

Two Wheels Balancing Robot with Line Following Capability

Nor Maniha Abdul Ghani, Faradila Naim, Tan Piow Yon

Abstract—This project focuses on the development of a line follower algorithm for a Two Wheels Balancing Robot. In this project, ATMEGA32 is chosen as the brain board controller to react towards the data received from Balance Processor Chip on the balance board to monitor the changes of the environment through two infra-red distance sensor to solve the inclination angle problem. Hence, the system will immediately restore to the set point (balance position) through the implementation of internal PID algorithms at the balance board. Application of infra-red light sensors with the PID control is vital, in order to develop a smooth line follower robot. As a result of combination between line follower program and internal self balancing algorithms, we are able to develop a dynamically stabilized balancing robot with line follower function.

Keywords—infra-red sensor, PID algorithms, line follower Balancing robot

I. INTRODUCTION

OVER the past decades, the research on two wheeled inverted pendulum mobile robot or commonly known as balancing robot have gain momentum in a number of robotic centers around the world due to natural unstable dynamics of the system. [1] Since a Two wheeled balancing robot need a good controller to maintain itself in upright position without the needs from external forces. Thus, providing a good platform for researcher to explore the efficiency of various controllers in control system based on the inverted pendulum model. Nowadays, various controllers are implemented on two wheeled balancing robot for example Linear Quadratic Regulator, Pole-Placement Controller, Fuzzy Logic Controller, and Proportional Integrated Derivative Controller. [3]. A two wheeled balancing robot is categorized by the ability to balance on its two wheels and spin on the spot. As a result from this additional maneuverability, it allows easy navigation on the various terrains, turn sharp corner, and traverse small step or curbs and ability to carry loads. Two wheeled robots also have a small footprint than three or four wheeled robots thus enable it to travel around corridors and tight corners more easily. [4] These capabilities have the potential to solve many challenges in industry and society. As the two wheeled balancing robot has been investigated and developed to become human transport machine. The Segway, Pegasus, and iBot models are the example of the design of two wheeled balancing robot as a human transport machine. In addition, a motorised wheelchair utilising this technology would give the operator greater maneuverability and thus access to places most able-bodied people take for granted. Small carts built utilising this technology allows humans

to travel short distances in a small area or factories as opposed to using cars or buggies which is more polluting. [2] In this project, the two wheels balancing robot will not only solve the balancing problem but also automatically move around and avoid basic obstacles through the implementation of Dual ground Sensor and looking forward infrared sensor respectively. The Dual ground sensor is located at the bottom of the Balbot which is the infrared IR distance sensor, where ground sensors are used to measure the tilting angle. Powerful ATMEGA32 processor is used to be the brain of the robot. BEMP motor velocity sensor is used to obtain the speed of the platform. The entire controller algorithm will be computed in C programming and stored inside the microcontroller. Without an active control system, the robot would just fall over. Thus, the controller plays an important role in this project. Lastly, three pairs of infra-red sensor are used to guide the line follower task during balance state.

II. LITERATURE REVIEW

Conducting literature review prior to begin a research project is vital in understanding two wheels balancing robot control technique, as this will supply the researcher with much needed additional information on the methodologies and technologies available and used by other research counterparts around the world. This chapter provides a condensed summary of literature reviews on key topics related to balancing a two-wheeled robot. [2] Comparisons between the present project and the related topics of existing information will also be discussed.

A. Previous Project Work

A research professor at Carnegie Mellon University, Ralph Hollis who has developed a totally unique balancing robot that balances on top of a bowling ball. He calls his robot design "BallBot" [6, p. 72]. Mr. Hollis and his research associates believe that robots in the future will play a vital role in the daily lives of humans. He believes that in order for robots to be productive in our daily lives, some key problems need to be solved first. One the important problem he states in his article about mobile self balancing robots is the overall structure of the robot itself. (R. Hollis 2009) A researcher, Steve Hassenplug has successfully constructed a balancing robot called Legway using the LEGO Mindstorms robotics kit. Two Electro-Optical Proximity Detector sensors from HiTechnic Sensors to provide the tilt angle information and detect lines. The controller is programmed in high level programming language specifically created for LEGO Mindstorms which was written in brickOS (LegOS) and uses EOPDs to maintain a constant distance from the ground. As the distance decreases, Legway moves forward. As the distance increases,

Legway moves backward. Every 50 ms, Legway attempts to recalculate the balance point by measuring the current distance and motor speed. To move forward for line following, Legway actually sets the motors to run backward, causing a tilt, which it automatically corrects by moving forward. When one sensor is over the line, it stops that motor, and Legway balances using only the other motor, causing it to turn. To spin in place, both motors are shifted "off center" in opposite directions, the same amount, but they still correct for tilting. Legway uses its two optical proximity detectors to balance the two wheel LEGO robot. (Hassenplug S. 2003)

nBot is a two-wheeled balancing robot built by David P. Anderson. This robot uses commercially available of the shelf inertial sensors and motor encoders to balance the system. Such inertial sensors that are used on nBot are an accelerometer and a gyroscope. The basic idea for a two-wheeled dynamically balancing robot is pretty simple which is driving the wheels in the direction that the upper part of the robot is falling. If the wheels can be driven in such a way as to stay under the robot's center of gravity, the robot remains balanced. In practice this requires two feedback sensors: a tilt or angle sensor to measure the tilt of the robot with respect to gravity, and wheel encoders to measure the position of the base of the robot. Four terms are sufficient to define the motion and position of this "inverted pendulum" and thereby balance the robot. (Anderson D.P. 2003)As invented by Dean Kamen, the Segway PT's dynamics are identical to the inverted pendulum. For added mobility, the Segway is also based on the two wheel platform design. The advanced control algorithms behind the Segway transporter are a company trade secret. The basics of a Segway are computers that process the control algorithms, two tilt sensors, five gyroscopes, and two electric motors. Only three of the five gyroscopes are used to balance the Segway. The remaining two gyroscopes are used as backup. These critical components that make up a Segway are important to keep the vehicle in perfect balance. Current models of the Segway personal transporter can achieve top speeds of 12.5 mph. The Segway is able to navigate thru rough terrain, while successfully carrying a human onto of the platform. The Segway is typically found in urban settings; used for guided tours and city government officials. (Dean Kamen 2001)A researcher, Priyank Patil has successfully design and constructed a Line Follower Robot which will sense a line and manoeuvring the robot to stay on course, while constantly correcting wrong moves using feedback mechanism forms a simple yet effective closed loop system. This robot is running on Atmel's AVR@ microcontrollers which have a RISC core running single cycle instructions and a well-defined I/O structure that limits the need for external components. Internal oscillators, timers, UART, SPI, pull-up resistors, pulse width modulation, ADC, analog comparator and watch-dog timers are some of the features you will find in AVR devices. AVR instructions are tuned to decrease the size of the program whether the code is written in C or Assembly. With on-chip in-system programmable Flash and EEPROM, the AVR is a perfect choice in order to optimize cost and get product to the market quickly. [8]

B. PID Control Systems

The implementation and complexity difficulty associated with the nonlinear method causes most control researchers to utilize the linear controller approach. The method that will be used to control the self-balancing two wheeled robot will be a linear controller. It will be applied through a Proportional, Integral, and Derivative also refer to as the PID. The PID has proven to be popular among the control engineering community. As stated by the author of article Vance J. Van Doren, "For more than 60 years after the introduction of Proportional-Integral-Derivative controllers, remain the workhorse of industrial process control" [9].

III.METHODOLOGY

The main purpose of this project is to upgrade a balancing robot to become a line follower Balbot. It involves the understanding of balancing system in the Balance Board which using PID controller as linear controller with Analog GP2D120 IR distance sensors as feedback input and the concept of line following system with the implementation of IR sensors as shown in Fig. 1 about the General System Block Diagram. The line following mode can be establish based on the algorithm programmed in the Brain Board once the robot achieved balance state or condition as described in Fig. 2 about the Robot's Internal System Flow. This project is dividing into two parts namely hardware and software implementation. Hardware part involves more in interfacing between Brain Board and Balance Board with Analog GP2D120 IR distance sensors and Infra Red Sensors and also brief introduction on the robot's components. While in the software part include programming software using Brain Board Code Editor and procedure in loading programs into the microcontroller on the Brain Board. Fig. 3 illustrated the front view of the robot.

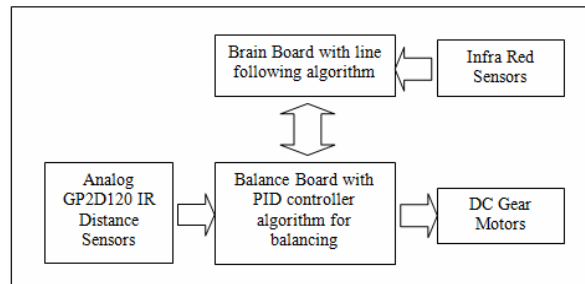


Fig. 1 Robot's General System Block Diagram

A. Balancing System

The Balbot balances by using the infrared range sensors aimed at the ground to calculate the absolute angle and the angular velocity of the robot. The primary function of the Balance Processor Chip (BPC) is to perform real time balancing control by using data from the GP2D120 IR Distance Sensors to drive the two motors in a manner that will keep the center of gravity above the wheels at all times. Thus, frees the master processor in the Brain Board to perform other tasks.

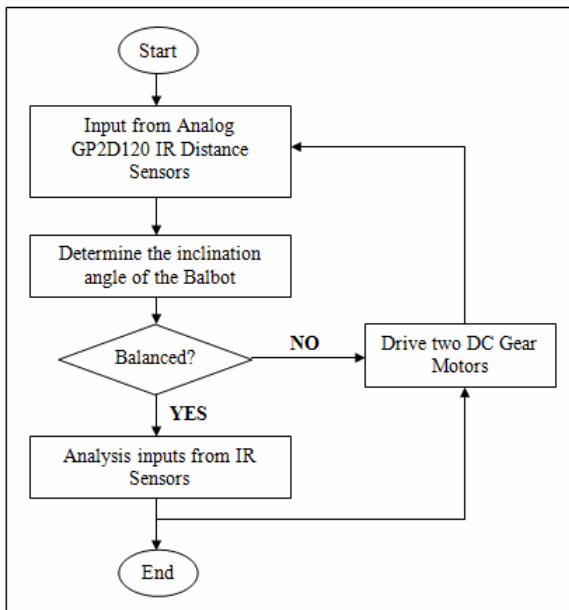


Fig. 2 Robot's Internal System Flow

The PID controller in balancing system is a linear controller that controls overall performance of robot. DC Gear motors are actuators for movement. The Balance Board executes the standard Digital Cascaded Control Loop (built into Balance Processor Chip) algorithm to balance the robot. The control loop runs at a frequency of approximately 40.2Hz. The actual velocity of robot will be detected by BPC using Back-Electromotive Force (BEMP). Then, BPC will calculate the error between the desired velocities with the actual velocity by considering also the tilt angle as the feedback from GP2D120 IR Distance Sensors, thus the robot will moving toward a falling direction (left side) at a specific angular velocity calculated by the system corresponding with the tilt angle in order to maintained the Balbot in balance condition as shown in Fig. 3. [10]

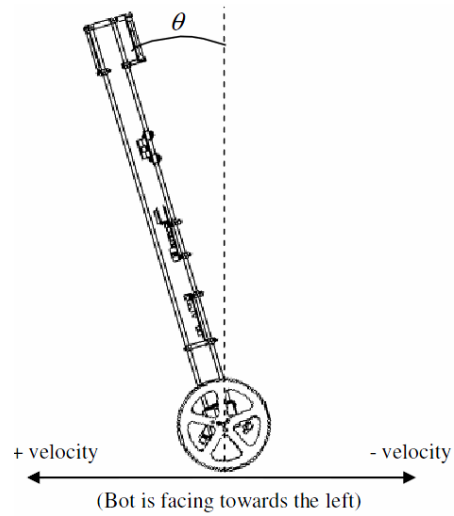


Fig. 3 The movement of Balbot due to the tilt angle with specific velocity

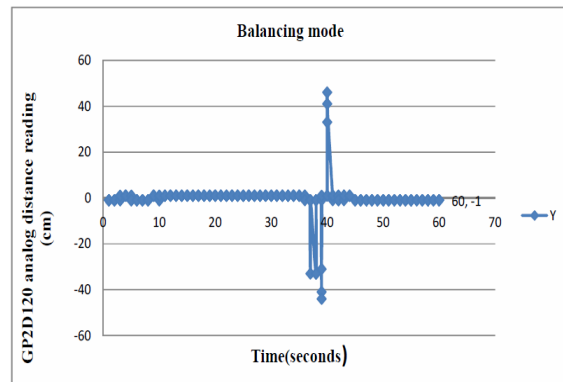


Fig. 4 GP2D120 Distance Reading for Balancing Mode

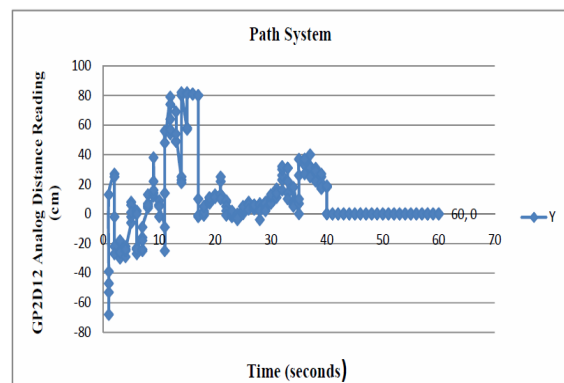
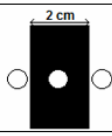
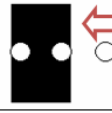

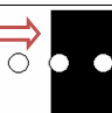
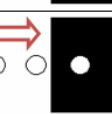


Fig. 5: GP2D120 Distance Reading for Path Navigation

B. Line Following System

Line following task is performed through the implementation of specified algorithm with the data from the Infra-Red sensors which are mounted at the bottom of robot. The data or status receive from the sensor will be processed by the programmed microcontroller to perform the continuous movement according to the line. The direction and speed of rotation is control by the microcontroller through PWM manipulation. The robot will conduct line following function when a balancing state is achieved. If the robot is in unstable condition, balancing task will be performed first, then follow by the line follower. Table 1 show that the designed line follower control algorithm for robot. There have five states for line following control in robot which are five possible condition might be occurs during line follower system execution in robot. All of the five states are expressed in five desired output direction in Balbot which are forward, shift to left, fast shift to left, shift to right, and fast shift to right. The inputs in this line follower are IR sensors which mounted at the bottom of robot. The logic "1" indicates no line detect while logic "0" indicates line detected. The algorithm can be adapted for situation in which the robot is continuously moving forwards. The line follower algorithm also can be expressed in flowchart for the decision of robot as shown in Fig 6. It is showed that the robot in this line follower algorithm will check on the state of the sensors first and based on the information from the three IR sensors, the desired motion of the robot will be determined. The programming that loaded in the ATMEGA32 is to defined function of a line follower control algorithm and decision of robot movement. First of all, the input ports for the sensors are been defined and initialized in Brain Board Code Editor Software.

TABLE I
LINE FOLLOWER CONTROL ALGORITHM

Status	Sensor Input			Output
	L	M	R	
	0	1	0	FORWARD
	1	1	0	SHIFT TO LEFT
	1	0	0	FAST SHIFT TO LEFT
	0	1	1	SHIFT TO RIGHT
	0	0	1	FAST SHIFT TO RIGHT

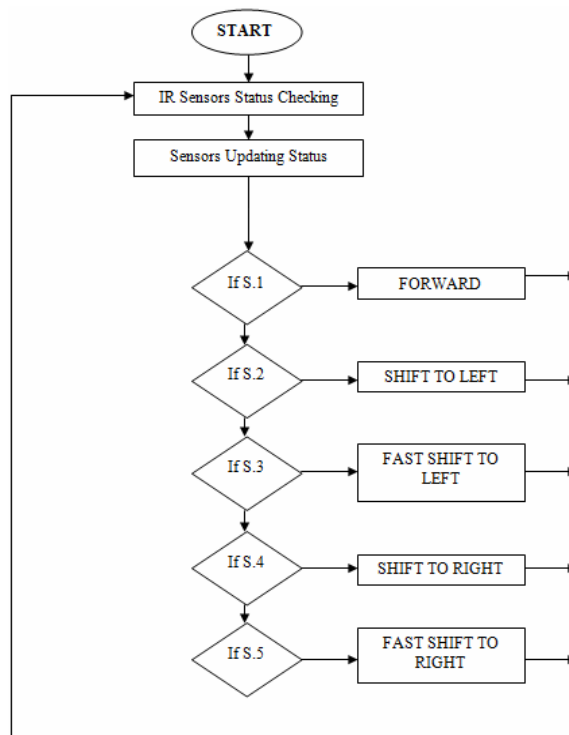


Fig. 6 Flow Chart for the Line Following System

Then all the input ports are enable pull-ups. There are two variables that will change the desired path and speed of robot which are tmp_fwd and tmp_steer. For the tmp_forward is function to move robot forward or reverse with specific speed, while tmp_steer function to determine the steering direction of robot to turn left or right at desired speed. The positive number in tmp_fwd and tmp_steer indicate that robot will move forward and turn right respectively and vice versa. The picture of the IR sensors and Controller Circuit Board are shown in Fig.7.

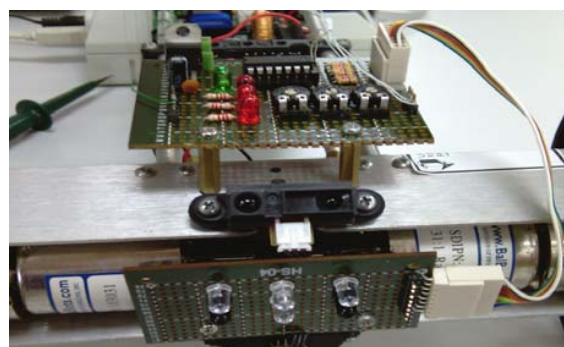


Fig. 7 IR Sensors and Controller Circuit Board

IV.RESULT AND DISCUSSION

In this section, result from the balancing and the line follower control algorithm obtained and observed from the robot has been described. The data were collected and recorded through the observation. The discussion section will highlight about initial calibration, efficiency in balancing mode, and the performance of line follower control system hardware and also algorithm. The analysis phase provides an opportunity to assess and evaluate the robot effectiveness and efficiency in maintaining stability and executing line following task. This allows a comparison to be undertaken between the actual system performance and the anticipated project objectives. The opportunity to calibrate and perform additional fine tuning of the design is also explored with the aim of achieving the best result possible in mind.

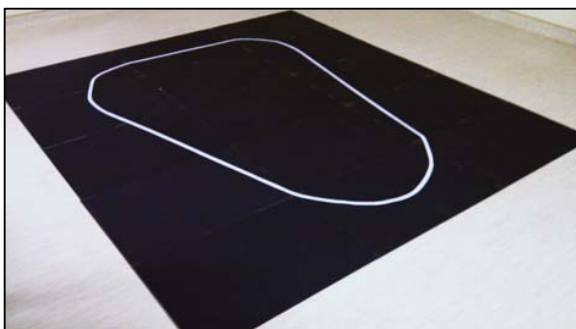


Fig. 8 Line Follower Track

V.CONCLUSION

This project as a whole was very successful in driving a line follower two wheeled balancing robot with all the considerations necessary in ensuring it could meet the required capabilities and objectives. The software and programming parts is achieving satisfaction level. For future work and suggested areas of further investigation, it is recommended that using a gyrosensor instead of IR sensor to make the robot balance on all types of surface including uneven surface. Thus, it will be more robust in terms of controllability and functionality.

REFERENCES

- [1] S.W. Nawawi, M.N. Ahmad, J.H.S Osman (2008). "Real-Time Control of a Two Wheeled Inverted Pendulum Mobile Robot", Proceedings of World Academy of Science, Engineering and Technology Volume 29 May 2008.
- [2] Ooi, Rich Chi (2003). "Balancing a Two-Wheeled Autonomous Robot". University of Western Australia. Thesis B. Mechatronics Engineering.
- [3] Ong, Yin Chee; Abidin, M.S.B (2006). "Design and Development of Two Wheeled Autonomous Balancing Robot", Center for Artificial Intell. & Robotic (CAIRO), University Technology Malaysia, Kuala Lumpur. 4th Student Conference on Research and Development (SCORED 2006), Shah Alam, Selangor, MALAYSIA, 27-28 June, 2006.
- [4] Vivien Coelho; Stanley Liew; Karl Stol; Guanyu Liu (2008). "Development of a Mobile Two-Wheel Balancing Platform for Autonomous Applications". Department of Mechanical Engineering, the University of Auckland. 15th International conference on Mechatronics and Machine Vision in Practice (M2VIP08), 2-4 Dec 2008, Auckland, New-Zealand.
- [5] José Miranda (2009), "Application of Kalman Filtering and PID Control for Direct Inverted Pendulum Control". Master Thesis, California State University, Chico Spring 2009.
- [6] R. Hollis, "BallBots," Scientific American, October 2006. Retrieved February 4, 2009. Available at the World Wide Web: <http://www.sciam.com/article.cfm?id=ballbots>
- [7] Pakdaman, M.; Sanaatiyan, M.M. (2009), "Design and Implementation of Line Follower Robot," Computer and Electrical Engineering, 2009. ICCEE '09. Second International Conference on , vol.2, no., pp.585-590, 28-30 Dec. 2009
- [8] Priyank Patil (2010), "AVR Line Following Robot," Department of Information Technology K. J. Somaiya College of Engineering Mumbai, India. Retrieved Mar 5, 2010. Available at: <http://www.kmitl.ac.th/~kswichit/ROBOT/Follower.pdf>
- [9] Swope Design. Inc., "Balancing Robots Made Easy", Available at: <http://www.Balbots.com>, 2004.
- [10] Miller Peter (2008), "Building a Two Wheeled Balancing Robot", University of Southern Queensland, Faculty of Engineering and Surveying. Retrieved Nov 18, 2009. Available at: <http://eprints.usq.edu.au/6168/>