Design And Development of Two Wheeled Autonomous Balancing Robot

Ong Yin Chee, Mohamad Shukri b. Zainal Abidin
Center for Artificial Intelligence And Robotic (CAIRO)
Universiti Teknologi Malaysia, City Campus,
Jalan Semarak 54100, Kuala Lumpur, Malaysia
Email : shukri@fke.utm.my, yinchee83@gmail.com

Abstract - **In this paper, the control of two-wheeled autonomous balancing robot is presented. The system is based on distance measuring sensor to detect the lean of the robot. PID control system is implemented to pilot the motors so as to keep the system in equilibrium. The navigation of the robot is controlled by operator by using RC remote controller. The robot has the ability to balance itself while moving and turning on a flat terrain.**

I. INTRODUCTION

Two wheeled balancing robots have gained momentum over the last decade in a number of control and robotics research centers around the world. Why? Two wheeled balancing robot is a good platform for researchers to investigate the efficiency of various controllers in control system. The research on two wheeled balancing robot is based on inverter pendulum model. Therefore, a two wheeled balancing robot needs a good controller to control itself in upright position without the needs from outside. Nowadays, various types of controllers were implemented on two wheeled balancing robot for examples Linear Quadratic Regulator, Pole-Placement Controller, Fuzzy Logic controller, Proportional Integrated Derivative Controller.

This project is to design and fabricate a two wheeled balancing robot that has the ability to balance itself on flat terrain and investigate the suitability and performance of PID controller in balancing system. The robot chassis must be robust, symmetrical & the center of gravity of the robot must be high. Two general purpose distance measuring sensors (Sharp GP2D12) are used to detect the current position and tilting angle of the robot. Microcontroller is used to be the brain of the robot. The entire PID controller algorithm will be compute into C programming and store inside the microcontroller. Without an active control system, the robot would just fall over. Therefore, PID controller plays an important role in this project. The navigation of the robot is controlled by operator by using RC remote controller. The robot has the ability to balance itself while moving and turning on a flat terrain.

II. DESIGN AND FABRICATION

Hardware Structure

The materials choose for the robot structure is an importance element because the frame of the robot must be robust, symmetrical and the center of the gravity of the robot must be high. Beside that, the frame design of the robot must have the space to house the two DC motors, sensors, circuitry and battery of the robot. The design of the frame determines the stability and centre of gravity of the robot in static position or when the robot is in motion. For this project, the robot's frame is making by raw material-Aluminum. Aluminum is used due to the light weight, rigid and robustness quality of the material which is most suitable for the balancing robot. Besides that, in order to make the center of the gravity of the robot to become high, the robot's battery location will be allocated on the top and center of the robot's structure. The wheels of the robot are made by PE because it is a rigid and light material.

The actuation to balance the robot is two unit of GM8712 gear motor, with a 60.5:1 reducing gearbox DC motor [19].

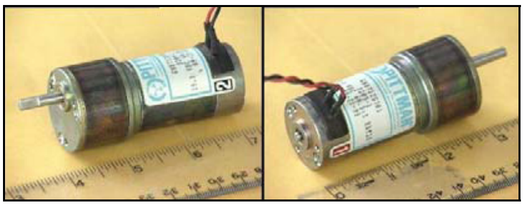

Fig.1: GM8712 DC motor

Fig. 2: The two wheeled balancing robot

Sensor and Electronic

Two general purpose distance measuring sensors (GP2D120) will be used on the balancing robot in order to detect the current state of the robot. The sensors are mounted on an aluminum strip placed at the front and back of the robot. The line of sight is toward the ground at an angle. The output of the sensors is an analog voltage. The GP2D12 can sense a height from 4 to 30 cm [17]. The sensor's analog output voltage is measured by the PIC on-board A/D register. The robot keeping balance by measuring the height at the front and back of itself and adjusts the wheels position to maintain equal height.

PIC 16F 877A will be used on this project due to its special functions registers like 8 Analog to Digital Converter module and 2 PWM generator module which are needed on this project. The output of the sensor which is in analog voltage will be connected to the PIC microcontroller A/D input pin. Beside that, two PWM generators will be used to activate the robot's motors and 20 MHz oscillator will be used to generate pulse for the microcontroller. To interface the DC motor and the microcontroller, a motor driver is needed. On this project, L293B will be used to do the interfacing. It provides total DC current up to 2 Ampere. In order to generate PWM pulse to two DC motors, the output pin of the PWM (CCP1/CCP2) will be connected to the L293B motor driver enable pin.

The 12V 1100mAH Ni-Mh rechargeable battery is used as the power source for the robot.

Control System

PID control algorithm will be implementing on this project for keeping the robot upright. The output of the sensor which is in analog voltage will convert into digital value first by using A/D converter of microcontroller then inputs it into PID control algorithm to determine the speed and direction of the motors in order to stabilize the robot and stand it upright.

There are three PID control systems implemented to stabilize the robot. One of the PID control systems is used to control the tilt and angle of the robot and the other two PID control systems are used to control the speed of both left and right DC motors.

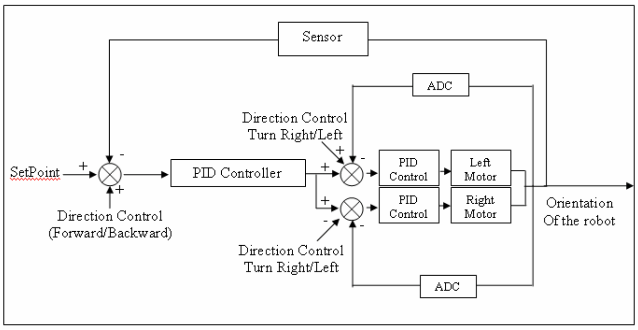

Fig. 3: Block Diagram of PID Control

The sampling rate of the program is 20ms. As its name suggests, the PID algorithm consists of a three part equation with proportional (P), integral (I), and differential (D) terms. These terms determine the controller output (power to the motors) from its input signal (the two sensor readings). The setpoint of the robot is the sensor's reading when the robot perpendicular to the flat terrain. The equation to calculate the PID controller output of the balancing system is simplified as follow:

$$\text{Error} = \text{Setpoint Reading} - \text{Current Sensor Reading} + \text{Input signal from RC receiver} \quad (1)$$

The input signals from RC receiver are: signal to move forward and backward

The proportional term increases the motor power as the robot leans further over and decreases the motor power as the robot approaches the upright position. A gain factor, Kp, determines how much power to apply to the motors for any given lean, as follows:

$$\text{Output Proportional Term} = Kp * \text{Error} \quad (2)$$

While the proportional term is effective at responding to the lean, once the robot reaches the upright position it will proceed to tip in the opposite direction until the proportional control term increases the motor power enough to reverse the robot's motion, rotating it back in the other direction. Therefore, the robot will oscillate back and forth, just as a car with worn out shock absorbers bounces for a long time when the car goes over a bump [11].

The differential term of the PID algorithm acts as a damper reducing oscillation. Another gain factor, Kd, determines how much power is applied to the motors according to the following equation:

Output Differential Term

*=Kp Kd * (Error – Last Error) / T* (3)
*=(Kp Kd/T) * (Error – Last Error)* (4)

Simplify as below:
*=KD * (Error – Last Error)* (5)

Finally, neither the proportional nor differential terms of the algorithm will remove all of the lean because both terms go to zero as the orientation of the robot settles near vertical.

The integral term sums the accumulated error (error summed over time) and applies power in the opposite direction indicated by the sum to drive the lean to zero, as follows:

Output Integral Term
*= KpKi * Sum of Error * T* (6)
= KpKiT (Sum of Error)* (7)

Simplify as below:
*= KI * (Sum of Error)* (8)

The output of the PID controller for balancing the robot will be:

Motor PWM = Proportional Term + Integral Term + Differential Term (9)

To tune the PID controller, Ki and Kd must be set to zero first and the Kp is slowly increase until the robot start to oscillate. Next, the Ki is slowly increased until the robot start to oscillate again. Then the Kd is slowly increased until the robot is stable and is not oscillating. Then the PID controller is tuned. Although the tuning method used is not optimized, but it does stabilized the robot to a level that it can balance in upright position.

The PID controller for both of the motor speed will use the same method as above. The output of the Motor PWM as equation (9) above will be used as the setpoint for left and right motors. The Back EMF method used to detect the current speed of both motors. The Back EMF refers to using the voltage generated by a spinning motor (EMF) to check the current speed of the motor's rotation [12].

Error of Right Motor Speed = Setpoint of right motor – Current Speed Reading of Right Motor (10)

Output Proportional Term of Right Motor
= Kp * Error of Right Motor Speed (11)

Output Differential Term of Right Motor
= Kd * (Error of Right Motor Speed – Last Error of Right Motor Speed) (12)

Output Integral Term of Right Motor
= Ki * Sum of Error for Right Motor Speed (13)

Right Motor Speed = Proportional Term of Right Motor + Differential Term of Right Motor + Integral Term of Right Motor (14)

For tuning the PID control of motor speed, the value of Kp Ki and Kd is get by trial and error method. Although this is not efficiency method but it can control the speed of motor very well. For the speed control of left motor, same method is using same as right motor speed control algorithm.

III. RESULT AND CONCLUSION

The Balancing Robot has successful balance itself in upright position and the navigation of the robot can control by a human operator. However, it has some drawbacks. The robot only can balance on a flat terrain. In order to improve this problem, more advance sensors and powerful algorithm must be implemented. Below figure is the actual output of the robot which show the tilting angle and PID controller output of the robot.

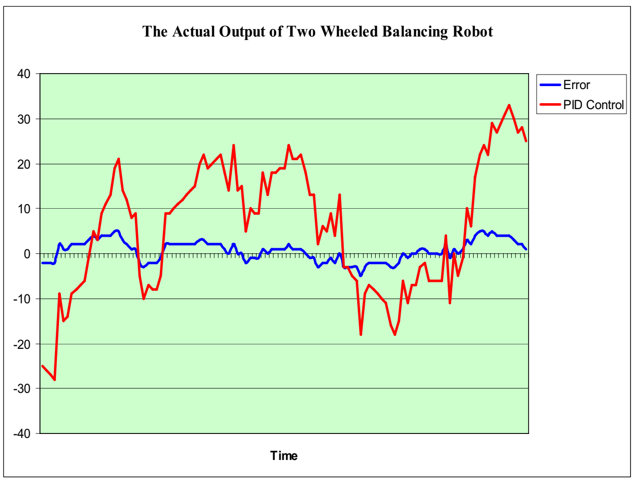

Fig. 4: The PID Output of the Robot.

IV. ACKNOWLEDGEMENTS

We would like to thank MOSTI and CAIRO for the funding support (IRPA Vot. 74256).

V. REFERENCES

[1] Ooi, Rich Chi (2003). Balancing a Two-Wheeled Autonomous Robot. University of Western Australia: Thesis B. Mechatronics Engineering.
[2] Christopher T. Killian, "Design, Modern Control Technology Components and System," Anne Arundel Community College, US.
[3] Bose, I. (2000). How to Make Good Homemade PCBs? Electronics for You Magazine. Pg 37-40.
[4] HI-TECH Software Pty. Ltd. PICC Lite C Manual. Australia: User Manual.
[5] Microchip Technology Inc. Brushed DC Motor Fundamentals. USA: Application Note.
[6] Microchip Technology Inc. RC Model Aircraft Motor Control. USA: Application Note.
[7] Microchip Technology Inc. Implementing a PID Controller Using a PIC18 MCU. USA: Application Note.
[8] http://leiwww.epfl.ch/joe/joerelatives.html
[9] http://www.geology.smu.edu/%7Edpa
[10] http://www.independencenow.com/?from=indetech. comindex.html
[11] http://www.teamhassenplug.org/robots/legway/
[12]http://www.acroname.com/robotics/info/articles/back-emf/back-emf.html
[13] http://www.prism.gatech.edu/~gth856d/buffalo/paper.pdf
[14] http://www.ridgesoft.com/articles/balancebot/BuildingABalancingRobot.pdf
[15] http://homepage.mac.com/sigfpe/Robotics/equibot.html
[16] http://www.balbots.com/articles
[17] http://info.hobbyengineering.com/specs/gp2d120.pdf
[18] http://www.tcnj.edu/~rgraham/PID-tuning.html
[19] http://www.pittmannet.com/122100.html