85	0.655	217.95	0.855	269.23
0.060	38.46	0.260	102.56	0.460	153.85	0.660	217.95	0.860	269.23
0.065	38.46	0.265	102.56	0.465	153.85	0.665	217.95	0.865	282.05
0.070	38.46	0.270	89.74	0.470	153.85	0.670	217.95	0.870	269.23
0.075	51.28	0.275	89.74	0.475	153.85	0.675	217.95	0.875	269.23
0.080	51.28	0.280	102.56	0.480	153.85	0.680	217.95	0.880	282.05
0.085	38.46	0.285	89.74	0.485	153.85	0.685	217.95	0.885	282.05
0.090	38.46	0.290	102.56	0.490	153.85	0.690	230.77	0.890	269.23
0.095	38.46	0.295	102.56	0.495	153.85	0.695	217.95	0.895	282.05
0.100	38.46	0.300	102.56	0.500	166.67	0.700	217.95	0.900	282.05
0.105	51.28	0.305	89.74	0.505	166.67	0.705	217.95	0.905	294.87
0.110	51.28	0.310	102.56	0.510	166.67	0.710	217.95	0.910	282.05
0.115	51.28	0.315	102.56	0.515	166.67	0.715	217.95	0.915	282.05
0.120	51.28	0.320	102.56	0.520	153.85	0.720	230.77	0.920	282.05
0.125	51.28	0.325	102.56	0.525	166.67	0.725	217.95	0.925	282.05
0.130	38.46	0.330	102.56	0.530	166.67	0.730	217.95	0.930	282.05
0.135	38.46	0.335	102.56	0.535	153.85	0.735	217.95	0.935	282.05
0.140	38.46	0.340	102.56	0.540	166.67	0.740	217.95	0.940	282.05
0.145	51.28	0.345	102.56	0.545	153.85	0.745	230.77	0.945	282.05
0.150	38.46	0.350	115.38	0.550	153.85	0.750	230.77	0.950	282.05
0.155	38.46	0.355	115.38	0.555	166.67	0.755	230.77	0.955	282.05
0.160	38.46	0.360	102.56	0.560	166.67	0.760	217.95	0.960	282.05
0.165	51.28	0.365	102.56	0.565	166.67	0.765	230.77	0.965	282.05
0.170	51.28	0.370	102.56	0.570	166.67	0.770	217.95	0.970	282.05
0.175	51.28	0.375	102.56	0.575	166.67	0.775	217.95	0.975	282.05
0.180	51.28	0.380	102.56	0.580	153.85	0.780	217.95	0.980	294.87
0.185	51.28	0.385	102.56	0.585	153.85	0.785	217.95	0.985	282.05
0.190	51.28	0.390	102.56	0.590	166.67	0.790	230.77	0.990	282.05
0.195	51.28	0.395	102.56	0.595	166.67	0.795	217.95	0.995	282.05

Secondly, a brushed DC motor's transfer function can be estimated with a first-order transfer function with a DC gain *K* and time constant τ (**Figure 5.14**).
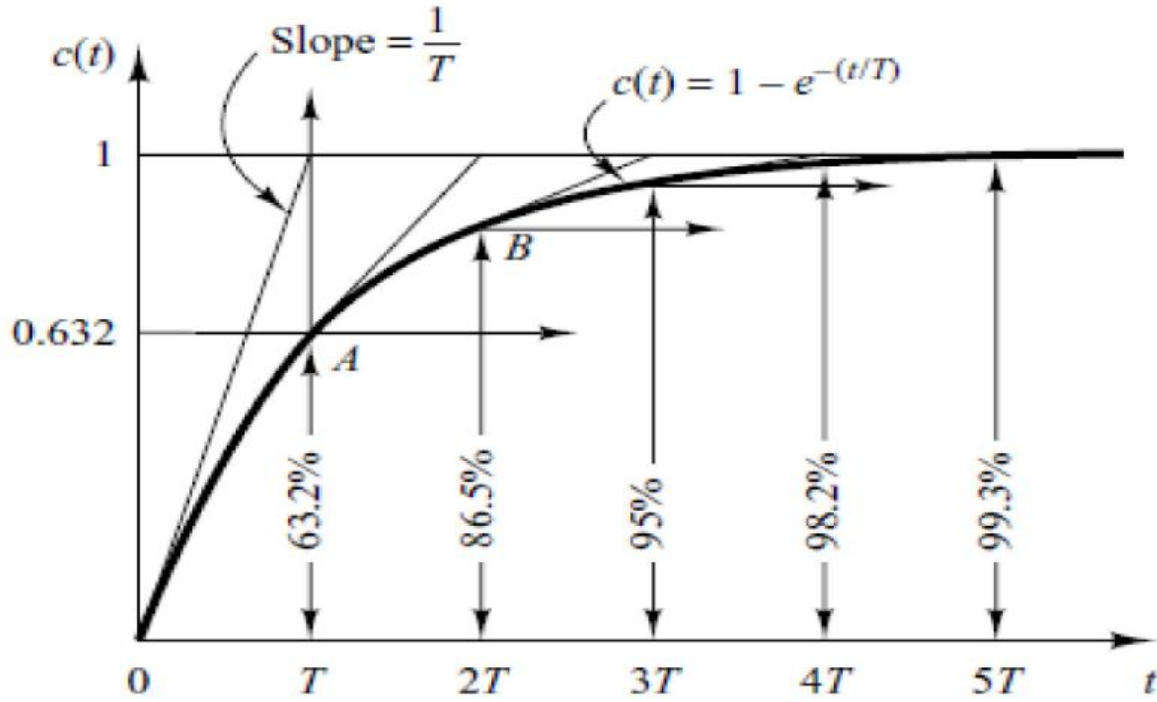

Figure 5.15 Step response of a first-order system

Using System Identification Toolbox in MATLAB and the dataset in **Table 5.5** , we have the final transfer function of the left motor (with TB6612FNG driver, no load):

$$G_{LM}(s) = \frac{\omega(s)}{T(s)} = \frac{2.8411}{0.0486s + 1}$$

The same procedure was done with the right motor. Its response was recorded in the following table and figure.
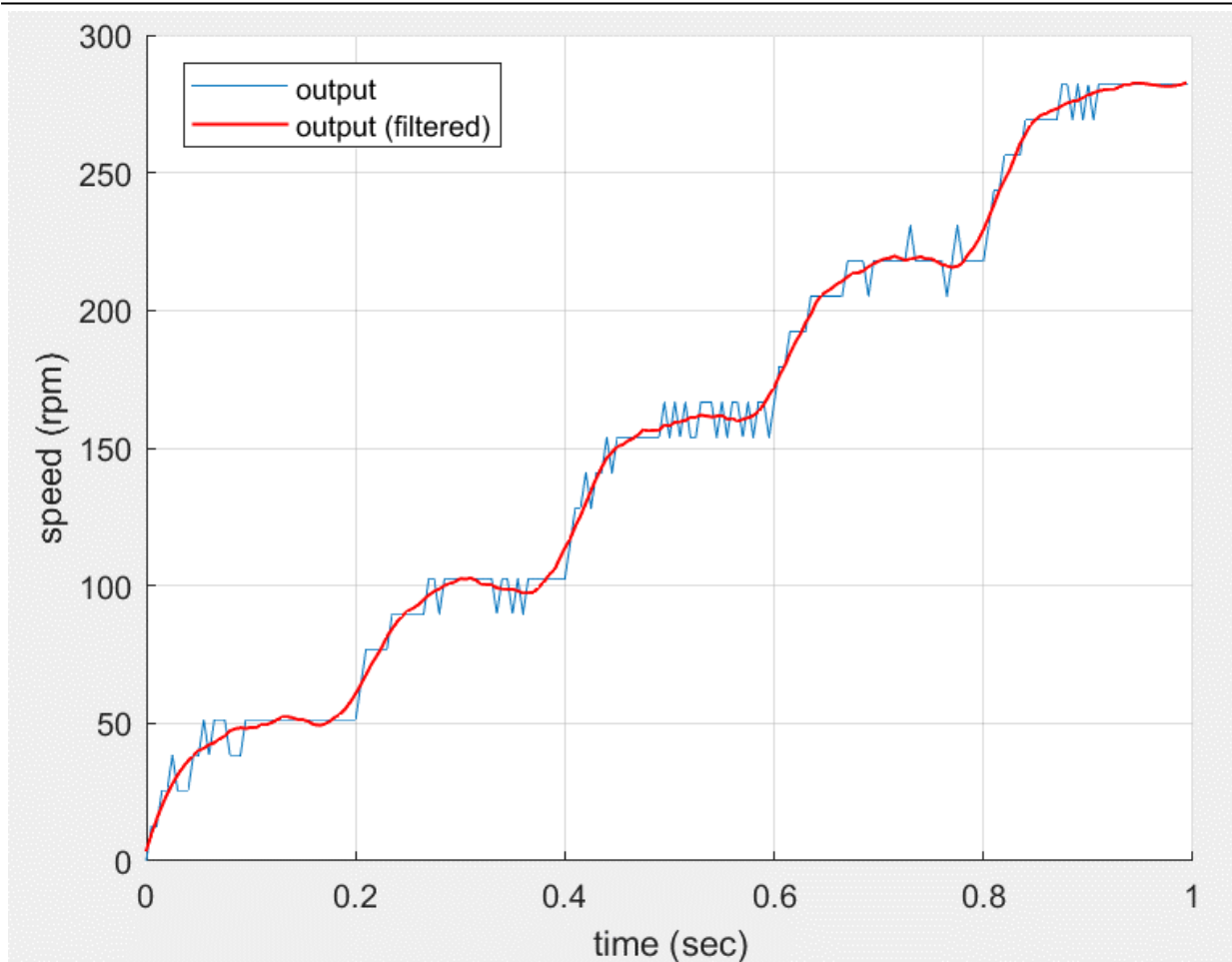
Figure 5.16 Response of the right motor

Table 5.6 Dataset of the right motor

Time	RPM	Time	RPM	Time	RPM	Time	RPM	Time	RPM
0.000	0.00	0.200	51.28	0.400	102.56	0.600	166.67	0.800	217.95
0.005	12.82	0.205	64.10	0.405	115.38	0.605	179.49	0.805	230.77
0.010	12.82	0.210	76.92	0.410	128.21	0.610	179.49	0.810	243.59
0.015	25.64	0.215	76.92	0.415	128.21	0.615	192.31	0.815	243.59
0.020	25.64	0.220	76.92	0.420	141.03	0.620	192.31	0.820	256.41
0.025	38.46	0.225	76.92	0.425	128.21	0.625	192.31	0.825	256.41
0.030	25.64	0.230	76.92	0.430	141.03	0.630	192.31	0.830	256.41
0.035	25.64	0.235	89.74	0.435	141.03	0.635	205.13	0.835	256.41
0.040	25.64	0.240	89.74	0.440	153.85	0.640	205.13	0.840	269.23
0.045	38.46	0.245	89.74	0.445	141.03	0.645	205.13	0.845	269.23
0.050	38.46	0.250	89.74	0.450	153.85	0.650	205.13	0.850	269.23
0.055	51.28	0.255	89.74	0.455	153.85	0.655	205.13	0.855	269.23
0.060	38.46	0.260	89.74	0.460	153.85	0.660	205.13	0.860	269.23
0.065	51.28	0.265	89.74	0.465	153.85	0.665	205.13	0.865	269.23
0.070	51.28	0.270	102.56	0.470	153.85	0.670	217.95	0.870	269.23
0.075	51.28	0.275	102.56	0.475	153.85	0.675	217.95	0.875	282.05
0.080	38.46	0.280	89.74	0.480	153.85	0.680	217.95	0.880	282.05
0.085	38.46	0.285	102.56	0.485	153.85	0.685	217.95	0.885	269.23
0.090	38.46	0.290	102.56	0.490	153.85	0.690	205.13	0.890	282.05
0.095	51.28	0.295	102.56	0.495	166.67	0.695	217.95	0.895	269.23
0.100	51.28	0.300	102.56	0.500	153.85	0.700	217.95	0.900	282.05
0.105	51.28	0.305	102.56	0.505	166.67	0.705	217.95	0.905	269.23
0.110	51.28	0.310	102.56	0.510	153.85	0.710	217.95	0.910	282.05
0.115	51.28	0.315	102.56	0.515	166.67	0.715	217.95	0.915	282.05
0.120	51.28	0.320	102.56	0.520	153.85	0.720	217.95	0.920	282.05
0.125	51.28	0.325	102.56	0.525	153.85	0.725	217.95	0.925	282.05
0.130	51.28	0.330	102.56	0.530	166.67	0.730	230.77	0.930	282.05
0.135	51.28	0.335	89.74	0.535	166.67	0.735	217.95	0.935	282.05
0.140	51.28	0.340	102.56	0.540	166.67	0.740	217.95	0.940	282.05
0.145	51.28	0.345	102.56	0.545	153.85	0.745	217.95	0.945	282.05
0.150	51.28	0.350	89.74	0.550	166.67	0.750	217.95	0.950	282.05
0.155	51.28	0.355	102.56	0.555	153.85	0.755	217.95	0.955	282.05
0.160	51.28	0.360	89.74	0.560	166.67	0.760	217.95	0.960	282.05
0.165	51.28	0.365	102.56	0.565	166.67	0.765	205.13	0.965	282.05
0.170	51.28	0.370	102.56	0.570	153.85	0.770	217.95	0.970	282.05
0.175	51.28	0.375	102.56	0.575	166.67	0.775	230.77	0.975	282.05
0.180	51.28	0.380	102.56	0.580	153.85	0.780	217.95	0.980	282.05
0.185	51.28	0.385	102.56	0.585	166.67	0.785	217.95	0.985	282.05
0.190	51.28	0.390	102.56	0.590	166.67	0.790	217.95	0.990	282.05
0.195	51.28	0.395	102.56	0.595	153.85	0.795	217.95	0.995	282.05

The final transfer function of the right motor – driver system:

$$G_{RM}(s) = \frac{\omega(s)}{T(s)} = \frac{2.7974}{0.0442s + 1}$$

5.5.4. Verify the transfer functions

With the obtained transfer functions, we use the MATLAB function *lsim* with the same input as in **Figure 5.17** to simulate the response of two motors and then compare it with the real response that we measured. The results are shown in the figures below.
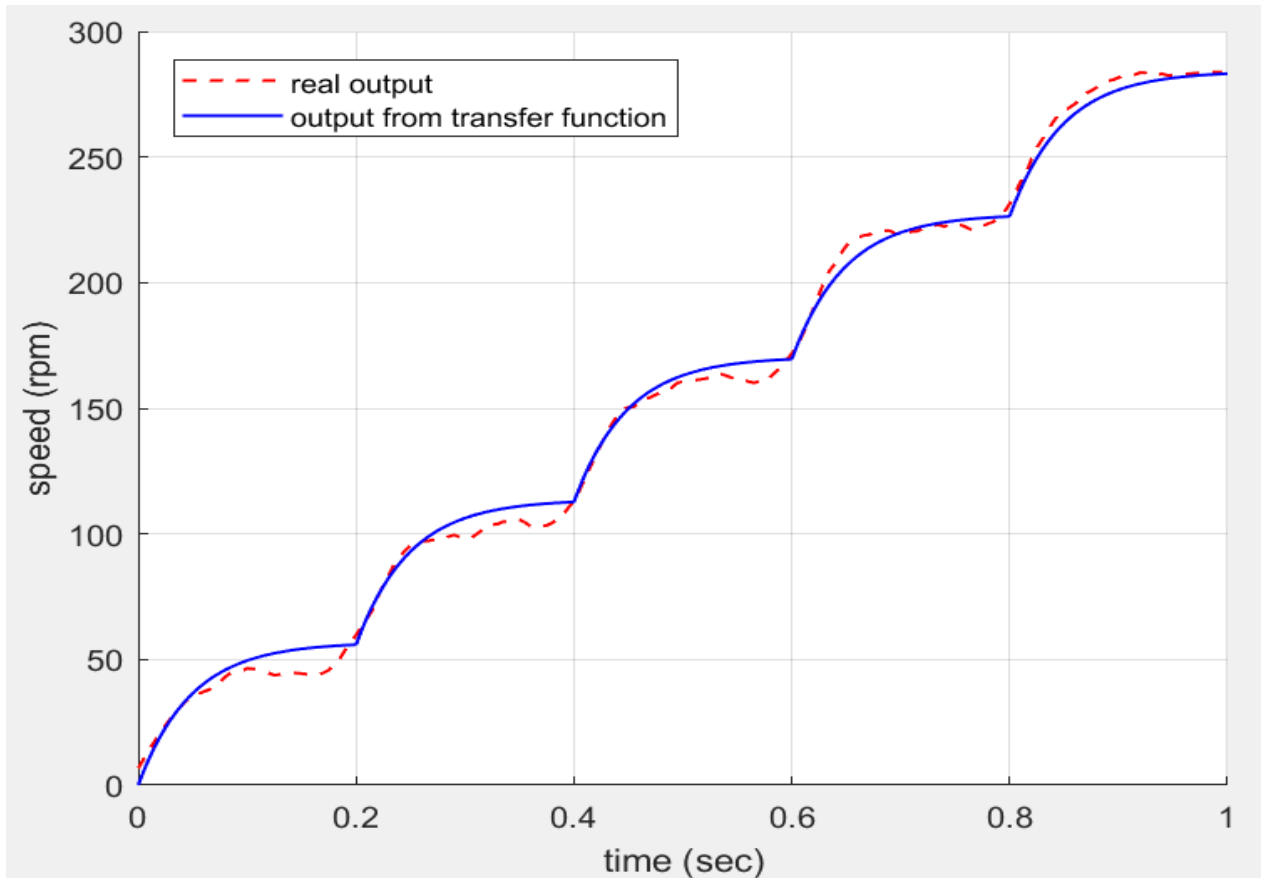

Figure 5.17 Real and simulated response of left motor – driver system
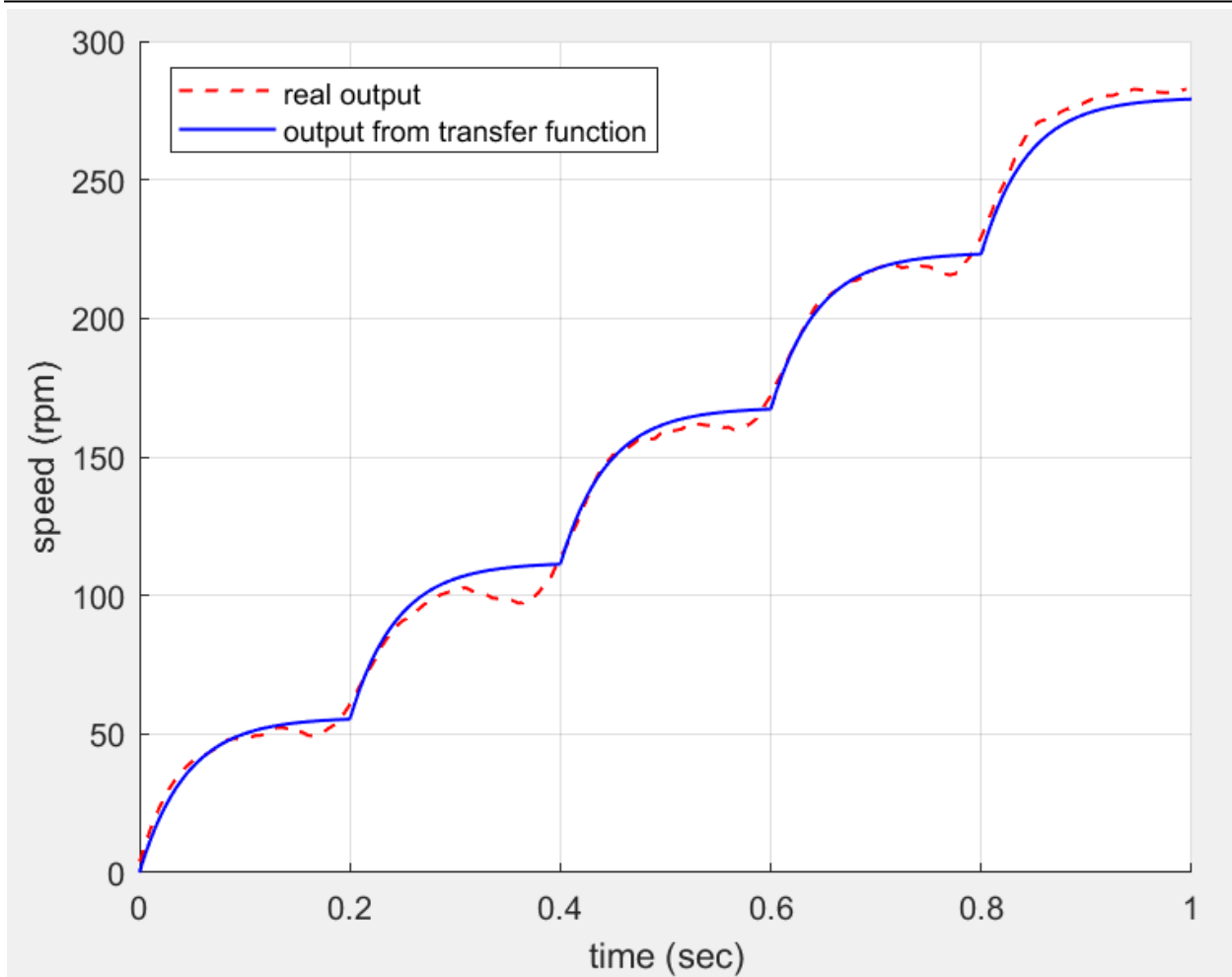
Figure 5.18 Real and simulated response of right motor – driver system

As shown in the figures, the simulated response closely matches with the real one. Therefore, the transfer functions are quite precise and can be used in simulation.

5.5.5. PID controller design

❖ **Requirement for design**

+   Overshoot: 10 %
+   Settling time: should be less than line-following sampling time (0.01s), $T_s <$ 0.01s
+   Small steady state error

The purpose of a PID controller is to ensure that the motors can reach the desired velocity under various load conditions and disturbances. The sampling time of the controller is chosen as 0.005s, which is twice as fast as that of the line-following controller.

Using PID Tuner Toolbox in MATLAB, we obtain the discrete PID controller with K_p = 2.1977, K_i = 75.2996 and K_d = 0.0050. With this controller, the left motor's speed can settle 0.02 second with zero overshoot.
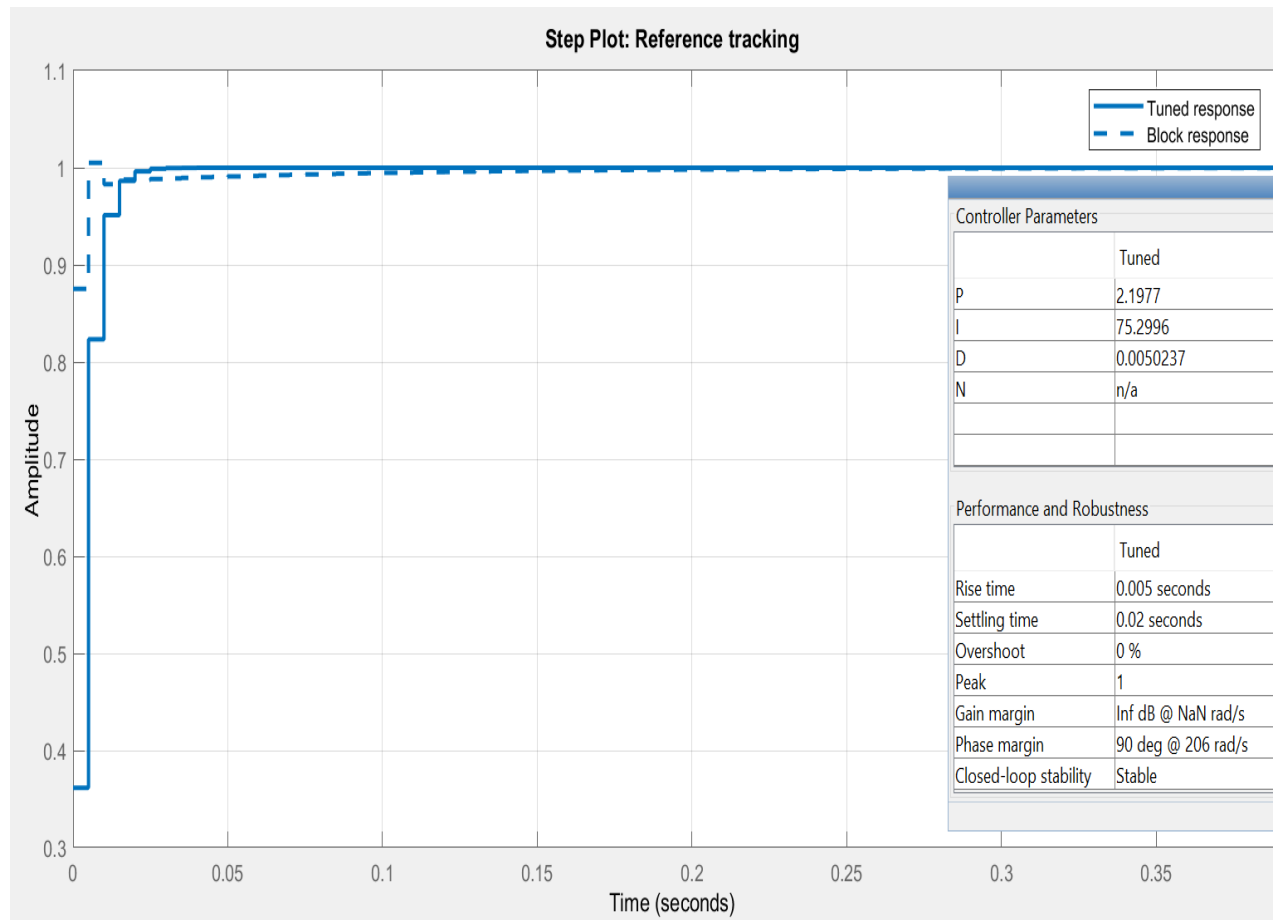

Figure 5.19 Left motor's PID controller with PID Tuner

Similarly, we obtain the discrete PID controller for the right motor with K_p = 2.352, K_i = 74.954 and K_d = 0.0054115. With this controller, the right motor's speed can settle in 0.02 second with zero overshoot and zero steady-state error.
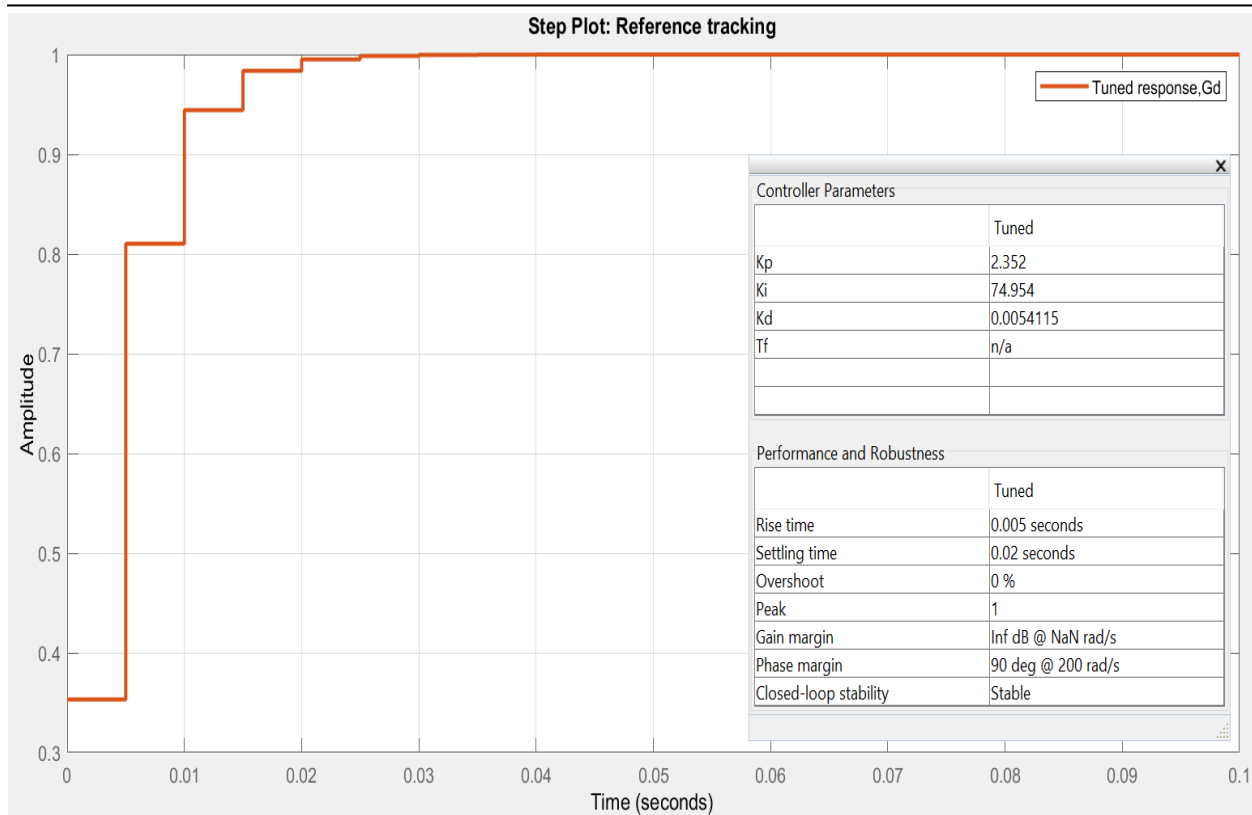
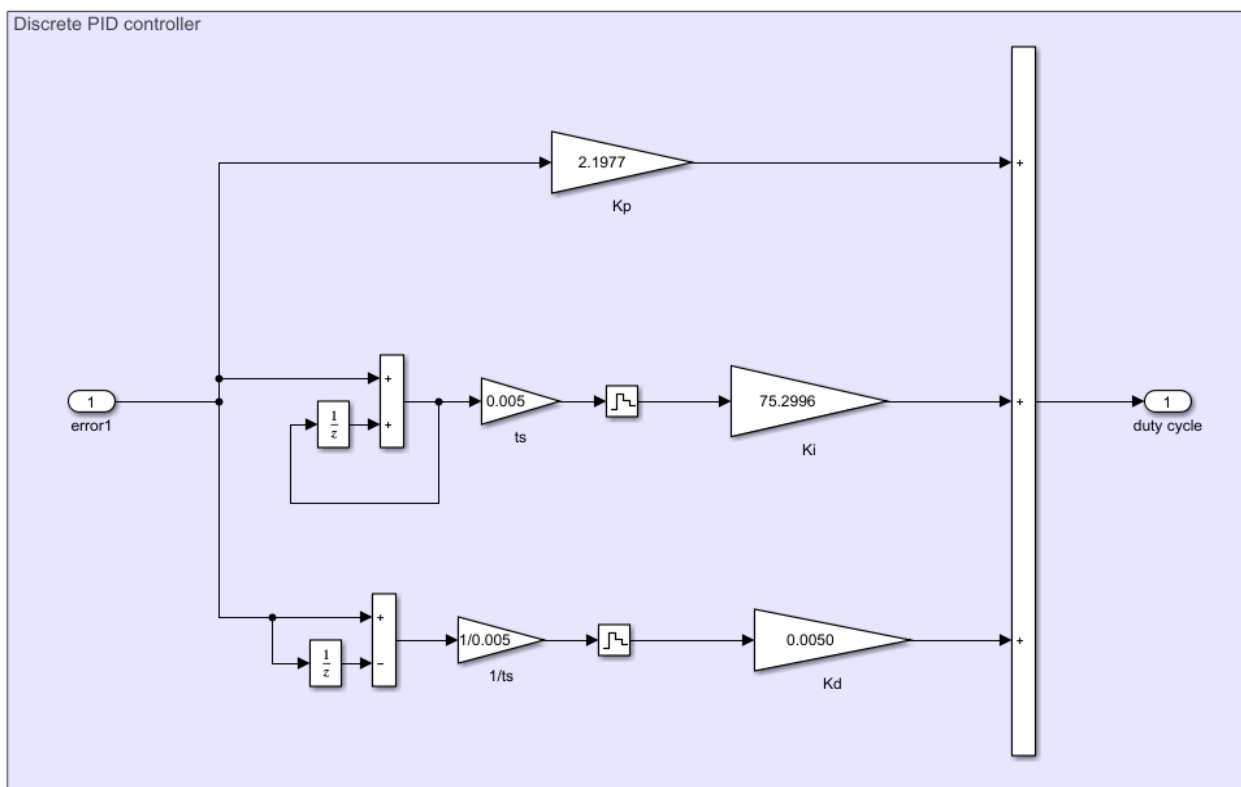**Figure 5.20** Right motor's PID controller with PID Tuner

Figure 5.21 The discrete PID controller block
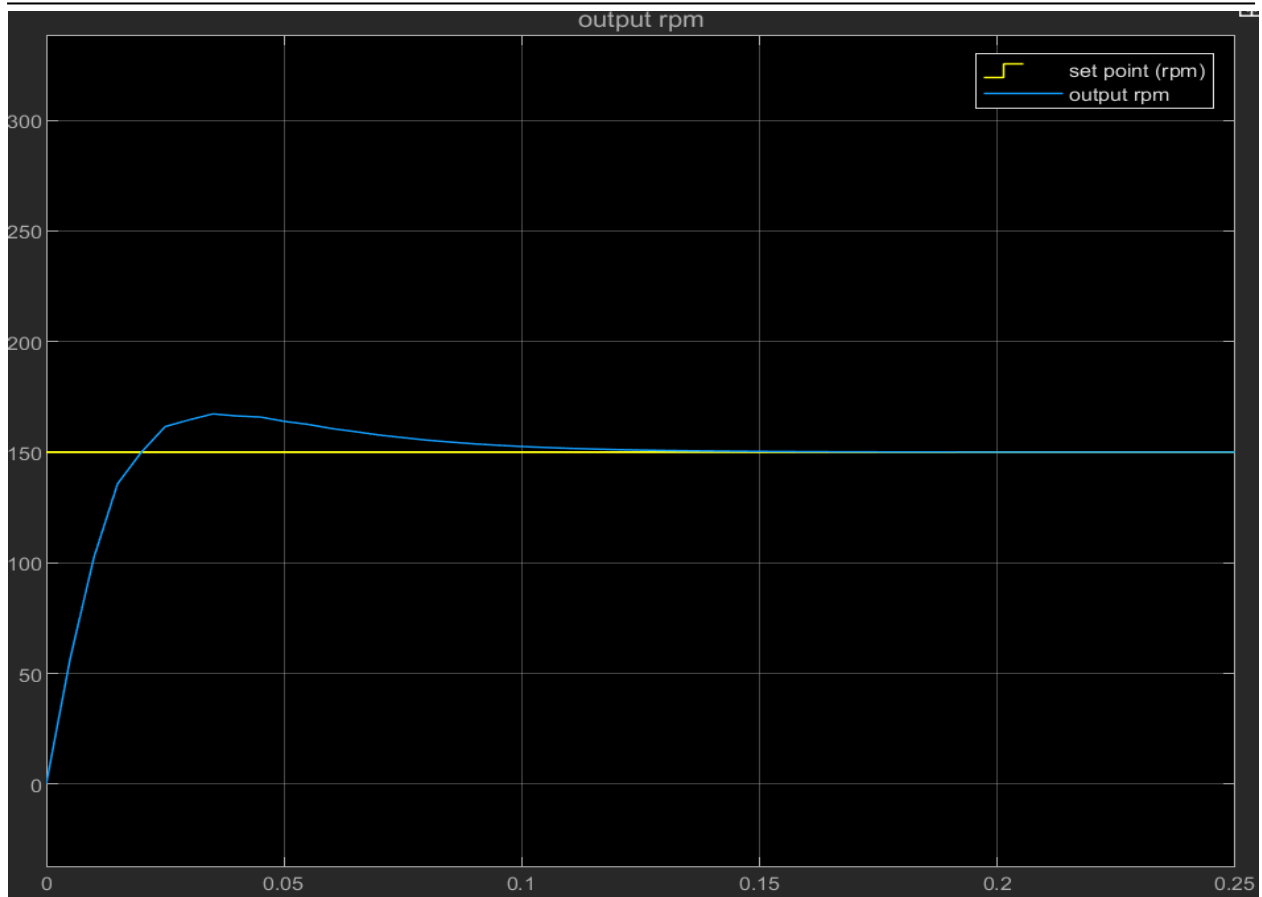
Figure 5.22 Left motor system's response with set point = 150rpm

CHAPTER 6: SIMULATION AND EXPERIMENTAL RESULT

6.1. SIMULATION RESULT

The line following algorithm is simulated in MATLAB with the track width 2b = 180mm, distance between the sensor array and the wheels' axle d = 100mm, the controller's parameters $[k_1 \ k_2 \ k_3] = [50 \ 2.10^{-4} \ 0]$, line-following sampling time $t_s = 0.01$s, motor controllers' sampling time $t_m = 0.005$s.

In simulation with average velocity 0.7 m/s, the robot completes the journey in roughly 16 seconds. Fig 6.1 shows the 3 tracking errors recorded throughout the whole path.

The error e_2 is small (<5mm) when the robot travels in a plain straight line or curve. However, the algorithm will introduce larger error (up to 17mm) when the robot steers from a line to a curve and vice versa (segments BCD, DEB, BFA, AGB).
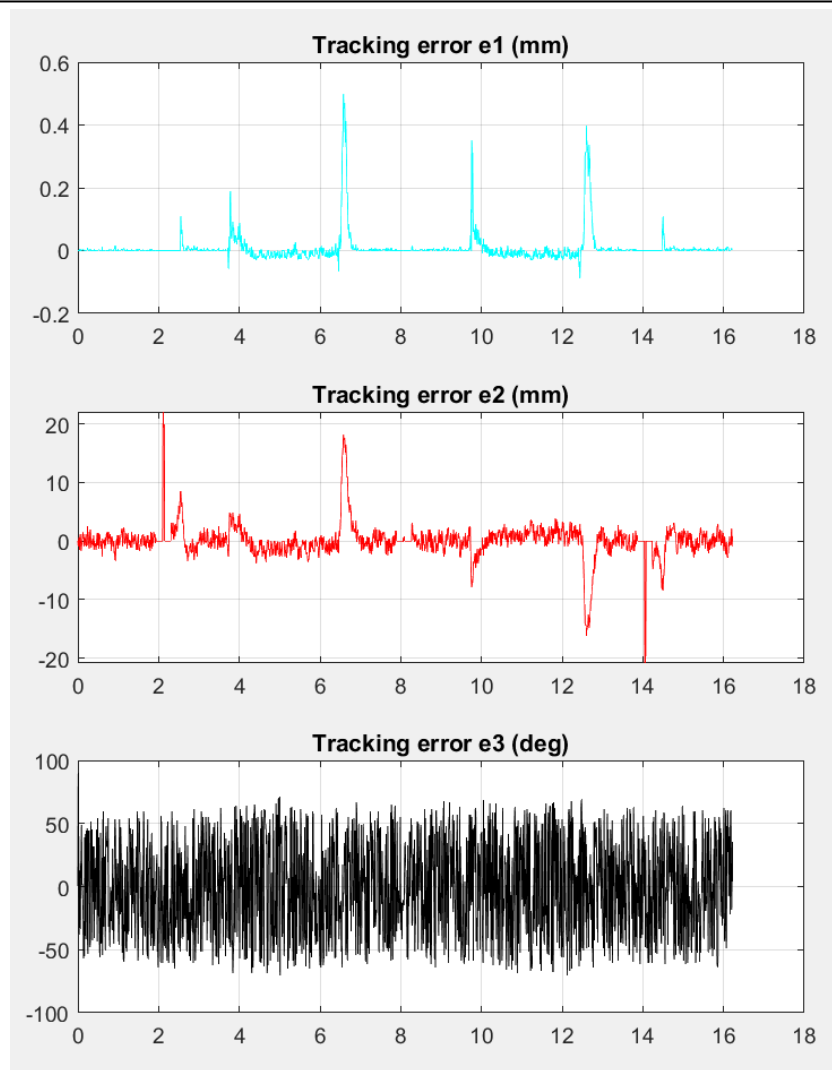
Figure 6.1 Tracking errors throughout the line-following process

The highest error e_2, which is roughly 22mm, is recorded at the junction B where the robot steers at an angle of 26 degrees. The error at other positions is always below 20mm; therefore, the maximum error requirement is satisfied.

Figure 6.2 shows the robot's linear and angular velocity throughout the path. During turning and steering, the robot's velocity will be reduced to prevent its inertia pushing it off the lines.
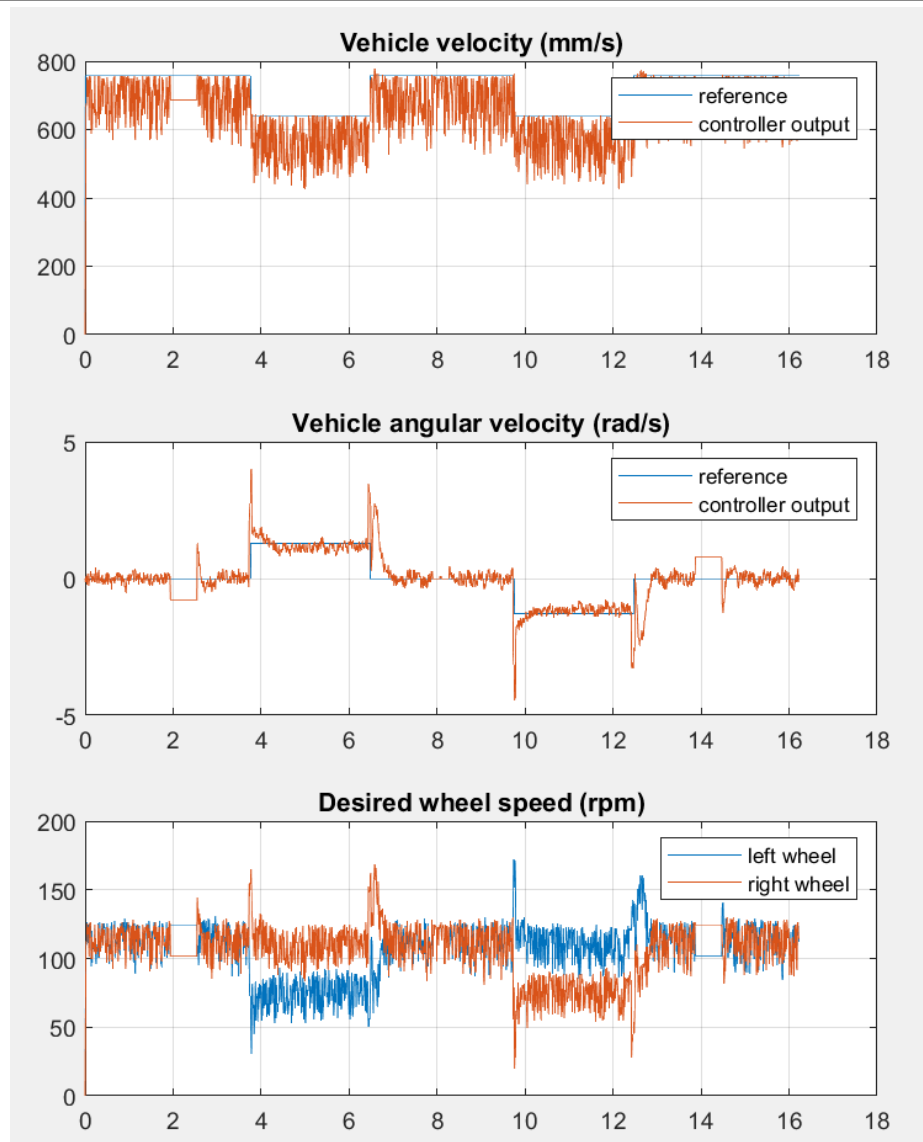
Figure 6.2 Velocity of the robot and its wheels

As shown in the graph, wheels' velocity is within the motors' speed range. The real response of the motors with PID controllers is demonstrated in (**Figure 6.3**.)
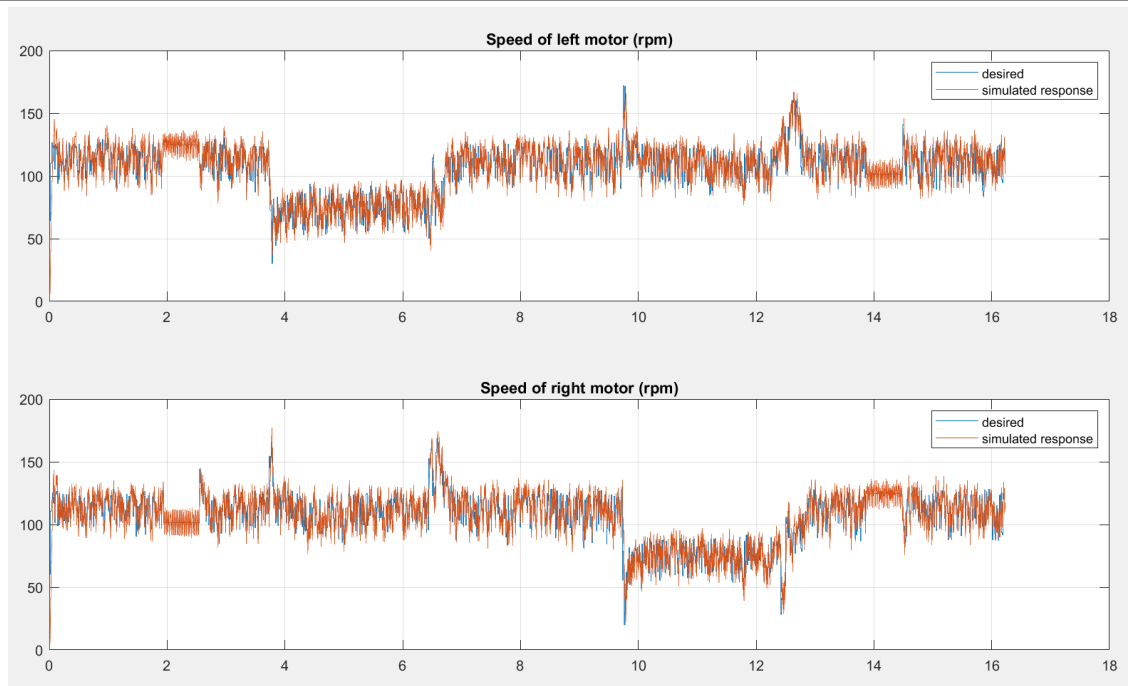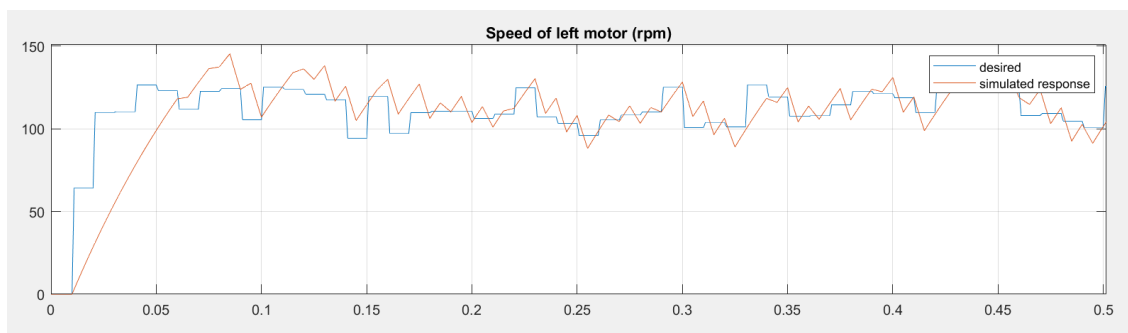
Figure 6.3 Response of the 2 motors throughout the path

A closer look:

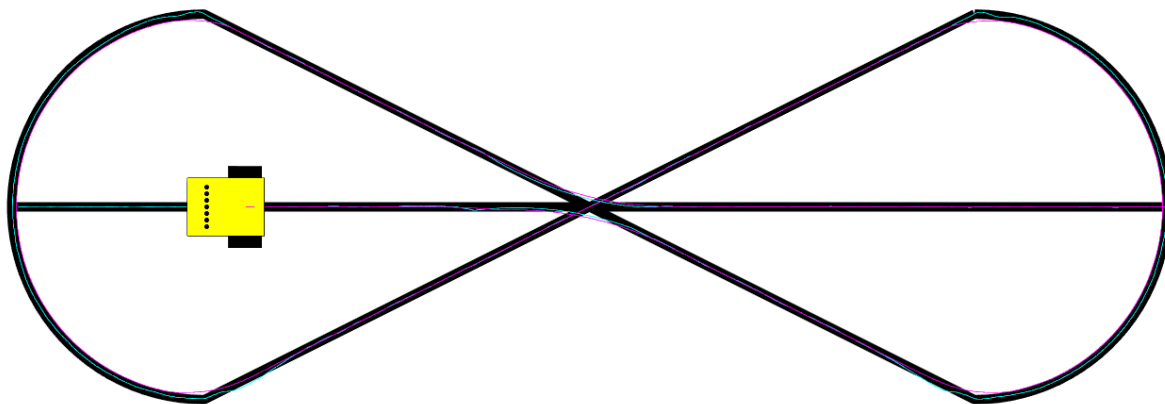The PID controller is sufficient in keeping the motor speed close to the desired value.

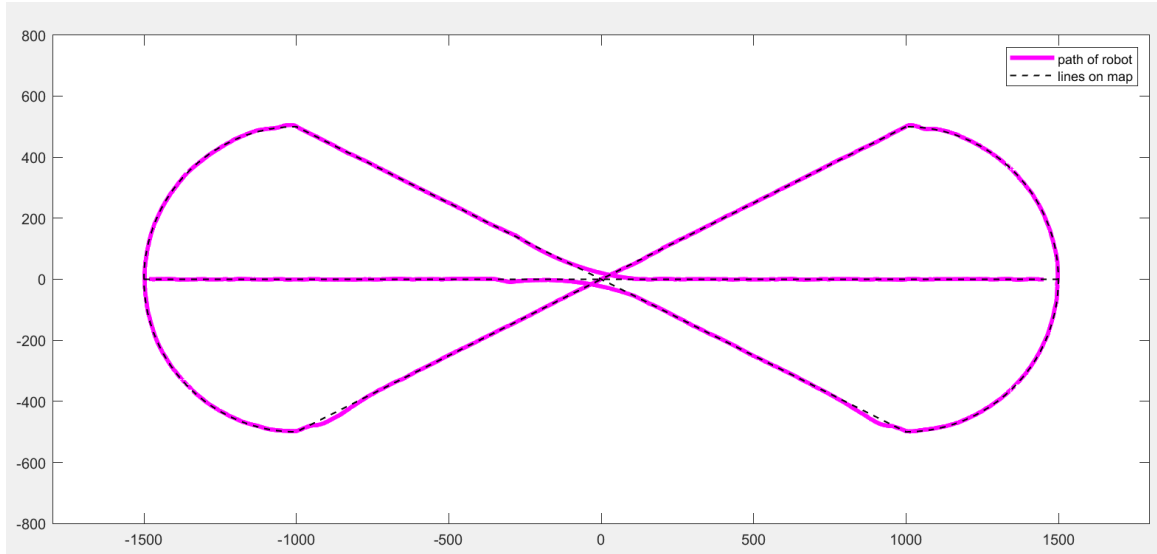**Figure 6.4** Simulation of the line-following process

Figure 6.5 Travel path of the robot

6.2. EXPERIMENT RESULT
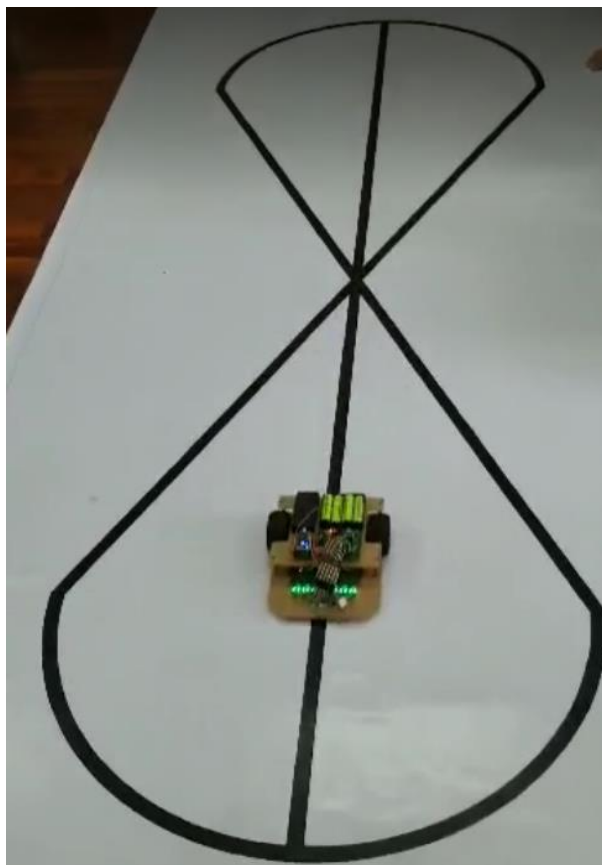
6.2.1. Images on testing robot

Figure 6.6 Experimental run (extracted from video)

6.2.2. Evaluation

TCRT5000 was disturbed by environment very easy.

Redraw the real map was not accuracy because the path was not a continuous path, we need to separate the path into many sub-equations.

The speed of motor depends on the error from sensors which was very easy to be disturbed by environment. The initial turning angel for each sub-function were calculated by hand and not as accuracy as we want.

❖ **Advantages:**
+ The robot followed well on most of the segments and curves of the map.
+ The robot speed was quite fast (about 0.7m/s).
+ The complete time was about 18s seconds.

❖ **Disadvantages:**
+ The robot oscillates more around the line.
+ The maximum error condition of 20mm is not satisfied

❖ **Causes of larger errors during testing include:**
+ Two wheels are not perfectly coaxial
+ Two motors' speed does not match
+ Friction of the steel-ball caster wheel
+ The robot's inertia
+ Error of the sensor array due to different lighting conditions
+ Error of the sensor array due to rough surfaces

❖ **Solutions to reduce these errors:**
+ The robot frame should be made of a stiffer material (e.g. aluminum)
+ Perform experiments to record the sensors' readings in various lighting conditions

❖ **Simulation did not take into account real-world factors such as:**
+ Friction and damping
+ The robot's dynamics
+ Noise and disturbances

REFERENCES

[1] Andrew Reed Bacha, Line Detection and Lane Following for an Autonomous Mobile Robot, MS diss., Virginia Polytechnic Institute and State University, 2005.

[2] G. H. Lee et. al., Line Tracking Control of a Two-Wheeled Mobile Robot Using Visual Feedback,International Journal of Advanced Robotic Systems, DOI: 10.5772/53729, received 4 Apr 2012; Accepted 24 Sep 2012.

[3] Huu Danh Lam et. al., Smooth tracking controller for AGV through junction using CMU camera,Hội nghị Toàn quốc lần thứ 7 về Cơ điện tử - VCM-2014.

[4] A. H. Ismail et. al.,Vision-based System for Line Following Mobile Robot,IEEE Symposium on Industrial Electronics and Applications (ISIEA 2009), October 4- 6, 2009, Kuala Lumpur, Malaysia.

[5] Mustafa Engin, Dilúad Engin, Path Planing of Line Follower Robot,Proceedings of the 5th European DSP Education and Research Conference, 2012.

[6] Nguyễn Tấn Tiến, Trần Thanh Tùng, Kim Sang Bong (2016), Giảng dạy thiết kế hệ thống cơ điện tử qua đồ án, Hội nghị toàn quốc lần thứ 8 về Cơ Điện tử, pp. 416- 422.

 [7] Supriadi el at, Line follower robot optimization based fuzzy logic controller using membership function tuning, International Joural of Engineering & Technology, 7, 112- 116.

[8] Vishay Semiconductors, Application of Optical Reflex Sensors, [online] Available at: https://www.vishay.com/docs/80107/80107.pdf [Accessed 9th December 2020].

[9] Steven Bell (2011). High-Precision Robot Odometry Using an Array of Optical Mice. 2011 IEEE Region 5 Student Paper Contest, Oklahoma Christian University.

[10] (2003, Oct.) The Electronic Lives Manufacturing. [Online]. http://elmchan.org/works/ltc/report.html

[11] D.Jet Cook. (2006) the ultra-fast line following robot. [Online]. www.robotroom.com/Jet.html

[12] James Vroman. (1998) James Vroman Wed site. [Online]. http://james.vroman.com/tecbot1a.htm

[13] VISHAY, TCRT5000, TCRT5000L Datasheet, Document number: 80112, Rev. 1.1, 02-Jul-09

[14] VISHAY, Application of Optical Reflex Sensors TCRT1000, TCRT5000, CNY70, Document number: 80107, Rev. 1.1, 02-02

[15] TOSHIBA Corporation, TB6612FNG Driver IC for Dual DC motor, 2014-10-01