Control System Implementation and Analysis for Omniwheel Vehicle

Andra Bramanta[1], Agus Virgono[2], Randy Erfa Saputra[3]
Computer Engineering, School of Electrical Engineering
Telkom University
Bandung, Indonesia
[1]andra.bramanta@gmail.com, [2] avirgono@telkomuniversity.ac.id, [3]resaputra@telkomuniversity.ac.id

***Abstract*—In this paper, we implement an embedded control system to a four-wheel omniwheel vehicle. PID control is used to stabilize the velocity of each wheel so the vehicle can move steadily. Color markers are used as navigation waypoints and to correct any deviation on the vehicle movement. The implemented system is capable to reduce the vehicle's movement deviation when move straight from one marker to another marker. The color markers can be used with various distance from each other.**

***Keywords*—omnidirectional drive; omniwheel; control system; PID control**

I. INTRODUCTION

Omnidirectional drive of wheeled vehicles is a control mechanism capable to move the vehicle to any direction on 2D plane without changing the vehicle's heading. Omnidirectional drive is used to overcome the problem of vehicle movement in narrow spaces [1].

Omnidirectional drive can be implemented using omniwheel. An omniwheel is a wheel capable to roll forward/backward and sideways, thanks to the presence of small wheels with rotary axes parallel to the plane of wheel's disc.

To drive an omniwheel vehicle, a control system is needed to manage the direction of rotation and speed of each wheel so that the vehicle can move in the desired direction. PID control can be used for that purpose [2][3].

The vehicle can navigate and correct any deviation while moving by using color marks as beacons. The marks can be read by a color sensor attached below the vehicle.

A control system can be implemented as an embedded system for an omnidirectional vehicle with four omniwheels. The vehicle must capable to move from one color mark to another and correct any deviation occured along the movement. The accuracy and the stability of the system will be discussed.

II. SYSTEM DESIGN

By using the color marks, the system receives input from the color sensor which then provide input for the control direction. The system receives a list of movement and will move from one color mark to another according to the list received. The system is also capable of receiving manual input from the operator. Fig. 1 shows overall system diagram.

A. Vehicle Mechanics

Four-wheel omniwheel vehicle used in implementation is made with chassis by using acrylic sheet. The vehicle consist of two levels supported by a spacer. Fig. 2 shows the vehicle used in implementation and Fig. 3 specifies its dimension.

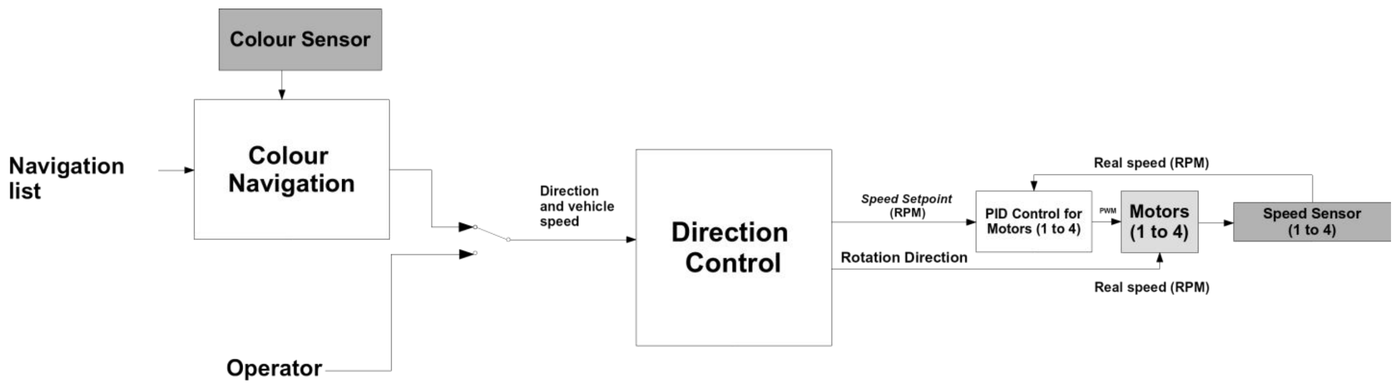

Fig. 1. System Diagram

978-1-5386-1667-3/17/$31.00 ©2017 IEEE

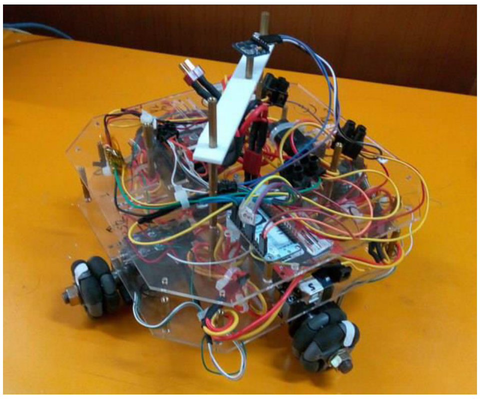

Fig. 2. Vehicle Used in Implementation

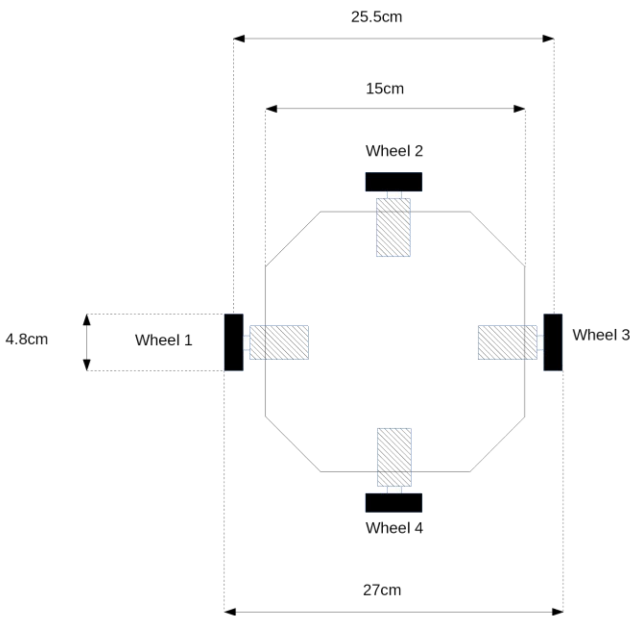

Fig. 3. Vehicle dimension and motor placement

The motor used is a High Power DC motor with a 1:75 gearbox ratio. Here are the motor specifications:

- Working voltage: 6-12v
- RPM output: 130 RPM
- Torque: 12 Kg.cm
- Current without load: 350mA
- Stall current: 4.5 A

Omniwheel used in implementation is using roller made from Thermoplastic rubber, with specification:

- Diameter: 50mm
- Number of rollers: 8
- Roller diameter: 13mm
- Maximum load: 2Kg

B. Direction Control

Direction control will determine the direction of motor rotation and set point for PID control. Direction control also able to control the vehicle into eight cardinal directions, as shown in Fig. 4.

This direction control can be described using formula provided from inverse kinematics from [3]. Based on Fig. 5, the formula can be described in the following:

- The positive y-axis is the front of the vehicle, the positive x-axis is the right direction of the vehicle. The angle value is calculated zero from the positive x-axis direction and increases on counterclockwise rotation.
- R is the radius of the vehicle.
- θ_i , i = 1, 2, 3, 4 is the angle of each wheel from the x-axis.
- Direction of a positive wheel rotation follows the direction of vehicle rotation using the right hand rule with the thumbs up.
- m_1, m_2, m_3, and m_4 is linear velocity of each wheel, which calculated in (1).

$$m_i = r_{\text{wheel}}\omega_{\text{wheel}} \quad (1)$$

The r_{wheel} is the wheel radius and ω_{wheel} is the angular velocity of wheel. The m_i can also be calculated in (2).

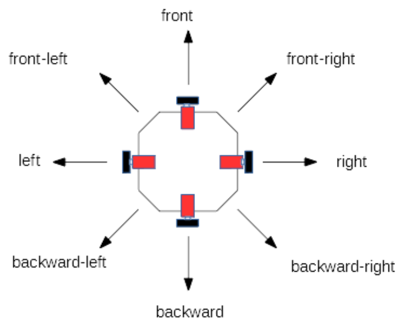

Fig. 4. Direction supported by vehicle direction control

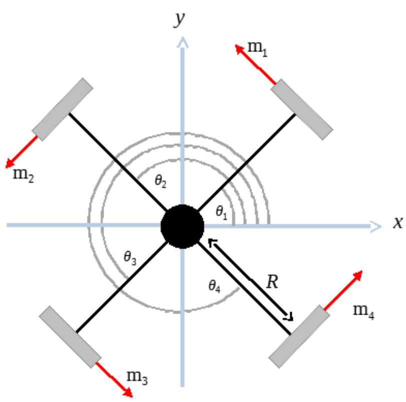

Fig. 5. Inverse kinematics for Omnidirectional Vehicle

$$m_i = -v_x \sin\theta_i + v_y \cos\theta_i + R\omega \quad (2)$$

The v_x and v_y are velocity component in x-axis and y-axis respectively, and ω is angular velocity of the vehicle. After then, the revolutions per minute (RPM) of the wheel is calculated in (3).

$$RPM_i = \frac{30}{\pi} \frac{m_i}{r_{\text{wheel}}} \quad (3)$$

C. PID Control

The PID control is implemented using the Arduino PID library with a sample time of 100ms. The Arduino PID library provides a good PID implementation for microcontroller devices using the Arduino platform [4][5]. Table 1 shows PID constant used in this implementation, and Fig. 6 shows the motor response.

PID controls are manually and online tuned with trial and error method, until satisfy the following targets:

- Capable of providing smooth response and almost the same for each motor, the difference in response will cause the vehicle to turn when it starts moving.
- No significant overshoot.

Manual tuning method used in this implementation:

1. Set *Kp* of motor #1 with small value.
2. Increase *Ki* of motor #1, then stop when overshoot occurs.
3. Reset *Kp* again, increase it, then stop when overshoot or undershoot happen.
4. Set *Ki* and *Kp* for all motors based on tuning of motor #1.
5. Tuning each motor sufficiently to have the same response as motor #1.

The *Kd* constant can not be used in the implementation, because unstable oscillations occurs even *Kp* is set at small value. This occurs due to unstable sample time or due to high noise from encoder reading [4] or requiring a higher resolution encoder [6].

D. Color Mark Navigation Control

Vehicle is designed to be able to navigate from one color mark to another color mark. Color marks are laid on ground to be read by the color sensor. The signal of the marks represent these commands:

1. forward
2. left
3. stop

Fig. 7 shows the vehicle movement by using color mark. It is supposed to move forward from the initial color mark to the next color mark and then stop. It continues to move left to another color mark and then stop. The navigation and correction utilizes omniwheel vehicle capabilities to move in different directions without having to rotate.

Color mark consists of five colors, main color at the center used for navigation and four colors at the sides used for deviation correction. These four colors provide clue for the vehicle where to move to the center of the mark as shown in Fig. 8. Color mark navigation control is described as flowchart in Fig. 9.

TABLE I. PID CONSTANT

Motor	*Kp*	*Ki*	*Kd*
1	0.3	4	0
2	0.3	4	0
3	0.3	4	0
4	0.3	4	0

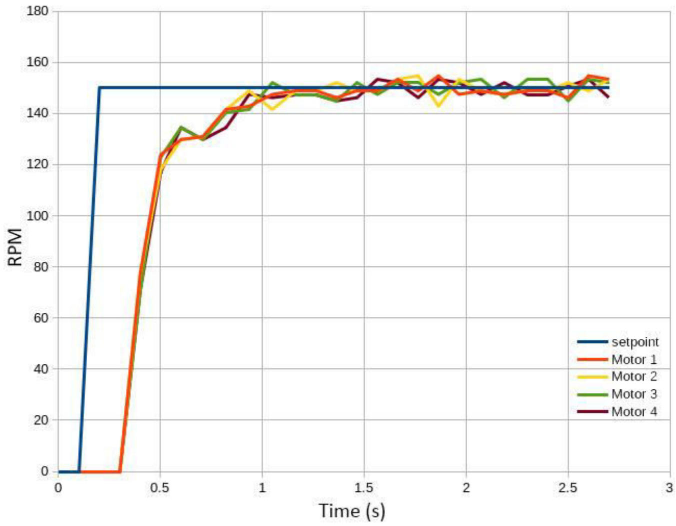

Fig. 6. PID control responses

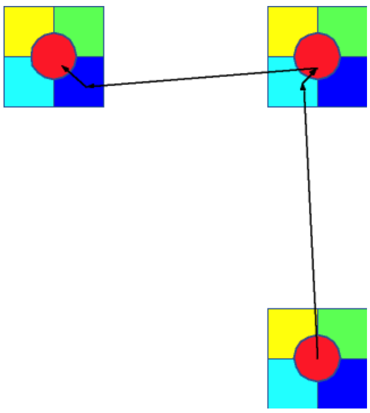

Fig. 7. Vehicle movement using color mark

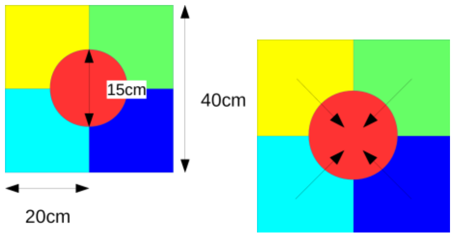

Fig. 8. Color mark dimension and correction direction in color mark

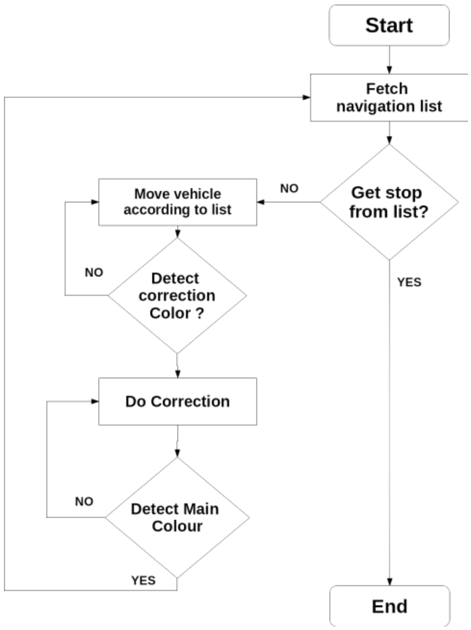

Fig. 9. Flowchart of color mark navigation control

E. *Embedded System*

Fig. 10 shows overall embedded system that has been implemented. The embedded system consists of a primary microcontroller where control system software is implemented and two secondary microcontroller that controls the motor driver and encoder readings. The primary microcontroller uses ESP8266 and the secondary microcontroller uses Arduino UNO.

Both primary and secondary microcontrollers are connected using TWI (Two Wire Interface) connection. The TWI speed used in this system is 100 kHz. The speed is sufficient for the purpose of data transmission and commands between the main and secondary microcontrollers.

Motor driver used is VNH2SP30 from STMicroelectronics. VNH2SP30 provides H-Bridge with PWM speed control. This motor driver is able to work up to 16V and 16A in continuous current. VNH2SP30 is also has overcurrent, overvoltage and undervoltage protection. The selection of this motor driver provides a safe margin for the operation of high power DC motors used in this vehicle.

The incremental encoder discs used is made from acrylic, using laser cutting with resolution of 90 count per revolution. Encoder disks are mounted on each wheel and read using optical interrupter QVE00112 from Fairchild semiconductor. QVE00112 is chosen because it has a small gap of 0.356mm so it can read encoders that have been created. Encoder pulse count is read using interrupt by secondary microcontroller. Primary controller read this pulse count every 100ms.

TCS3200 is used as color sensor in this implementation, which uses three channels (R, G, B) as color to frequency converter. The sensor is connected to the main microcontroller and is able to recognize 5 color classes. Color recognition is done by reading a color sample and then taking a range of values for each color class.

III. TEST AND ANALYSIS

A. *Deviation Test*

Vehicles is moved straight from point to point with distance of 1m and 3m while the deviation from end point is measured as in Fig. 11. Testing without PID control is also done to compare the ability of the control system in reducing the deviation. Fig. 11 also show vehicle headings used in this test.

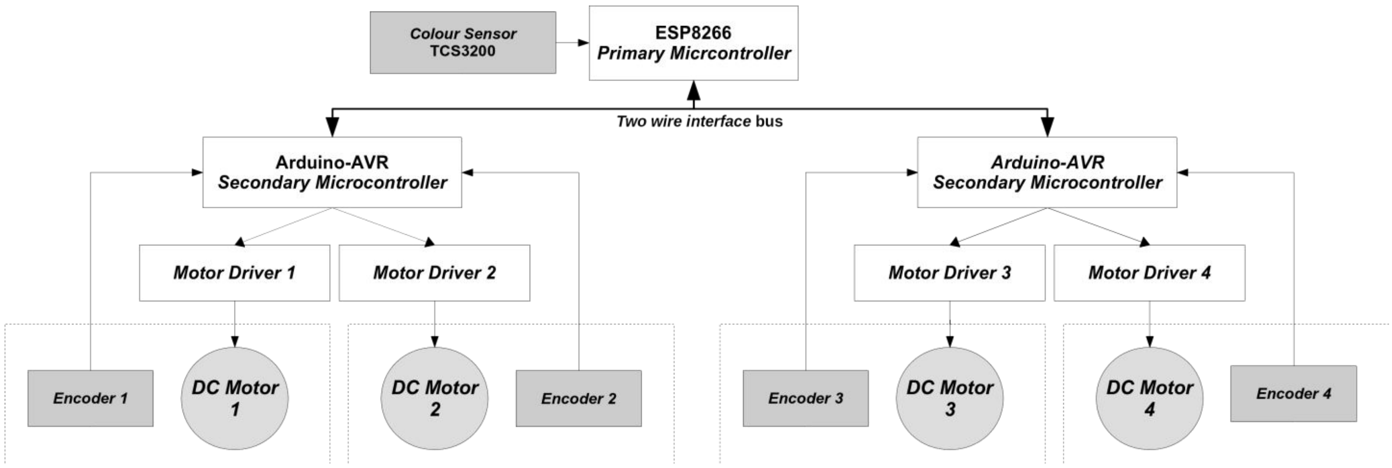

Fig. 10. Embedded system diagram

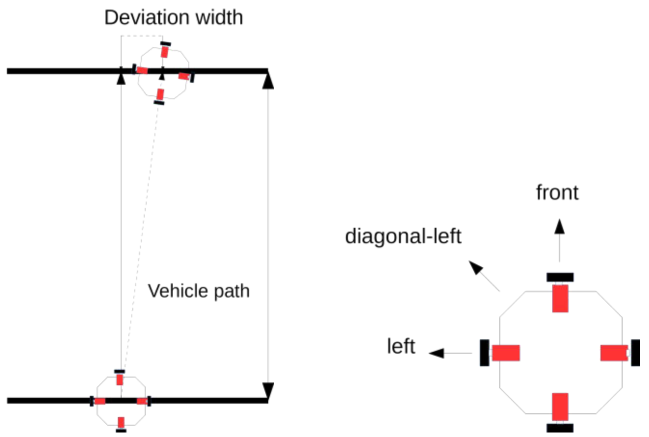

Fig. 11. Vehicle deviation while moving and heading used in deviation test

Table II and III show the result of this test with difference in deviation at 1m distance, meanwhile Table IV and V show the result at 3m distance.

Speed control implemented in the vehicle is able to reduce vehicle deviation from 65.85% to 86.85% with an average accuracy of 12.36cm in 3m straight movement. During testing, the vehicle sometimes slip. This happens because of the height difference between the wheels. The slip may cause the vehicle to turn when it starts to accelerate and PID control on each wheel fails to stabilize the vehicle.

In addition, sometimes PID control is unstable due to errors of encoder speed readings from main microcontroller to secondary microcontrollers. This can result an unstable vehicle. There are some other analysis of why deviation occur:

- PID speed control still has a different responses when the motor starts to move simultaneously,
- PID response is not fast enough to correct vehicle speed in case of slippage or interference

B. *Color Mark Navigation Test*

The test is performed to determine the ability of the vehicle to move from the color mark to another color mark and to know the ability of using the color mark to correct the deviation of the vehicle so that it can go back to the center of the mark before moving to the next. Test is done by using three color mark with the distance of 1.5m, 2.5m and 3.5m. The vehicle will move forward, right and diagonal-left backward to its original position, as shown in Fig. 12.

At the distance of 1.5m, all navigation marks are passed well, resulting a maximum deviation of ~10cm at 1.5m and the color mark can still correct the deviation. Besides, deviation can accumulates in each color mark until the deviation becomes very large and vehicle cannot get to the next color mark. Table VI, VII, and VII show the test results in detail.

TABLE II. DEVIATION TEST AT 1M DISTANCE

Test	Vehicle Headings					
	Front		Left		Diagonal Left	
	PID	Non-PID	PID	Non-PID	PID	Non-PID
1	1.50 cm	9.30 cm	1.70 cm	2.70 cm	1.10 cm	9.20 cm
2	2.50 cm	8.20 cm	1.30 cm	0.30 cm	2.70 cm	5.30 cm
3	2.70 cm	9.70 cm	0.90 cm	0.50 cm	1.10 cm	11.30 cm
4	2.40 cm	7.20 cm	1.30 cm	3.80 cm	3.40 cm	11.10 cm
5	1.90 cm	8.70 cm	1.90 cm	0.80 cm	3.40 cm	7.90 cm
6	1.30 cm	9.40 cm	1.50 cm	6.90 cm	2.30 cm	11.80 cm
7	0.60 cm	9.40 cm	1.10 cm	4.60 cm	2.20 cm	9.10 cm
8	0.80 cm	6.30 cm	1.60 cm	3.20 cm	0.70 cm	11.20 cm
9	2.10 cm	5.40 cm	0.40 cm	7.30 cm	3.60 cm	7.80 cm
10	1.30 cm	4.20 cm	1.10 cm	0.40 cm	1.50 cm	10.70 cm
Average	**1.71 cm**	**7.78 cm**	**1.28 cm**	**3.05 cm**	**2.20 cm**	**9.54 cm**

TABLE III. DIFFERENCE IN DEVIATION AT 1M DISTANCE

Difference in Deviation		
Front	Left	Diagonal Left
6.07 cm	1.77 cm	7.34 cm
78.02%	70.52%	76.94%

TABLE IV. DEVIATION TEST AT 3M DISTANCE

Tests	Vehicle Headings					
	Front		Left		Diagonal Left	
	PID	Non-PID	PID	Non-PID	PID	Non-PID
1	16.60 cm	45.30 cm	2.90 cm	4.80 cm	7.20 cm	73.20 cm
2	9.40 cm	25.40 cm	16.80 cm	7.40 cm	8.20 cm	95.40 cm
3	24.00 cm	53.60 cm	4.30 cm	45.30 cm	6.90 cm	81.40 cm
4	27.00 cm	58.60 cm	7.70 cm	31.20 cm	7.40 cm	68.70 cm
5	7.50 cm	104.70 cm	0.90 cm	64.20 cm	9.90 cm	78.30 cm
6	3.20 cm	78.80 cm	4.60 cm	7.30 cm	2.30 cm	87.30 cm
7	4.40 cm	10.40 cm	10.30 cm	5.70 cm	9.80 cm	47.10 cm
8	5.00 cm	37.90 cm	5.60 cm	2.50 cm	4.10 cm	26.20 cm
9	3.20 cm	20.40 cm	6.40 cm	18.20 cm	8.30 cm	28.70 cm
10	23.30 cm	31.30 cm	10.40 cm	18.10 cm	16.10 cm	23.80 cm
Average	**12.36 cm**	**46.64 cm**	**6.99 cm**	**20.47 cm**	**8.02 cm**	**61.01 cm**

TABLE V. DIFFERENCE IN DEVIATION AT 3M DISTANCE

Difference in Deviation		
Front	Left	Diagonal Left
34.28 cm	13.48 cm	52.99 cm
73.50%	65.85%	86.85%

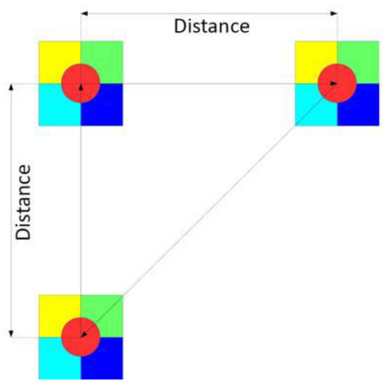

Fig. 12. Colour Mark Navigation Test Case

TABLE VI. COLOR MARK NAVIGATION AT 1.5M DISTANCE

Tests	Colour Mark Passed		
	Forward	Right	Diagonal-left backward
1	Pass	Pass	Pass
2	Pass	Pass	Pass
3	Pass	Pass	Pass
4	Pass	Pass	Pass
5	Pass	Pass	Pass
6	Pass	Pass	Pass
7	Pass	Pass	Pass
8	Pass	Pass	Pass
8	Pass	Pass	Pass
10	Pass	Pass	Pass

TABLE VII. COLOR MARK NAVIGATION AT 2.5M DISTANCE

Tests	Colour Mark Passed		
	Forward	Right	Diagonal-left backward
1	Pass	Pass	Fail
2	Pass	Pass	Pass
3	Pass	Pass	Pass
4	Pass	Fail	Fail
5	Pass	Pass	Pass
6	Pass	Pass	Pass
7	Pass	Fail	Fail
8	Pass	Fail	Fail
8	Pass	Fail	Fail
10	Pass	Pass	Pass

TABLE VIII. COLOR MARK NAVIGATION AT 3.5M DISTANCE

Tests	Colour Mark Passed		
	Forward	Right	Diagonal-left backward
1	Pass	Fail	Fail
2	Pass	Fail	Fail
3	Fail	Fail	Fail
4	Pass	Fail	Fail
5	Pass	Fail	Fail
6	Pass	Pass	Pass
7	Pass	Fail	Fail
8	Pass	Fail	Fail
8	Pass	Fail	Fail
10	Pass	Fail	Fail

Several analysis drawn from this test:

- Navigation marks can be installed in every 1.5 meters to get a pretty good accuracy of 3 color marks.
- The accuracy that can be corrected with the color mark is at the size of the correction mark, which is 20 cm.
- The color marks cannot correct altered vehicle heading due to the accumulated deviation.
- Color sensor is sensitive to ambient lighting.

IV. CONCLUSION

From the conducted testing and analysis, it can be concluded that:

1) The control system is successfully implemented as an embedded system.

2) PID speed control is able to reduce the deviation of vehicle with a ratio of 65.85% to 86.85% with an average accuracy of 12.36 cm in 3 meters distance of motion.

3) The color marks capable of correcting the deviation of the vehicle, so the vehicle can move from the color mark to another color mark up to 1.5 meters.

Suggestions that can be given by the author for further development are:

1) Use encoder with higher resolution for PID control to make more accurate and responsive system. All PID controls may use a higher sample rate.

2) Implement PID tuning for each direction of vehicle movement, so that each wheel can move together with higher accuracy.

3) PID can be tuned with different method, such as, by using numerical algorithm, or using different sampling time and controller's parameter.

4) Color sensors are not suitable for use in environments that have a various light intensity, such as outdoor, so alternative sensors are required.

5) Use error detection for communication between microcontrollers to improve the reliability of the control system as a whole.

REFERENCES

[1] S. Soni, T. Mistry, and J. Hanath, "Experimental analysis of mecanum wheel and omni wheel," International Journal of Innovative Science, Engineering & Technology, vol. 1, issue 3, May 2014.

[2] M. West and H. Asada, "Design of a holonomic omnidirectional vehicle," in Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on, 1992, pp. 97–103.

[3] R. Rojas and A. G. Förster, "Holonomic control of a robot with an omnidirectional drive," KI-Künstl. Intell., vol. 20, no. 2, pp. 12–17, 2006

[4] T. Wescott, "PID without a PhD," Embed. Syst. Program., vol. 13, no. 11, pp. 1–7, 2000.

[5] B. Beauregard, "Improving the Beginner's PID." [Online]. Available: http://brettbeauregard.com/blog/2011/04/improving-the-beginners-pid-introduction/. [Accessed: 13-May-2017]

[6] M. Johnson and M. Moradi, PID Control: New Identification and Design Methods. London: Springer, 2005.