



GLOBAL JOURNAL OF RESEARCHES IN ENGINEERING: F
ELECTRICAL AND ELECTRONICS ENGINEERING
Volume 14 Issue 3 Version 1.0 Year 2014
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 2249-4596 & Print ISSN: 0975-5861

Design of PID Controller for Direction Control of Robotic Vehicle

By C. S. Mala & S. Ramachandran

Jawaharlal Nehru Technological University, India

Abstract- A PID Controller has been designed and incorporated into the Robot based Agricultural System. This system comprises of a Robotic Vehicle which navigates through the field in the desired direction performing tilling or de-weeding as it moves. The PID control system has been designed for automatic steering and speed control of the Robotic Vehicle. The PID control system continuously checks and corrects the direction of travel to keep the vehicle on desired track. PID Control equations for Discrete Time Domain have been derived and validated by using MATLAB. Functional tests were carried out using ModelSim. The hardware results were finally validated using MATLAB again. The direction accuracy achieved is better than 0.01%.

Keywords: *PID controller, robot control, agricultural automation, speed control, steering control.*

GJRE-F Classification : *FOR Code: 090699*



Strictly as per the compliance and regulations of :



© 2014. C. S. Mala & S. Ramachandran. This is a research/review paper, distributed under the terms of the Creative Commons Attribution-Noncommercial 3.0 Unported License (<http://creativecommons.org/licenses/by-nc/3.0/>), permitting all non commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Design of PID Controller for Direction Control of Robotic Vehicle

C. S. Mala ^α & S. Ramachandran ^σ

Abstract- A PID Controller has been designed and incorporated into the Robot based Agricultural System. This system comprises of a Robotic Vehicle which navigates through the field in the desired direction performing tilling or de-weeding as it moves. The PID control system has been designed for automatic steering and speed control of the Robotic Vehicle. The PID control system continuously checks and corrects the direction of travel to keep the vehicle on desired track. PID Control equations for Discrete Time Domain have been derived and validated by using MATLAB. Functional tests were carried out using ModelSim. The hardware results were finally validated using MATLAB again. The direction accuracy achieved is better than 0.01%.

Keywords: PID controller, robot control, agricultural automation, speed control, steering control.

I. INTRODUCTION

The Robotic Vehicle is primarily designed to till the field to the specified length and breadth. The movement of the vehicle in any direction be it forward or reverse, should be in a straight line. Otherwise, the vehicle may move haphazardly, not tilling the ground in a straight line. If this operation is carried out manually, the throughput is painfully low. If the entire agricultural operation is carried out using Robotic vehicle, such as the proposed one, which is unmanned, then great care has to be taken to ensure that the movement of the vehicle is in the desired direction always. This calls for an automated feedback system, which keeps track of the wheel direction of the vehicle, compares it with the set direction, and takes corrective measure in case of any deviation from the set direction. There are many techniques reported to achieve the direction control or speed control of the wheels of automatic unmanned vehicles using PID controller.

A tele-operated, farmland information collecting robot [1] was developed using PID Controller and PWM principle [2]. An ideal PID controller is described by Wang Hongxing [3] for a farm land information collecting robot. For tuning of the PID controller parameters, critical proportioning methods are used [4]. A PID Controller developed using 8051 microcontroller [5-6] has been briefly explained with programming done in embedded C.

Suying Yang, Miaomiao Gao, Jianying Lin, and Zhuohan Li have worked on PID controller IP Core for temperature control in System on a Programmable Chip (SOPC). The authors have described the PID Control algorithm using HDL on FPGA [7]. They have designed PID controller IP core to make temperature tests to the control object with features of inertia and pure delay. Avalon bus is used for connecting processor to the peripherals.

Guoshing Huang and Shuocheng Lee have explained the PC based PID speed control in DC Motor [8] using Lab view. It uses Vis Sim software simulator to analyze its response. The signals have been acquired through NI DAQ USB-6008 card. The parameters are adjusted to control the motor speed by using the Lab view aided PID controller.

Zhen Wang and Lingshun Liu [9] have designed a control system using the microcontroller TMS320LF2407A for sampling and regulating the speed of the DC Motor and have used AT89C51 microcontroller for inputting and displaying the speed of the DC Motor.

Junyao Gao et al. [10] have designed control methods for unmanned vehicle. For steering control of the wheels, the authors have applied a complex mathematics using a multistep neural network. Fuzzy Genetic Algorithm (GA) PID control method is used for the turning control of wheels.

An Intelligent Control for a Crawling Unmanned Vehicle [11] have been designed by Dr. Kwang Hwa Lee. An adaptive feedback controller is designed by using a combination of PID and Adaptive Neural network Fuzzy Logic.

Xiaogang Ruan, Qiyuan Wang and Naigong Yu, have designed Dual-loop Adaptive Decoupling Control for Single Wheeled Robot based on Neural PID Controller [12]. They have used Newton-Euler equations in their design to achieve wheel control.

Migara H. et al. have designed a Robotic Arm with Hydraulic actuators operated by servo valves [13]. Each joint position is controlled by a PID based system. The system was modeled using MATLAB-SIMULINK tool box.

M. S. Tsoeu and M. Esmail, have carried out research on Unconstrained Model Predictive Control (MPC) and PID Evaluation for Motion Profile Tracking Applications [14]. The authors have tested optimally tuned Model Predictive Control (MPC) and PID

Author ^α: Jawaharlal Nehru Technological University, Hyderabad, India. e-mail: csmala63@yahoo.co.in

Author ^σ: SJB Institute of Technology, Bangalore, India. e-mail: ramachandr@gmail.com

controller for motion profile tracking and have compared them using Pareto optimality. They have found that the results of PID Controller are reasonable, whereas MPC gives unanticipated outcomes.

Xiaogang Ruan, Qiyuan Wang and Naigong Yu have designed Dual-loop Adaptive Decoupling Control for Single Wheeled Robot based on Neural PID Controller [12]. They have used Newton-Euler equations in their design to achieve wheel control.

Migara H. et al. have designed a Robotic Arm with Hydraulic actuators operated by servo valves [13]. Each joint position is controlled by a PID based system. The system was modeled using MATLAB-SIMULINK tool box.

M. S. Tsoeu and M. Esmail have carried out research on Unconstrained Model Predictive Control (MPC) and PID Evaluation for Motion Profile Tracking Applications [14]. The authors have tested optimally tuned Model Predictive Control (MPC) and PID controller for motion profile tracking and have compared them using Pareto optimality. They have found that the results of PID Controller are reasonable, whereas MPC gives unanticipated outcomes.

For tracking stabilization of Robot Motion, a new variable structure PID controller design is considered [15]. A certain PID sliding motion of a controller with PID sliding surface is substantiated for a robotic manipulator. Lyapunov full quadratic form and upper and lower matrix norm inequalities are formulated for sliding and global stability. This control algorithm is applied to robot arm through simulations and the authors have found that the control function is satisfactory.

An approach for a remote PID controller design have been presented by Hui Zhang et al [16]. They have first developed the design method for Static Output Feedback (SOF) Controller, and using this PID controller has been designed. The SOF Controller has been designed using Bernoulli random binary distribution and matrix inequalities.

Chaoraingern J et al. [17] have designed PID Controller using Coefficient Diagram Method (CDM) for track following control of hard disk drive. It is based on proper selection of the characteristic transfer function of the closed loop control system. They have shown through MATLAB simulation results that CDM design is quite stable.

Marino R et al. [18] have designed a nested PID control for lane keeping autonomous vehicles which are vision based. The steering angle is the control input and it is designed taking into account yaw rate and the central offset measured by a gyroscope and vision system respectively. The simulations were carried out on Standard Big Sedan CarSim Vehicle model.

Saidonar M. S. et al. [19] have considered only differential steering control with proportional control methods for a mobile robot. The current velocities of the

right and left wheel are used to determine the current position of the mobile robot. The velocity and angular velocity of the robot helps to determine the number of encoder pulses that have to be fed to the DC motors connected to the right and left wheel. The proportional controller is used to produce the same speed for both right and left wheel to ensure the movement of the robot in a straight line.

Veerachari Mummadi [20] has proposed a digital PID controller for H bridge soft switching boost Converter. The author has formulated the mathematical models of the H bridge using the system identification tool and has used in the PID design. Designing the PID Controller has been done using Pole Placement Technique (PPT) together with Sensitivity Function Shaping (SFS).

PID Controller has been designed in this work for automatic steering and speed control of the Robotic Vehicle. PID Control equations for Discrete Time Domain are presented and validated by using MATLAB in this section.

The paper is organized as follows. Section 2 presents the PID Controller and Derivation of Discrete Domain PID Controller Equations and MATLAB realization of the same. Section 3 presents the Block Diagram of the Proposed Robotic Vehicle with PID Controller and the Algorithmic state Machine Charts. Section 4 presents the architecture of the PID Controller. Section 5 presents the Simulation results and the Validation of Verilog RTL Design Using MATLAB and Section 6 presents the Conclusion.

II. PID CONTROL SYSTEM

The PID controller produces an output signal comprising three terms-one proportional to error signal, another proportional to integral of error signal and the third one proportional to the derivative of error signal. Fig.1 presents the PID controller in its expanded form.

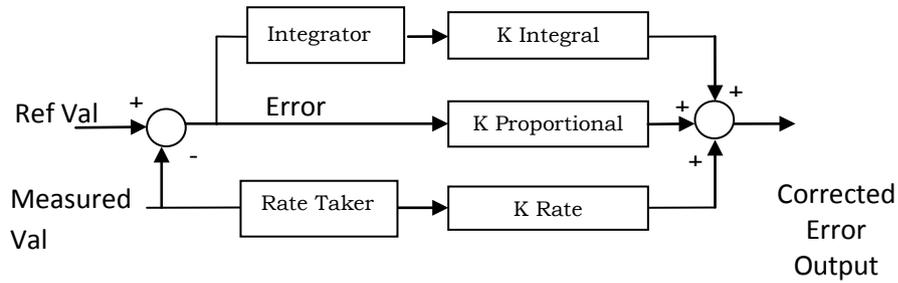


Figure 1 : Block Diagram of PID Controller

PID control may be achieved in continuous time domain by computing an error term such as:

$$u(t) = K_p e(t) + (K_p/T_i) \int e(t) dt + K_p T_d (d/dt) e(t) \quad (1)$$

or

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d (d/dt) e(t) \quad (2)$$

where,

$u(t)$ = Total Corrected Error,

K_p = Proportional Gain,

T_i = Integral Time,

T_d = Derivative time,

K_i = Integral Gain = K_p/T_i and

K_d = Proportional Gain = $K_p T_d$.

The proportional control stabilizes the gain but produces a steady state error. The integral control reduces or eliminates the steady state error. The derivative control reduces the rate of change of error. For example, in steering control, if the reference or set direction is 180 degrees and if the robot is moving off from the set direction, say, to 182 degrees, then the controller dynamically corrects the wheel speeds in such a manner that the robot moves closer to 180 degrees. The frequency at which the measured direction is input is called the sampling rate. Higher the sampling rate, greater will be the precision in correcting the error. The PID control is also achieved at the same rate.

$$dirn(n) = set_dir + (K2/K0) * integral(n) - u(n)/K0 + [(u(n-1)/K0)/(K2/(K1*SR) + K0/K1)] \quad (6)$$

Equation (6) gives the corrected direction.

In order to get the value of the corrected direction closer to the reference set direction, various values of K_0 , K_1 and K_2 were experimented with in the above equation using MATLAB. Optimum values arrived at for these gain constants are presented in the next section.

b) MATLAB Realization of PID Controller

The PID Controller equations derived in the previous section has been coded in MATLAB and

a) Derivation of Discrete Domain PID Controller Equations

Fundamental PID Controller equations were presented in the previous section. In this section, discrete domain equations are so derived that it is amenable for Verilog RTL coding. To start with, the following assumptions are made:

- Number of samples = n ,
- Sample rate (say, 25 Hz depending on the sensor employed) = SR ,
- Desired direction of the Robot = set_dir , and
- Actual measured direction from a digital compass = $dir(n)$.

The difference between the set and the measured directions naturally is the error expressed as Eq. 3.

$$e(n) = set_dir - dir(n) \quad (3)$$

Overall Error computed by the Proportional, Integral and Differential terms of the PID Controller may be derived from Eq. (2) and is presented in Eq. (4).

$$u(n) = K_0 * e(n) + K_1 * e(n-1) + K_2 * integral(n) \quad (4)$$

where K_0 , K_1 and K_2 are constant gain factors for the Proportional, Differential and Integral terms respectively and

$$integral(n) = integral(n-1) + (set_dir - dir(n))/SR \quad (5)$$

As an approximation, we get the corrected direction:

results of the corrected direction and the set direction for different gain constant values have been presented in Fig. 2 for the Optimum Gain constants. For this set of gain constants, the corrected direction is closest to the set direction with the least overall error. The values for average direction, average error and overall error for different sets of K_0 , K_1 , and K_2 have also been computed and presented in Table 1.

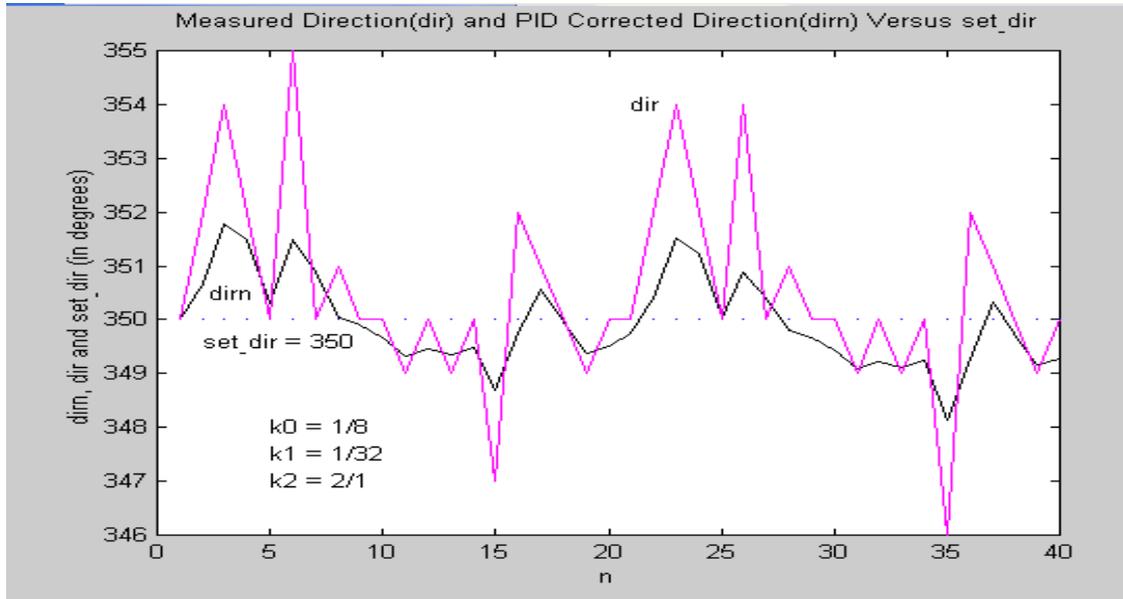


Figure 2 : Matlab Results: Comparison of PID Corrected Direction Vs Set Direction for Optimum Gain Constants – Least Error

Table 1 : Comparison of Average Error and Overall error for different values of Gain Constants

Set dir = 350 degrees, No. of samples = 40, Sample rate = 25 Hz, Length of Field = 300 ft

Sl. No.	K0	K1	K2	Avg. Direction	Avg. Error	Overall error in feet
1	1/8	1/32	2/1	350.0024	-0.0024	-0.0128
2	1/8	0	2/1	350.1875	-0.1875	-0.9817
3	1/8	1/32	0	350.5859	-0.5859	-3.0680
4	1/4	1/32	2/1	350.3446	-0.3446	-1.8042
5	1/16	1/32	2/1	348.0040	1.9960	10.4512
6	1/8	1/16	2/1	348.9060	1.0940	5.7284
7	1/8	1/32	1/4	350.4825	-0.4825	-2.5265
8	1/16	1/32	1/4	350.0093	-0.0093	-0.0488
9	1/8	1/32	1/16	350.5591	-0.5591	-2.9276

III. BLOCK DIAGRAM OF THE PROPOSED ROBOTIC VEHICLE WITH PID CONTROLLER

The block diagram of PID controller designed for the Robotic Vehicle is presented in Fig. 3. The system comprises the following functional blocks: Digital Compass, PID Controller, Robotic Controller and DC Motor Controller.

A digital compass is used for measuring the earth's magnetic field. It has a built in UART interface and can be readily used with any FPGA, ASIC or Microcontroller based design. The measuring range of the compass is 0 to 360 degrees and it has a resolution of 0.1 degree and an accuracy of 1 degree. The frequency response is 25 Hz. The initial reading is set as is convenient for the field to be tilled. It is the reference direction in which the vehicle is continuously expected to navigate.

The Digital compass output is read every 40 msec. The output is compared with the reference direction. The difference (error) is applied to PID controller for error computation. The error output of the PID controller is input to Pulse Width Generator to produce pulses

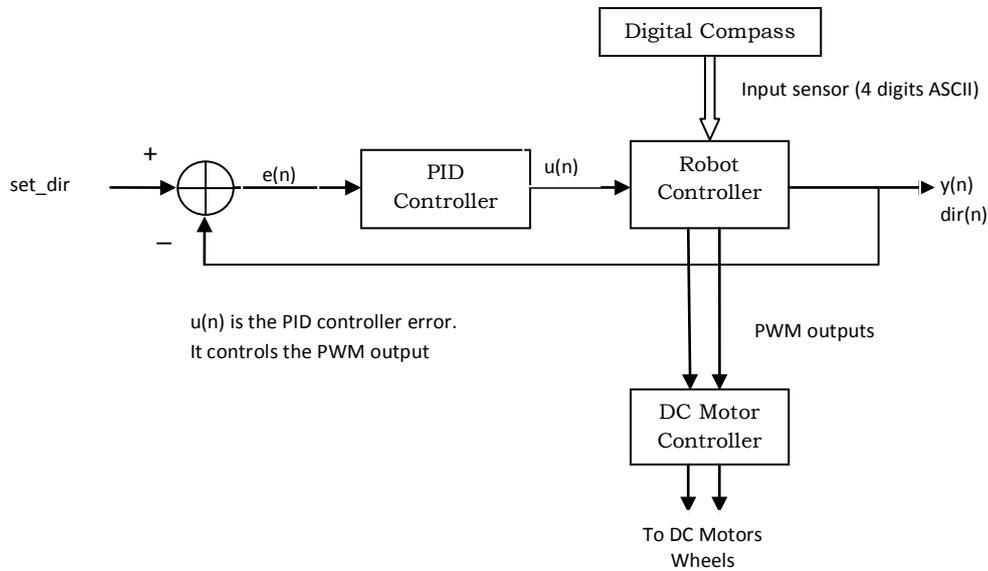


Figure 3 : Block Diagram of the Robotic Vehicle with PID Control

proportional to the error computed and is applied to the DC Motor Drivers to correct the error and align the wheels close to the reference direction. The communication with the Digital compass is RS232 based.

The Robotic Controller is responsible for the movement of the vehicle in the set direction. It generates the Pulse Width for the DC Motors that drives the Wheels of the vehicle according to the correction required in the direction of motion. The pulse width generated is fed to the DC motor driver.

a) Development of Algorithms

The Robotic Vehicle Controller may be realized by developing Algorithmic State Machine Charts for various blocks. The Digital compass senses the direction of motion and passes on that information serially to the Digital Compass ASCII module of the Robotic Controller. The sensor will begin to send out the measured direction only on receiving the command 31 H on its RX line. The value 31 H is transmitted serially at 9600 baud rate by the Digital Compass ASCII module.

The digital compass starts transmitting the measured value serially on its TX line. The reception baud rate is also 9600. One direction value is received as 7 bytes. The First and second Bytes are 0D H and 0A H respectively. Then the three bytes of data representing the measured direction in ASCII format is received. This is followed by 2E (being Byte 6) indicating decimal point. Byte 7 is the digit after the decimal point. As an example, for 350.0 degrees, the third byte to seventh byte is 33 H, 35 H, and 30 H respectively.

Since the ASM charts for the entire design are numerous, the ASM chart for the PID Controller only is

presented as an example in Fig. 4. The main function of this module is to read the current direction of the robotic vehicle, check for deviation from the set direction at 25 Hz sample rate, calculate the error and correct the direction of the vehicle. The vehicle is kept at the desired direction by controlling the pulse width of the right wheels, whereas the pulse width of the left wheel is kept constant.

The PID Controller starts functioning only after receiving a valid "enable_pidc" signal from the main controller. Once the module is enabled, it waits for the direction data output from the Digital Compass module. It acquires the direction data only after getting a valid "data_valid" signal. The first direction data received is treated as the "set_dir" (meaning set direction) for the vehicle. The successive "dir" data received is then compared with the "set_dir" and the corresponding error "e(n)" is computed. Thereafter, the PID corrected error "u(n)" is calculated which is the sum of proportional, integral and the differential errors to compute the corresponding corrected direction "dirn".

Once the corrected direction is calculated, the next task of the PID Controller is to derive the pulse widths for the left and the right wheels. For this purpose, a free running counter "pwidthm" is used, whose terminal count corresponds to 100 % duty cycle. For the left wheels, the counter value is compared with the "set_dir" value, which is equivalent to about 67% duty cycle pulse width and a high pulse output is issued at output pin "pwidthm_l" for the condition: (start_pwidthm = 1) & (pwidthm <= set_dir). Otherwise, Logic "0" is output. Similarly, for the right wheels, a high pulse output is issued at output pin "pwidthm_r" for the condition: (start_pwidthm = 1) & (pwidthm <= dirn).

Otherwise, Logic "0" is output. The output pins "pwidth_l" and "pwidth_r" are connected to enable pins of DC Motor Driver IC L298, not presented here.

Reset Conditions:

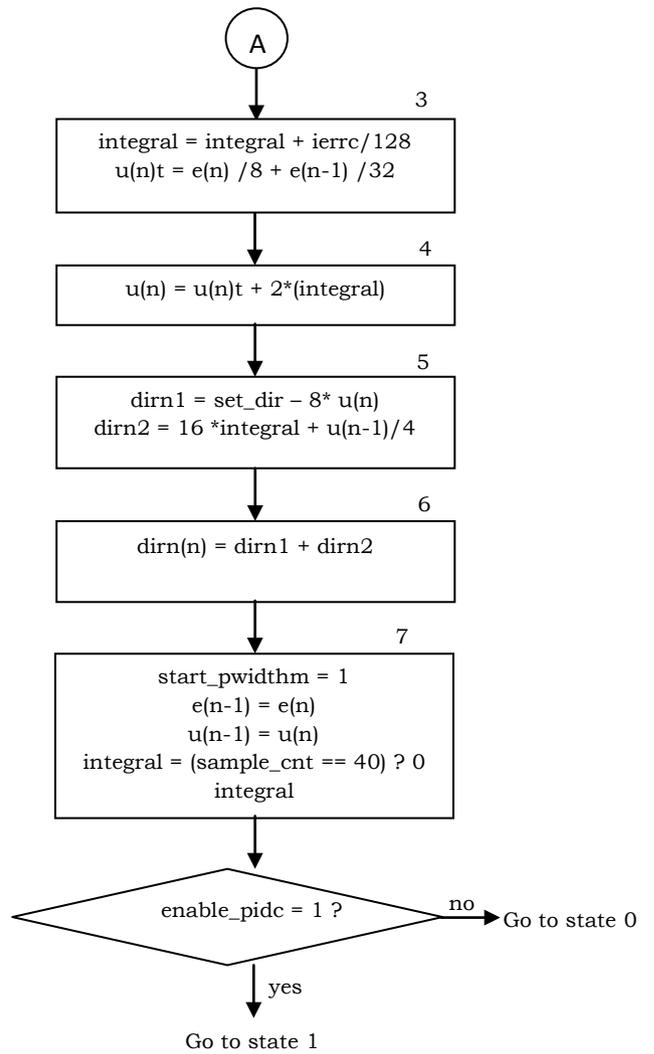
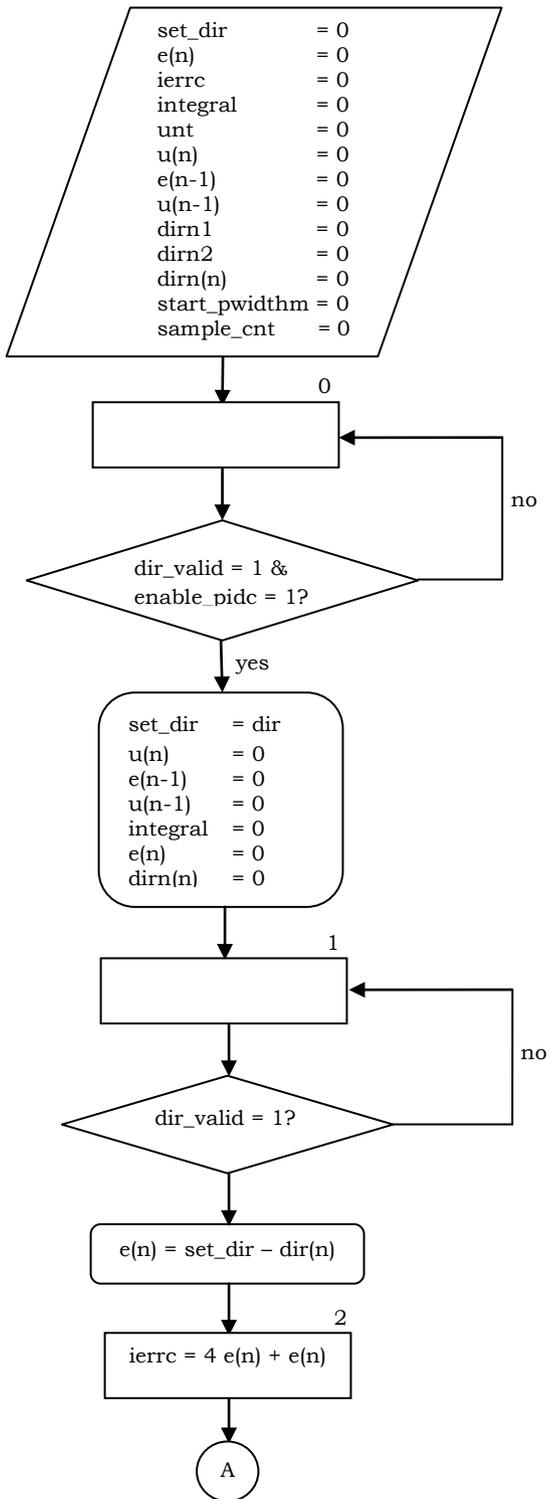


Figure 4 : ASM Chart for PID Controller Module

IV. ARCHITECTURE OF PID CONTROLLER

The Architecture of PID Controller comprises the following modules: Digital Compass ASCII, Digital Compass, BCD to Binary conversion and PID controller that have been briefly explained earlier. They are presented in Fig. 5 to Fig. 8.

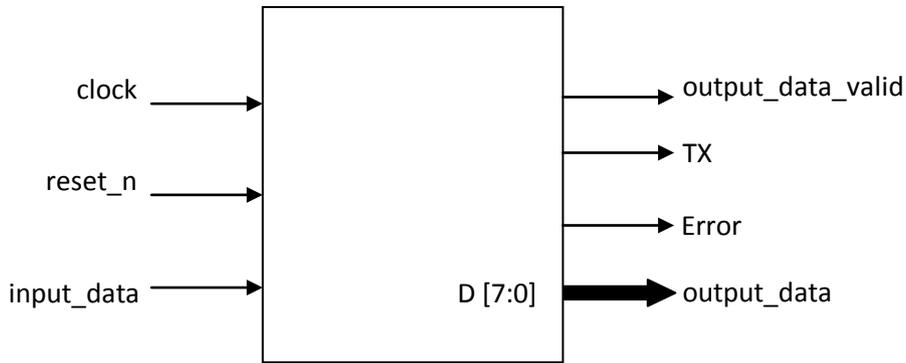


Figure 5 : Digital Compass ASCII RTL Module

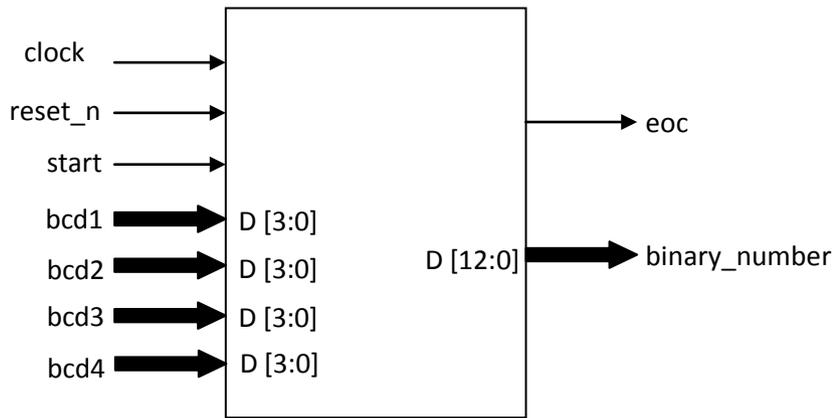


Figure 6 : BCD to Binary Converter Module

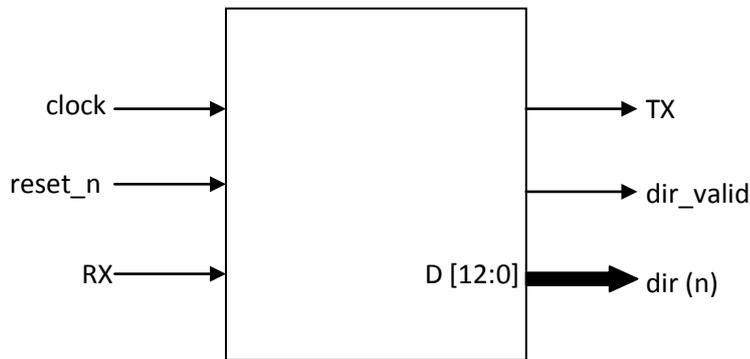


Figure 7 : Digital Compass Module

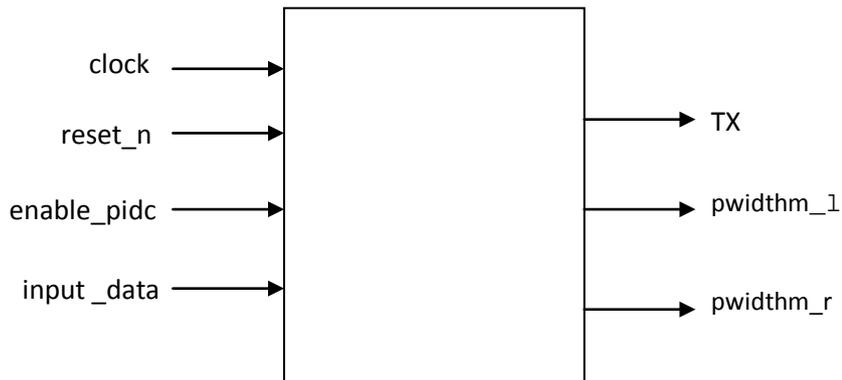


Figure 8 : PID Controller Module

V. SIMULATION RESULTS

The PID Controller RTL Codes were designed and simulated using Modelsim and tested

for several samples of direction data. A test bench was written for the purpose of playing the role of Digital Compass Sensor

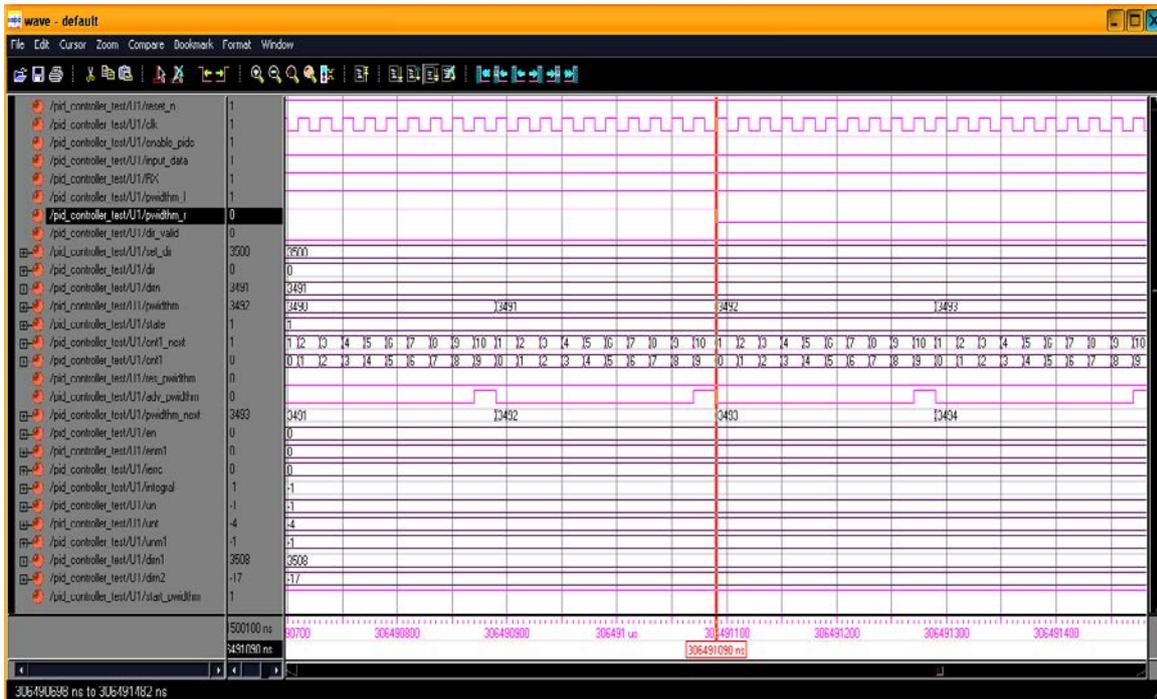


Figure 9 : PID Controller Waveform showing Robot Direction Correction

for transmitting the Robot direction and to carry out functional tests. Since the waveforms are too numerous, only the most essential one is presented here. The waveform for the generation of pulse width signals “pwidthm_l” and “pwidthm_r” for left and right wheels are presented in Fig. 9. The signals are generated based on set direction and corrected values of the direction. The generation of the two pulses always starts simultaneously.

The direction is controlled by only the duty cycle of the two pulses. This is achieved by keeping the duty cycle of left wheel constant (“pwidthm_l” low for “pwidthm = 3500”) with respect to set direction (set_dir = 3500), whereas the duty cycle of right wheels is determined based on the corrected direction value “dirn”. When the set direction and the corrected direction values are equal, the pulse width for all the four wheels is same. In this example, the measured direction “dir” is 350.0. It may be noted that when the current PID computed Robot direction “dirn” matches (3491 meaning 349.1 degrees) the current running counter value “pwidthm”, then the right wheels pulse width output “pwidthm_r” is forced low, thus correcting the Robot direction to 349.1 degrees. Although the actual error “e(n)” is zero in this case, the PID computed error “u(n)” is -1. This error is made up in the next sample.

VI. VALIDATION OF VERILOG RTL DESIGN USING MATLAB

Various directions such as the set direction (set_dir), measured direction (dir), PID corrected direction (dirn) and the corresponding pulse width (pwidthm_r) that controls the right wheels of the Robot are plotted using MATLAB for 40 samples of Digital Compass and is presented in Fig. 10. The waveform reveals that the direction correction effected by the RTL code is more effective than even MATLAB in spite of using less precision in Verilog coding, thus validating the Robotic Controller Design.

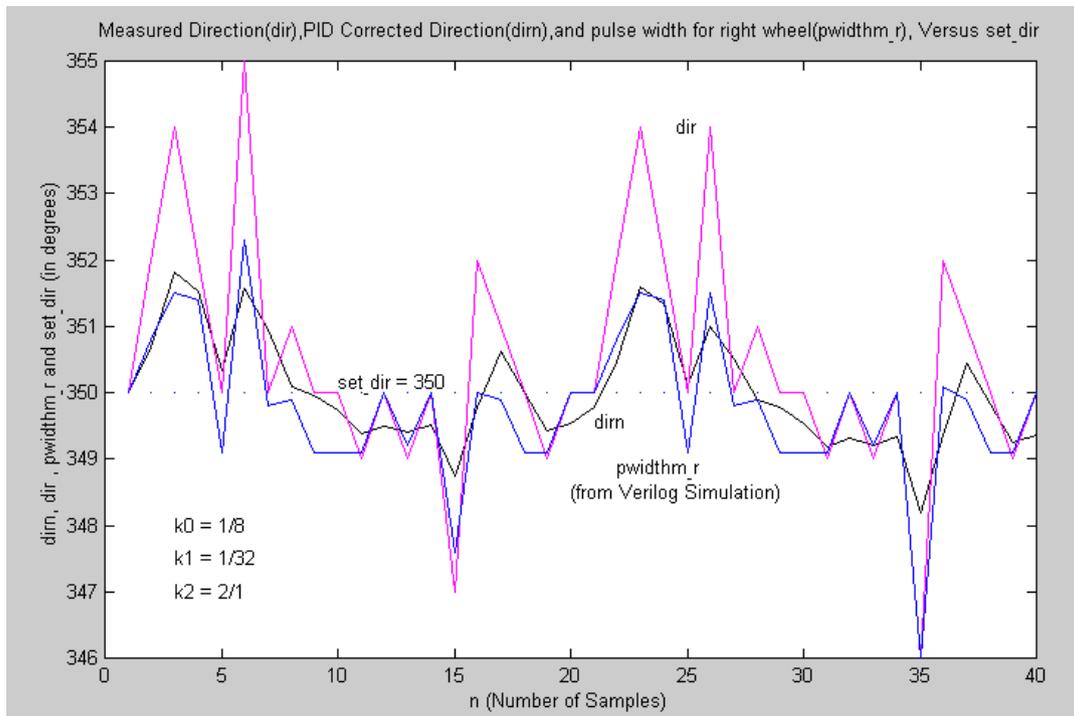


Figure 10 : Robot Controller Verilog RTL Design Validation Using Matlab

VII. CONCLUSION

A PID Control System for Robot based Agricultural System has been designed. As an example, the set direction is taken to be 350 degrees and the Digital Compass continuously gives out the measured direction at the sample rate of 25 Hz. The number of samples taken is 40 for testing. The PID control system continuously checks and corrects the direction of travel to keep the vehicle on the desired track. PID Control equations for Discrete Time Domain have been derived and coded in Verilog RTL. Functional tests were carried out using ModelSim. The hardware results were finally validated using MATLAB. The direction accuracy achieved is better than 0.01%.

REFERENCES RÉFÉRENCES REFERENCIAS

1. Yuzhu Cheng, Yong Chen and Hongxing Wang, Design of PID controller based on Information Collecting Robot in Agricultural Fields, 978-1-4244-9763-8/11, IEEE 2011, pp. 345-348.
2. Blas M. Vinagre, Concepción A. and Monje, Fractional PID Controllers for Industry Application-a Brief Introduction, Journal of Vibration and Control, 2007, Vol. 7, Issue 13, pp. 1419-1429.
3. Zhou Yijun, The Improved critical proportioning method used for self-tuning of PID parameters, Process Automation and Instrumentation, 2002, Vol. 1, Issue 23, pp. 14-16.
4. Wang Hongxing, Research on the Control System of Farmland Information Collecting Robot, NanJing Forestry University, 2008.
5. Hu Na, Development of an information collection robot body in agricultural fields, Nan Jing Forestry University, 2009.
6. Liu Xiaokai, Design and Research of Reactive Control System of DC Motor Based on MCU of C8051F, He Bei University of Technology, 2005, Vol. 12, pp. 72-76.
7. Suying Yang, Miaomiao Gao, Jianying Lin, and Zhuohan Li, The IP Core Design of PID Controller based on SOPC, International Conference on Intelligent Control, IEEE 2010, pp. 363-366.
8. Guoshing Huang and Shuocheng Lee, PC based PID speed control in DC Motor, IEEE, ICALIP 2008, pp. 400-407.
9. Zhen Wang and Lingshun Liu, The design of a speed regulator of DC Motor based on TMS320LF2407A and AT89C51, IEEE 2010.
10. Junyao Gao, Jianguo Zhu, Boyu Wei and Shilin Wang, Unmanned Vehicles Intelligent Control Methods Research, The Ninth International Conference on Electronic Measurement & Instruments ICEMI'2009 Vol3 pp. 736-741.
11. Dr. Kwang Hwa Lee, An Intelligent Control for a Crawling Unmanned Vehicle , MTS 0-933957-28-9, IEEE 2009, pp. 490-499.
12. Xiaogang Ruan, Qiyuan Wang and Naigong, Dual-loop Adaptive Decoupling Control for Single

- Wheeled Robot based on Neural PID Controller, 11th Int. Conf. Control, Automation, Robotics and Vision, December 2010 pp. 2349- 2351.
13. Migara H. Liyanage, Nicholas Krouglicof and Raymond Gosine Design and Control of a High Performance Scara Type Robotic Arm with Rotary Hydraulic Actuators, IEEE 2009, pp. 827-832.
 14. M. S. Tsoeu and M. Esmail, Unconstrained Model Predictive Control (MPC) and PID Evaluation for Motion Profile Tracking Application, IEEE Africon 13 - 15 September 2011, The Falls Resort and Conference Centre, Livingstone, Zambia.
 15. E. M. Jafarov, M. N. A. Parlaksi, and Y. Istefanopulos, A New Variable Structure Pld-Controller Design for Robot Manipulators, IEEE transactions on Control Systems Technology, Vol.13, Issue 1, Jan 2005, pp. 122-130.
 16. Hui Zhang, Yang Shi and Aryan Saadat Mehr, Robust Stastic Output Feedback Control and Remote PID design for networked motor Systems, IEE transactions on Industrisl Electronics, Vol. 58, Issue 12, Dec 2011, pp. 5396-5405.
 17. Chaoraingern J, Ladkrabang, Vaidee W, Trisuwannawat T and Numsomran A, 11th International Conference on Control, Automation and Systems, 2011, pp. 72 – 77.
 18. Marino R., Scalzi S., Orlando G, and Netto M, A nested PID steering control for lane keeping in vision autonomous vehicles, American Control Conference 2009, pp. 2885 – 2890.
 19. Saidonar M. S., Desa H , and Rudzuan M. N, A differential steering control with proportional controller for an autonomous mobile robot, 7th International Colloquium on Signal Processing and its Applications, 2011, pp. 90-94.
 20. Veerachary Mummadi, Design of Robust Digital PID Controller for H-Bridge Soft-Switching Boost Converter, IEEE Transactions on Industrial Electronics, Vol. 58, Issue 7, 2011, pp. 2883-2889.