Volume2, Issue 12, December-2015, pp.1091-1095 ISSN (O): 2349-7084

International Journal of Computer Engineering In Research Trends

Available online at: www.ijcert.org

SELF BALANCING ROBOT

Mrs.LEKSHMY.S[1], ALEESHA GEORGE[2], ATHIRA C.V[3]

[1,2,3] Department of ECE, Vimal Jyothi Engineering College,
Chemperi, Kannur

E-mail:[1]lekshmyhari@vjec.ac.in , [2]aleeshageorge20@gmail.com and [3]athiracv90@gmail.com

Abstract:-In this paper, we presented the Balance model as a two wheeled self balancing robot that is capable of adjusting itself with respect to changes in weight and position. We developed the Balance System from a single gyroscope and a single accelerometer. The stability of the system is to show the capabilities of the ATmega328P in doing PID loops even with limited accuracy in position readings. PID control system is designed to monitor the motors so as to keep the system in equilibrium. It should be easily reproducible given the right parts and code.

Keywords – Self balancing robot, Accelerometer, Gyroscope, Kalman filter, PID controller, feedback control

1. INTRODUCTION

Two wheeled balancing robots are based on inverted pendulum configuration which relies upon dynamic balancing systems for balancing and maneuvering. This robot basis provides exceptional robustness and capability due to their smaller size and power requirements. Such robots find their applications in surveillance and transportation purpose. In particular, the focus is on the electro-mechanical mechanisms & control algorithms required to enable the robot to perceive and act in real time for a dynamically changing world.

In Self balancing robot, if the bot gets tilt by an angle, the centre of mass of the bot will experience pseudo force which will apply a torque opposite to the direction of tilt. This thesis presents a development self-balancing mobile robot using PID controller. The platform has been designed using mobile robot kits including IMU and two servos, and controlled by an open source microcontroller with PID. An Arduino microcontroller, hobby grade servos, and a six-degree of freedom (axis) accelerometer and gyroscope have been used to create the controlled platform. The controller has been designed to maintain the platform at an initially selected angle when the support structure orientation changes.

The value of PID parameters i.e. Kp, Ki and Kd have been obtained and applied to the Arduino.The software has been written with logic to convert the digital data from the accelerometer to an acceleration magnitude vector.The magnitude is then compared to a predetermined mathematical function to infer the angle of tilt of the platform. The angle of tilt is then converted to angle of rotation for the servos to act on.

2. BLOCK DIAGRAM AND DESCRIPTION

The block diagram consists of mainly:-

- Accelerometer
- Gyroscope
- Kalman filter
- 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˚.For measuring the angle, two sensors, accelerometer and gyroscope are used.

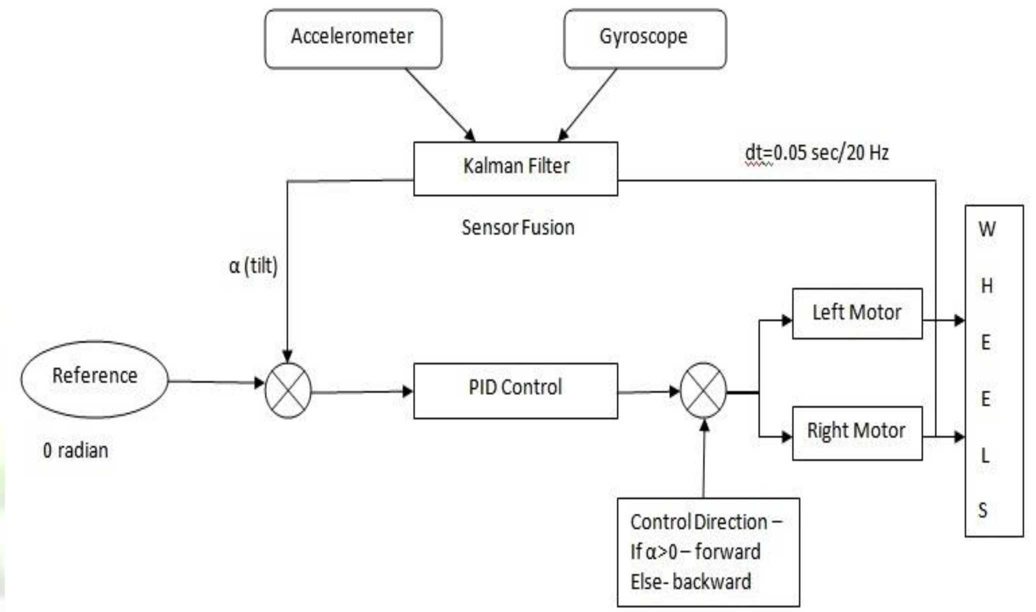

Fig1.Block diagram of self balancing robot

Accelerometer can sense either static or dynamic forces of acceleration.Static forces include gravity, while dynamic forces can include vibrations and movement.Gyroscope measures the angular velocity,so if this measure is integrated,we obtain the angle the robot is moved.The outputs of the sensors are fused using a Kalman filter.Sensors measure the process output say α which gets subtracted from the reference set-point value to produce an error. Error is then fed into the PID where the error gets managed in three ways. After the PID algorithm processes the error, the controller produces a control signal μ.PID control signal then gets fed into the process under control. Process under PID control is two wheeled robot.PID control signal will try to drive the process to the desired set-point value that is 0˚ in vertical position by driving the motors in such a way that the bot is balanced.

Here in this section, the components and techniques used for building the model which is composed of Arduino microcontroller, gyroscope, accelerometer, PID controller, motor driver and motors.

3. ACCELEROMETER AND GYROSCOPE

The sensor used in this project is IMU. The IMU is an electronics module consisting of more than one module in a single unit, which takes angular velocity and linear acceleration data as inputs and sent to the main processor. The IMU sensor actually contains mainly two separate sensors. The first 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 and thruster system, the significant output sensed of these accelerations is for gravity. The second sensor is the gyroscope. It also gives three analog signals. These signals describe the vehicle angular velocities about each of the sensor axis. It is not necessary to place IMU at the vehicle centre of mass, because the angular rate is not affected by linear or angular accelerations. The data from these sensors is collected by the microcontroller attached to the IMU sensor through a 12 bit ADC board. The sensor information communicates via a RS422 serial communications (UART) interface at a rate of about 10 Hz. The accelerometer is used on the balancing system in order to detect the current state of the model.

Here IMU sensor used is MPU-6050.This chip contains a 3-axis gyroscope and 3-axis accelerometer. This makes it a “6 degrees of freedom inertial measurement unit”.The InvenSense MPU-6050 chip is a 3.3V IC, with

Volume 2, Issue 12, December-2015, pp. 1091-1095

a working 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 rail.This is recommended ,as due to the voltage drop-out of the regulator on the VCC line, using the Arduino 3.3V rail may not provide enough voltage. Other features include a built in 16-bit analog to digital conversion on each channel and a proprietary Digital Motion Processor input. The DMP combines the raw sensor data and performs some complex calculations onboard to minimize the errors in each sensor. The DMP has a built in auto-caliberation function.The biggest advantage of the DMP is that it eliminates the need to perform complex and resource intensive 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 over time. Accelerometer can measure the force of gravity and with that information, the angle of robot can be obtained. Kalman filter is used for the fusion of outputs of two sensors. It is a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process, in a way that minimizes the mean of the squared error. The filter is very powerful in several aspects: it supports estimations of past, present, and even future states, and it can do so even when the precise nature of the modelled system is unknown.

4. CONTROL SYSTEM

The control algorithm that is used to maintain the balance on the autonomous self-balancing robot is the PID controller. The proportional, integral, and derivative (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 referred 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 will be used on the PID controller to execute the proportional term, integral term for reduction of steady state errors, and the 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 had been used to calculate the correction, the robot would have reacted in the same way as in the classical line following algorithm. The second term forces 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 approaches 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 acts as a damper reducing oscillation. 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:

Output Differential Term=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 system 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:

Output Integral Term= Ki * (Sum of Error) (8)

The output of the PID controller for balancing the model is

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

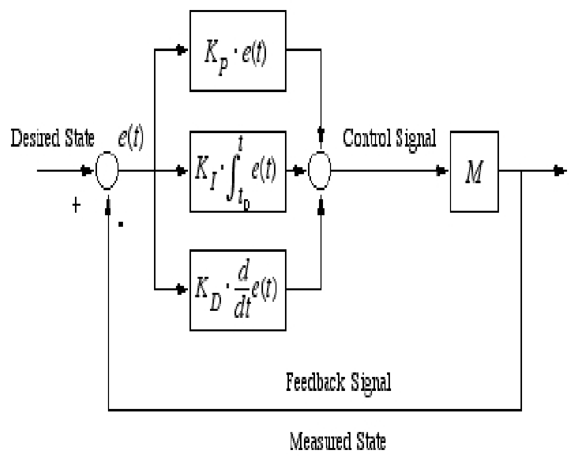

Fig2.PID Controller

Fig2 shows the working principle 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 oscillate. Next, the Ki is slowly increased until the system start to oscillate again.Then the Kd is slowly increased until the system is stable and is not oscillating.The output of the Motor PWM as equation (9) above will be used as the set-point for the motor.

Error of Motor Speed = Set-point of motor – Current Speed Reading of 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 efficiency method but it can control the speed of motor very well.

5. CONCLUSION

As performance limits in mobile robotics are increasing, dynamic effects are becoming ever more important.Self Balancing System could balance in limited conditions without much complex circuits.One of the major limitations was the sensing of balance. The time taken to attain the stable position 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 extremely interdisciplinary and as such one of 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 autonomy of the robot by adding a vision system,thus allowing the robot to avoid obstacles. Segway and ball bot are applications of self-balancing bot.Also,by improving the components of the robot we hope to achieve higher speeds.

REFERENCES

[1] Nawawi, S. W., Ahmad, M. N. and Osman, J. H. S (2008) "Real-Time Control of Two-Wheeled Inverted Pendulum Mobile Robot" World Academy of Science, Engineering and Technology 29, 214 - 220.

[2] 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.

[3] 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.

[4] R.S.Meena,Vikas Kumawat(2011) .Controller Design For Servo Motor Using MATLAB .In Proceeding of National Conferences on Advances & Research in Electrical System Technology (AREST 2011).

[5] Popescu, Cristina; Paraschiv, Nicolae; Cangea, Otilia.(2011) Comparison Between PID And Fuzzy Controllers Used In Mobile Robot Control annals of daaam & proceedings,jan2011, pp223.

[6] A. Warnasch, and A. Killen, Low cost, high G, micro electro-mechanical systems (MEMS), inertial measurements unit (IMU) program, IEEE Position Location and Navigation Symposium, 2002, pp. 299-305.

[7] Andrea Demetlika, Tomislav Tomašić, Mladen Crneković (2012),Self balancing Mobile Robot Tilter.FAMENA issue 3, volume 36.

[8] Young Soo Suh.(2003) .Attitude Estimation Using Low Cost Accelerometer And Gyroscope Proceedings KORUS 2003. The 7th Korea-Russia International Symposium on Volume: 2, Page(s): 423 - 427.

[9] D. Simon, Kalman Filtering With State Constraints: A Survey Of Linear And Nonlinear Algorithms, IET Proceediings in Control Theory & Applications,Volume 4, no. 8, pp. 1303-1318.