Analysis on Modeling and Control of Two-wheels Drive Mobile Robot in Three Different Surfaces using PID

Su Myat Htay[1], Lu Maw[2], Theint Theint Soe[3]
Faculty of Electronic Engineering [1,2,3],
University of Technology (Yatanarpon Cyber City)[1,2,3]
Pyin Oo Lwin, Myanmar
sumyathtay1983@gmail.com [1], *lumaw29@gmail.com* [2], *theint.soe@gmail.com* [3]

Abstract

This paper proposes a simple linear model structure to generate transfer functions of the DC encoder motor using two wheels mobile robot. In three different surfaces such as smooth floor, rough floor and mat floor, transfer function is estimated by black-box modeling. The mobile robot, known as unmanned ground vehicle (UGV) is built with DC encoder motors using Arduino Uno board and driver. The transfer functions of the two dc motors that are implemented in the left and right sides of the mobile robot are different so two PID controllers are designed. All of the testing and simulation results are verified in Matlab through detail explanations.

Keywords - Arduino Uno, Matlab, DC Motor, L298motor driver, PID controller

1. Introduction

Having a robot assistant surely seems like a dream for most of us. For a robot that performs autonomously, the robot has three wheels, the two wheels are motor driver and one is free wheel. The algorithms for speed control are implemented in Arduino uno played as controller. In there, the model of the dc motor is calculated from the measured input-output data by using the Matlab system identification toolbox. The left and right wheels are controlled by PID, to the desire speed. . In this paper, system identification is used to model the control object. There are three different types of modeling methods [1]. They are white-box modeling, grey-box modeling and black box modeling. A black box is a system which can be viewed in terms of its inputs and outputs without any knowledge of its internal workings. Black-box modeling technique is used because there is no need to measure internal parameters of DC motor that are difficult to know. To use the black-box method (system identification), it is required measured input-output data of the dc motor. The black-box modeling is used to generate transfer function with PID control for three different floors.

2. The System Design

The design of mobile robot in this paper is depicted in Figure 1 and the block diagram of mobile robot is shown in Figure 2. The robot used is three wheeled vehicle. One is free wheel and the other two are dc motor drive. An optical encoder is implemented in each motor (left and right) to measure the speed of the wheel. The dynamic transfer functions of the two motors are estimated through system identification. PID (Proportional- Integral- Differential) based feedback stability control compensates robot's speed to maintain the desired performance.

The mobile robot is built with hardware components such as Two DC motors with Encoders (JGA25-370), DC motor driver (L298N), three wheels UGV (one is free wheel), Arduino Uno board (ATmega328P) and Battery (12V). After generating the transfer function used black-box modeling, Raspberry Pi 3Model B and Pi Camera need for the image processing with camera vision. So, when the mobile robot is built, Pi and Pi-camera are shown in design of mobile robot because there is no use in this work. In this paper, PID control with black-box modeling and analyzing with simulation results for three floors.

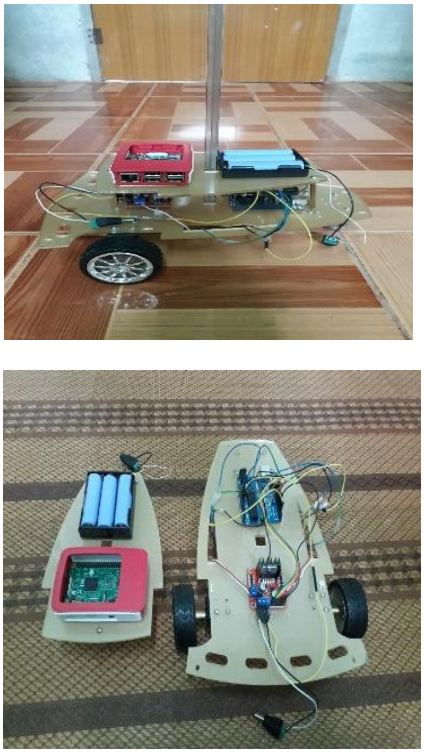


Figure 1. Design of Mobile Robot

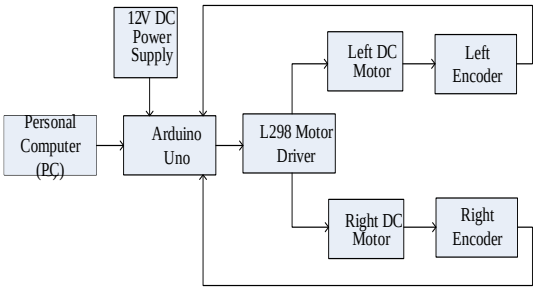


Figure 2. Block Diagram of Mobile Robot

3. Modeling

The mobile robot (UGV) is built with the hardware components as the following DC motor shown in Figure 3. The two 12V DC motors are implemented for left and right sides of the robot and which have the specifications of the gear ratio of 1:34 and maximum output speed of 280 RPM.

Figure 3. JGA25-370 DC Motor

The two motors of the robot are connected with L298 Dual motor driver which allows speed and direction of two DC motors at the same time. The L298N motor driver connecting with the two motors in which 8-bit PWM signal from Arduino must be applied to pins of Enable 1 and Enable 2.Arduino Uno is implemented to drive the motors by applying PWM signal to the motor driver and to read encoders data for the feedback loop. The software used for this controlling is Arduino software (IDE).The PWM signals are applied motor input, then the output signals of encoder are radian per second.

3.1. Data Acquisition

In data acquisition, the input data to the two dc motors is 8-bit PWM values and the output data is the speed measurement from the encoder. Data acquisition is the collecting of input and output data to find the dynamic models for left and right sides of the robot[2].In this paper, sampling time (T_s) of 0.1 second (F_s=10Hz) is used during data acquisition. This sampling frequency is high enough for UGV dynamic.

Firstly, Robot Car is driven with Arduino uno board, driver, motor with encoder for speed (radian per second) using program (PWM input). The connections between robot and laptop give data of encoder speed on the serial monitor shown in Figure 4. On the serial monitor of the laptop, the input PWM and the output rad/s are saved to notepad. This data are imported to the Excel, and divided into Left wheel and Right wheel data.

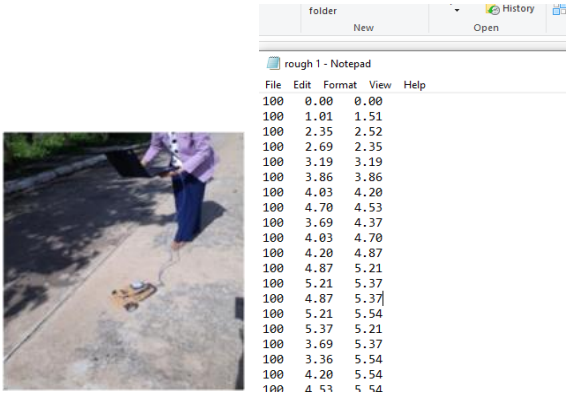


Figure 4. Data Acquisition on Rough Floor

Secondly, the result from Arduino serial monitor are processed in Excel for right and left motor. Again these data from Excel are continued to Matlab import data for only Right or Left motor. When the data from serial monitor are processed in excel, if the data are delimited, choose text to columns from data box. Then, space is defined for data columns to be convenient on import data as shown in Figure 5.

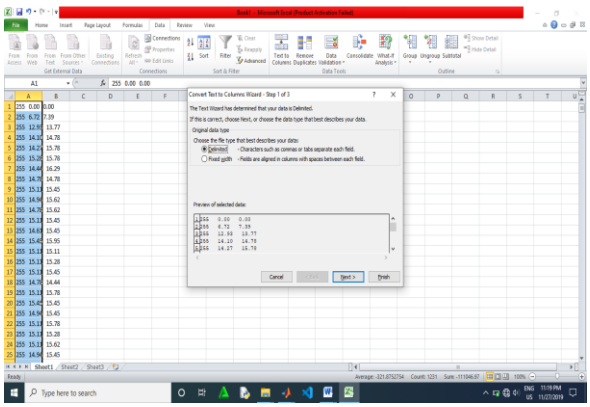


Figure 5. Delimited Data on Excel File

So, the data is saved using notepad from serial monitor on laptop. The data are processed in the Excel. Then Matlab software is used to import data from excel named for each role as shown in Figure 6.

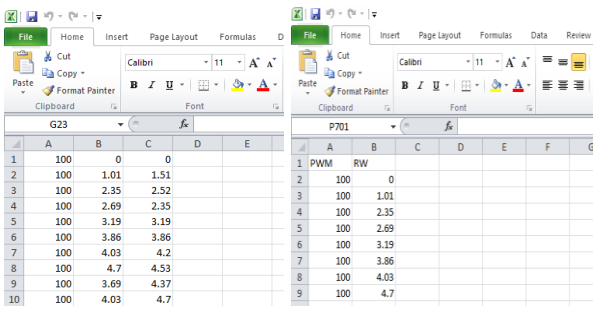


Figure 6. Data Named for Each Role

Figure 7. Data Acquisition on Mat Floor and Smooth Floor

Lastly, each wheel data are imported on Matlab Home at import data. After choosing import data from import selection, system identification on Matlab Application is processed to generate transfer function. Select Time-Domain signals, then the input and output are appeared below workspace variable, data name is written below data information.

3.2. System Identification

The system identification toolbox in Matlab application is used to estimate transfer function model. The starting time is zero and the sample time is 0.1s are imported. The data format is as shown in Figure 8.

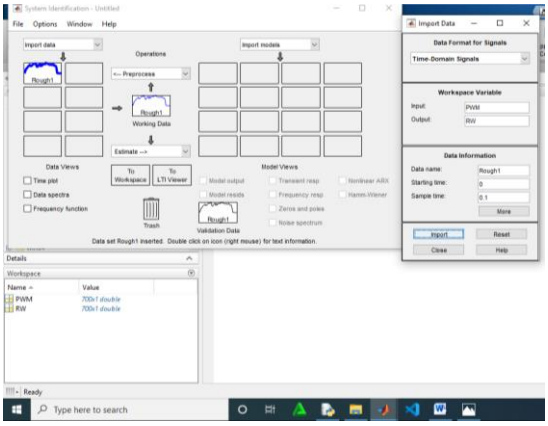


Figure 8. Data Format for Signals

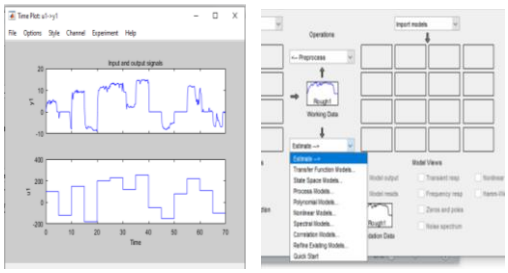


Figure 9. (a) Input and Output Waveform (b) Various Model under Estimates

The input and output signals are shown in Figure 9(a) choosing Time plot. Choose transfer function models in Estimate among state-space model, process model, etc. Define the numbers of poles and zeros and press estimate, wait a few second, when press close, appear transfer function. Repeat various transfer functions are achieved by choosing number of poles and zeros. The various models are described under Estimates as shown in Figure 9(b).

The model output of that TF and best fits are shown in Figure 10.The number of poles is chosen second order and the best fit is chosen highest value. For example,the best fit of tf1 60.27 is bigger than tf2 58.4 shown in Figure 11, so double clip on tf1 and transfer funcion is estimated.Although the transfer function is appeared,the accuracy is measured with other input data. After validating data, TF model is carried out the percentage of accuracy with measured data.

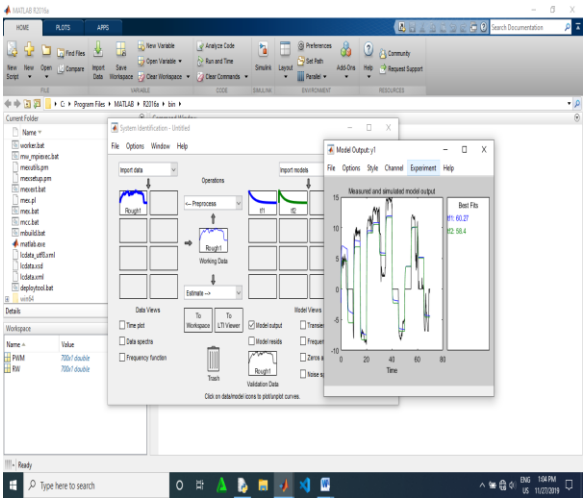


Figure 10. Model Output of Transfer Function and Best Fits

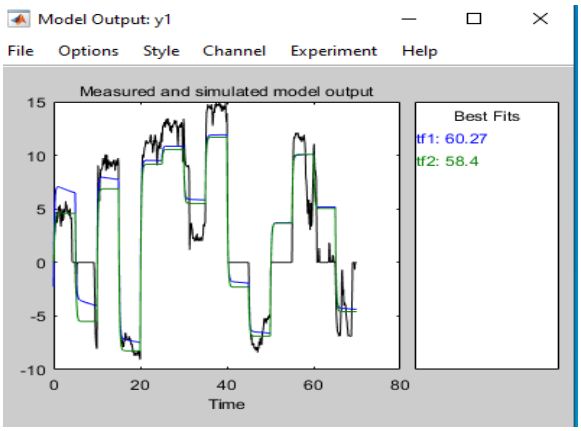


Figure 11. Choosing the Highest Best Fits

The transfer functions are described in transient response, frequency response and the zeros and poles respectively. The phase and amplitude of transfer function are represented in frequency response. These data description below model views can describe if require. Procedure of Modeling in System Identification is shown in Figure 12.

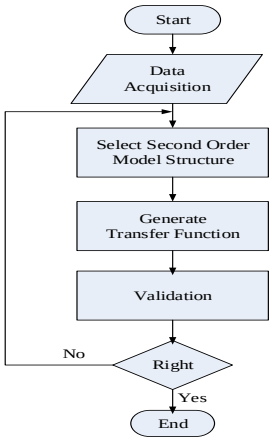


Figure 12. Procedure of Modelling in System Identification

The best fit for system identification of right wheel and left wheel is described according three floors: smooth floor, rough floor, mat floor. The best fit is the percentage of accuracy when comparing different input and output data with various orders. In these motor, the modeling of transfer functions is limited by second order. Without the use of filtering, the default system is used to calculate the best fit. After validation, the best fit is also described for right and left wheels. The comparisons of best fit characteristics are shown in Table1.The transfer function is generated for described best fit under system identification. For examples: for smooth floor; the highest best fit is 67.86 of right wheel and 69.78 of left wheel, for rough floor; the highest best fit is 60.27 of right wheel and 56.71 of left wheel, for Mat floor; the highest best fit is 61.14 of right wheel and 60.86 of left wheel.

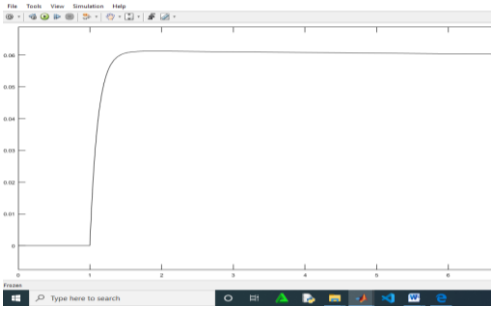
Table 1. Comparisons of Best Fit Characteristics

Types of floor	Best fit for system identification		Best fit for validation	
	Right wheel	Left wheel	Right wheel	Left wheel
Smooth floor	67.86%	69.78%	69.72%	68.36%
Rough floor	60.27%	56.71%	56.7%	52.6%
Mat floor	61.14%	60.86%	59.44%	61.36%

3.3. Simulation

Simulink is a graphical extension to MATLAB for modeling and simulation of systems. When a transfer function is built, the initial conditions are assumed to be zero. In Simulink, systems are drawn on screen as block diagrams. Many elements of block diagrams are available, such as transfer functions, summing junctions, etc., as well as virtual input and output devices such as function generators and oscilloscopes.

(a)



(b)

Figure 13. (a) Simulink block without PID Controller (b) Closed -Loop Step Response

3.3.1. Tuning with PID Controller Proportional-Integral-Derivative Control (PID) obtain a closed-loop system with no overshoot, fast rise time, and no steady-state error. Figure 14 shows PID controller with Simulink block on smooth floor inputs.

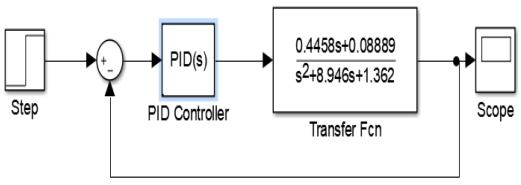
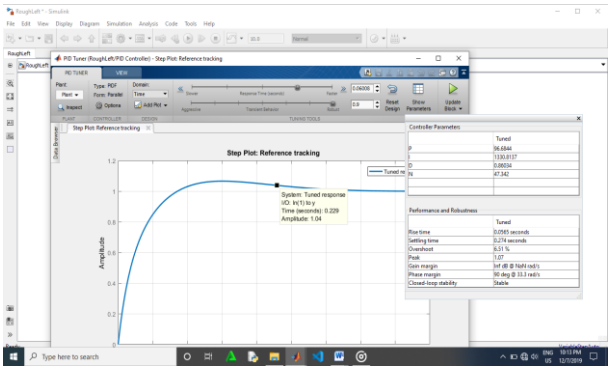
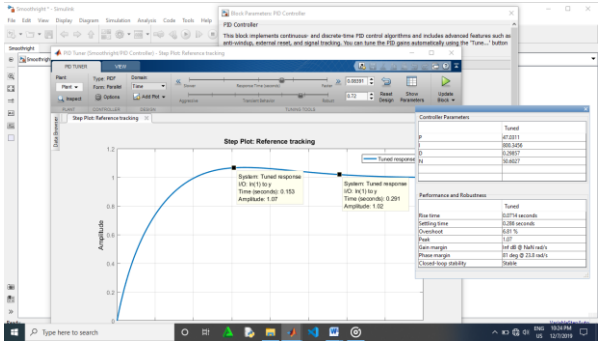


Figure 14. PID Tuning on Smooth Floor

The overshoot of the desire response is at most 10 percent. Before reaching the settling time is limited under 0.5s, reduce the overshoot by increasing the response time. Move the response time slider to the left to increase the closed loop response time. When adjust response time, the response plot, the controller parameters and performance measurements update[3].The PID controller parameters and response curve are shown in Figure 15 for left and right wheel on smooth floor.For rough and mat floors, these parameters are described only in table .



(a) Right Wheel



(b) Left Wheel

Figure 15. PID Controller Parameters on Smooth Floor

4. Analysis the Simulation Result

Figure 16 shows the block diagram of overall system, there are modeling transfer function, PID tuning and analyzing of three floors: smooth floor, rough floor, mat floor.

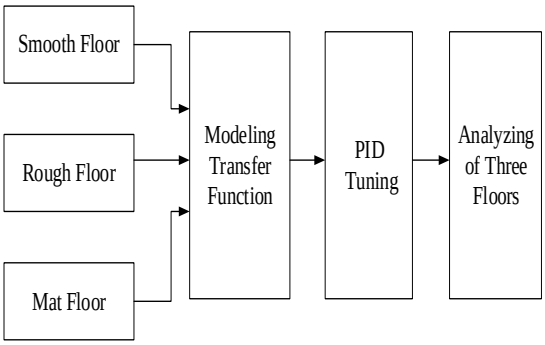


Figure 16. Block Diagram of Overall System

For Smooth floor, the chosen transfer functions of left and right wheels of mobile robot are shown in Eq.(1) and Eq.(2):

$$\frac{\omega}{V} = \frac{0.4458s + 0.08889}{s^2 + 8.946s + 1.362} \quad (1)$$

$$\frac{\omega}{V} = \frac{0.3022s + 0.08739}{s^2 + 6.58s + 1.707} \quad (2)$$

Table 2. Comparisons of Closed Loop Characteristics for Smooth Floor

Wheel	Smooth Floor					
	K_p	K_i	K_d	Settling time	Percent Overshoot	Steady State Error
Left	47	800	0.3	0.286	6.8 %	0
Right	96.68	1331	0.86	0.274	6.51%	0

For Rough floor, the chosen transfer functions of left and right wheels of mobile robot are shown in Eq.(3) and Eq.(4):

$$\frac{\omega}{V} = \frac{03492s + 0.005803}{s^2 + 12.28s + 2.586} \quad (3)$$

$$\frac{\omega}{V} = \frac{0.2594s + 0.01968}{s^2 + 6.521s + 0.1482} \quad (4)$$

Table 3. Comparisons of Closed Loop Characteristics for Rough Floor

Wheel	Rough Floor					
	K_p	K_i	K_d	Settling time	Percent Overshoot	Steady State Error
Left	45.7	533	0.373	0.384	5.59%	0
Right	103.4	1307	1.002	0.297	6.48%	0

For Mat floor, the chosen transfer functions of left and right wheels of mobile robot are shown in Eq.(5) and Eq.(6):

$$\frac{\omega}{V} = \frac{0.5369s + 0.1546}{s^2 + 12.2s + 2.586} \quad (5)$$

$$\frac{\omega}{V} = \frac{0.4115s + 0.1152}{s^2 + 9.558s + 2.152} \quad (6)$$

Table 4. Comparisons of Closed Loop Characteristics for Mat Floor

Wheel	Mat Floor					
	K_p	K_i	K_d	Settling time	Percent Overshoot	Steady State Error
Left	28.91	625.1	-0.023	0.285	6.22%	0
Right	27.05	471.51	-0.137	0.368	7.99%	0

5. Conclusions

In this paper, the transfer functions of three different floors are obtained by black-box modeling using system identification on Matlab software. The best modeling condition is smooth floor analysis among these three different surfaces. In smooth floor analysis, the settling time to reach the desire value and overshoot are

convenient for both left and right wheels. Measure and test the transfer function of the DC motor and calculate the transfer function of the motor. This system is a technique of controlling the speed of DC motor using PID controller on three different floors. PID control is a very practical and effective control method for the control system. It reduces the error of the system to get the desired speed of the motor. This system helps in maintaining the stability of the system. By analyzing the transfer function of the DC motor on three different floors, the desire outputs are compared to propose future plan.

6. Acknowledgment

First of all, the author would like to express her special thanks to Dr. Aung Win, Rector, University of Technology (Yadanarpon Cyber City), for initiating the Master programme at the University of Technology (Yadanarpon Cyber City). I would like to express his deep gratitude to Dr. Htin Kyaw Oo, Professor, Head of Faculty of Electronic Engineering, for his effective guidance, constructive comments and criticism. I would like to express my special thanks to my supervisor Dr. Lu Maw, Professor, Faculty of Electronic engineering, for her helpful suggestions, constructive advice and editing this research paper. Then, I am also greatly indebted to my co-supervisor, Dr. Theint Theint Soe, for her enthusiasm, guidance and encouragement. Finally, a special note of thank is also intended to all who supported her with necessary help for this paper.

7. References

[1] Htet Htet Shin, Nay Min Tun, 'Modeling and Performance Comparison of Different PID Configurations for DC Motor Speed Control with Unknown Parameters, Department of Electronic Engineering Mandalay Technological University.

[2] Hsu Myat Thein Lwin, Cho Thet Paing, " Modeling and Speed Control of Ground Mobile-Robot" Faculty of Electronic Engineering, University of Technology (Yatanarpon Cyber City), Pyin Oo Lwin, Myanmar .

[3] 'MATLAB & Simulink Tutorial' *Principles of Automatic Control & Dynamics*, Violeta Ivanova, Ph.D. Educational Technology Consultant MIT Academic Computing, violeta@mit.edu

[4] Bharat Joshi, Rakesh Shrestha 'Modeling, Simulation and Implementation of Brushed DC Motor Speed Control using optical incremental encoder feedback', University of South Carolina, Conference Paper, October 2014

[5] http://www.mathworks.com/help/index.html 2

[6] Fatiha Loucif "DC motor speed control using PID controller" , Department of Electrical Engineering and information, Hunan University, ChangSha, Hunan, China (E mail:fatiha2002@msn.com) [J. Breckling, Ed., *The Analysis of Directional Time Series: Applications to Wind Speed and Direction*, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.

[7] Johan Schoukens, "System Identification: what does it offer to electrical engineers", Vrije Universiteit Brussel, dep. ELEC, Pleinlaan 2, B1050 Brussels, Belgium

[8] Khaled Sailan and Klaus-Dieter Kuhnert, "Speed Control of Unmanned Ground Vehicle for Non Autonomous Operation", Real time System Institute, Siegen University

[9] Richard C. Dorf, Robert H. Bishop, "Modern Control System", twelfth Edition

[10] Stephen A. Jacklin*, Johann M. Schumann, and Pramod P. Gupta NASA Ames Research Center, Moffett Field, CA, 94035 ' Development of Advanced Verification and Validation Procedures and Tools for the Certification of Learning Systems in Aerospace Applications '.

[11] Than Myint Kyi, Kyaw Zin Latt, ''The Speed Control for Camera Movement Using PID Algorithm in Lab VIEW'', Department of Electronic Engineering Mandalay Technological University.

[12] Yamin Nyein , Than Htike Aung, 'Color Sensor Based Line Follower Robot Using PID Control', Department of Electronic Engineering Mandalay Technological University ,The Twelfth National Conference on Science and Engineering, NCSE-2019

Kinematics Modeling and Simulation of the 4-DOF Robotic Manipulator

Myat Noe Wai[1], Lu Maw[2], Thwe Thwe Htoo[3]
Faculty of Electronic Engineering[1,2,3]
University of Technology (Yatanarpon Cyber City) [1,2,3]
Pyin Oo Lwin, Myanmar
myatnowai617@gmail.com[1], lumaw29@gmail.com, thwethwehtoo.ec@gmail.com[3]

Abstract

This paper proposes a kinematic modeling of a 4-DOF articulated robotic arm design that would become a solution for picking and placing objects from one place to another. Standard DH parameter convention is used to calculate forward kinematics. An analytical method for a closed-formed solution of inverse kinematics is applied to solve the unknown joint angles required for the autonomous position of a robot arm. This paper focuses on the design and simulation of the four degrees of freedom manipulator robot (4-DOF) along with its geometric Jacobian and reachable workspace under a GUI interface designed in MATLAB. The angular rotation of each joint is powered by DC-geared motors with hall-effect encoders for getting maximum joint limits. Finally, simulation results for the kinematic model using MATLAB are presented validating the inverse kinematics algorithm.

Keywords - 4-DOF robot, DH convention, kinematics, workspace, Jacobian.

1. Introduction

Nowadays, robots are very powerful elements of today's industry. They are capable of performing many tasks and operations precisely and do not require common safety and comfort elements humans need. Robotics is a rich area of research, according to their kinematics, dynamics, and control. Kinematics plays a significant role in robotics and especially for the study of industrial manipulators' behavior [1]. It can be generally divided in forward and inverse kinematics. The motion takes place in the Cartesian space; but most of the industrial robots, especially the articulated robotic arm, are controlled in rotary joint spaces. Therefore, a kinematic transformation between the Cartesian space and joint space is needed. Jacobian is the most widely proposed method for solving the inverse kinematic problem for redundant manipulators. In other words, it is the process of obtaining a configuration space that corresponds to a given workspace. This paper focuses on modeling and simulation of the 4-DOF robotic arm, a small special robot with four revolute joints.

2. Kinematic Analysis

Kinematics in robotics is a statement form about geometrical description of a robot structure. From the geometrical equation, relationship among joints spatial geometry concept on a robot can be got with ordinary co-ordinate concept which is used to determine the position of an object. In other word, kinematics is the relationships between the positions, velocities, and accelerations of the links of a robot arm. The aim of kinematics is to define position relative of a frame to its original co-ordinates. Kinematics problem consists of forward and inverse kinematics and each type of the kinematics has its own function. Forward kinematics is used for transferring joint variable to get end-effector position. On the other hand, inverse kinematics will be applied to find joint variable from end-effector position.

2.1. DH Convention

Denavit-Hartenberg (DH) convention is commonly used in the kinematics analysis of the robotic manipulator.

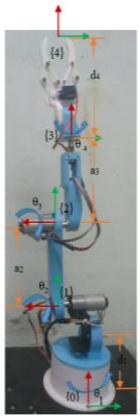


Figure 1. Mechanical Design and Assignment of 4-DOF Robotic Arm

It is based on attaching a coordinate frame at each joint and specifying four parameters known as DH parameters for each link, and DH table is constructed utilizing this parameter. These parameters are the twist angle α_i, link length a_i, link offset d_i, and joint angle θ_i. In this 4-DOF robotic manipulator, all the joints are revolute type and corresponding ranges of each joint are

specified as from -120˚ to 120˚. The relationship between the joint variables and the position, and also orientation of the gripper, is driven using following DH parameters as illustrated in Table 1. This table involves determining four parameters to build a complete homogeneous transformation matrix.

DH parameter is computed manually or using computer programs such as MATLAB programs [2]. As shown in Figure 1, its corresponding DH parameters are determined according to the physical robot configuration. At first, joint coordinate frames are assigned at each joint and then, moving from frame to frame, the specific link parameters are determined as the standard DH notation. Since all joints are revolute in this robot system, joint variables are revolute joints' angles (θ_i).

Table 1. DH Parameter

Links	angles	Link length a(cm)	Link offset d(cm)	Twist angle (α)	Range of Joint (θ)
1	θ_1	0	13.6	-90	-120,120
2	θ_2	12	0	0	-120,120
3	θ_3	11.5	0	90	-120,120
4	θ_4	0	16.5	0	-120,120

2.2. Forward Kinematic Model

The forward kinematic model represents the relations calculating the operational coordinates, giving the location of the end-effector, in terms of the joint coordinates. After establishing D-H coordinate system for each link as shown in Table 1, a homogeneous transformation matrix can easily be developed considering frame {*i*-1} and frame {*i*} transformation. So, the link transformation matrix between coordinate frames {*i*-1} and {*i*} has the following form [3]:

$$^{i-1}T_i = Rot(z,\theta_i).Trans(z,d_i).Trans(x,a_i).Rot(x,\alpha_i) \quad (1)$$

$$T_i = \begin{bmatrix} \cos\theta_i & -\cos\alpha_i\sin\theta_i & \sin\alpha_i\sin\theta_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\alpha_i\cos\theta_i & -\sin\alpha_i\cos\theta_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The basic transformations are calculated using Equation 1 as follows:

$$^0T_1 = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad ^1T_2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2c_2 \\ s_2 & c_2 & 0 & a_2s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$^2T_3 = \begin{bmatrix} c_3 & -s_3 & 0 & a_3c_3 \\ s_3 & c_3 & 0 & a_3s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad ^3T_4 = \begin{bmatrix} c_4 & 0 & s_4 & 0 \\ s_4 & 0 & -c_4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Therefore, the final homogeneous transformation is got via the product of the four basic transformations.

$$^0T_4 = {}^0T_1.{}^1T_2.{}^2T_3.{}^3T_4 \quad (2)$$

$$= \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} Q^{3x3} & p^{3x1} \\ 0\ 0\ 0 & 1 \end{bmatrix}$$

where, Q and p are the orientation matrix and position vector, respectively; their parameters are as follows:

$$n_x = c_1c_{23}c_4 - s_1s_4$$
$$n_y = s_1c_{23}c_4 + c_1s_4$$
$$n_z = -s_{23}c_4$$
$$o_x = -s_1c_4 - c_1c_{23}s_4$$
$$o_y = c_1c_4 - s_1c_{23}s_4$$
$$o_z = s_{23}s_4$$
$$a_x = c_1s_{23}$$
$$a_y = s_1s_{23}$$
$$a_z = c_{23}$$
$$p_x = c_1(a_3c_{23} + a_2c_2 + d_4\sin_{23})$$
$$p_y = s_1(a_3c_{23} + a_2c_2 + d_4\sin_{23})$$
$$p_z = d_1 - a_3s_{23} - a_2s_2 + d_4c_{23}$$

$s_1 = \sin\theta_1$, $c_1 = \cos\theta_1$, $s_{23} = \sin(\theta_2 + \theta_3)$ $c_{23} = \cos(\theta_2 + \theta_3)$.

3. Analytical Inverse Kinematics

The conversion of the position and orientation of the manipulator's end-effector from Cartesian space to joint space is known as inverse kinematics. In general, inverse kinematics solutions are nonlinear. To find those equations can be complicated and sometimes there is no solution for the problem. Analytical approach is provided in this section for solving inverse kinematics of a robot arm.

Given that the position of the robot arm is

$$P = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix}.$$

From the orientation matrix, the RPY (roll, pitch,

yaw) matrix is

$$RPY=\begin{bmatrix} r_{11} & r_{21} & r_{31} \\ r_{12} & r_{22} & r_{32} \\ r_{13} & r_{23} & r_{33} \end{bmatrix} \quad (3)$$

where, $r_{11}= \cos\beta\cos\gamma\cos\alpha - \sin\gamma\sin\alpha$

$r_{12}= \cos\beta\cos\gamma\sin\alpha + \sin\gamma\cos\alpha$

$r_{13}= - \sin\beta\cos\gamma$

$r_{21}= - \cos\beta\sin\gamma\cos\alpha - \cos\gamma\sin\alpha$

$r_{22}= - \cos\beta\cos\gamma\sin\alpha + \cos\gamma\cos\alpha$

$r_{23}= \sin\beta\sin\gamma$

$r_{31}= \sin\beta\cos\gamma$

$r_{32}= \sin\beta\sin\gamma$

$r_{33}= \cos\beta$

. The roll-pitch-yaw angle of the end-effector is defined as follows:

- Pitch: Pitch is the angle of rotation about y_5 axis of end-effector as follows:

$$\text{pitch, } \beta = \theta_{23} = \text{atan2}\left(\pm\sqrt{r_{13}^2 + r_{23}^2}, r_{33}\right)$$

atan2 is used because its range is $[-\pi, \pi]$, whereas the range of atan is $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$

- Roll: The roll $\gamma = \theta_4$ is got via the orientation matrix.
- Yaw: yaw is not free and bounded by θ_1 as follows:

$$\text{Yaw, } \alpha = \theta_1 = \text{atan2}(Y,X)$$

Table 2. Inverse Kinematics Solution

1.	From the position matrix, get X, Y and Z
2.	From the orientation matrix, get the roll, pitch and yaw angles $\text{Roll} = \beta = \theta_{23} = \theta_2 + \theta_3$ $\text{Pitch} = \gamma = \theta_4$ $\text{Yaw} = \alpha = \theta_1$
3.	Calculate the second joint angle: $\theta_2 = \text{atan2}(s_2, c_2)$ $\text{where}, c_2 = \left(\pm\sqrt{P_x^2 + P_y^2} - a_3c_{23} - d_4s_{23}\right)/a_2$ $s_2 = \left(P_z - d_1 + a_3s_{23} - d_4c_{23}\right)/\left(-a_2\right)$
4.	Calculate the third joint angle: $\theta_3 = \theta_{23} - \theta_2$

To find the second and third joint angles, the position vector is used to calculate analytically. After performing all the calculations, the inverse closed-form kinematic solutions are obtained as shown in Table 2. Using the inverse kinematic equations, Cartesian space is converted into joint space for driving the specific position and orientation of the robotic pose.

4. Velocity Analysis: The Jacobian Matrix

The Jacobean defines the transformation between the robot hand velocity and the joint velocity. The Jacobian matrix is the mapping of the relationships between the joint velocities and the corresponding end-effector linear and angular velocities. It is useful for finding singular configurations, analyzing redundancy, determining the inverse kinematics algorithms for velocity analysis, describing the relationship between the forces applied to the end-effector and the resulting torques at the joints, and deriving the dynamics algorithms.

Knowing the joint velocity, the joint angles and the parameters of the arm, the Jacobean can be computed and the hand velocity calculated in terms of the hand Cartesian coordinates. The kinematic Jacobian matrix is obtained through a direct calculation using the relation between the linear and angular velocity vectors, V_n and ω_n, of the frame R_n and the joint velocity $\dot{q}$ as follows:

$$v = J(q)q^{\square} \quad (4)$$

The matrix $J(q)\hat{I}R^6$ is the manipulator Jacobian or the geometric Jacobian which depends on the robot configuration and robot degrees of freedom (DOF).

Describing the Jacobian components, the Jacobian matrix can be written as:

$$J = \begin{bmatrix} J_\omega \\ J_v \end{bmatrix} \quad (5)$$

Since there are four revolute joints in the robot manipulation configuration, the 6 × 4 Jacobian matrix of the robot is defined as in Equation 6:

$$J=\begin{bmatrix} e_1 & e_2 & e_3 & e_4 \\ e_1\times a_{1e} & e_2\times a_{2e} & e_1\times a_{3e} & e_1\times a_{4e} \end{bmatrix} \quad (6)$$

$$=\begin{bmatrix} 0 & s_1 & -s_1 & c_1s_{23} \\ 0 & -c_1 & c_1 & s_1s_{23} \\ 1 & 0 & 0 & -c_{23} \\ r_{41} & r_{42} & r_{43} & 0 \\ r_{51} & r_{52} & r_{53} & 0 \\ r_{61} & r_{62} & r_{63} & 0 \end{bmatrix}$$

where, the components of J_v useful for analysing singular configurations are:

$$r_{41} = c_1(d_2 - d_3) - a_1 s_1 - s_1(a_2 c_2 + c_2(a_3 c_3 - d_4 s_3) + s_2(a_3 s_3 + d_4 c_3))$$
$$r_{51} = s_1(d_2 - d_3) + a_1 c_1 + c_1(a_2 c_2 + c_2(a_3 c_3 - d_4 s_3) + s_2(a_3 s_3 + d_4 c_3))$$
$$r_{61} = 0$$
$$r_{42} = -c_1(a_2 s_2 - d_4 c_{23} + a_3 s_{23})$$
$$r_{52} = -s_1(a_2 s_2 - d_4 c_{23} + a_3 s_{23})$$
$$r_{62} = a_2 c_2 + d_4 s_{23} + a_3 c_{23}$$
$$r_{43} = c_1(s_2(a_3 c_3 - d_4 s_3) - c_2(a_3 s_3 + d_4 c_3))$$
$$r_{53} = s_1(s_2(a_3 c_3 - d_4 s_3) - c_2(a_3 s_3 + d_4 c_3))$$
$$r_{63} = -a_3 c_{23} + d_4 s_{23}$$

5. Workspace Analysis

Generally, the workspace of a robotic manipulator is the total volume swept out by the end-effector as the manipulator executes all possible motion. The workspace is determined by the geometry of the manipulator and the limits of the joint motions.

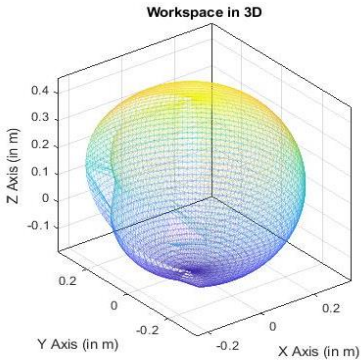


Figure 2. Workspace of the Manipulator

The robotic manipulator provides a wide region of operation of the end effector. Its volume of work is limited by the length of its links and the articular limitations of the manipulator. Since the limits of all joint angles are specified as from -120 to 120, the corresponding workspace are discovered driving each joint between these joint limits. For the calculation of workspace, programming code is implemented using MATLAB for the 3D representation of the arm's workspace. In Figure 2, the workspace of the manipulator is presented in 3D view.

6. Simulation Results

In this section, the simulation results are illustrated as the corresponding simulation cases; driving the robot arm in joint and Cartesian trajectories, workspace discovering and validating the forward and inverse kinematics in MATLAB GUI.

Firstly, the robot is driven using forward kinematics as shown in Figure 3. In this test, the joints coordinated are used as inputs; starting from [0, pi, 0, 0] to [pi/4, pi/2, pi/4,0]. The "jtraj" command in Robotic Toolbox is used for the joint trajectory. Next, the robot is driven using inverse kinematics as illustrated in Figure 4. Therefore, the inverse kinematic algorithm of the robot plays an important role in driving the robot in Cartesian space.

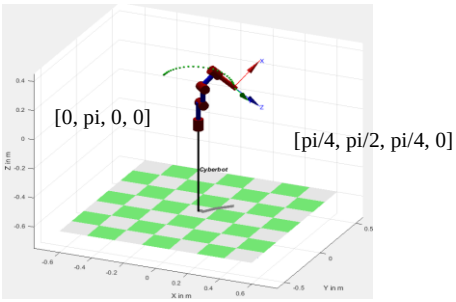


Figure 3. Joint Trajectory

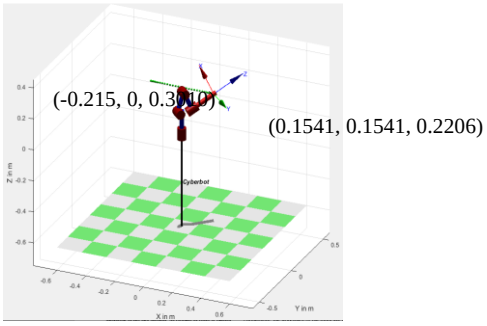


Figure 4. Cartesian Trajectory

In this test, the Cartesian coordinates used as inputs; starting from (-0.215, 0, 0.3010) to (0.1541, 0.1541, 0.2206), and the required joint angles are calculated with the inverse kinematic algorithm in section 3. The "ctraj" command in Robotic Toolbox is used for the Cartesian trajectory.

Finally, it is done to validate the accuracy of the kinematic equations in MATLAB GUI as shown in Figure 6. In this GUI, the forward and inverse kinematics can be easily checked whether they contribute an exact solution or not. In this system, the forward kinematics will provide the position and orientation of the robot according to the DH configuration. For the inverse kinematics, the required inputs are the XYZ position and RPY orientation to easily compute the exact joint angles. After doing the various joint-angle data, it provides an exact solution as the joint angles of the forward kinematics.

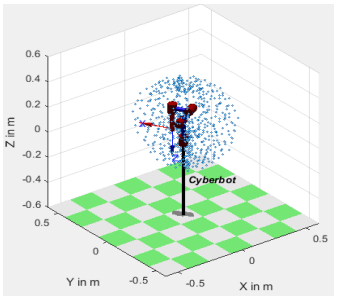


Figure 5. Workspace Simulation

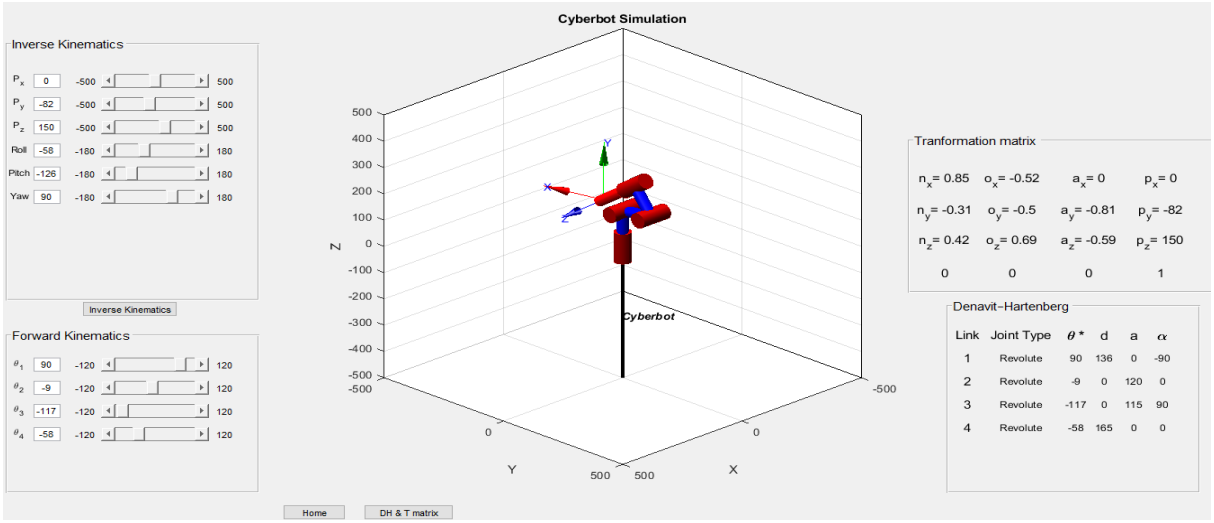


Figure 6. Simulation for Validating the Forward and Inverse Kinematics in MATLAB GUI

7. Conclusion

In this paper, the forward and inverse position analyses and the Jacobian matrix that appears in velocity analysis are presented for the 4-DOF robotic arm. Furthermore, the spherical workspace of the robot is analyzed. In addition, the simulation of the robot manipulator is carried out using MATLAB Robotics toolbox, and the exact solution of the inverse position is validated in MATLAB GUI for user-friendly views. In the future, the robotic dynamic modeling cooperated with motor specifications will be conducted. Furthermore, position and speed controllers will be implemented through the dynamic model using the state-space control method.

8. References

[1] Amin A. Mohammed, M. Sunar, "Kinematics Modeling of a 4-DOF Robotic Arm", IEEE International Conference Paper, King Fahd University in Saudi Arabia, 19 May 2017.

[2] Adelhard Beni Rehiara, "*Kinematics of Adept Three Robot Arm"*, June 9th 2011, University of Papua Indonesia,2011.

[3] Aung Myat San, Wut Yi win, "Modelling and simulation of a 5-DOF Stationary Articulated Robot using Matlab and Root Analyzer", 7th ICSE, Mandalay Technological University, 2016.

[4] Gómez, G. Sánchez, J. Zarama, M. Castañeda Ramos, J. Escoto Alcántar, J. Torres, A. Núñez, S. Santana, F. Nájera, J. A. Lopez, "Design of a 4-DOF Robot Manipulator with Optimized Algorithm for Inverse Kinematics'', Vol:9, No:6, International Journal of Mechanical and Mechatronics Engineering ,2015.

[5] Jorge Luis Aroca Trujillo, Ruthber Rodriguez Serrezuela, Vadim Azhmyakov, Roberto Sagaro Zamora, "Kinematic Model of the Scorbot 4PC Manipulator Implemented in Matlab's Guide'', Vol. 11, Contemporary Engineering Sciences, 28 February 2018.

[6] J. Kaur and V. K. Banga, "Simulation of Robotic Arm having three link Manipulator," International Journal of Research in Engineering and Technology (IJRET), vol. 1, ISSN: 22774378, March, 2012.

[7] Ali ROSHANIANFARD, Noboru NOGUCHI, "Kinematics Analysis and Simulation of A 5DOF Articulated Robotic Arm Applied to Heavy Products Harvesting", JOURNAL OF AGRICULTURAL SCIENCES 24 (2018) 91-104 Hokkaido University, 17 September 2017.

[8] Atique, M. M. U., Ahad, M. A. R., "Inverse Kinematics solution for a 3DOF robotic structure using Denavit-Hartenbe Convention", IEEE International Conference on Informatics, 2014.

Modeling and Simulation of Two Wheeled Self-Balancing Robot using LQR Control

Pann Ei Hlaing[1], Kyu Kyu Mar[2], Pa Pa Win San[3]
Faculty of Electronic Engineering[1,2,3]
University of Technology (Yatanarpon Cyber City)[1,2,3]
Pyin Oo Lwin, Myanmar
panneipeh@gmail.com[1], thankyumar@gmail.com[2], papawinsan.1@gmail.com[3]

Abstract

This paper intends to balance the two wheel self-balancing robot when the wheel attached to body is applied by a force. This is based on the inverted pendulum model. Modeling of the system is used with the mathematical modelling. After modeling, the system parameters are exported to MATLAB. In this paper, Linear-Quadratic-Regulator (LQR) is used for controlling the linearized system of the two wheeled self-balancing robot. Finally, results are shown angle body and wheel position of the two wheel self-balancing robot.

Keywords – Linear-Quadratic-Regulator (LQR), Two Wheel Self-Balancing Robot, MATLAB

1. Introduction

Two wheel self-balancing robot based on the inverted pendulum is an unstable mechanical system with nonlinear dynamics [1]. These robots are used in narrow or dangerous working spaces, transportation field, medicine field, agriculture field and so on.

Most of the vehicles are affected by external disturbances, errors in the estimated parameters, unmodeled and noise associated with the measurements. There are many different techniques for control method of this system. Among these techniques, we found the Newton-Euler equations, Lagrange equations and State Space equation that can obtain a good understanding of the mathematical Model [1] [2].

Moreover, CAD software is used to modeling and exported to Matlab/Simmechanics that is multi-body modeling toolbox of the Matlab. So, the moment of inertia is not considered in their system [2]. This paper presents the performance of modern control method (LQR) with respect to body's angle and wheel's position of an inverted pendulum. It is assumed that the system has an optimal full-state feedback. So, the system can measure all of the states. In this system, mathematical modeling is used for modeling and Matlab software is used for both linearization and controller design.

Generally, linear control and non-linear control strategies have been applied to achieve the control objective of the two wheeled self-balancing robot. Linear controllers depend on the linearization process are presented a limited operation range that they can limit the behaviour of the robot. Therefore, it has been applied nonlinear control techniques that allow achieving desired outcomes and overcoming these restrictions [3]. In this paper, linear control process has been applied to the system.

2. Modelling and Control

In this section, three sections are divided. Firstly, modelling and control of the two wheels self-balancing robot is described by using mathematical modelling based on the parameters of the system. Secondly, theses parameters and result equations from mathematical modelling are exported to the MATLAB. Finally, Linear Quadratic Regulator (LQR) controller designed that is approached the state space equations to get the desired angle and position of the system. Figure 1 is shown the modelling and control strategy of this paper.

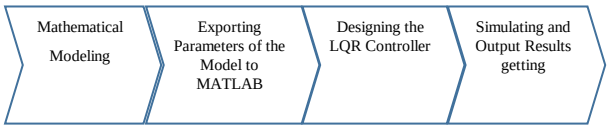


Figure 1: Modeling and Control Strategy of the Paper

A. Modeling

These sections are divided into three, a short description on the modeling of the two wheeled self-balancing robot system is provided. Figure 2 is shown the free body diagram of the Two Wheel Self-Balancing Pendulum system. In order to obtain the dynamic model of the system, the following conditions have been made:

Step-I: The system will start in a state of equilibrium so that the initial condition is assumed to be zero.

Step-II: Desired body angle and wheel position are assigned so that the body and wheel of the robot do not move more than a few degrees away from the vertical to satisfy a linear model.

Step-III: A step input is applied.

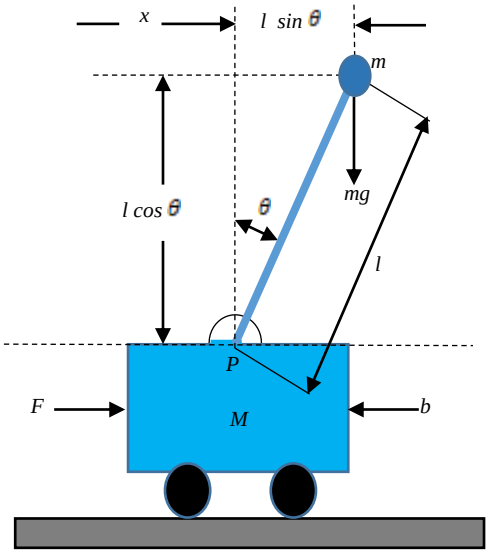


Figure 2: Free Body Diagram of the Two Wheel Self-Balancing Pendulum System

Parameters	Values	Description
M	0.35	Mass of the wheel (Kg)
m	0.128	Mass of the body (Kg)
I	0.0019	Moment of inertia of body (Kgm^2)
l	0.2	Distance of wheel to center of mass (m)
g	9.81	Gravitational constant (m/s^2)
b	0.1	Friction constant (N/m/s)

Table 1: The Specifications of the Propose System

The specifications of the propose system are shown in Table 1. In this system, Force F is the input of the system that is applied the wheel for moving horizontally to the x-axis direction on the ground to balance body angle. Angle of the body and position of the wheel are outputs in this system. The following equation is the moment of inertia of body.

$$I = \frac{1}{3}ml^2$$

From the free body diagram of the system, the following dynamic equations are determined. Summing the forces of the cart along horizontal direction, following equation of motion was obtained [7, 8]:

$$M\ddot{x} + b\dot{x} + N = F \quad (1)$$

Summing the forces along the horizontal direction as shown in the free body diagram, following equation for N was obtained:

$$N = m\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta \quad (2)$$

After substituting eqn. 2 into eqn. 1, the first equation of motion for the system was found as follows:

$$(M+m)\ddot{x} + bx + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F \quad (3)$$

To acquire the second equation of motion, the forces along the perpendicular direction of the pendulum were summed up and found the following equation:

$$P\sin\theta + N\cos\theta - mg\sin\theta = ml\ddot{\theta} + m\theta\ddot{x}\cos\theta \quad (4)$$

To get rid of P and N terms from the eqn. 4, the moments around the centroid of the pendulum was taken which resulted following equation:

$$-Pl\sin\theta - Nl\cos\theta = l\ddot{\theta} \quad (5)$$

Combining eqn. 4 and 5, the second dynamic equation was obtained as follows:

$$(I + ml^2)\ddot{\theta} + mgl\sin\theta = -ml\ddot{x}\cos\theta \quad (6)$$

From eqn. 3 and 6, two linear equations of the transfer function were found, where $\theta = \pi$. Assume that, $\theta = \pi + \emptyset$ ($\emptyset$ represents a small angle from the vertical upward direction).

Therefore, $\cos\theta = -1, \sin\theta = -\emptyset$ and $\frac{d^2\theta}{dt^2} = 0$.

Thus, after linearization the following 2 equations of motion were appeared (where *u* has been substituted for the input F). [5][6]

$$(I + ml^2)\ddot{\emptyset} - mgl\emptyset = ml\ddot{x} \quad (7)$$

$$(M+m)\ddot{x} + b\ddot{x} - ml\ddot{\emptyset} = u \quad (8)$$

To obtaining the transfer functions of the linearized system equations, first assume the Laplace transforms from the system equations assuming zero initial conditions. The following equations (3) and (4) are the resulting Laplace transforms.

$$(I + ml^2)\emptyset(s).s^2 - mgl\emptyset(s) = mlX(s).s^2 \quad (9)$$

$$(M+m)X(s).s^2 + bX(s).s - ml\emptyset(s).s^2 = U(s) \quad (10)$$

To find the transfer function, X(s) from equations (3) and (4) are eliminate. After eliminating, rearrange the transfer function of the system which is described in equation (5) and (6).

$$\frac{\emptyset(s)}{U(s)} = \frac{\frac{ml}{q}s^2}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mgl}{q}s^2 - \frac{bmgl}{q}s} \quad (11)$$

$$\frac{X(s)}{U(s)} = \frac{\frac{(I+ml^2)s^2 - gml}{q}}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mgl}{q}s^2 - \frac{bmgl}{q}s} \quad (12)$$

In above equations (11) and (12) are the transfer function of the body angle and the transfer function of the wheels position.

The following equations (13) and (14) are representing state space form and output equation of the system transfer functions.

$$\begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \\ \dot{\emptyset}(t) \\ \ddot{\emptyset}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-(I+ml^2)b}{I(M+m)+Mml^2} & \frac{m^2gl^2}{I(M+m)+Mml^2} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{-mlb}{I(M+m)+Mml^2} & \frac{mgl(M+m)}{I(M+m)+Mml^2} & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \emptyset(t) \\ \dot{\emptyset}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{(I+ml^2)}{I(M+m)+Mml^2} \\ 0 \\ \frac{ml}{I(M+m)+Mml^2} \end{bmatrix} U(t) \quad (13)$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \emptyset(t) \\ \dot{\emptyset}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} U(t) \quad (14)$$

B. Exporting System Parameters to the Matlab

By substituting the system parameters to the input equation (7) and output equation (8), the following results are obtained.

$$\begin{bmatrix} \dot{x}(t) \\ \ddot{x}(t) \\ \dot{\emptyset}(t) \\ \ddot{\emptyset}(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -0.26 & 2.379 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -0.9481 & 44.41 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \emptyset(t) \\ \dot{\emptyset}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 2.6 \\ 0 \\ 9.481 \end{bmatrix} U(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ \dot{x}(t) \\ \emptyset(t) \\ \dot{\emptyset}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} U(t)$$

From equation (7), the stability of the system can be determined by calculating the open-loop poles using equation (9).

$$det\ (SI-A) \quad (9)$$

In equation (9), A is a system matrix and I is the unit matrix. The following result is the open-loop poles by solving equation (9).

Open-loop poles: 0 -0.2092 -6.6396 6.6396

Above these poles can be seen, one of the four poles, 6.6396 locates on the right hand side of the s-plane which stated that the system is unstable. So, a controller has to be designed in order to stabilize the system.

C. LQR Controller Design

LQR is a method of modern control theory that uses state-space approaches to analyze such a system. Using state-space methods, it is relatively simple to work with a multi-output system. The system can be stabilized using full state feedback. The schematic of this type of control system is shown in Figure 3.

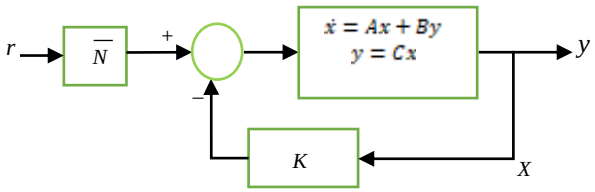


Figure 3: The LQR Control Structure

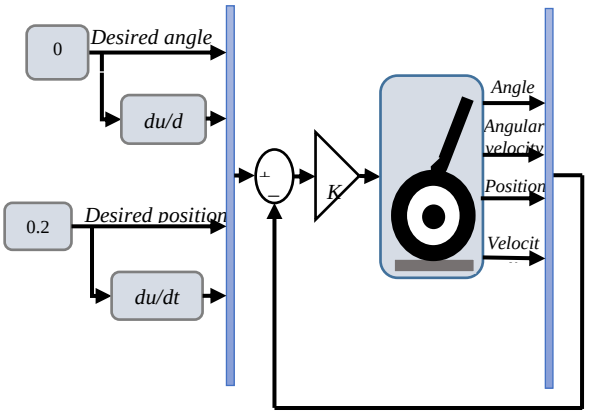


Figure 4: Control Block Diagram

Control block diagram is shown in Figure 4. In this system, angle of body and position of wheel is controlled by using Linear-Quadratic-Regulator (LQR) controller that is appropriate to control body angle and wheel position of the two wheeled self-balancing robot.

In designing LQR controller, LQR function of MATLAB m-file can be used to determine the value of the vector K which determines the feedback control law. This is done by choosing two parameter values, input R = 1 and Q = C'*C where C is from state equation (14). The controller can be tuned by changing the nonzero x and y elements in Q matrix which is done in m-file

code. Consequently, by tuning the values of x = 5000 and y = 100, the following values of matrix K are obtained.

K = [-70.7107 -31.5980 73.4343 12.0140]

In order to reduce the steady state error of the system output, a value of constant gain, *Nbar* should be added after the reference. With a full-state feedback controller all the states are feedback. The steady-state value of the states should be computed, multiply that by the chosen gain K, and use a new value as the reference to computing the input. *Nbar* can be found by using the user-defined function which can be used in m-file code.

The value of constant gain, *Nbar* are found to be:

Nbar = -70.7107

3. Simulation Result

In this section, the simulation results of the system using LQR controller which is performed on the model of two wheel self-balancing robot are described. All figure to express the body angle in radians and the wheel position in meters representing angle in red colour and position in blue.

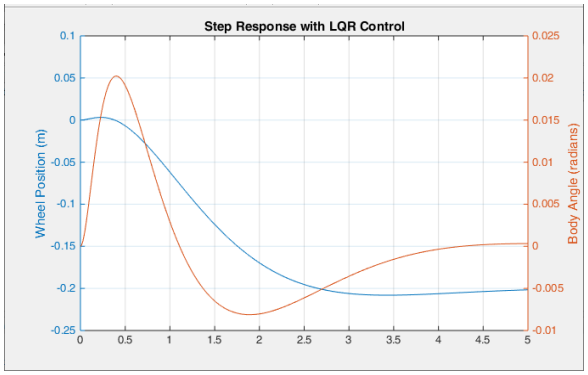


Figure 5: Step Response with LQR Control with Input R=1 and Q = C'*C

In Figure 5, this plot is not satisfactory for desired parameters. The body angle and wheel position overshoot appear fine, but their settling times need improvement and the wheel position rise time needs to be reduced. Moreover, the wheel's final position is not also near the desired location but it has in fact moved in the opposite direction.

Table 2: Settling Time and Overshoot of Angle and Position with R=1 and Q = C'*C

Parameters	Settling Time (second)	Overshoot (%)
Body Angle	4	0.0025
Wheel Position	3	0.002

Therefore, Q matrix in m-file must be modified to get a better response. The (1, 1) element of Q matrix is changed into 5000 and the (3, 3) element to 100.

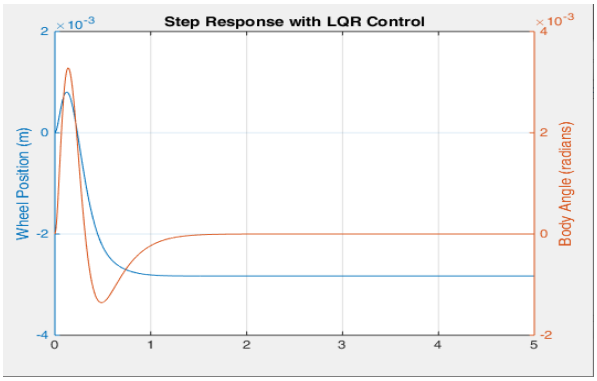


Figure 6: Step Response with LQR Control with Input R=100 and Q = 5000

After changing the R value and Q value of the system, the body angle is reached to the desired angle between 1.3 sec but the wheel position is not reached to the desired position and overshoot has too much. Although it makes the tracking error smaller, this condition is not satisfied. So, pre-compensation is added to the system to get more satisfied design.

Table 3: Settling Time and Overshoot of Angle and Position with R=100 and Q = 5000

Parameters	Settling Time (second)	Overshoot (%)
Body Angle	1.3	0.00003
Wheel Position	1	0.00002

The response achieved above is good, but was based on the assumption of full-state feedback, which is not necessarily valid. To address the situation where not all states variables is measured, a state estimator must be designed. A schematic of state-feedback control with a full-state estimator is shown in below:

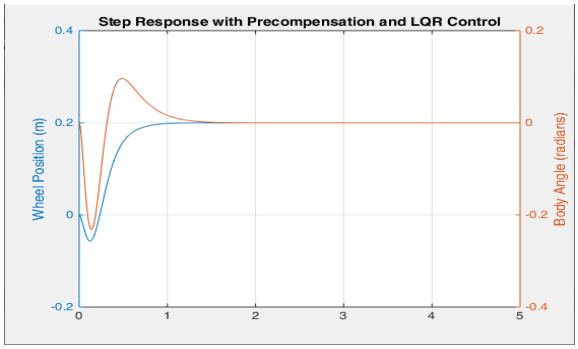


Figure 7: Step Response with Pre-compensation and LQR Control

Table 4: Settling Time and Overshoot of Angle and Position with Pre-compensation

Parameters	Settling Time (second)	Overshoot (%)
Body Angle	1.2	0.001
Wheel Position	0.8	0

After adding the pre-compensation process, the body angle is reached to the desired angle between 1.2 sec and the wheel position is also reached to the desired position and has not to overshoot. So, this condition is satisfied.

4. Conclusion

This paper presents modeling, body angle control and wheel position control of a self-balancing robot. Modern controller (LQR) is designed in MATLAB/Simulink for body angle and wheel position control. According to the simulation results, the body angle is reached to the desired angle between 1.2 sec and the wheel position is also reached to the desired position and do not have to overshoot by adding pre-compensation process. The LQR controller Q and R matrices are determined by trial and error method and A and B matrices generated by linearization in MATLAB. As a result, the responses from LQR controller were plotted in window as.

5. Acknowledgment

The author would like to express sincere appreciation to the Rector of University of Technology (Yatanarpon Cyber City) for kind permission to prepare for this paper. The author would also like to give special thanks to her supervisor and all teachers in Faculty of Electronic Department and all who willingly helped the author throughout the preparation of the paper. This paper is dedicated to the author's parents for continual and full support on all requirements and moral encouragement.

6. References

[1] A. A Bature, S.Buyamin, M.N.Ahmad, and M. Muhamad, "A Comparison of Controllers for Balancing Two Wheeled Inverted Pendulum Robot", International Journal of Mechanical & Mechatronic Engineering IIMME-IIENS, 2014.

[2] Abdullah Cakan, Fatih Mehmet Botasli, "Modeling and LQR control of a wheeled self-balancing robot", 1st International Mediterranean Science and Engineering Congress (IMSEC 2016), Cukurova University, Congress Center, October 26-28, 2016

[3] B. Mabler and J.Haase, "Mathematical model and control strategy of a two-wheeled self-balancing robot', in 39th Annaual Coference of the IEEE Industrial Electronics Society, (IECON), Nov 2013, pp. 4198-4203.

[4] A.N.K. Nasir, M.A. Ahmad and M.F.Rahmat, "Performance Comparison between LQR and PID Controller for an Inverted Pendulum System", International Conference on Power Control Optimization, Chiang May, Thailand, 18-20, July 2008.

[5] Control Tutorials for MATLAB and Simulink – Inverted Pendulum: System Modelling.

[6] Kasruddin , A. N., April 2007, Modeling and Controller Design for an Inverted Pendulum System, Master's Thesis, Faculty of Electrical Engineering, University Teknologi Malaysia.

[7] ME-51015, Vibration and Control, Fifth Year Textbook, Mechanical Engineering.

[8] Rodrigo Or6stica, Manuel A. Duarte-Mermoud, Senior Member, "Stabilization of Inverted Pendulum Using LQR, PID and Fractional Order PID Controllers", IEEE, y Cristian Jauregui.

Performance Analysis of Parallel Coupled Microstrip Bandpass Filter

Hla Thida Nyein[1], Thae Su Aye[2], Cho Thet Paing [3]
Faculty of Electronic Engineering [1,2,3]
University of Technology (Yatanarpon Cyber City) [1,2,3]
Pyin Oo Lwin, Myanmar
nnyein80@gmail.com[1], thaesuaye1483@gmail.com[2], ayecho1@gmail.com[3]

Abstract

This paper presents the performance analysis of parallel couple line microstrip bandpass filter with center frequency at 2.45GHz for wireless LAN(2.4-2.5GHz) application of S band. It expresses the analysis of 5^{th} order and 3rd order parallel coupled line bandpass filter in the insertion loss, return loss and stopband attenuation parameters. The insertion loss method of Chebyshev response (0.5 dB passband ripple) is used to design the filter. The design is based on the use of half wave long resonators and admittance inverters. The filter is implemented using the dielectric constant 4.4 of FR-4 substrate with loss tangent 0.02 and 1.6 mm height. ADS software is used to calculate for physical dimensions of the parallel coupled lines and simulation results.

Keywords- bandpass filter, Chebyshev, Parallel coupled line, Advanced Design System (ADS), Insertion loss method

1. Introduction

A microwave filter is a two port network of frequency selectivity device. Applications can be found in virtually any type of RF or microwave communication, radar, or test and measurement system. The microwave bandpass filter is an essential part of the wireless communication systems. A band-pass filter allows passing the frequencies within a certain desired band and rejects other signals whose frequencies are either below or above the edges of pass-band frequencies [1].

This paper described the performance analysis of parallel coupled microstrip bandpass filter. The design method is based on the use of half wavelength resonator and admittance inverters.The advantages of the parallel coupled microstrip bandpass filters are small physical size, simple design procedures, low cost and easy to fabricate [2]. Its drawbacks are difficult to obtain a narrow passband with low insertion loss and often shift the filter's center frequency.

The center frequency is selected to 2.45 GHz for WLAN application. The order of the filter is calculated by Chebyshev response (equal ripple) with a stopband attenuation of -30 dB at 2.7 GHz and the passband ripple of 0.5 dB. The proposed system includes the design procedures of the filter, the transformation of lumped element to microstrip filter and the simulation and test result includes optimization.

The proposed microstrip filter's parameters (such as space gap between lines, the line width and lengths) are received based on design equations by using LineCalc tool in ADS software. The coupling gaps correspond to the admittance inverters in the lowpass prototype circuit. The even and odd mode characteristic impedances of parallel coupled half wavelength resonators are calculate by using admittance inverters. These even and odd modes impedances values are used to compute physical dimensions of the filter.

2. Filter Specification and Synthesis

The steps followed for designing parallel-coupled band pass filter are: determination of the order of filter and type of approximation functions, determination of the corresponding low-pass prototype, transformation of the lumped circuit element into distributed elements. The insertion loss method of filter design provides lumped element circuits.

The filter specification is,

Center frequency f_c	: 2.45 GHz
Insertion loss	: < 3 dB
Return loss	: > 10 dB
Stopband attenuation	: 30 dB at 2.7GHz (5^{th} order)
Response type	: Chebyshev Response (0.5 dB passband ripple)

Source impedance = Load impedance = 50 Ω

The first step is the calculation of fractional bandwidth (FBW), the normalize frequency and the number of order by Chebyshev response.

Here, f_1 = 2.3 GHz and f_2 = 2.6 GHz, thus

$$\omega_0 = \sqrt{\omega_1 \omega_2} \quad (1)$$

$$\Delta = \frac{\omega_2 - \omega_1}{\omega_0} \quad (2)$$

$$\frac{\omega}{\omega_c} = \frac{1}{\Delta}\left(\frac{\omega}{\omega_0} - \frac{\omega_0}{\omega}\right) = 1.6219 \quad (3)$$

Then the order of the filter is obtained by using equation (4).

$$n \geq \frac{\cosh^{-1}\sqrt{10^{0.1L_{As}-1}/10^{0.1L_{Ar}-1}}}{\cosh^{-1}\left(\frac{\omega}{\omega_c}\right)} \quad (4)$$

where $\omega = 2\pi f$,

ω_0 = the center frequency

f_1 , f_2 = edges of the passband frequencies
L_{As} = the minimum stop band attenuation
L_{Ar} = the passband ripple

3 .Design and Simulation

According to the insertion loss method, lowpass prototype parameters values are required to design the filter. These parameters values are calculated by the following equations(5 to 11) or received with the aid of Table (1), which gives such element values for equal ripple low-pass filter prototypes parameters values for *N* = 1 to 5[1].

A.Lumped Element Prototype

$$g_0 = 1.0 \quad (5)$$

$$g_1 = \frac{2}{\gamma}\sin\left(\frac{\pi}{2n}\right) \quad (6)$$

$$g_i = \frac{1}{g_{i-1}} \frac{4\sin\left[\left(\frac{2i-1}{2n}\right)\pi\right]\sin\left[\left(\frac{2i-3}{2n}\right)\pi\right]}{\gamma^2 + \sin^2\left[\left(\frac{i-1}{n}\right)\pi\right]}, \quad i = 2, 3\ldots n \quad (7)$$

$$g_{n+1} = 1.0, \quad \text{for n = odd} \quad (8)$$

$$\coth\left(\frac{\beta}{4}\right), \quad \text{for n = even,} \quad (9)$$

where $\beta = \ln\left[\coth\left(\frac{L_{Ar}}{17.37}\right)\right]$ (10)

$$\gamma = \sinh\left(\frac{\beta}{2n}\right) \quad (11)$$

Table1. Equal Ripple Lowpass Prototype Parameters Values for 0.5db Passband Ripple [1]

N	g_1	g_2	g_3	g_4	g_5	g_6
1	0.6986	1.0000				
2	1.4029	0.7071	1.9841			
3	1.5963	1.0967	1.5963	1.0000		
4	1.6703	1.1926	2.3611	0.8419	1.9841	
5	1.7058	1.2296	2.5408	1.2296	1.7058	1.0000

The low pass prototype parameters value of 5th order and 3rd order are obtained from the Table (1) and used these values to transform parallel coupled transmission lines.

B. Lumped Element to Microstrip Transformation

The low pass prototype values of 5th order and 3rd order are used in equations (12 to 14) to get the inverter constants for a coupled line filter with N+1 section. The complete equivalent circuit of N+1 coupling with the admittance inverted converter and the transmission line segments connector is shown in Figure (1) [3].

$$Z_0 J_1 = \sqrt{\frac{\pi\Delta}{2g_1}} \quad (1^{st} \text{ coupled line}) \quad (12)$$

$$Z_0 J_n = \frac{\pi\Delta}{2\sqrt{g_n g_{n-1}}}, \quad n = 2,3,\ldots..N \quad (13)$$

(Intermediates coupled lines)

$$Z_0 J_{N+1} = \sqrt{\frac{\pi\Delta}{2g_N g_{N+1}}} \text{ (last coupled line)} \quad (14)$$

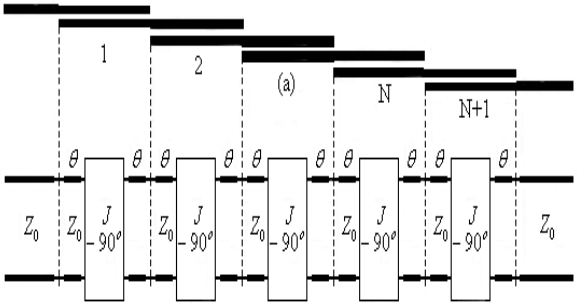


Figure 1: Parallel-Coupled Microstrip Bandpass Filter by Using J Inverters

The calculated values of the inverter constant are used to receive the even and odd mode characteristic impedances. Now the even and odd mode impedances are calculated by the following equations (15-16) and then used to compute the physical dimensions of the filter.

$$Z_{oe} = Z_0[1 + JZ_0 + (JZ_0)^2] \quad (15)$$

$$Z_{oo} = Z_0[1 - JZ_0 + (JZ_0)^2] \quad (16)$$

The lowpass prototypes values, the even and odd mode characteristic impedance values for 3rd order and 5th order are described in Table (2) and Table (3).

Table 2. 5th Order Parallel Coupled Filter's Parameters

n	Value of g	$Z_0 J_n$	Z_{oe}	Z_{oo}
1	1.7058	0.3324	72.1445	38.9045
2	1.2296	0.1302	57.3576	44.3376
3	2.5408	0.1066	55.8982	45.2382
4	1.2296	0.1066	55.8982	45.2382
5	1.7058	0.1302	57.3576	44.3376
6	1.0000	0.3324	72.1445	38.9045

Table 3. 3rd Order Parallel Coupled Filter's Parameters

n	Value of g	$Z_0 J_n$	Z_{oe}	Z_{oo}
1	1.5963	0.3436	73.0830	38.7230
2	1.0967	0.1424	58.1338	43.8939
3	1.5963	0.1424	58.1338	43.8939
4	1.0000	0.3436	73.0830	38.7230

C. Design Simulation Result

The physical dimensions of even and odd mode characteristic impedances values in Table (2) and Table (3) are used to design the parallel coupled line bandpass filter in ADS software and then simulated.

By using the substrate's parameters and the even and odd mode characteristic impedance values in LineCalc Tool of ADS software, the physical dimensions of filter are received. The parameters of substrate set in MSub controller are

Dielectric constant ε_r = 4.4
Metal conductivity = 5.81 e7
Thickness of metal layer = 0.035 mm
Substrate thickness = 1.6 mm
Dielectric loss tangent = 0.02

Figure (2) shows the 5th order parallel coupled line bandpass filter schematic circuit diagram.

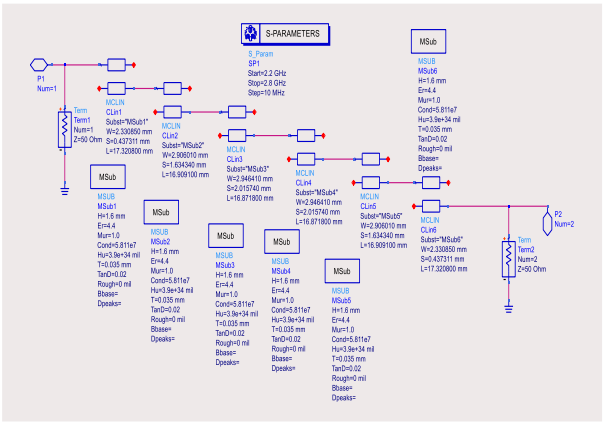


Figure 2. Schematic Diagram of 5th Order Parallel Coupled Line Bandpass Filter

Figure (3) is the EM simulation result of the 5th order parallel coupled bandpass filter. These show the return loss, insertion loss and the stopband attenuation. Besides the center frequency shift to the left and it requires to optimize.

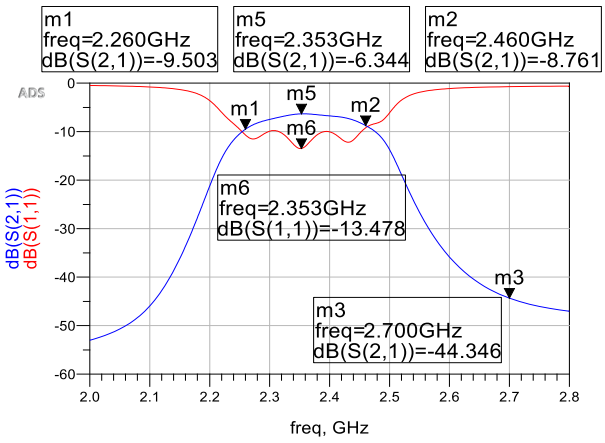


Figure 3.The EM Simulation Result of 5th Order Filter before Optimized

In optimization experiments, decreasing the length (the width and space are kept constant) shifts the frequency band to the right and increasing the length shifts the frequency band to the left. The change of length is not affected on attenuation and insertion loss.

When the length and space are kept constant, the width is increased and decreased in simulation; the insertion loss is increased as the width is increased.

When the length and width are kept constant, the space is increased and decreased in simulation, the insertion loss increases as the space increases.

The optimized simulation result of 5th order is shown in Figure (4). After optimization the insertion loss and return loss at 2.45 GHz are -5.855 dB and -15.155 dB. The stopband attenuation at 2.7 GHz is about -31 .287 dB and 3dB bandwidth range is 200 MHz.

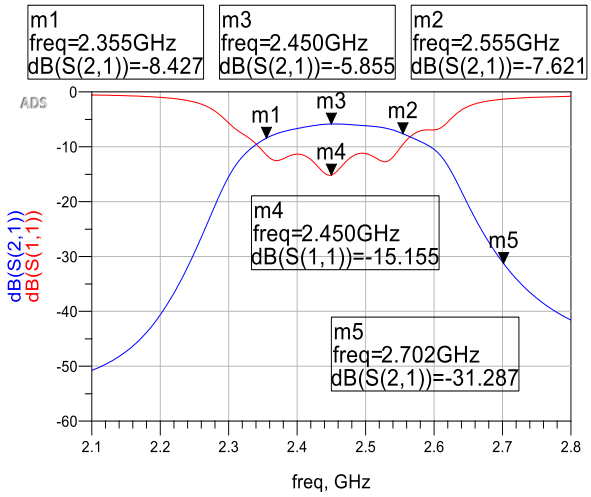


Figure 4.The Optimized Simulation Result of 5th Order Filter

Similarly Figure (5) and Figure (6) shows the schematic diagram and simulation result of 3rd order parallel coupled line filter. Figure (7) is the optimized simulation result.

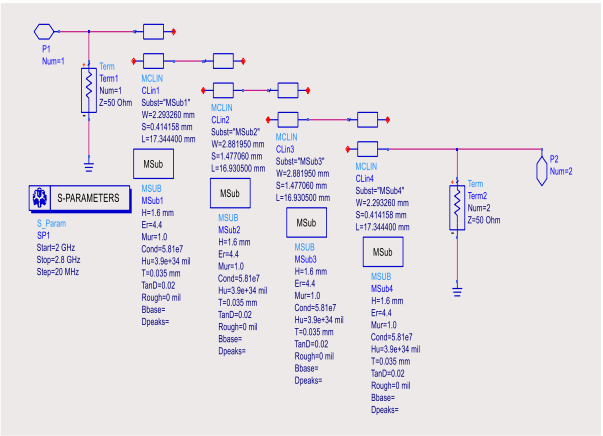


Figure 5.Schematic Diagram of 3rd Order Parallel Coupled Line Bandpass Filter

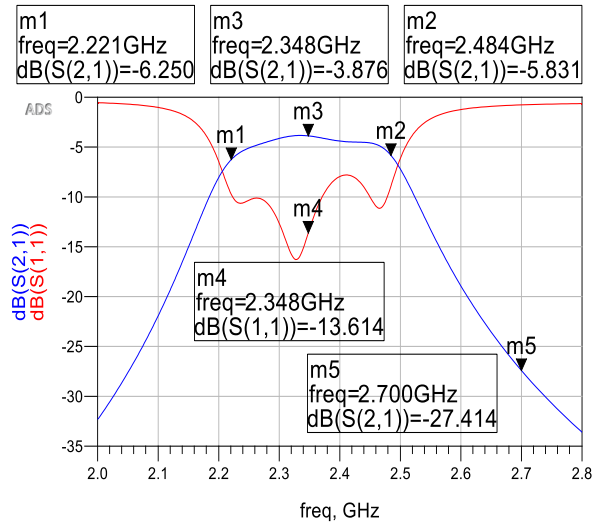


Figure 6.The EM Simulation Result of 3rd Order Filter before Optimization

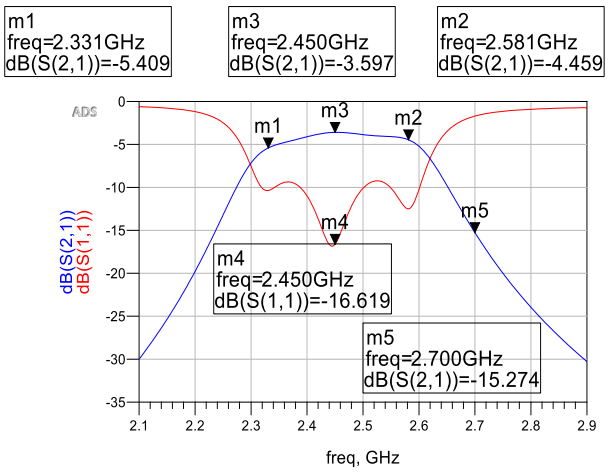


Figure 7.The Optimized Simulation Result of 3rd Order Filter

In figure (7), the insertion loss and return loss at 2.45 GHz are – 3.597dB and -16.619 dB. The stopband attenuation at 2.7 GHz is -15.274 dB and -2 dB bandwidth ranges is about 250 MHz.

Before and after optimizing of the length, width and space of microstrip's parameters values are shown in Table (4) and Table (5).

Table 4.Physical Dimensions of 5th Order Coupled Line Resonators

Lines (nos.)	Before Optimization			After Optimization		
	L (mm)	W (mm)	S (mm)	L (mm)	W (mm)	S (mm)
1	17.32	2.33	0.437	16.628	2.238	0.419
2	16.909	2.906	1.634	16.233	2.789	1.602
3	16.872	2.946	2.015	16.197	2.829	1.835
4	16.872	2.946	2.015	16.197	2.829	1.835
5	16.872	2.906	1.634	16.233	2.789	1.602
6	17.32	2.33	0.437	16.628	2.238	0.419

Table 5. Physical Dimensions of 3rd Order Coupled Line Resonators

Lines (nos)	Before Optimization			After Optimization		
	L (mm)	W (mm)	S (mm)	L (mm)	W (mm)	S (mm)
1	17.344	2.293	0.414	16.45	2.201	0.390
2	16.931	2.882	1.447	16.25	2.756	1.418
3	16.931	2.882	1.447	16.25	2.756	1.418
4	17.344	2.293	0.414	16.45	2.201	0.390

The physical layout of 5th order and 3rd order parallel coupled microstrip filter are shown in Figure (6) and (7).

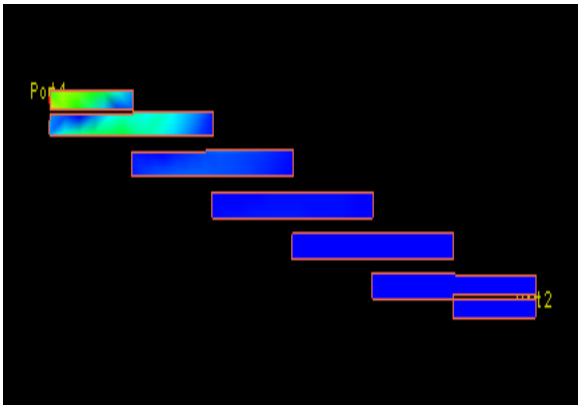


Figure 6.The Physical Layout of 5th Order Parallel Coupled Microstrip Filter

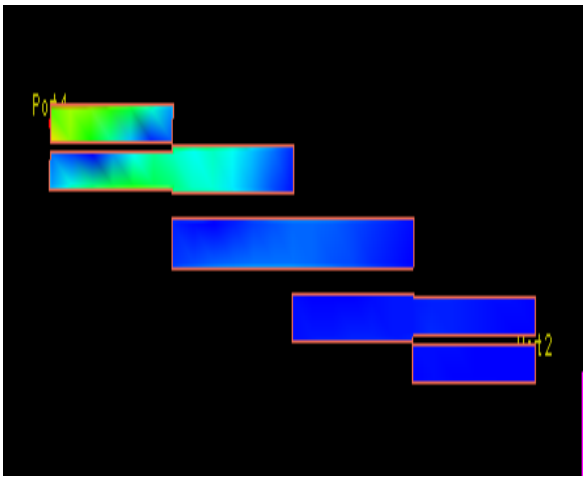


Figure 7.The Physical Layout of 3rd Order Parallel Coupled Microstrip Filter

4. Performance Analysis of the Filters

Figure (8) expresses comparing the insertion loss result of the filters. This shows the 5th order parallel filter has 200 MHz of 3 dB bandwidth ranges and the stop band attenuation is about -31 dB at 2.7 GHz. It can be seen that in Figure (8), the 3rd order parallel coupled filter has 250 MHz of 2 dB bandwidth ranges and the stop band attenuation is about -16 dB at 2.7 GHz .

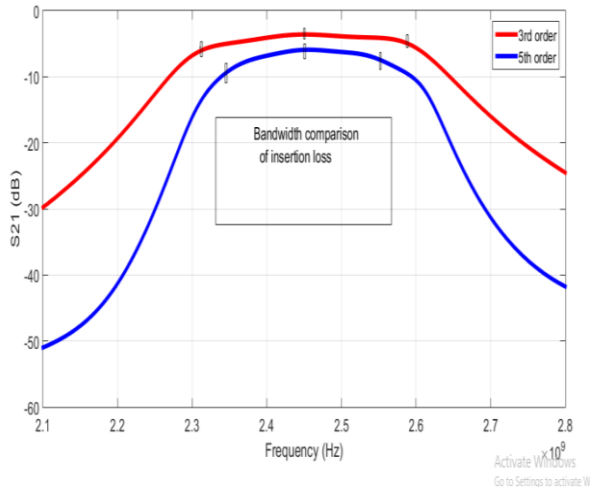


Figure 8.The Result of Insertion Loss in 5th And 3rd Order Parallel Coupled Line Filters

The return loss in 5th order and 3rd order parallel coupled bandpass filter's simulation result is shown in figure (9). In this figure the return loss of 5th order filter is -14.867 dB at centre frequency and 3rd order filter is 16.619 dB at 2.45GHz.The percent of return loss at -10 dB point is 10.2 % in 3rd order and 8.16 % in 5th order.

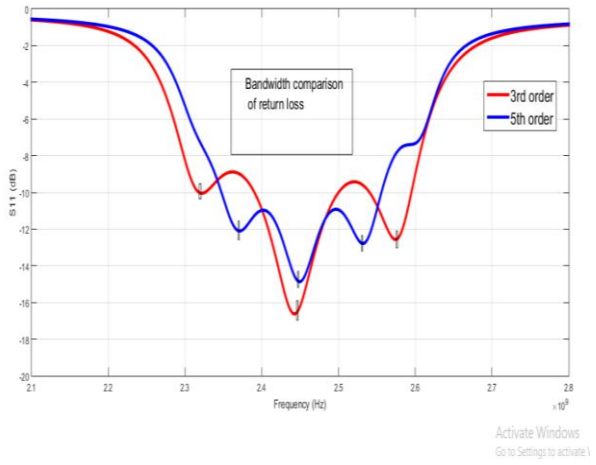


Figure 9.The Result of Return Loss in 5th and 3rd Order Parallel Coupled Filters

The simulation result values are shown in Table (6). The result of the insertion loss and return loss in 5th order are -5.927 dB and -14.867 dB at 2.45 GHz. The insertion loss and return loss in 3rd order is -3.597 dB and -16.619 dB at 2.45 GHz. The stopband attenuation in 5th order is about -31.289 dB and in 3rd order is -15.274 dB at 2.7 GHz.

Each order of filter has their respective advantages depend on the application areas. 3rd order filter has low insertion loss, better return loss and greater bandwidth than the 5th order filter but the stopband rejection is significant in 5th order.

Table 6.The Parameters Values of the Simulation Result

Order of filter	Insertion loss (S_{21}) at 2.45GHz	Return loss (S_{11}) at 2.45GHz	Stopband attenuation at 2.7 GHz
5th order	-5.927 dB	-14.867 dB	-31.289 dB
3rd order	-3.597 dB	-16.619 dB	-15.274 dB

The schematic result of insertion loss in 3rd order filter is better to compared with in [9].The return loss in both is desirable values depending on passband ripple. The percentage of bandwidth for insertion loss at half power point (-2 dB) and the percentage of bandwidth at -10 dB return loss are 10.2% in 3rd order filter. So the bandwidth of 3rd order filter in this system is better than that of [9].

5. Conclusion

In this paper, the parallel coupled bandpass filter has been designed, performance analyzed and proposed for wireless LAN application of microwave systems. The proposed system of 3rd order has the insertion loss -3.597 dB and the return loss -16.619 dB at center frequency with 250 MHz bandwidth. The 5th order filter has the insertion loss -5.927 dB and the return loss -14.867 dB at center frequency with 200 MHz bandwidth. The insertion loss is slightly high because of the substrate's loss tangent 0.02 but both filters have desired insertion loss bandwidth ranges and this is cover for wireless LAN frequency band

.

6. Acknowledgment

The author would like to express sincere appreciation to the Rector of University of Technology (Yatanarpon cyber city) for kind Permission to prepare for this paper. The author would also like to give special thanks to my supervisor and all teachers in Faculty Electronic Engnieering and all who willingly helped the author throughout the preparation of the paper. This paper is dedicated to the author's parents for continual and full support on all requirements and moral encouragement.

7. References

[1] Pozar_David_M.]_Microwave Engineering, Fourth Edition

[2] Salima Seghier,Nasreddine Benahmed and Fethi Tarik Bendimerad , 'Design *of Parallel Coupled Microstrip Bandpass Filter for FM Wireless Applications* '978-1-4673-1658- 3/12/$31.00 ©2012 IEEE

[3] Abdulmajid E. Ferh, hitham Jleed 'Design, *Simulate and approximateParallel Coupled Microstrip BandPass Filter at 2.4 GHz'* 978-1-4799-3351-8/14/$31.00 ©2014 IEEE

[4] Khyati Patel,Brijesh Shah,Vipul Dabhi , '*Design an Performance Analysis of 1 GHz Edge Coupled Microstrip Bandpass Filter'* 978-1- 4799-808-9/ 5/$31.00 © 2015 IEEE

[5] S.Srinath , '*Design of 3rd Order Coupled Line Bandpass Filter for Wireless Application using Agilent ADS' An ISO 3297: 2007 Certified Organization)* Vol. 5, Issue 7, July 2016

[6] Hong, J.S., M.J, Microstrip Filter for RF/Microwave Applications, A Wiley- Interscience Publication,Canada, 2001

[7] Zulkifli Ambak, '*The microstrip coupled line bandpass filter using LTCC technology for 5GHz wireless LAN application*'978-1-4244- 2867-0/08/$25.00 ©2008 IEEE

[8] Agilent EEsof EDA, Advanced Dessign System,Circuit Design Cookbook 2.0. www.AtlantaRF.com, Parallel Coupled Bandpass Filters (Stripline and Microstrip)

[9] Dipak C.Vaghela,*'Design,Simulation and Development of Bandpass Filter at 2.5 GHz'*2015 IJEDR .Vol.3. Issue 2,ISSN;2321-993

Modelling and Speed Control of Unmanned Ground Mobile Robot Using PID Controller

Thinzar Oo[1], Htin Kyaw Oo[2], Lu Maw[3]
Faculty of Electronic Engineering[1,2,3]
University of Technology (Yatanarpon Cyber City)[1,2,3]
Pyin Oo Lwin, Myanmar
thinzaroo282328@gmail.com[1], htinkyawooece@gmail.com[2], lumaw29@gmail.com[3]

Abstract

This paper presents modelling and speed control of differential derived wheeled mobile robot. The vehicle has three wheels. The two are motor drives and one is free wheel. Black-box modelling is used to generate the transfer function of two-wheel derived ground vehicle with system identification toolbox in Matlab. Simulink software is used to implement the Proportional-Integral-Differential (PID) controller that can be used to control the speed of DC motor. Proportional – Integral – Differential (PID) controller designs the desired stability criterions for the speed control of the system. All of testing, simulation results and performances of controller are verified in Matlab.

Keywords- DC motors, Black-box modelling, System identification, PID Controller, Matlab.

1. Introduction

The wheeled mobile robots have been increasingly presented in industrial and service robots such as transportation, surveillance, mining and military missions, Domestic and so on. This research aims to develop of robot technology and to maintain the speed of the motor at the desired stability criterions. A variety of research has been proposed for modelling and stability control for UGV (unmanned Ground Vehicle).There are difficulties for the experiments of the data collected in the system modelling in mathematical form of UGV system identification and control [1]. Different kind of controllers are used for the speed control of DC motor such as FLC (Fuzzy Logic Controller), PID (Proportional-Integral-Differential), IMC (Integral Model Controller) etc. [2]. In this research PID control algorithm is used for speed control process.

1.1. Literature Review

Many of research paper have been discussed different kinds of modelling and control methods for various dynamic systems in different roles. Paper [3] provides detail design and experimental data logging in the system modelling of UGV control. Paper [4] provides to control the motors using Optical Incremental Encoder feedback. Paper [5] describes the PID tunning methods and speed control of DC motors.

In this system, speed control is researched with PID controller and modelling is accomplished with Black-box modelling methods through System Identification toolbox in Matlab.

1.2. System Overview

The proposed system, mobile robot design is a type of wheeled UGV (unmanned ground vehicle). In this research, the mobile robot consists of raspberry Pi3 Model B, Pi 8MP Camera, Arduino UNO, DC 12V power supply and l298N H-bridge module. Raspberry Pi3 Model B and Pi Camera are applied for vision based target tracking of the object for future work. The dynamic transfer functions of two-wheel derived ground vehicle are estimated with Black-box modelling algorithm through System identification toolbox in Matlab. Arduino UNO is supported for input and output data logging for system modelling and speed control of the robot. PID (proportional-integral-differential) based stability control maintains the robot's speed with the desired performance.

1.3. Organization

This paper presents modelling and speed control of two-wheel derived ground vehicle UGV (Unmanned Ground Vehicle) .It is composed as the followings. Section 2 presents system architecture and main concepts, required essential information for this research. Section 3 describes the system design with modelling and closed-loop analysis of two-wheel derived ground vehicle. Section 4 includes PID controllers design for speed control with simulation results. Finally, Section 5 is about the discussions and conclusions.

2. Mobile Robot and Configuration

In this section, system architectures of wheeled UGV (unmanned ground vehicle) are described. Hardware requirements are YFRobot GM25-13CPR JGA25 Gear Motor, Arduino UNO, L298N Dual H-ridge Motor Driver and 12V Battery. Arduino IDE and Matlab software are required for this research.

The robot is made up of three wheels, the two are motor derives and one is free wheel. Two YFRobot GM25-13CPR JGA25 Gear Motors are used. The two motors operate between 6V-12V. RPM values are 240 for 6V and 352 for 12V.Gear Ratio is 1:34 and Load Torque and Stall Torque are 0.85-1.2 kg.cm and 5.6-8 kg.cm respectively. Maximum current value is 250 mA. This motor type is described in Figure 1. The mechanical structure of this system is depicted in Figure 2.

Figure.1 YFRobot GM25-13CPR JGA25 Gear Motor with Encoder

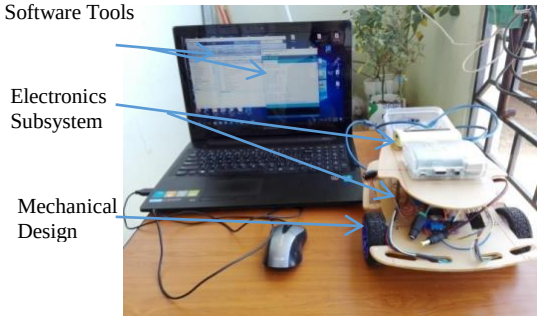


Figure 2. Overall Parts of UGV (Unmanned Ground Vehicle)

The two motors are connected to L298N Dual H-bridge motor driver for controlling speed and direction of motors. It is motor controller breakout board and can control by pulse width modulation (PWM).

Arduino UNO board is a microcontroller based on the ATmega328.It is implemented to drive the motors by applying PWM signal to the motor driver and to read sensors (encoders) data for the feedback loop. The software used for this controller is Arduino Software (IDE).

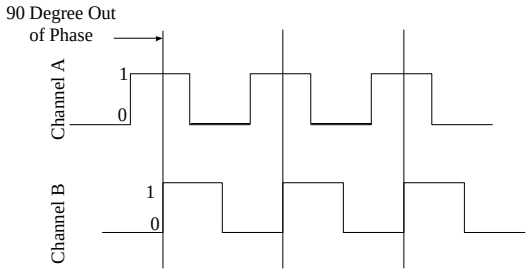


Figure 3. Quadrature Optical Encoder Pulses

A quadrature encoder is used to get feedback for controlling the motors. The encoders are implemented with 2 out-of-phase output channels (Channel A and Channel B).Direction of movement can be determined based on the phase relationship between two channels[5].These tracks or channels are coded ninety electrical degree out of phase to each other in Figure 3.

3. System Modelling and Closed-loop Analysis

This section describes black-box modelling and open-loop analysis of two DC motors.

3.1. System Modelling

For Black-box modelling processes, the System Identification toolbox is used to generate the transfer function of the dynamic system. Black-box modelling is usually a trail-and-error process, where the parameters of various structures are compared with the results. This process does not need to measure physical parameter of the system. It needs only experimental input and output data in time-domain at a uniform sampling interval. The sampling time (T_s =0.1) second is used during the data acquisition. This type of modelling is less accuracy compared with mathematical modelling. The flowchart the procedure of modelling in system identification describes in Figure 4.

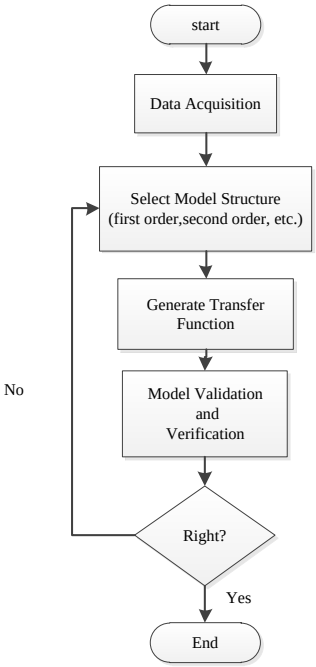


Figure 4. Procedure of Modelling in System Identification

In modelling process, four steps are required. Firstly, data acquisition process, input output data of the system must catches in sampling time 0.1 second. Figure 5 shows data logging of input and output data on smooth floor. And then, the second step is selected

model structures with the best fits to accurate the collected data in data acquisition. Among them second order with one zero and two poles are chosen for both left and right motor. The next step is estimation method is applied to generate the approximate transfer function for the system. Finally, validate the simulation model and output of real system and then verify outputs.

Figure 5. Photo of Testing the System for Data Acquisition Process on Smooth Floor

Quadrature Encoder pulses are counted and converted to the rad/s by Eq. (1).

$$\text{Velocity}, \omega \left(\frac{\text{rad}}{\text{s}}\right) = \frac{\frac{\text{Encoder Pulses}}{\text{Pulses/Revolution}} \times \frac{60\ \text{Sec}}{\text{min}}}{\text{Fixed time interval (second)}} \times \frac{2\pi}{60} \quad (1)$$

For this system, maximum speed is approximately 30 rad/sec at a maximum PWM (Pulse Width Modulation) pulse.

After data acquisition, input and output data are imported for each motors at System Identification toolbox in Matlab. The input data as (PWM) and the output data as (rad/sec) of left and right motors are described in Figure 6.

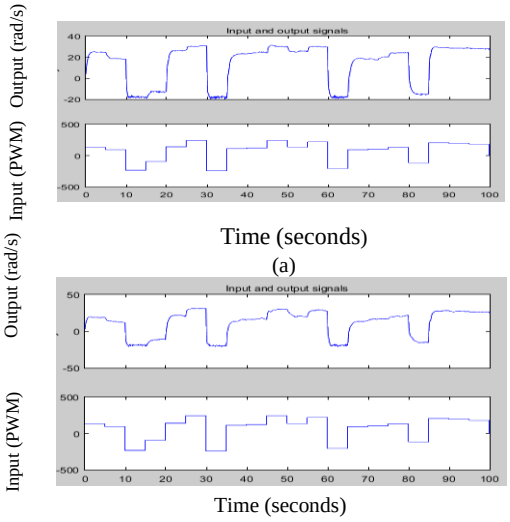


Figure 6. Output (rad/s) and Input (PWM) Data Logging (a) Left Motor (b) Right Motor

Approximate model orders (first and second order, etc.) are estimated in System Identification. Second order transfer functions with one zero and two poles are chosen for left and right motors. Left motor and right motor best fits are 78.73% and 84% respectively. They are shown in Figure 7.

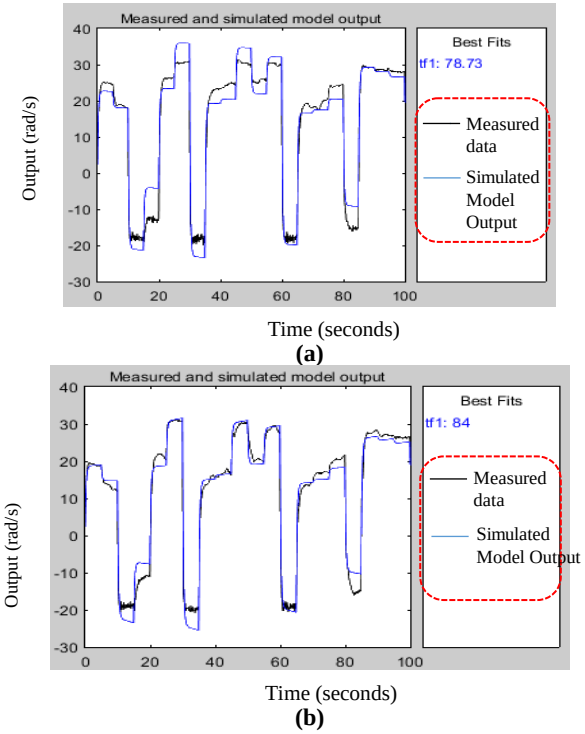


Figure 7. Transfer Function Estimation in System Identification Toolbox (a) Left Motor (b) Right Motor

After generating transfer functions, model validation is performed by comparing simulated transfer function and real system measured data output. In this process different input and output data are imported to the system identification toolbox again. After validation, the second order with one zero and two poles model is the most convenient for both left and right motors.

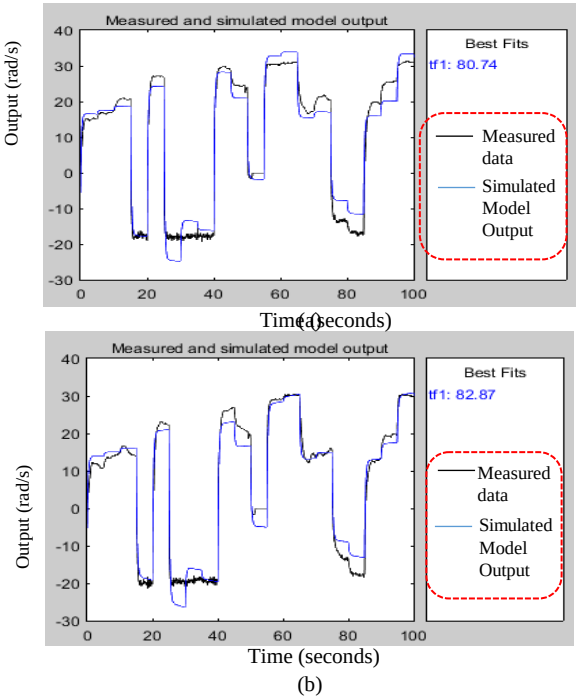


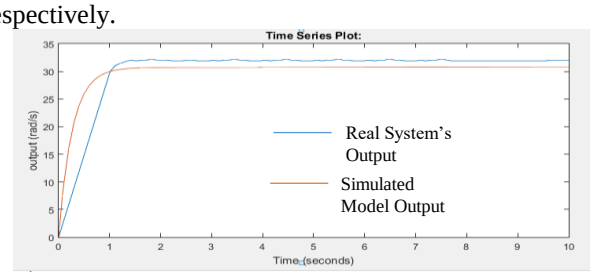
Figure 8. Model Validation at System Identification Toolbox (a) Left Motor (b) Right Motor

Left motor and right motor transfer functions of UGV are described in Eq. (2) and Eq. (3).

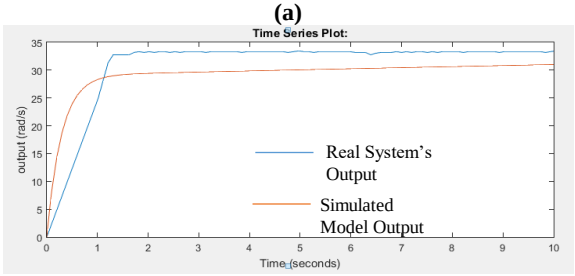
$$\frac{\omega}{V} = \frac{0.4508s + 0.002839}{s^2 + 3.753s + 0.02125} \quad (2)$$

$$\frac{\omega}{V} = \frac{0.3924s + 0.00944}{s^2 + 3.456s + 0.05751} \quad (3)$$

Finally, verifying the simulated model output with the real system's output. Both the input values of two models are applied by PWM (255). According to this process, the comparing results are shown in Figure 9.Accuracy of left motor is 98% and right motor is 95% respectively.



(a)



(b)

Figure 9. Verification Results with PWM value 255 (a)Left Motor (b) Right Motor

3.2. Closed-Loop Analysis

For the closed-loop analysis, step input is fed to closed-loop system without controller is shown in Figure 10,

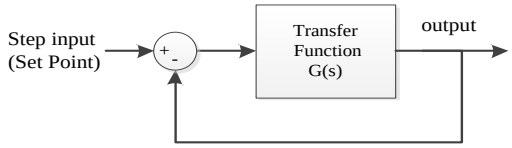
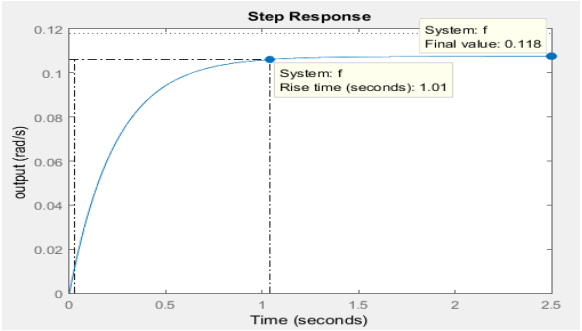
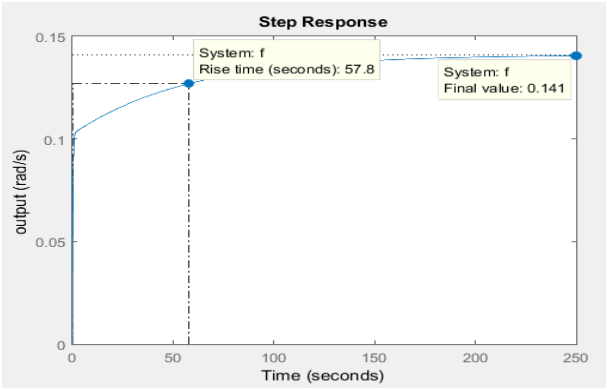


Figure 10. Block Diagram of Closed-Loop System without Controller

Both the output step responses of left and right motor are analyzed in Matlab Simuink. The closed-loop system is unstable which can be seen in Figure 11.



(a)



(b)

Figure 11. Closed-Loop Step Responses (a) Left Motor (b) Right Motor

According to the step responses of left and right motors, these are also not satisfied the desired specifications in Table 1.

Table 1.Desired Performance Specifications

Parameter	Symbol (Unit)	Desired Performance
Settling Time	T_s (s)	<0.5s
Rise Time	T_r (s)	<2%
Maximum Percent overshoot	M_p(%)	<0.8s
Steady State Error (Ess)	E_{ss}	<2%

So the controller must be needed to compensate for this problem. PID controller is chosen for left and right motors in my system. PID controller is sufficient and discuss about it in Section 4.

4. PID Controller Design for Speed Control with Simulation Results

In this section presents PID controller design and speed control of two dc motors. The simulation results of the speed are verified with Matlab Simulink to obtain the desired specifications.

4.1. PID Controller

Closed-loop control, also known as feedback control is needed to achieve and maintain the desired output by comparing it with the actual output.

In this system, PID controller is performed to control the motor speed. There are three parameters are determined, proportional gain by K_p, integral gain K_i and derivative gain by K_d to obtain the desired response. The control block of PID controller is shown in Figure 12.

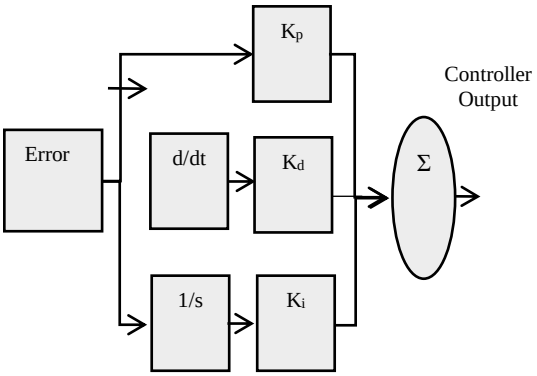


Figure 12. Control Block of PID Controller

The continuous PID controller equation is shown in Eq. (4).

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (4)$$

The error between the desired speed and the actual speed is given as input to a PID controller. The PID controller depend on the error changes its output and control the input of the dynamic model. Proportional controller K_p reduces the steady state error and rise time but does not eliminate the error. So Integral controller, K_i is used to solve it. It also increases overshoot and settling time. Derivative controller K_d reduce these effects. Feedback system architecture with controller is shown in figure 13. The effects of each controller are shown in Table 2.

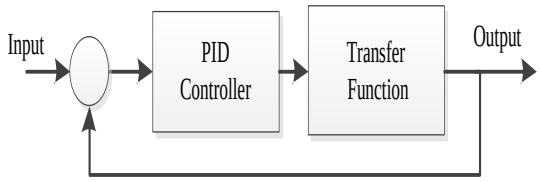


Figure 13. Feedback System Architecture

Table 2.Effects of Parameters in PID Controller

Parameter	Rise Time (T_r)	Maximum Overshoot (M_p)	Settling Time (T_s)	Steady State Error (E_{ss})
K_p	Decrease	Increase	Small Change	Decrease
K_i	Decrease	Increase	Increase	Eliminate
K_d	Minor Change	Decrease	Decrease	No effect

4.2. Speed Control

Block diagram of speed control is shown in Figure 14. In this system two separately PID controllers are used for two motors. The output speeds are felt back to their inputs through the left and right encoders of two DC motors.

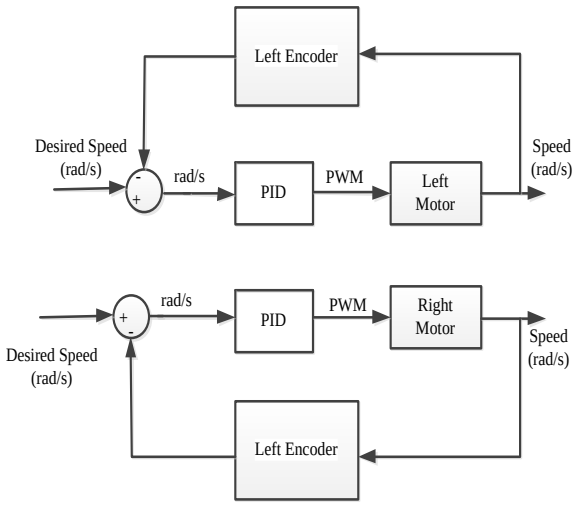


Figure 14. Block Diagram of Speed Control

In controlling method, comparing the performances of Proportional (P), Proportional-Integral (PI), and Proportional-Integral-Differential (PID) controllers are shown in Table 3 and Table 4 for both auto and manual tunings.

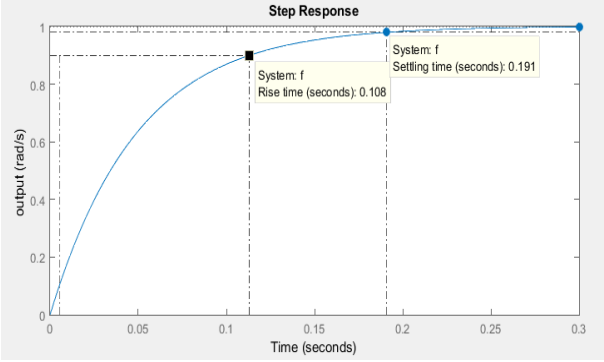
Table 3. PID Tuning for Speed Control of Left Motor

Parameter	PID (Manual)	PID (Auto)	P (Manual)	PI (Manual)
K_p	45	115.1102	600	45
K_i	170	1640.4705	0	170
K_d	0.01	0.39719	0	0
Rise Time (sec)	0.1084	0.0328	0.0081	0.1079
Settling Time (sec)	0.1914	0.198	0.0145	0.1906
Maximum Precent Overshoot (%)	0	8.7	0	0
Steady State Error	0	0	1	0

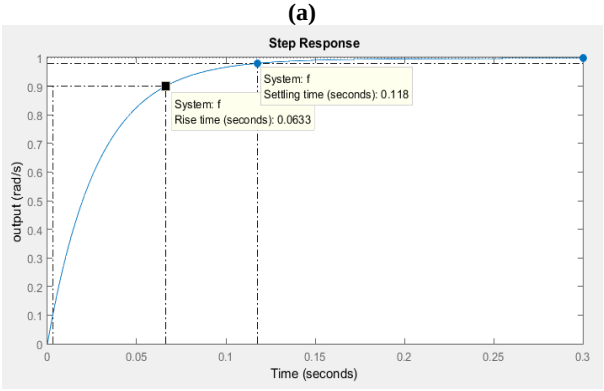
Table 4. PID Tuning for Speed Control of Right Motor

Parameter	PID (Manual)	PID (Auto)	P (Manual)	PI (Manual)
K_p	90	93.5349	1000	90
K_i	290	989.945	0	290
K_d	0.01	-0.50714	0	0
Rise Time (sec)	0.0635	0.0356	0.0056	0.0633
Settling Time (sec)	0.1182	0.2286	0.0102	0.1177
Maximum Precent Overshoot (%)	0	9.3	0	0
Steady State Error	0	0	1	0

According to Table 3 and Table 4, PI controller gives the best response for this system. So PI controller is chosen for two DC motors. The result curves of left and right motors are shown in Figure 15.



(a)



(b)

Figure15. Step Responses of Speed Control for Two DC Motors (a) Left Motor (b) Right Motor

5. Discussion and Conclusion

This paper presents modelling for two-wheel derived vehicl*e* by using Black-box modelling method and the speed control by using PID controller. Black-box modelling is performed in System Identification Toolbox in Matlab. The two transfer functions of two-wheel derived vehicl*e* are achieved second order with one zero and two poles with the Best Fits 78.73% for left motor and 84% for right motor respectively. Closed-loop analyses using Simulink. It does not satisfy the required performances. Closed-loop feedback control is used to satisfy these problems. PID controller is used for speed control of two-wheel derived ground vehicl*e*. In closed-loop control system, speed control is an inner loop control. In this paper, speed control is only verified with simulation in Matlab, but identification and model verification is obtained from the experiment.

In PID controller design, different controllers (P, PI, PID) are discussed and analysed with their performances.

All of the simulation results are verified with comparison and discussion. Among them, PI controller is chosen for speed control of this system .It is satisfied with the desired performances for this system.

6. References

[1] Johan Schoukens, "System Identification: what does it offer to electrical engineers", Vrije Universiteit Brussel, dep. ELEC, Pleinlaan 2, B1050 Brussels, Belgium.

[2] Md Akram Ahmad, Kamal Kishor, Pankaj Rai,"Speed Control of a DC motor using Controllers", Automation, Control and Inteligent Systems , Volume-2, Issue-6-1,December 2014.

[3] Seung Min Lee, Wei Phing Ang, Stephanie Sumcad, and Charles Tytler, "Unmanned Ground Vehicle System Identification and Control", Department of Aeronautics and Astronautics, Purdue University.

[4] Bharat Joshi, Rakesh Shrestha, Ramesh Chaudhary, "Modeling, Simulation and Implementation of Brushed DC Motor Speed Control Using Optical Incremental Encoder Feedback", IOE, Central Campus, Pulchowk, Tribhuvan University, Nepal, November, 2015.

[5] Hang Wu, Weilhua Su, Zhiguo Lium,"PID Controller Design and tuning methods", Industrial Electronics and Applications (ICEIA), 2014 IEEE 9th Conference on Vol., no.,pp.808,813,9-11 June 2014.

[6] K. Rajanwal, R. Shakya, S. Patel, R.K. Maurya "Comparative Analysis of PI, PID and Fuzzy Logic controllers for Speed Control of DC Motor", International Journal of engineering Research and Technology, Vol. 3-Issue 1,pp. 1319-1324,Jan 20,2014.

[7] Mario Gavran, "PI Controller for DC Motor Speed Realized with Arduino and Simulink", Faculty of Electrical Engineering and Computer Science, Maribor, Slovenia.

[8] R. K. Munje, M.R. Roda, B.E. Kushare, "Speed Control of DC Motors using PI and SMC" Proc. of IEEE Int. Conf. on Power and Energy, Singapore, Oct 2010, pp.945-950.

[9] Fatiha Loucif "DC Motor Speed Control Using PID Controller", Department of Electrical Engineering and Information, Hunan University, Chang Sha, Hunan, China, 2005.

[10] A. M. Karnik, N. K. Sinha, "Modelling A Robot Arm from Sampled Input-Output Data",Department of Electrical and Computer E ngineering, McMaster University, Hamilton, 1985.

[11] Wei Wu, "DC Motor Parameter Identification Using Speed Step Responses", Flight Control and Navigation, 3, September, 2012

[12] William T. Kirchner, Steve C. Southward, "Adaptive Vehicle Traction Control: Combined Longitudinal and Lateral Motion", Springer-Verlag Berlin Heidelbreg, 2013.

Design and Analysis of Microstrip Patch Antenna for Wireless LAN Application by Using Inset Feed Technique

Pyae Phyo Swe[1],Cho Thet Paing[2], Kyu Kyu Mar[3]
Faculty of Electronic Engineering[1,2,3]
University of Technology (Yatanarpon Cyber City) [1,2,3]
Pyin Oo Lwin, Myanmar
pyaephyoswe0036@gmail.com[1], ayecho1@gmail.com[2], thankyumar@gmail.com[3]

Abstract

Antenna is the critical part of wireless communication's applications. In this paper, the microstrip patch antenna is designed for Wireless LAN application using microstrip inset feed. The antenna is designed on Fr4 epoxy substrate material with dielectric constant 4.4. The operating frequency of 2.45 GHz is achieved with the return loss value, about 30.1dB. By using the microstrip inset feed technique and substrate thickness of 3.2mm, the proposed patch antenna obtained the increment of bandwidth and return loss. The significant improved performance of microstrip inset feed patch antenna can be used in (2.412GHz-2.484GHz) Wireless LAN application and real time implementation. The patch antenna's parameters such as Return Loss, VSWR, Radiation Pattern and Impedance Matching are achieved by analyzing the width, length and thickness of substrate using on Feko software.

Keywords- Wireless LAN, FR4 epoxy, FEKO

1. Introduction

For communication system, wireless technology provides less expensive alternative and a flexible way. Through the years, this technologies have been different standard. Today, there are more and more interest in research on multipurpose systems in the wireless communication[10]Wireless LAN is one in which a mobile user can connect to a local area network (LAN) through a wireless (radio) connection[2].If the radiation is not achieved, wireless transmission is not attainable. Antenna is needed for this transmission to be achieved the radiation [4].So, low profile antenna of microstrip patch is designed.

Low profile antenna is required in high performance communication applications where size, weight, cost, performance, ease of installation and aerodynamic profile are constraints [8, 3].These applications are many other government and commercial applications such as mobile radio and wireless communications that have similar specifications. Microstrip patch antennas are used in wireless communication [4, 10]. In spite of many advantages, these antennas suffer from some disadvantages which include their low efficiency, low power, high Q, spurious feed radiation, and very narrow bandwidth. [9]There have been considerable efforts made by researchers from all over the world toward increasing its efficiency .The choice of antenna selection is based on the requirements of the application such as frequency band, gain, cost, coverage, weight, etc. [5].

In this paper, 2.45GHz frequency band microstrip patch antenna for WLAN application is designed and simulated using FEKO software [1].By analyzing the width, length and thickness of substrate, patch antenna's parameters are obtained to design an efficient patch antenna for Wireless LAN application.

The rest of paper is composed of antenna design in section 2, simulation results are discussed in section 3. Finally, the paper is concluded in section 4.

2. Antenna Design

To design the patch antenna some parameters are necessary such as resonant frequency, dielectric constant, substrate height[6].The proposed antenna is designed on a FR4 epoxy substrate with a thickness (h) equals to 3.2mm, a relative permittivity ε_r of 4.4.The patch antenna design is calculated by using the following formula.Conception and simulations are realized using FEKO.

Patch Width

$$W = \frac{c}{2f_r\sqrt{\frac{\varepsilon_r+1}{2}}} \quad (1)$$

Where,
W = width of the patch
f_r = resonating frequency (or) operating frequency
ε_r = relative permittivity (or) dielectric constant

Effective Dielectric Constant

$$\varepsilon_{reff} = \frac{\varepsilon_r+1}{2} + \frac{\varepsilon_r-1}{2}\left[1+\frac{12h}{W}\right]^{-0.5} \quad (2)$$

Where

ε_{reff} = effective dielectric constant

h = thickness of substrate

Effective Length

$$L_{eff} = \frac{c}{2f_r\sqrt{\varepsilon_{reff}}} \quad (3)$$

Where,

L_{eff} = effective length

Length Extension

$$\Delta L = 0.412h\frac{(\varepsilon_{reff}+0.3)(\frac{W}{h}+0.264)}{(\varepsilon_{reff}-0.258)(\frac{W}{h}-0.8)} \quad (4)$$

Where,

ΔL = length extension (or) fringing length

Patch Length

$$L = L_{eff} - 2\Delta L \quad (5)$$

Where,

L = patch length

Substrate Material and Ground Plane Dimension

$$W_s \times L_s = 0.49\lambda_{2.45GHz} \times 0.49\lambda_{2.45GHz}$$

$$W_g \times L_g = 0.49\lambda_{2.45GHz} \times 0.49\lambda_{2.45GHz}$$

Where,

W_s = width of substrate

L_s = length of substrate

W_g = width of ground plane

L_g = length of ground plane

Feeding Techniques

Microstrip feeding techniques: The fabrication of microstrip feed technique is easy. And, inset feed is simple to contest by controlling the feed position. But, the more increase the thickness of substrate, the better performance of feed radiation [7].

- *Width Of 50Ω Feeding Line:*

$$W_f = \frac{8e^{1.529}}{e^{2A}-2} \quad (6a)$$

$$\left\{A = \frac{Z_0}{60}\sqrt{\frac{\varepsilon_r+1}{2}} + \frac{\varepsilon_r-1}{\varepsilon_r+1}(0.23+\frac{0.11}{\varepsilon_r})\right\} \quad (6b)$$

- *Length of 50Ω Feeding Line:*

$L_f \leq$ Patch of length (L)

- Inset Feed Depth (y_0)

$$R_{in}|_{(y=y_0)} = R_{in}|_{(y=0)}\cos^2(\pi \times \frac{y_0}{L}) \quad (7a)$$

$$R_{in} = 90\frac{\varepsilon_r^2}{\varepsilon_r-1}(\frac{L}{W})^2 \quad (7b)$$

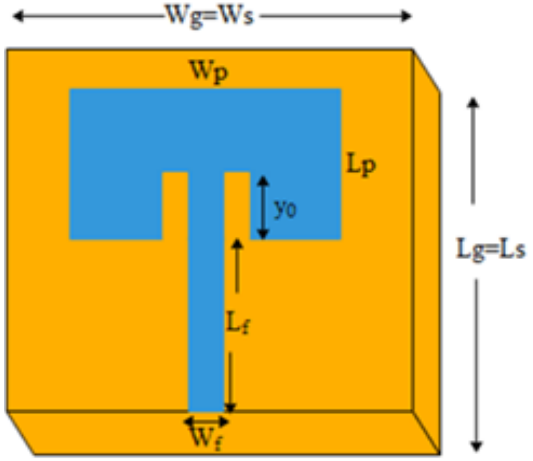


Figure 1. Design Evolution of Proposed Antenna

So, considering the feeding techniques of the proposed antenna designs is shown in Figure 1.which is printed on FR4 substrate of thickness 3.2 mm and permittivity 4.4. The resonating frequency is 2.45GHz.

Table1. Parameters of Proposed Patch Antenna

Parameters	Dimensions
patch width, w_p	50mm
patch length, l_p	26.8mm
50Ω feed width ,w_f	6mm
50Ω feed length, l_f	24mm
inset feed width,x_0	1mm
inset feed depth,y_0	4mm
substrate width (w_s)× substrate length, (l_s)	60mm×60mm
ground plane width, (w_g)× ground plane length, (l_g)	60mm×60mm

3. Simulation Results & Discussion

By using the parameters mentioned in Table1.the proposed patch antenna will be designed with FEKO software.

The microstrip inset feed patch structure designed on FEKO software is shown in Figure 2. microstrip patch antenna can be designed with various shapes. But, the proposed antenna design is the rectangular shape. This shape is suitable for the 2.45GHz Wireless LAN applications.

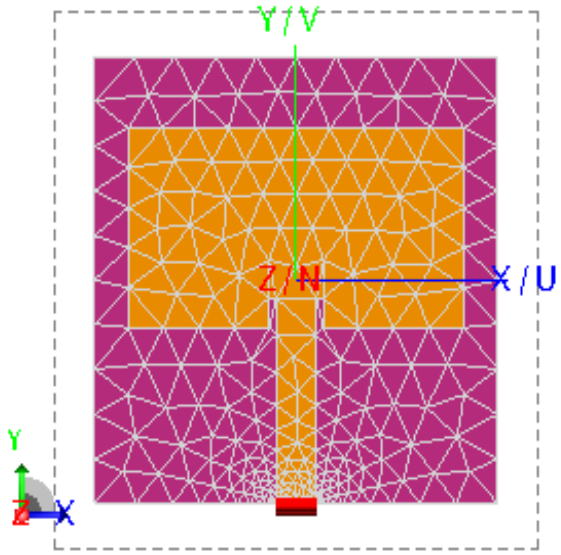


Figure 2. Design Implementation of Proposed Antenna Using Feko Software

A.Performance comparison with difference width and length of microstrip inset feed patch antenna

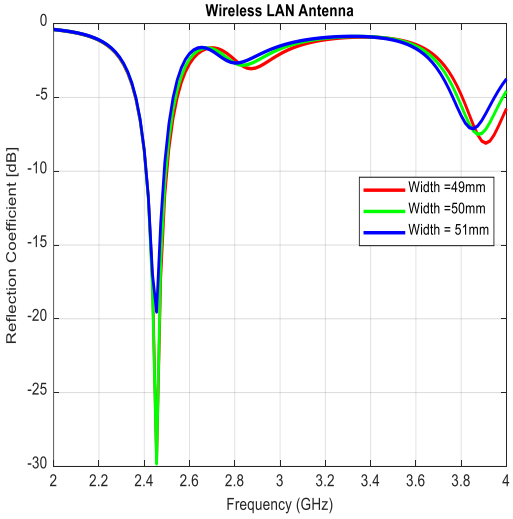


Figure 3. Performance Comparison with Difference Width

The Figure 3.shows the result of reflection coefficient(dB) , by considering the difference width of patch antenna .At the patch width value 50mm,the reflection coefficient (dB) is around -30 dB . The other patch width value is not good for the reflection coefficient value. So, the proposed antenna's patch width value 50mm is selected.

In Figure 4.the difference length of patch antenna is considered. At the patch length 26.8mm, the operating frequency have 2.45GHz.The other parameter length do not coverage the wireless LAN (2.412-2.484) frequency band application [11].

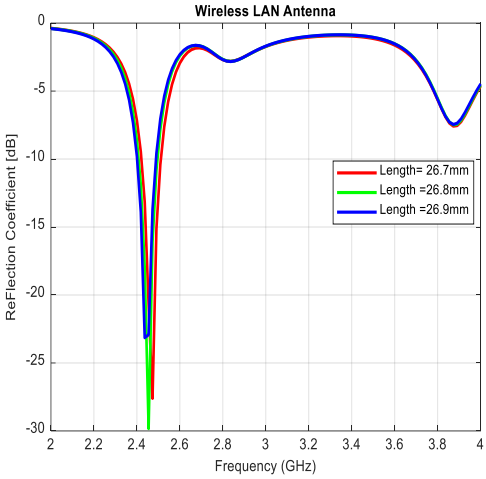
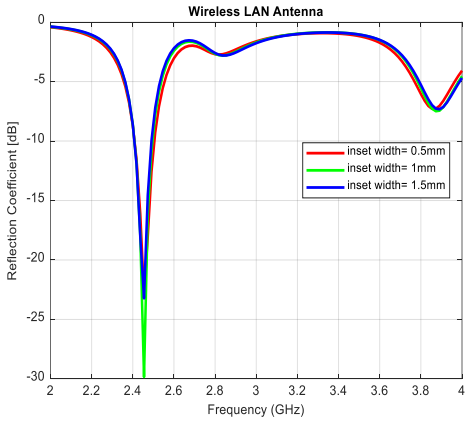


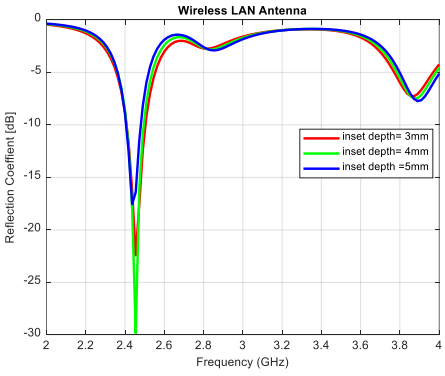
Figure 4. Performance Comparison with Difference Length

Wireless LAN communication is need to achieve the coverage area .Hence, the proposed patch antenna length have been chosen as 26.8mm.

B. Performance comparison with difference parameter of inset fed width and inset fed depth



(A). Performance Comparison with Difference Inset Fed Width



(B) Performance Comparison with Difference Inset Fed Length

Figure 5. Performance Comparison with Difference Parameter of Inset Fed Width and Inset Fed Depth

Figure 5.shows the result of antenna performance with difference parameter of inset fed width and inset fed depth. At the inset feed width value 1mm and inset feed length value 4mm is achieved good performance of reflection coefficient value and coverage bandwidth.

C. Comparison of Bandwidth Performance base on the Substrate thickness

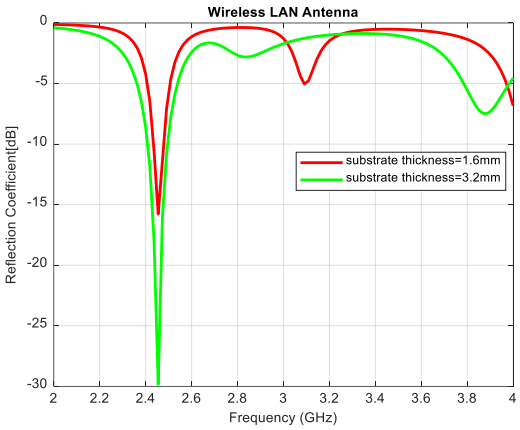


Figure 6. Comparison of Bandwidth Performance Base on the Substrate Thickness

Figure 6.shows the bandwidth performance of patch antenna. The proposed patch antenna is used with the substrate thickness 3.2mm.Substrate thickness 1.6mm cannot support the frequency coverage for the proposed patch antenna Wireless LAN application [11].And the reflection coefficient value is not achieved the good performance. Hence, the proposed patch antenna of the substrate thickness 3.2mm is picked. According to the substrate thickness 3.2mm, the antenna bandwidth is coverage for the Wireless LAN application [11].Moreover, the performance of reflection coefficient value is better than the substrate thickness 1.6mm.

D. Simulation results of proposed patch antenna

According to the simulation results mentioned above, 50mm in width, 26.8mm in length and substrate thickness 3.2mm are selected to design proposed system for simulation.

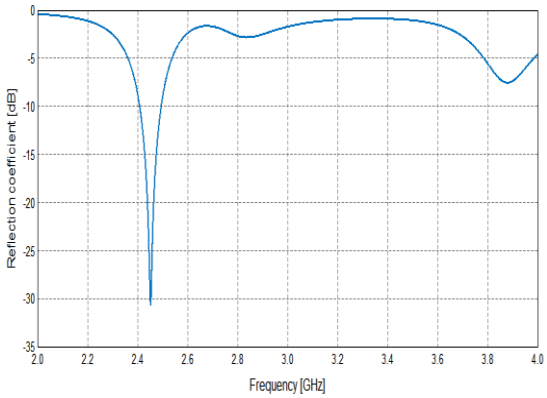


Figure 7. S Parameter Vs. Frequency Plot

The S11 parameter for the proposed antenna was calculated and the simulated reflection coefficient results are shown in Figure 7. The value of reflection coefficient value is -30.1 dB in this proposed antenna. The achieved reflection coefficient value is small enough and frequency is very closed enough to the specified frequency band for 2.45 GHz Wireless LAN application. S_{11} parameter which has value less than -10dB is obtained at (2.409-2.493) GHz frequency range. At operating frequency 2.45GHz, the return loss is achieved with 30.1dB.

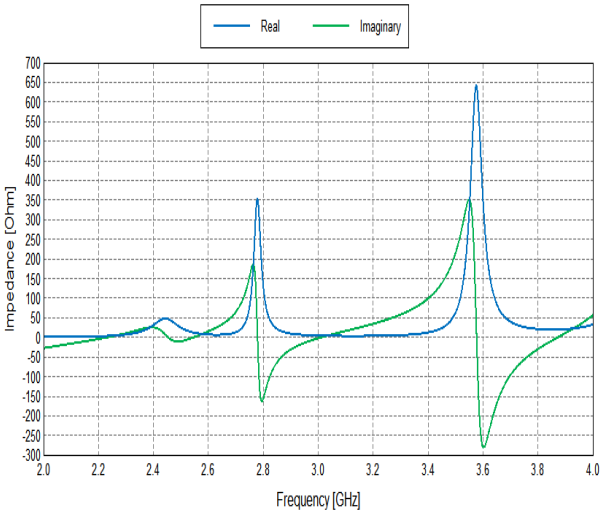


Figure 8.Impedance Vs. Frequency Plot

Impedance matching between source and load is also important parameter for designing the antenna. In this research, the impedance matching is achieved. Because the proposed patch antenna is used with the microstrip inset feeding techniques. The impedance matching of proposed patch antenna is described in the Figure 8.

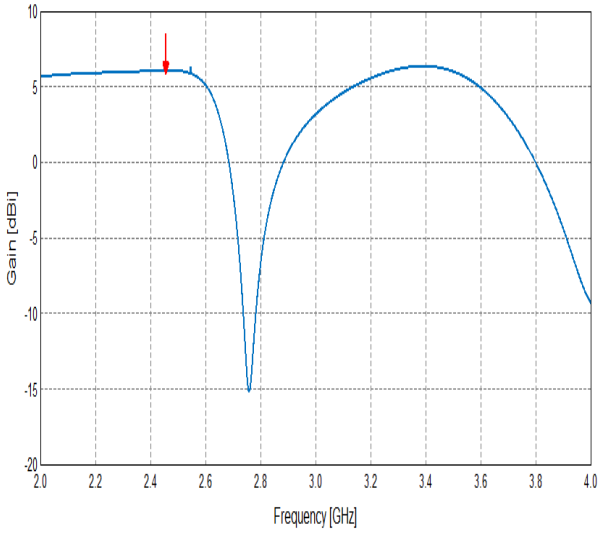
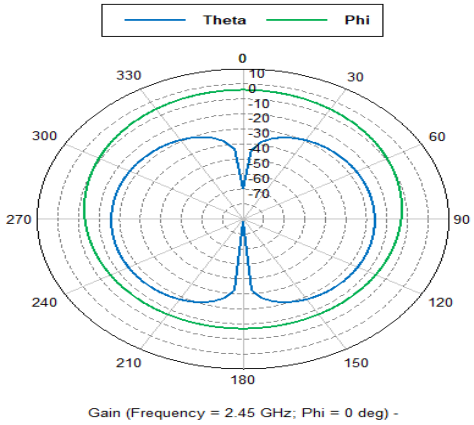
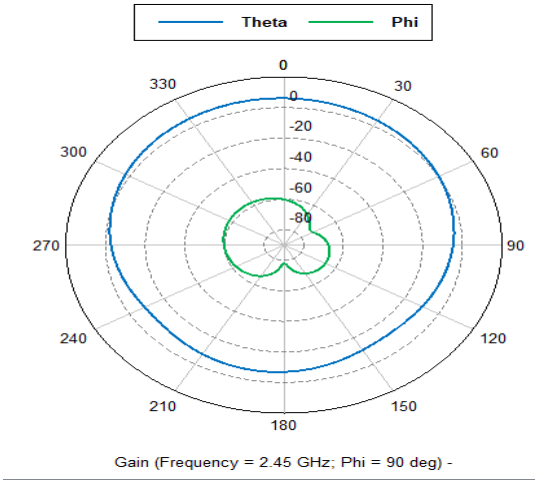


Figure 9. Gain Vs. Frequency Plot

The gain versus frequency plot of proposed patch antenna is shown in Figure 9.The gain of the antenna is lowest between 2.7GHz and 2.95GHz.In this frequency, there is no radiation in the positive Z direction ($\theta = 0°, \emptyset = 0°$) .The proposed patch antenna gain value is achieved 6dBi at 2.45GHz by using the substrate of loss tangent value 0.001.This gain value is enough for the wireless LAN application.



(A)Radiation Pattern in XZ Plane



(B)Radiation Pattern in YZ Plane

Figure 10. Radiation Pattern of Proposed Patch Antenna

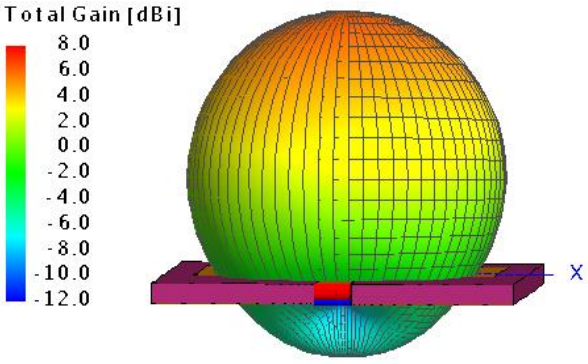


Figure 11. 3D Radiation Pattern of Proposed Patch Antenna

Figure 10 and 11 show the radiation pattern of proposed patch antenna for Wireless LAN applications. According to the radiation pattern, these type of antenna is the directional antenna.

The surface current distributions of proposed patch antenna is shown in Figure. 12. This figure shows the significant confluence of the current in the rectangular parts of the patch.

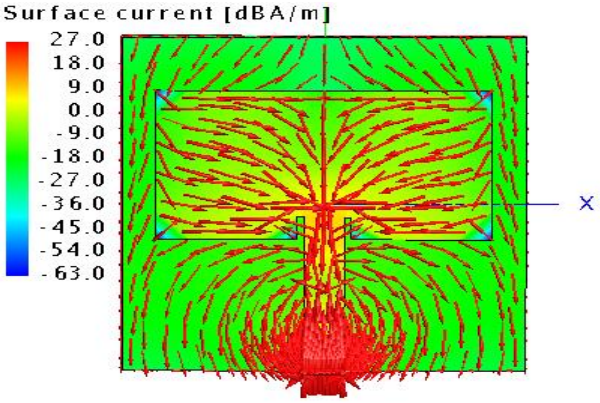


Figure 12. Current Distribution of Proposed Patch Antenna

5. Conclusion

In this paper, the single band microstrip patch antenna operating at 2.45GHz by using the microstrip inset feeding techniques has been designed .The results are accomplished by analysing the design parameters (width, length, substrate thickness and inset fed parameters) using on Feko software for the simulated patch structure. According to the simulation results, the proposed patch antenna provides a good impedance matching (|S11| < -10dB) for operating frequency in the wireless LAN band. In our case, we can observe from the S11 curve that the bandwidth is equal to 84 GHz. So, this frequency range is enough for Wireless LAN application. The fractional bandwidth of patch antenna is 3.3%.These type of antenna is covered for (2.412-2.484) GHz Wireless LAN application and implemented in real time.

5. Acknowledgment

The author would like to express sincere appreciation to the Rector of University of Technology (Yatanarpon Cyber City) for kind Permission to prepare for this paper. The author would also like to give special thanks to my supervisor and all teachers in Faculty of Electronic Engineering and all who willingly helped the author throughout the preparation of the paper. This paper is dedicated to the author's parents for continual and full support on all requirements and moral encouragement.

6. References

[1] Asokan V,Thilagam S,Vinoth kumar K Design and Analysis of Microstrip patch antenna for 2.4GHz ISM band and WLAN application, IEEE sponsered 2nd international conference on electronic and communication system978-1-4788-7225-8/15/2015IEEE

[2]D. Orban and G.J.K. Moernaut, The Basics of Patch Antennas, IEEE Antennas and Wireless Propagat. Lett. Vol. 1, 2002, pp 56-59.

[3]Gerard Djengomemgoto1, Reha Altunok1, Cem Karabacak1 'Dual-Band Gemini-Shaped Microstrip Patch Antenna for C-Band and X-Band Applications'

[4]Indrasen Singh, Dr. V.S. Tripathi Micro strip Patch Antenna and its Applications: a Survey ISSN:2229-6093 [5]John_Willey [Cropped_fixed] (Antenna Theory Analysis_and_ Design Constantine_A._Balanis)_(2nd_Ed)

[6]Kukunuri Suraj and M. Neelaveni Ammal Design and Development of Microstrip Patch Antenna at 2.4 GHz for Wireless Applications Vol 11 (23), June 2018

[7]Ms.Varsharani Mokal1, Prof S.R.Gagare2, Dr.R.P.Labade3Analysis of Micro strip patch Antenna Using Coaxial feed and Microstrip line feed for Wireless Application ISSN: 2278-8735.Volume 12, Issue 3, Ver. III (May - June 2017), PP 36-41

[8]Ruchi Kadwane, Vinaya Gohokar 'Design and Characteristics Investigation of Multiband Microstrip Patch Antenna for Wireless Application 'Volume 2, Issue 3, June 2014, PP 61-66 ISSN 2349-4395 (Print) & ISSN 2349-4409

[9]Seyed Ehsan Hosseini, Amir Reza Attari, and Aref Pourzadi, A Multiband PIFA with a Slot on the Ground Plane for Wireless Applications, International Journal of Information and Electronics Engineering, Vol. 3, No. 4, July 2013.

[10]Supratha C,Robison S,Design and analysis of microstrip patch antenna for WLAN application ,proceeding of 2018 IEEE conference978-1-5386-3702-9/18

[11]www.electronics-notes.com