International Journal of Current Trends in Engineering & Research (IJCTER)
e-ISSN 2455–1392 Volume 2 Issue 3, March 2016 pp. 1-5
Scientific Journal Impact Factor : 3.468
http://www.ijcter.com

A ROBUST FEEDBACK CONTROLLED SELF BALANCING ROBOT

Samana Jafri[1], Sohrab Shah[2], Shaazaan Bagasrawala[3], Zeeshan Shaik[4]

[1,2,3,4] *Department of Electronics and Telecommunication,*
M.H. Saboo Siddik College of Engineering, Byculla,
Mumbai-400008, India

Abstract—The research on balancing robot has been increased over the last decade in a number of robotics research laboratories around the world. Due to the unstable dynamics of the system such robots have a lot of ability to balance on its two wheel and is also capable to twirl on the spot. This additional maneuverability allows easy navigation on various area, turn sharp corners and cover small steps . These type of robots capabilities to solve a number of problems of in industry and society. An futuristic use of this robot a motorized wheelchair utilizing this technique would give the user greater skillful movement and thus access to places where most able-bodied people take for granted. Such robots build can help humans to travel short distance in small area or factories where as to car buggies which is more polluting The robot utilizes a Proportional-Integral-Derivative (PID) controlled differential method used for steering trajectory control. A gyroscope and accelerometer is used to measure the tilt from the center position of the robot and the encoders on the motors to measure the wheel's rotation.

Keywords—self-balancing robot, Accelerometer, Gyroscope , PID Controller , Feedback Control

I. INTRODUCTION

Two wheeled balancing robots are based on inverted pendulum configuration relying on the dynamic balancing systems for balancing and maneuvering. These robot bases deliver exceptional robustness and capability due to their smaller size and power requirements. The research in this field had led to the invention of robots such as Segway, Murata boy etc. These type of robot are widely use in surveillance & transportation use. This project is based on development of a self-balanced two wheeled robot which has a configuration same to a bicycle. In particular, the main idea is on the electro-mechanical mechanisms & control algorithms enable the robot to sense and act according in real time for a dynamically changing world. While these are use in many other application of different robots but the construction of sensors, and filters and actuators and the synchronous working of the all is a learning experience.

BLOCK DIAGRAM AND DESCREPTION

The diagram consists of mainly:

- Accelerometer
- Gyroscope
- PID controller
- Motors

The whole bot gets balanced on two wheels having the required grip providing sufficient friction. In order to obtain the verticality of robot two things must be done, in one hand the angle of inclination must be measured, and in the other hand motors must be controlled to move forward or backwards to make an angle 0˚ .This it will maintain the center of gravity of the robot thus balancing the whole system**.** For measuring the angle, two sensors, accelerometer and gyroscope are used.

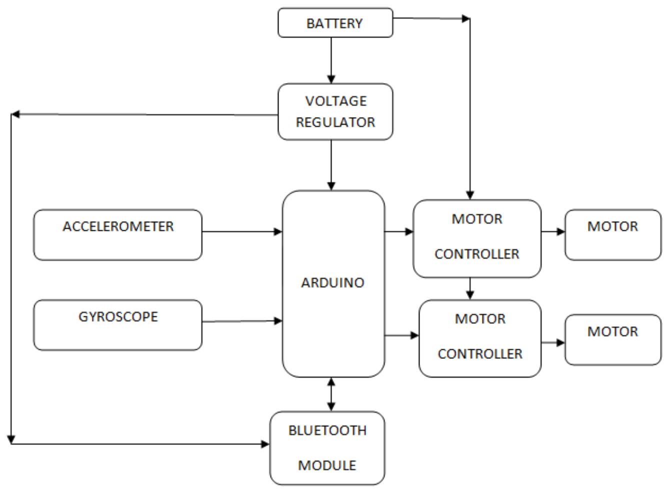

Figure 1: Block diagram of self-balancing robot

In the block diagram shown in the fig 1 we observe that when the supply is initiated the robot gets active and initially is in a slightly tilted position. Here the supply is given by the battery and a voltage regulator regulates the supply and gives it to the Arduino UNO and also to the MPU sensor which consist of mainly two sensors that is the gyroscope and the accelerometer. The accelerometer is used to measure the angle of tilt and the gyroscope will provide the angular velocity to the wheels of the robot via the Arduino which will further give it to the motor- driver to drive the wheels of the robot. The wheels of the robot move in the direction in which the robot is falling thus preventing it falling. The direction of the movement of the robot is provided by the Bluetooth control by which we can control the complete movement of it. This will be controlled by an application made by us so it can be made to use at any place where we want it to work and on any surface. Thus we see that all the components are interfaced and are working together in a synchronous manner.

II. ACCELEROMETER AND GYROSCOPE

The sensor used in this project is MPU 6050. The MPU is an electronics module consisting of more than one module in a single unit, which measures the angular velocity and linear acceleration values changed as inputs and sent to the main processor. The MPU sensor actually contains mainly two separate sensors as mentioned above. The initial one is the accelerometer. To describe the acceleration about three axes it generates three analog signals and acting on the planes and vehicle. Because of the physical limitations the significant output sensed of these accelerations is for gravity. The other sensor is the gyroscope. It also gives us three analog signals. These signals describe the vehicle angular velocities across each of the sensor axis. It is not necessary to place MPU at the vehicle center of mass, because the angular rate is not affected by linear or angular acceleration. The values from these sensors is collected by the aurdino attached to the MPU sensor through a 12 bit ADC (Analog to digital) board. The accelerometer is used on the balancing system in order to detect the current state of the model. Here MPU sensor used is MPU-6050.This chip contains a 3-axis gyroscope and 3-axis accelerometer. This makes it a “6 degrees of freedom measurement”. The MPU-6050 chip is a 3.3V IC, with a voltage range of 2.375V-3.46V, according to its datasheet. It has a built in low drop-out voltage regulator, so it is safe to power the chip through the Arduino 5V in built supply. This is recommended, as due to the voltage drop-out of the regulator on the VCC line, by using the Arduino 3.3V may not provide enough voltage. Other features include 16-bit ADC conversion on each channel and a Digital Motion Processor input. The DMP combines the rough sensor data and performs some complicated calculations in board to minimize the errors generated in each sensor. The DMP has a built in auto-calibration function. The biggest advantage of the DMP is that it eliminates the need to perform complex calculations on the Arduino side.

Gyroscope measures the angular rate around an axes. Tilt angle can be obtained by integrating angular rate over sampled time. An estimate of angular displacement is obtained by integrating velocity signal in time. Accelerometer can measure the force of gravity and with that information, the angle of robot can be obtained. Filter is used for the fusion of outputs of two sensors. It is a set of equations that provides an efficient computational means to estimate the process, in a way that minimizes the mean of the squared error. The filter is very significant in several aspects it takes care of the estimations of past, present, and future states. a It can also do so even when the accurate nature of the modelled system is unknown.

III. CONTROL SYSTEM

The control algorithm that is used to maintain the balance on the autonomous self-balancing robot is the PID controller. The PID controller is well known as a three term controller. The input to the controller is the error from the system. The Kp, Ki, and Kd are called as the proportional, integral, and derivative constants (the three terms get multiplied by these constants) respectively. In the PID controller the error gets managed in three ways. The error generated will be used by the PID controller to implement the proportional term, integral term for reduction of steady state errors, and derivative term to handle overshoots. The PID control algorithm can be modelled in a mathematical representation. The equation given is to calculate the PID controller output of the balancing system is simplified as follow

Error = Set-point Reading – Current accelerometer reading- Current gyro reading (1)

If only the first term was used to calculate the correction, the robot would have reacted in the same way similar to the classical line following algorithm. The second term helps the robot to move towards the mean position faster. The third term resists sudden change in deviation. The proportional term increases the motor power as the system leans further over and decreases the motor power as the system route towards the upright position. A gain factor, Kp, determines how much power to apply to the motor for any given lean, as follows:

Output Proportional Term = Kp *Error (2)

The differential term of the PID algorithm is used as a damper for reducing oscillation. One of the another gain factor, Kd, determines how much power is applied to the motor according to the following equation:

Output Differential Term = KpKd * (Error – Last Error)/T (3)

= (KpKd/T) * (Error – Last Error) (4)

Simplify as below:

O/P Differential Term = Kd*(Error–Last Error) (5)

Finally, neither proportional nor differential terms of the algorithm will be able to remove all of the lean since both terms go to 0 as the orientation of the system settles near vertical. The integral term sums the accumulated error and applies power in the opposite direction indicated by the sum to drive the lean to zero, as given below:

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

= KpKiT*(Sum of Error) (7)

Simplify as below: Output Integral Term = Ki * (SumofError) (8)

The output of the PID controller for balancing the model is Motor PWM = Proportional Term + Integral Term+ Differential Term (9)

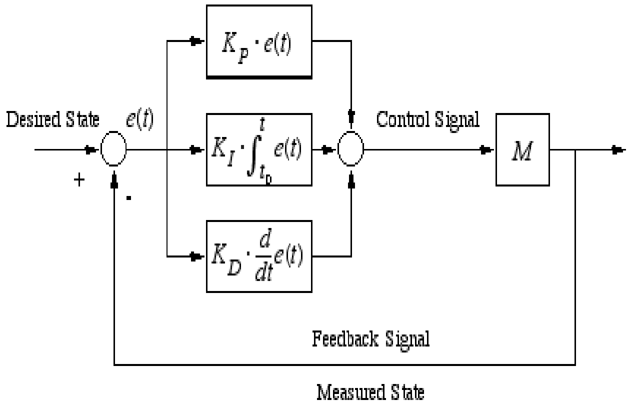

Figure 2: PID Controller

Fig2 shows the working of PID Controller. To tune the PID controller, Ki and Kd must be set to zero first and the Kp is slowly increased until the system start to swing. Next, the Ki is slowly increased until the system start to oscillate again. Then the Kd is slowly increased until the complete system is stable and is not oscillating. The output of the Motor as equation (9) above could be used as the set-point for the motor.

Error of the Motor Speed = Set-point of motor – Current speed of the motor (10)

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

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

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

Motor Speed =Proportional Term of Motor + Differential Term of Motor + Integral Term of 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 capable method but it can control the speed of motor very well.

IV. CONCLUSION

As performance of the mobile robotics is incrementing, dynamic effects are becoming ever more important. Self-Balancing System could balance in limited conditions without much complex circuits. The major limitations were the sensing of balance. The time taken to attain the stability is done within limited time and accuracy after the load is being placed. Because of the need to use the knowledge in fields of mechanics, electronics, programming and control, this project is highly interdisciplinary and as such the most representative mechatronic problems. The stability of the Self Balancing Robot may be improved if a properly designed gearbox that is having negligible gear backlash is used. So by implementation all of these concepts and by avoiding the errors that we came across the self-balancing bot is completely build. Further work will include increasing the level of self-determination of the robot by adding a vision system, thus allowing the robot to avoid obstacles. Segway and ball bot are applications of self-balancing Robot. Also, by improving the components of the robot we might be able to achieve higher speed.

REFERENCES

[1] He Bin, Liu Wen Zhen, Lv Hai Feng Shanghai University Shanghai Key Laboratory of Mechanical Automation and Robotics "The Kinematics Model of a Two-wheeled Self-balancing Autonomous Mobile Robot and Its Simulation", IEEE International Conference on computer engineering and application,P.R.China, June 2010.

[2] Junfeng Wu, Yuxin Liang," A Robust Control Method of Two-Wheeled SelfBalancing Robot", Zhe Wang College of Automation Harbin University of Science & Technology, The 6th International Forum on Strategic Technology,2011.

[3] Kui Peng Xiaogang Ruan, and Guoyu Zuo " Dynamic Model and Balancing Control for Two-Wheeled Self-Balancing Mobile Robot on the Slopes", School of Electronic Information and Control Engineering Beijing University of Technology, Beijing 100124 Proceedings of the 10th World Congress on Intelligent Control and Automation July 6-8, 2012, Beijing, China.

[4] Wei An and Yangmin Li, Senior Member, IEEE," Simulation and Control of a Two-wheeled Self-balancing Robot", Proceeding of the IEEE International Conference on Robotics and Biomimetic (ROBIO) Shenzhen, China, December 2013.

[5] Hau-Shiue Juabng and Kai-yew Lum, "Design and control of a Two-Wheel Self -balancing Robot using the Arduino Microcontroller Board", Proceeding of 10th IEEE conference in control and Automation Hangzhou, China, June 12-14,2013.

[6] Xiaogang Ruan, Wangbo Li, School of Electronic Information and Control Engineering Beijing University of Technology " Ultrasonic Sensor Based Two-wheeled Self-balancing Robot Obstacle Avoidance Control System", Proceedings of IEEE International Conference on Mechatronics and Automation, Tianjin, China, 3 - 6-2014.

[7] Lei Guo , Hong Wang , "PID controller design for output PDFs of stochastic systems using linear matrix inequalities", IEEE Transactions on Systems, Man, and Cybernetics, Part B 01/2005; 35:65-71,pp.65-71.

[8] K. Loffler, M. Gienger, F. Pfeiffer, and H. Ulbrich, "Sensors and control concept of a biped robot," IEEE Trans. Ind. Electron., vol. 51, no.5, pp.972–980.

