Driving control module for low cost industrial automated guided vehicle

ES Maarif¹ and T Moyo²

¹,²Automation Laboratory, Astra Manufacturing Polytechnics, Jakarta, Indonesia

E-mail: eka.samsul.maarif@gmail.com

Abstract. Automatic Guided Vehicles (AGVs) are computer-controlled “driverless” mobile vehicles equipped with optical, magnetic or laser guidance systems for automated functionality, categorized as either load carrying or load towing. AGV consists of several parts, such as the frame for pulling or lifting weights, movement balancer wheels, battery housing, collision avoidance sensors, driving module and others. But, the most important part of AGV is the driving module which is a module that functions as a movement guide, consisting of path detection sensors, driving motors and controllers. This Paper presents low cost a driving control module for low cost AGV that used in automotive manufacture industry. The AGV uses Raspberry Pi instead of PLC as controller, infrared sensor to detect line and industrial standard brushless DC (BLDC) motor as main actuator. PID control applied in the movement of the BLDC motor speed so that the AGV can follow the path smoothly and precisely. As a result, the AGV can follow the path with speed 30 m/min on a straight road and 30 m/m at a turn with a deviation 5 mm off the track.

1. Introduction

Material handling problem in industrial field consists of how to transport raw materials, partially manufactured products and goods between different locations of manufacturing systems, warehouses, etc. Depending on the kind of products to handle and the transport to perform, there are different solutions in the market. For instance, we can mention belt, roller and vertical conveyors, elevators, material handling robots and automated guided vehicles (AGVs) [1].

AGVs also have several advantages inherent to their design, such as the reduction of product damage from removal of human error, the ability to travel into hazardous areas without concern for operator safety, the ability to automatically track and record product movement, the reduction of labour, and the flexibility and adaptability especially prevalent in laser guided systems. AGV also plays main role in Flexible manufacturing systems (FMS), which are equipped with several CNC machines and AGV-based material handling system are designed and implemented to gain the flexibility and efficiency of production [2].

The basic principle of the operation of traditional AGV system is in a pre-defined route placed on the floor or located in the floor, which is followed by reading device, which is a part of each traditional AGV. In addition, other support devices are moving toward the basic functionality “tracking pre-defined route “to a higher level. They are safety bumpers, optical scanning devices, obstacle sensors, audio and visual warning devices, Wi-Fi connection, GPS modules, remote controls, communication via smartphone, camera systems and others [3]. There are several ways of navigation and guidance technologies of AGVs, which range from physical guide-paths using laser, magnetic
tape, optical sensors, wire, to gyroscope based inertial guidance and wireless. The latter provides the advantage that can be easily modified [4].

AGV generally has PLC or microcontroller as the main control device. Suman Kumar [5], at his paper Increase in Efficiency using PID Control of an Automated Guided Vehicle for Product Ware House, has develop microcontroller based AGV for his thesis. He uses ATmega16 to read line sensors that produce analogue signals and the microcontroller controls the speed of DC motors with L298 motor diver. He also applies PID control to adjust the motor rotation speed [5]. Martin Kajan [6], at his paper “Control of Automated Guided Vehicle with PLC SIMATIC ET200S CPU”, has been researching the use of PLC SIMATIC ET200S as AGV controller. The AGV communicates with the human machine interface (HMI) via industrial wireless LAN with the use of PROFINET and PROFISAFE communication standards. Direct connection of distributed control systems through industrial Ethernet can be easily implemented via PROFINET [6].

This paper focuses on developing a driving module at a low price but can still be used in industrial environments, especially in automotive manufacture industries and has a drive that is strong enough to pull loads up to 500 kg. This AGV uses a mini PC as an alternative to expensive PLC prices. Raspberry Pi is chosen, as it has flexibility in programming languages, sufficient input output, able to be communicated with the touch screen panel and allows further development of wireless communication.

2. Method

2.1 Design

2.1.1 Driving Design. A general AGV system essentially consists of vehicle peripheral on site component as well as stationary control system. The main components of AGV system are [7]: (1) Vehicle; (2) Guidance path system; (3) Controller.

The driving unit module in this paper covers the discussion of vehicle and guidance path systems. In the sub-section vehicle describes the AGV work system as a puller or load carrier. This will have an impact on the discussion of the method of controlling the direction of the AGV movement, AGV’s speed and the motor drive that is suitable for the job. In the sub-section guidance path describes the path sensors system used on the driving module and motor control to be able to follow a predetermined path. The last sub-section discusses the controller and the motor control method on AGV’s driving module.

The load that must be moved by AGV in the automotive manufacturing industry is relatively heavy, with a consumption of up to 500 kg. Thus, it is more likely to design AGV as a drag load rather than as a heap load. Because by giving a heap load it requires more powerful motor and greater power consumption. Figure 1 shows the frame design of AGV.

![Figure 1. Frame Design of AGV](image-url)
When designing the vehicle, there are four basic ways of providing drive and steering [8]. These are shown in Figure 2 below. Some types are easier to build, others have better steering characteristics. Two of these designs are fully symmetrical and may be operated in both directions.

Table 1 gives list the characteristics of each design.

| Design                | Simplicity   | Steering  | Reversible |
|-----------------------|--------------|-----------|------------|
| 4 Wheel Drive         | Simplest     | Coarse    | Yes        |
| Centre Drive & Caster | Simple       | Precise   | Yes        |
| Steer Drive Motor     | Medium       | Very Precise | No        |
| Rear Drive Rack Steer | Complex      | Precise   | Difficult  |

In this project center drive was chosen because of the ease of control and accuracy of movement. The prime mover of AGV can be selected between using a DC motor or brushless DC (BLDC). DC motors can be controlled with PWM drivers and signals [6][10], but DC motors does not have enough torque to load up to 500 kg. BLDC motors are high torque that the most commonly used motors in the industrial world, such as automotive, medical automation and instrumentation equipment. The BLDC motor does not use brush (brush) for magnetic field changes (commutation), but done electronically commutated.

Driving module system of the AGV movement which is designed in the drive unit sub assembly as in Figure 3 and the parts of the sub unit driving module unit and its function explained in Table 2.
Figure 3. Drive Unit AGV (a) Top view, (b) Down view, (c) Isometric view

Straight movements or turn movements is regulated by the rotation speed of each motor. For straight movements, the motor speed of the right and left must be the same, while for the movement turns right and left motor speed is distinguished. For example, if the AGV will turn right then the motor speed is set so that the right motor is slower than the left motor so that the driver’s mechanical movement will turn right, and vice versa.

Table 2. Components of Driving Module

| No | Component     | Function                                      |
|----|---------------|-----------------------------------------------|
| 1  | Brushless Motor | Actuator for wheel (2500 rpm)                 |
| 2  | Gearhead Motor | Speed reducer and torque multiplier           |
| 3  | Shaft wheel    | Motor and wheel connector                     |
| 4  | Wheel          | AGV mover                                     |
| 5  | Base 3         | Driving unit to Chassis connector             |
| 6  | Base 2         | Stabilizer base for right and left wheel      |
| 7  | Shaft 2        | Base 2 and 3 joint                           |
| 8  | Shaft 1        | Joint unit and base 2 connector              |
| 9  | Joint Unit     | Base connector                                |
| 10 | Base gearhead  | Gearhead attachment                           |
| 11 | Spacer         | Base 2 to Sensor’s bracket attachment         |
| 12 | Sensor’s Bracket | Sensor attachment                           |

2.1.2 Path Guidance. Selecting the guidance path system is based on the floor conditions that AGV will go through and how often the route changes will be taken. Fixed path (floor wire, magnetic tape, reflective tape, colour line) guidance is beneficial when the system path is well defined, simple and permanent. Open path (laser, inertia) guidance provides significant benefit with regard to path flexibility. Some AGV uses Radio-frequency identification (RFID) which monitors the AGV systems
report amount, location, and identify of inventory located on the AGV. It can also report the location of products in real time [11].

This driving module uses line lines as paths because it is the cheapest and easiest to determine which sensors are used. Therefore, a sensor is needed that can detect color differences. Color sensor usually contains an infrared transmitter and detector in one integrated components, they are infrared (transmitter) and photodiode (receiver).

2.1.3 Controller. PLC as controller which is commonly used in industry, offers system stability and ease of use [12]. However, to get complete features like analogue input, High Speed Input readings (to read encoder), analogue outputs, PWM or pulse output, the prices will be very high. Moreover, if the AGV system will be developed into wireless communication, the price will be more expensive.

In this project, Raspberry Pi mini PC was chosen as an alternative to PLC because it has a cheaper price and complete features including analogue input and output, PWM output, communication to the touch screen panel to built-in WIFI. Raspberry also offers the ease of selecting programming languages such as C, C++, or Python. The language used in Raspberry Pi is the Python programming language and the help of several studies that have been carried out by several people who have uploaded to social media and Google. Figure 4 shows control block diagram of the system.

**Figure 4. System Block Diagram**

Block Power Supply has battery and power supply module. Block Input has TCRT5000 Sensor as optical sensor to detect color line. Block Control has Bidirectional Voltage, Raspberry Pi and Relay module for switching. Block Output has BLDC motors and the drivers

2.2 Development

2.2.1 Speed of the AGV. The AGV criteria that will be made includes a number of things, such as 70 I/O including backup I/O, can be connected to touch panel and can carry a pull load of 500kg. The allowed speed standard on standard safety in automotive industry shown at Figure 5.
In the picture above, to draw a load of 500kg, the maximum speed is only 1.8 km/h.

\[
1.8 \text{ km/h} = \frac{(1.8 \times 1000) \text{ m}}{(1 \times 60) \text{ min}}
\]
\[
= 1800 \text{ m} / 60 \text{ min}
\]
\[
= 30 \text{ m/min}
\]

Whereas, in the manufacturing industry, the average allowable speed is 30 m/min when on a straight path and only 30% speed at turning path.

With a wheel diameter size of 140 mm, a rotational speed of 2500 Rpm motor and a gear head scale of 30:1 it can be seen that the maximum speed of the AGV will be made, namely: Wheel = 140 mm, Wheel circumference = 440 mm = 0.44 m; \( v \) motor = 2500 Rpm; gear head = 30:1 ratio.

\[
v \text{ motor} / v \text{ wheel} = \text{gear head comparison}
\]
\[
2500 / x = 30/1
\]
\[
x = 2500/30
\]
\[
x = 83 \text{ Rpm}
\]

So, speed the wheel now after going through the gearhead is 83 Rpm. With the wheel speed that has been known, it can be seen that the maximum speed of the AGV that will be made is:

\[
v \text{ AGV} = v \text{ wheel} \times \text{around the wheel}
\]
\[
= 83 \text{ Rpm} \times 0.44 \text{ m}
\]
\[
= 36.52 \text{ m/min}
\]

From the speed 30m/min, it can be seen that the maximum PWM signal that can be given to the driver is as follows:

\[
100 / \text{PWM} = \frac{36.52}{30}
\]
\[
\text{PWM} = \frac{(30 \times 100)}{36.52}
\]
\[
= 3000 / 36.52
\]
\[
= 82.14
\]

The maximum PWM signal that can be given to each driver is 82.14% of the maximum voltage which is 5volt or equal to 4.1volt.

2.2.2 Electrical Control. In designing the input device consists of 16 TCRT5000 sensor modules which are used to read the AGV movement path. The input device will be connected to the IO expander as the IO unit of Raspberry PI 3, which will then be connected to the pin contained in the process device, namely Raspberry PI 3. The input device read line path and controller execute the
program that was created for activate the output device, that is, the brushless motor driver by giving
the rotation direction command and the speed used by each motor in the condition or data from the
read input device, Figure 6.

![Figure 6. Electrical Controller Design](image)

The TCRT5000 sensor is connected to the expander IO which will then be connected to Raspberry
PI 3 through bidirectional voltage first. This is done because the input for Raspberry PI 3 must not
exceed 3.3 volt and with the bidirectional voltage module, this original voltage of 5 volts will be
changed to 3.3 volt so that it can be used by Raspberry PI 3. Then the output of the relay module will
be directly connected to Raspberry PI 3 because the 3.3 volt voltage is enough to activate the relay
module that will be used to control the motor driver.

2.2.3 Programming PID Controller. Programs written in Python are typically much shorter than C or
C++ programs, due to several considerations: (1) High-level data types are used to declare complex
operations in a single statement; (2) Statement grouping has been completed with indentation instead
of starting and ending suffixes; (3) No arguments or variable declarations are needed.

Separate program blocks with an indentation arrangement. So to enter sub-programs in a block, the
sub-programs are placed one or more spaces from the column of a program block. A command
sentence is an instruction that can be executed by a Python interpreter. Previously there were two
types of command sentences, namely print and value declaration. Generally, comments contain
information about the usefulness of a function. The syntax is the "#" fence sign.

Variable is a storage value that can be used in a program. Variables can be changed according to
instructions made: (1) Variable Name, the programmer uses the variable name in accordance with the
description of the contents of the variable and the variable is also a symbol that represents a certain
value. Creating variables in Python is very simple; (2) Keywords define the rules and structure of the
language, and they cannot be used as variable names.

The stability of AGV movement in running following the line is very important because it affects if
the AGV movement is unstable, it will be in terms of security of AGV usage. For this reason, PID
tuning is needed. Tuning this PID control aims to determine the proportional, integrative, and
derivative control action parameters of AGV. PID (Proportional – Integral – Derivative controller)
controller applied to PWM motor signal. It is a feedback mechanism controller that is usually used in
industrial control systems. A PID controller continuously calculates the error value as the difference
between the desired set point and the measured process variable. The controller tries to minimize the
error value at any time by setting the control variable to a new value determined by the PID formula.
The output of the PID control signal is formulated as below [5]:

\[ u(t) = K_p \cdot e(t) + K_i \int_0^t e(t) \, dt + K_d \cdot \frac{de(t)}{dt} \]  \hspace{1cm} (5)

In a discrete time the control signal output is formulated:

\[ PID = K_p \cdot e(k) + K_i \cdot \{ e(k-1) + e(k) \} \cdot Ts + K_d \cdot [e(k) - e(k-1)] / Ts \]  \hspace{1cm} (6)

where: PID is Control output signal; Kp is Proportional gain, tuning parameters; Ki is Integral Gain, tuning parameters; Kd is Gain Derivatives, tuning parameters; Ts is sampling time; e (k) is Error (Sp - PV); e (k-1) is Last Error; Sp is Setpoint; PV is Process variable (reading sensor weight).

At this project, tuning done without identify the AGV mechanism or dynamic analysis. Figure 7 is an illustration of the sensor weighting (sensor reading error value) on the AGV to be created:

![Figure 7. Sensor Weighting Error Value](image)

The ideal condition of AGV occurs when the AGV condition in PV = 0 (e.g., the condition of the sensor value = 111110000111111, the value 0 represents the sensor about the line). Or SP = 0 is the ideal condition of AGV. If PV is not equal to 0, it means that the robot is not in an ideal condition and that means there is a signal error. In this error conditions the PID formula will determine the results of the control signal.

PID drive program controls motor speed with reference to sensor reading data on the lines tracking program. It is divided into several function programs, namely: (1) Motor driver control program: This program is used to provide input to the motor driver in controlling the right and left motor rotation directions rotating in CW, CCW, or stop which will be used by the PID program; (2) Sensor weight mapping program: This program is used to map the weight of error values from sensor readings and determine the ideal conditions of AGV. The weight of the sensor reading must be different between sharp turns and not then this weight will be an error condition of the AGV position. The error condition of this program will be processed by the PID formula (1) Switching speed program: This program is used to adjust the speed used by AGV in running so that the AGV speed in the straight line will be different from the path when turning. By using the sensor reading conditions, a switching speed system on AGV can be designed. The following reference paths are used in switching AGV speed programs.
Figure 8. Switching speed reference path

Condition (a) will be used for minimum speed requirements or speed when turning, condition (b) will be used to change AGV speed to maximum speed or speed when on a straight line; (2) PID program: This program is used AGV to stabilize itself in walking following the line that has been made. The PID formula will process the error weight given by the sensor weight mapping program and will determine the results of its own control signal according to the weight of the sensor that is read.

2.3 Testing

2.3.1 Test Line Tracking Program. This test aims to find out the line tracking program that is functioning properly and can provide data that can be used as parameters to give commands to the driving program. There are 2 decimal numbers where the decimal number above is the result of the left sensor reading (SL) and the bottom is the result of conversion from the right sensor reading (SR). The picture below is a sample image of the sensor reading that has been converted to decimal and displayed with the print command on IDLE Python.

Figure 9. Testing the tracing line program

The picture above is a test image of a tracking line program displayed in Python shell where there is a reading of SL = 240 and SR = 255. These results are decimal values of sensor readings SL =
11110000 (sensors 5, 6, 7 and 8 detect the line) and SR sensor = 11111111 (right sensor does not detect line).

2.3.2 PID Motor Drive Test. The driving program testing aims to find out whether the data from the TCRT5000 sensor that is available can be processed by the driving program and the PID program so that it can run the motor to follow the line as expected. The results of the sensor readings will be processed by the PID program which will determine the results of its own control signal in accordance with the weight of the sensor that is read, then it will control the movement of AGV by giving a PWM signal to each motor driver in running following the available line. In this test, conditions that are read by the TCRT5000 sensor will be displayed in the input column where the red colour is a sensor that is reading the black line and the green colour is a sensor that does not read the black line. Then there is a speed column which is the AGV wheel speed from the control results of each motor driver controlling the speed of the wheel in following the available line. The picture below is the driving test data by showing the actual speed given to each wheel based on the weight of the sensor that is read.

![Figure 10. Testing of Motor Drive programs 1](image)

The picture above is a picture of testing the driving program where there are TCRT5000 sensor readings displayed on the AGV HMI which are sensors 7, 8, 9, and 10 detecting the line (AGV means in ideal conditions or PV = 0) and the speed used in each right and left motor driver is equal to 10.9 m / min so that the AGV will run straight along the line. This speed will still be given and does not change before the error condition changes or AGV is not in the ideal condition. The following are the results of testing on AGV when it is not in its ideal condition. An error condition that will be used as a test sample is PV = -5 so that the PID calculation and the speed should be given to each wheel as in the following figure.
Figure 11. Testing of Motor Drive programs 2

The picture above shows the results of TCRT5000 sensor readings displayed on the AGV HMI namely sensors 4, 5, 6 and 7 detect the line with the sensor weight value used in the condition is -5 and the speed used on the right wheel is 13.1 m / min and the left wheel is 8.7 m / min so the AGV will turn left to get the ideal AGV condition.

3. Result and Discussion

The test results from the previous section explain that the control program of the AGV driving module using Raspberry PI 3 which has been made has indicated the conformity of the actual working principle of AGV with the principles of AGV's work on the design concept that has been made. AGV can walk following the line as expected, using Raspberry PI 3 on a line colour using the TCRT5000 sensor and the selection of replacement components can be used on the next AGV because it has been proven to replace the previous component. Figure 12 shows sensor of the AGV.

Figure 12. Line Sensor

The AGV can run at a speed of 30 m / min on a straight line and 10m / min on a path with a large deflection of 1cm. this is influenced by the distance of each sensor used on AGV, which is 1 cm. the following explanation about AGV deflection can be seen in the picture.
In the picture above it can be seen that the AGV deflection made is affected by the distance between the sensors used. When the AGV runs straight, sensors 7, 8, 9 and 10 will detect the path, but with uneven floor conditions will also affect the movement of AGV in following the line. If the AGV turns left at 0.5 cm it will make the sensor 6 detect the line and the program will correct the AGV condition and will move to the right to get the ideal condition again. Similarly, if the AGV moves to the right and the 11 sensor detects the line, the program will correct the AGV condition and move to the left to get the ideal condition again. So it can be concluded that the AGV movement deflection in following the line is 1 cm seen from the outer distance of the line with the next sensor which is 0.5 cm right deflection and 0.5 left deflection. Figure 14 shows the AGV.

4. Conclusion

Control programs for low cost Automated Guided Vehicle (AGV) driving modules using Raspberry Pi 3 with TCRT5000 sensor input and PWM output for BLDC driver as actuators can run as expected.
The stage of making the program using Python 3.4 software, the program is adjusted to the AGV work function, namely: manual AGV activation, AGV automatically follows the line color, and can be adjusted to AGV speed so as to improve the AGV performance.

The AGV speed that is 30 m/min in the straight and 10 in turns with deflection about 5 mm.

5. References

[1] H. Martinez-Barbera, D. Herrera-Perez, “Development of a Flexible AGV for Flexible Manufacturing Systems”, Industrial Robot: An International Journal 37/5 (2010) 459–468, Emerald Group Publishing Limited. [DOI 10.1108/01439911011063281]

[2] Y. Seo, P. J. Egbelu, ” Integrated Manufacturing Planning for an AGV-Based FMS”, Int. J. Production Economics 60D61 (1999) 473D478

[3] G. Fedorko, S. Honus2, R. Salai,” Comparison of the Traditional and Autonomous AGV Systems”, MATEC Web of Conferences 134, 00013 (2017) DOI: 10.1051/matecconf/201713400013 LOGI 2017

[4] Llopis-Albert et al, “Designing Efficient Material Handling Systems Via Automated Guided Vehicles (AGVs)”, Multidisciplinary Journal for Education, Social and Technological Sciences ISSN: 2341-2593, Vol. 5 Nº 2 (2018): 97-105 | 97, http://polipapers.upv.es/index.php/MUSE/

[5] S.K. Das, “Increase in Efficiency using PID Control of an Automated Guided Vehicle for Product Ware House”, Thesis, Department of Mechanical Engineering, National Institute of Technology, Jamshedpur-831014, June 2016

[6] M. Kajan, L. Mraňko, F. Duchoň, P. Hubinský, J. Šovčík, “Control of Automated Guided Vehicle with PLC SIMATIC ET200S CPU”, American Journal of Mechanical Engineering, 2013, Vol. 1, No. 7, 343-348 Available online at http://pubs.sciepub.com/ajme/1/7/38

[7] S.K. Das, M.K. Pasan, “Design and Methodology of Automated Guided Vehicle-A Review”, IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE, Special Issue - AETM'16, Page 29

[8] A. Kumar, “Development of An Automated Guided Vehicles In Industrial Environment ”, Int. J. Mech. Eng. & Rob. Res. 2014 ISSN 2278 – 0149 www.ijmerr.com, Vol. 3, No. 1, January 2014

[9] H. Dudeja, L. Bagal, N. Junjur, S.S. Jagadale, “Mechanical Design of an Automated Guided Vehicle (AGV)”, International Journal of Research In Aeronautical And Mechanical Engineering, ISSN (Online): 2321-3051 Vol.3 Issue.5, May 2015. Pgs: 32-40

[10] R. Kapolyo, Bing, Z. Gang, “Research and Design of AGV System Application Based on PLC and RFID”, International Journal of Science and Research (IJSR)

[11] S.S. Shingare, O. Swami, K. Mahajan, Y. Junawane, “Modification of Material Handling Process Using Automated Guided Vehicle (AGV)”, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056 Volume: 04 Issue: 02 | Feb -2017