Design and Hardware Implementation of Ball & Beam Setup

Bilal Ahmad[1]
Dept. of Electrical Engineering
University of Engineering and Technology, Lahore
Lahore, Pakistan
bilalrouf@gmail.com

Iftikhar Hussain[2]
Dept. of Electrical Engineering
University of Engineering and Technology, Lahore,
Pakistan
Iftikharhussain.1994@gmail.com

***Abstract*—Balancing a ball on beam is the classic control problem just like inverted pendulum. Undergraduate students are used to develop the mathematical models of such classic control problems. In this project we theoretically design the PID controller and implement it on a very cheap and cost effective hardware. PID controller for this project is a dedicated design and implementation of a system which balances a ball at a particular point on a beam using a PID controller. The ball is free to move on the beam in any direction due to force of gravity. Here, the PID controller set the angle of the beam to make sure that the ball gets stable at our specified set point by eradicating the error between the current position and the reference point. The whole system could be implement by employing a servo motor, an ultrasonic distance sensor, a microcontroller and a support assembly. We used "MATLAB R2013a" to draw the theoretical response of multiple controller and "Arduino 1.6.8" to draw the real time response of ball & beam setup. This system is prior edge towards correlating our theoretical concepts with real time experimental environment and helps towards sophisticated system in grasping and development.**

Keywords—Ball & Beam Setup; Hardware Ball & Beam; PID Controller Design

I. INTRODUCTION

Ball and beam system is a non-linear [1] an open loop unstable system [2].The output ,ball-position, increases rapidly for a constant input (tilt angle of beam). Thus it's an open loop unstable system. It is a herculean task to maintain the ball at a fixed position because of its moving with an acceleration which is changing with amending the input (tilt angle). One end of the beam is coupled with the servo motor and the control signal was applied to servo to adjust the angle of the beam. Here by applying an appropriate feedback we stabilize this highly unstable system. This system is two degree of freedom system (Ball-position and Beam-angle) which is controlled by the voltage input to the servo motor.

Tasks such as system modeling, design and feedback control using P, PD and PID controller can be performed by implementing this system and we will be able to compare the performance of different controller which we applied.
This system has close uniformity to various application in the field and industry. A key example of the open loop unstable system is the control of the air craft during vertical take-off. The angle of thruster and diverters must be controlled continuously to avoid tipping and with proper feedback highly unstable system like F-35B can be stabilized. F-35B is an American modern fighter jet with ability of vertical takeoff without running on runway.

Here the control systems aims to calibrate the angle of the beam to make the ball stable at specific set point. As we know when ball moves on beam, it has an acceleration. Thus, we can control the position of ball by adjusting its acceleration. It demonstrates that there are two integrator and dynamical properties of beam which provides an important attribute which is open-loop unstable and a non-linear system [2] [1].

As ball and beam system is an open loop unstable system so first, we did the mathematical modeling of the system. After that using SISOTOOL we design the PID controller for the current system and draw the step response of the system using MATLAB. We applied three different controllers which are Proportional (P), Proportional Derivative (PD) Proportional Integral Derivative (PID) controller and conclude from practical results that PD and PID gives us required response with required parameters and stable this unstable system. We linearized this non-liner system by using linear approximation and then applied a linear feedback control such as PID controller. Some intelligent controllers are also present in literature like Neural Network [6], Fuzzy Logic [7], sliding mode fuzzy control, fuzzy neural control [8] and LQR [9].

II. HARDWARE DESIGN

There are too many arrangements which are possible to support the beam. One is to fix the beam in the middle of actuator (servo motor) which is shown below in figure 1 and it rotates against the central axis [4] [13]. In second experimental arrangement, the beam is lifted by two level arms which is shown below in figure 2. One level arm is affixed with the support assembly and other is jointed with the liver-arm of servo motor. This setup has drawbacks such as it demands more elements and has more frictional losses. The advantage to use 2nd setup is that it requires small servo motor to actuate the beam due to lever-arm effect. We also found some afflictions in driving the transfer function

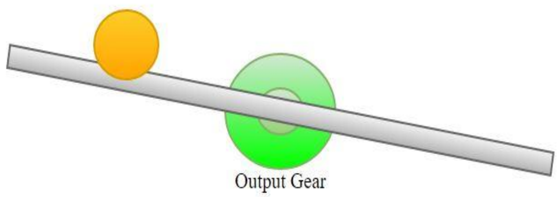

Figure 1. A simple ball & beam system realization when beam is fixed in the center.

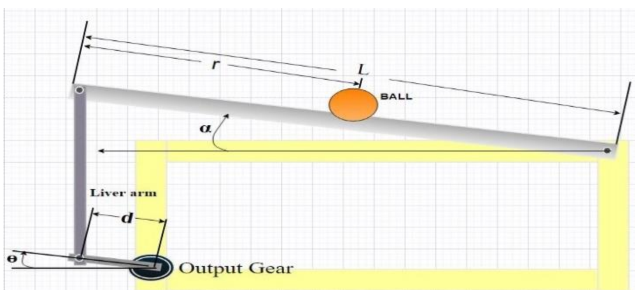

Figure 2. Ball & beam system with one end fixed.

Finally, we decide to use 2nd configuration to implement our current control system in which one end of beam is fixed. Our salient intention of design was to make a model which is balanced, light weight, easily transferable and economically affordable. The picture of the system which we implemented is shown in figure 3. Each component is labeled in the picture and described below in relative headings.

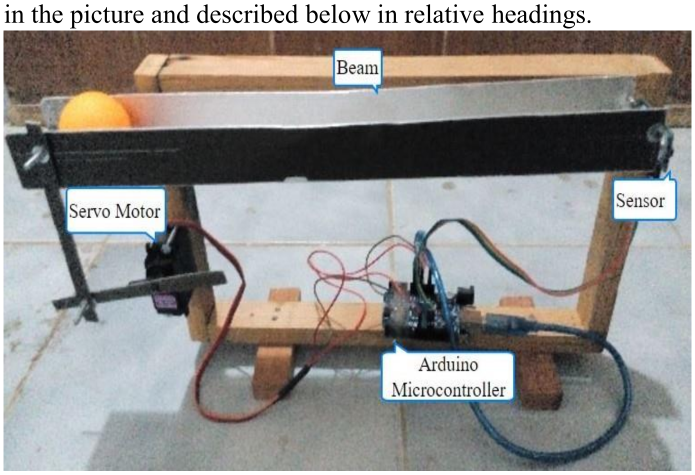

Figure 3. Implemented ball & beam hardware setup.

1. *Beam*
   We have to choose a light weight and low density material which is facile to actuate for a low power servo without overloading. Beam made of wood was creating the overloading effect to servo motor. Thus we chose an Aluminum (Al) beam which proved an appropriate choice later. The ball is moveable freely along the axis and inside the beam to ensure smoothness.
2. *Servomotor*
   As shown in prototype picture, the output shaft of servo is connected with the liver arm and this arm is further coupled with the level arm through a moveable joint. We used MG996R-Tower Pro model of servo motor.
3. *Support Assembly:*
   The support assembly is of rectangular-tower shape, made of wood to make our model light weight and easy to carry.
4. *Ultrasonic Sensor*
   We used an ultrasonic sensor (HCSR-04) to measure the position of ball on beam continuously. As ball moves inside the beam, thus we placed the distance sensor to the fixed end of beam. The inside movement of ball through the beam makes the performance of the sensor easy and abates the noise.

5. *Microcontroller*
   We used an Arduino UNO [5] for computing purposes which is economic and easy to code. The digital pin 4 and 5 are trig and echo pins respectively and pin 9 was assigned a task to give control signal to servo motor. The system we have designed is economical, well balanced and light in weight.

III. SYSTEM MODELING

A free body diagram of ball & beam system is given below where one end of beam is fixed. While modeling the system mathematically work done in the University of Michigan and the Milwaukee School of Engineering helped us a lot [14].

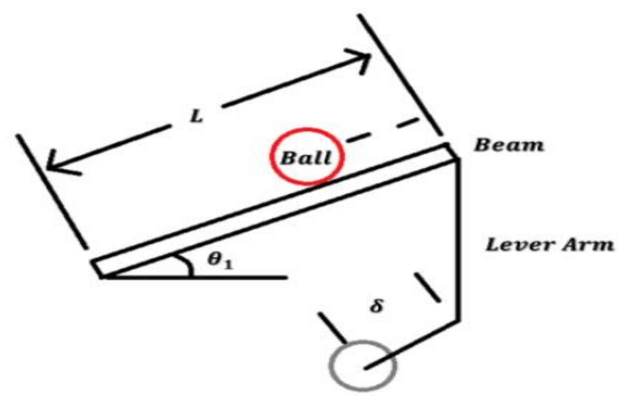

Figure 4. Ball & beam system used for mathematical modeling.

In this free body diagram the system parameters are as follows:

$$m = mass\ of\ ball$$

R = *radius of ball*
δ = *offset*
L = *length of beam*
x = *ball position*
$\theta 1$ = *beam angle*
$\theta 2$ = *gear angle*

In this system, the equations of motion can be used to solve for the relationship between the angle of the beam and the ball movement. The equations of motion can be developed into the following form.

$$M + \frac{\delta^2 x(\mathrm{t})}{\delta \mathrm{t}^2} + C\frac{\delta \mathrm{x}(\mathrm{t})}{\delta \mathrm{t}} + kx(t) = F(t)$$
$$M + \frac{\delta^2 X(\mathrm{t})}{\delta \mathrm{t}^2} + C\frac{\delta \mathrm{X}(\mathrm{t})}{\delta \mathrm{t}} + kX(t) = F(t)$$
$$M\ddot{X} + C\dot{X} + kX = F(t)$$

From this equation, it is possible to solve for M and for theta. The reason for doing this is to develop a relationship between the movement of the ball and the angle of the beam. Once this relationship is developed, it is possible to code, model, and predict the position of the ball based on the angle of the beam. The next step in the system modeling is to solve for M. This is shown below in figure.5.

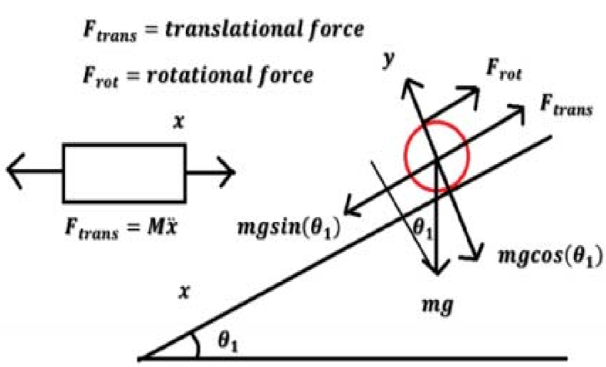

Figure 5. Ball & beam system used for mathematical modeling.

$$F_{rot} = \frac{Tr}{R} = \frac{J}{R}\frac{\delta w_b}{\delta t} = \frac{J}{R}\frac{(\frac{V_b}{R})}{\delta t}$$
$$F_{rot} = \frac{\frac{J}{R}\delta^2(\frac{x}{R})}{\delta \mathrm{t}^2} = \frac{J}{\mathrm{R}^2}\ddot{X}$$
$$F_x = F_{trans} + F_{rot} = m\ddot{x} + \frac{J}{R}\ddot{x} = (m + \frac{J}{R})\ddot{x}$$
$$M = m + \frac{J}{\mathrm{R}^2}$$

After solving for M, it is necessary to solve the angle of the beam. This can be done by equating the value for M to the force of the ball and gravity. This is shown below.

$$(m + \frac{J}{\mathrm{R}^2})\ddot{x} = -mg\sin(\vartheta_1)$$
$$(m + \frac{J}{\mathrm{R}^2})\ddot{x} + mg\sin(\vartheta_1) = 0$$
$$mg\sin(\vartheta_1) = mg\Theta_1$$

Equation of motion is:

$$\boxed{\left(m + \frac{J}{\mathrm{R}^2}\right)\ddot{x} + mg\Theta_1 = 0}$$

$$J_{sphere} = \frac{2}{5}m\mathrm{R}^2$$
$$m + \frac{J}{\mathrm{R}^2} = m + \frac{\frac{2}{5}m\mathrm{R}^2}{\mathrm{R}^2} = m + \frac{2}{5}m = \frac{7}{5}m$$
$$\left(\frac{7}{5}m\right)\ddot{x} = mg\Theta_1$$
$$\frac{7}{5}\ddot{x} = g\Theta_1$$
$$\ddot{x} = \frac{5}{7}g\Theta_1$$
$$Since\colon \Theta_1 = \frac{d}{L}\Theta_2$$
$$\left(m + \frac{J}{\mathrm{R}^2}\right)\ddot{x} + mg\frac{d}{L}\Theta_2 = 0$$
$$\boxed{\left(m + \frac{J}{\mathrm{R}^2}\right)\ddot{x} = -mg\frac{d}{L}\Theta_2}$$

Taking the Laplace transform and rearranging the equation gives:

$$\left(m + \frac{J}{\mathrm{R}^2}\right)R(s)\mathrm{s}^2 = \frac{-mgd}{L}\Theta(s)$$
$$\boxed{P(s) = \frac{R(s)}{\Theta(s)} = \frac{-mgd}{L\left(m + \frac{J}{\mathrm{R}^2}\right)}\frac{1}{\mathrm{s}^2}\ [\frac{m}{rad}]}$$

Table I. Values of constants used in this paper.

Abbreviation	Quantity	Value	Units
(g)	gravitational acceleration	-9.8	m/s^2
(R)	Radius of ball	0.02	m
(L)	Length of beam	0.34	m
(m)	mass of the ball	0.0026	Kg
(d)	Lever arm offset	0.05	m
(J)	Moment of inertia of ball	2*m*R^2/5	Kg.m^2

IV. PID MODELING

A proportional-integral-derivative controller is a linear feedback controller which is being used at large scale in

industry. It compiles the error (difference between set point and current value) and endeavor to alleviate it by varying the control parameter. Particularly, in our control system the control parameter is the angle of beam which is regulated by the PID controller [10].
The transfer function of PID controller is shown below [12]

$$G_c(s) = K_1 + \frac{K_2}{s} + K_3 s = \frac{K_1 s + K_2 + K_3 s^2}{s} = \frac{K_3\left(s^2 + \frac{K_1}{K_3}s + \frac{K_2}{K_3}\right)}{s}$$

Here k1, k2 and k3 are proportional, derivative and integral gains respectively. Onward in paper, we denote K1 as Kp, K2 as Ki and K3 as Kd.
The block diagram of our proposed system with PID controller and a unity feedback of ball-position on beam is shown below in figure 6 and 7 respectively.

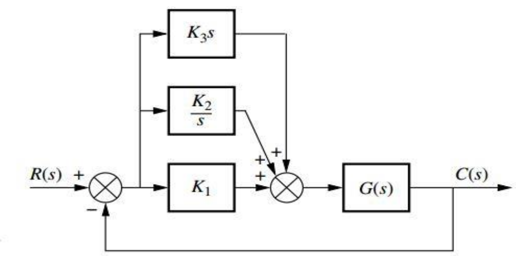

Figure 6. General system with P, PI and PID controllers.

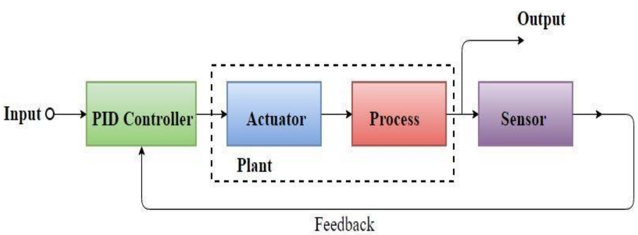

Figure 7. General system with P, PI and PID controllers unity feedback.

We use MATLAB to design the transfer functions and to draw the step responses of P, PD and PID controllers.
First, we obtain the transfer function of P controller and draw its step response by taking Kp=6. We can draw the conclusion from the response that system is unstable it can be seen in figure 8.

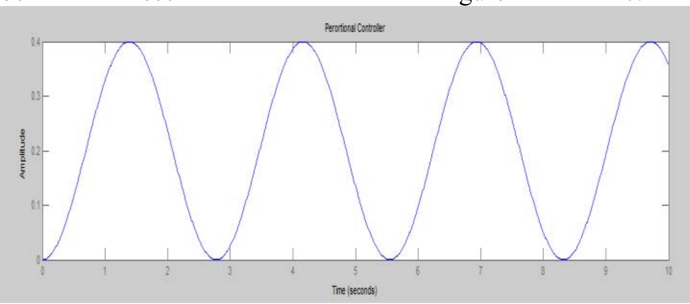

Figure 8. Simulation result of proportional controller.

After that, we designed a PD controller with the help of SISOTOOL and get Kp=7 Kd=5.5 We got a stable response which is shown in fig (9) but our required parameter of settling time and %age overshoot are not achieved. Objective was to design a system with %OS less than 15% and settling time less than 2 second. With the help of PD controller we obtained %OS=14.6 and settling time =2.2 sec.

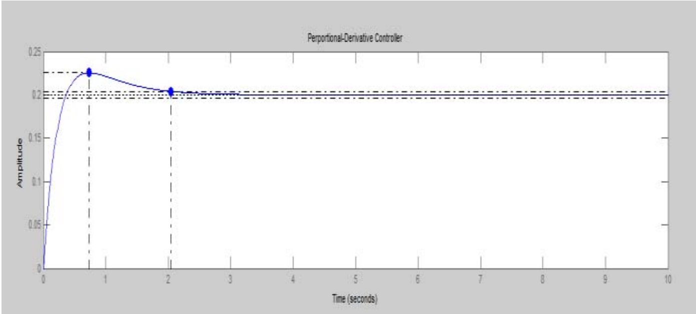

Figure 9. Simulation result of PD controller.

Finally, we designed a PID controller with Kp= 15.1, Kd= 8.4 and Ki = 2.9. The system gets stable and required appropriate response is achieved. The response is shown in figure (10). Results are summarized in table 2 for various parameters like %OS, Settling time and steady state error.

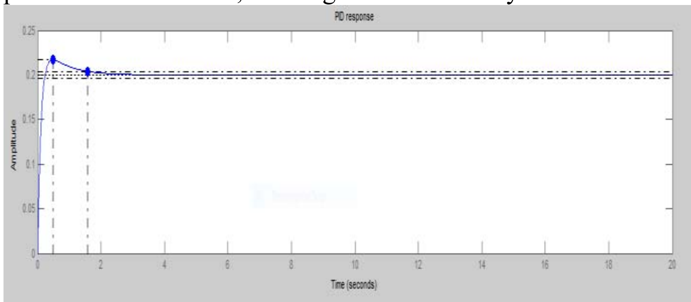

Figure 10. Simulation result of PID controller.

V. Experiment and Modeling

In this section we thoroughly demonstrate that how we can implement the prototype model. If we intend to control the acceleration of the ball, then we will have to control the angle of the beam. For this purpose, position of ball on beam is to be measured in real time. To obtain the position an ultrasonic sensor is localized at the fixed end, inside of the beam. The sensor generate a feedback signal which is given to the microcontroller. The figure 11 elaborate the portion of hardware setup where ultrasonic sensor is used.

Figure 11. Closed loop system with PID controller.

The subtractor is given with input and feedback signal. It calculates the difference (error) and sends it to the PID controller. After that the PID controller generates a control signal and fed it to the servo to calibrate the angle of beam with reference to the error in ball-position from set point.

Practically, we get the feedback signal from an ultrasonic distance sensor and transfer it to the Arduino UNO microcontroller. It calculates the error signal then PID algorithm uses it as input and calibrates the angle of beam.

We utilize serial communication between the Arduino and the host PC to monitor the real-time position of ball on beam. This task can be performed by awaking 'serial plotter' using command "Ctrl + Shift + L" in Arduino 1.6.8 environment.

After the implementation phase the system was likely to take a test. We use trial & error and Ziegler-Nichols method to adjust the gains of PID controller [11]. The trial and error gains are often different from theoretically obtained gains. The reason is that we neglect some non-linearities and friction in the system. After drawing out the particular values of gains of PID controller, we code and uploaded it to the Arduino software. The real time position of ball is measured by the sensor and sends it to the host PC by using serial communication with baud rate 9600. We apply P, PD and PID controllers and get their real time position graph in Arduino 1.6.8 by activating serial plotter. The serial plots of P, PD and PID controller are shown below in figure 12, 13 and 14 respectively.

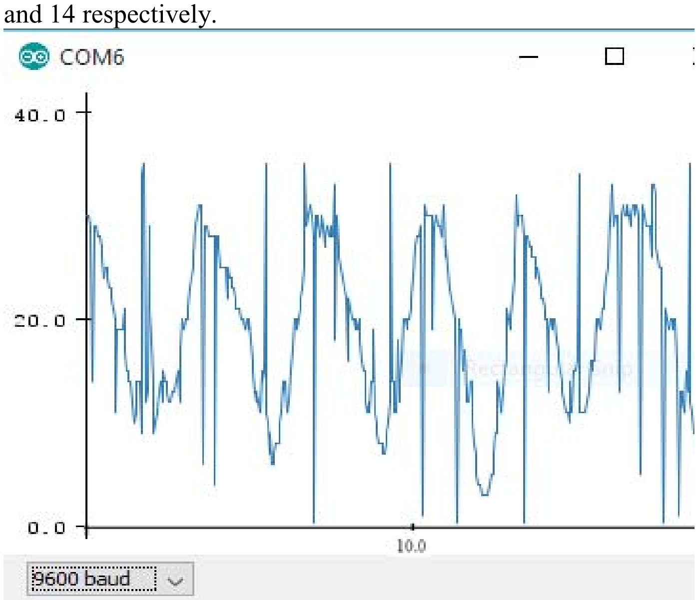

Figure 12. Experimental result when proportional controller is used only.

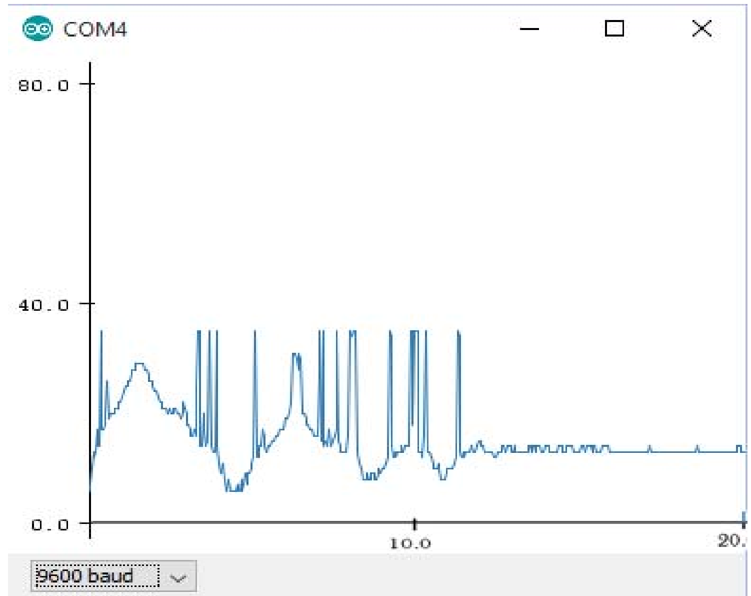

Figure 13. Experimental result when proportional & derivative controller is used.

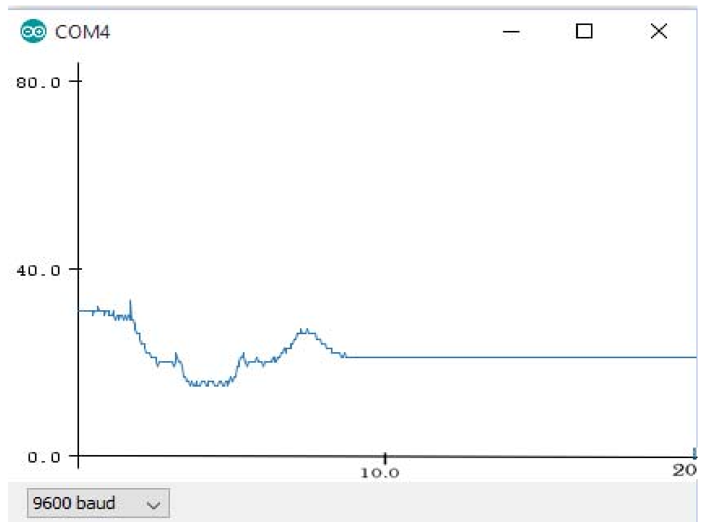

Figure 14. Experimental result when proportional, derivative and integrator controller used.

After circumspective observation, we realize that the system is imbalanced and unstable when P controller is used. The response of PD and PID controller is stable but PID controller handles the system most appropriately and meet our requirements.

We noticed ball get stable at desired position but still shaking about the desired position. This effect can be seen in real time PID controller plot. The cause may be the light weight of ping pong ball, because of which, there may not present enough force of friction. It might be, due to the noise of distance sensor or the unsuitable tuning of the PID controller. Inspite of neglecting some non-idealities, PID controller provides us desired results but the system needs to be ameliorate as yet.

VI. CONCLUSION

This research project was to familiarize the undergraduate students with hardware implementation of their own design. Theoretically designed PD controller was not meeting the requirements of settling time but PID controller provide the desired percentage overshoot and settling time. Graphs obtained from hardware are similar in shape with theoretically obtained graphs. Our designed hardware making the system stable almost after 5, 6 seconds whereas this should be stabilized before 2 seconds.

VII. REFERENCES

[1] Jeremiah Gertler, F-35 Joint Strike Fighter (JSF) Program Specialist in Military Aviation report, April 2014.

[2] Kuo, B. C. Golnaraghi, M. F. 2003. Automatic Control System, John Wiley & Sons New York.

[3] A. Ibrahim, 'Mechatronics Tutorials: Balancing of a Ball on Beam using Arduino as a PID controller', Mechatronicstutorials.blogspot.ca, 2014. Available: http://mechatronicstutorials.blogspot.ca/2014/07/balancing-ofball-on-beam-using-arduino.html.

[4] Evencio A. Rosales, Bennett T. Ito, Katie A. Lilienkamp, Kent H. Lundberg, "An Open Ended Ball Balancing Laboratory Project for Undergraduates", Proceeding of the 2004 American Control Conference Boston. Massachusetts, July 2004.Y

[5] Blogs.furman.edu, 'Arduino-controlled Ball-on-Beam with Interactive PID Interface « Furman Physics STEM Initiative', 2015. http://blogs.furman.edu/physicssteminitiative/2012/08/19/arduinocontrolled-ball-on-beam-with-interactivepid-interface

[6] H. Verrelst, K. Van Acker, J. Suykens, B. Motmans, B. De Moor and J. Vandewalle, "NLq Neural Control Theory: Case Study for a Ball and Beam System", Proc. of the European Control Conference (ECC'97), Brussels, Belgium, July 1–4, 1997.

[7] S.K Oh, H.J. Jang and W. Pedrycz, "The Design of a Fuzzy Cascade Controller for Ball and Beam System: A Study in Optimization with the Use of Parallel Genetic Algorithms", Engineering Applications of Artificial Intelligence, Vol. 22, pp. 261–271, 2009.

[8] P.H.Eaton, D.V.Prokhorov, D.C. Wunsch II, Neurocontroller alternatives for "fuzzy" ball-andbeam systems with nonuniform nonlinear friction, IEEE Trans. Neural Networks, Vol.11, No.2, 423-435, 2000.

[9] Z.H Pang, G. Zheng and C.X. Luo, "Augmented State Estimation and LQR Control for a Ball and Beam System", Proc. of the 6th IEEE Conference on Industrial Electronics and Applications (ICIEA), pp.1328-1332, June 21-23, 2011.

[10] https://en.wikipedia.org/wiki/PID_controller

[11]] K. Ogata. Modern Control Engineering. 3rd ed., New Jersey: Prentice Hall, 1997.

[12] Norman S. Nise. Control Systems Engineering. 6th ed., New York: John Wiley & Sons, 2010.

[13] Jeff Lieberman, "A Robotic Ball Balancing Beam", MIT Media Lab Publications, February 2004.

[14] Dr. Hadi Saadat. "BALL AND BEAM DESIGN PROJECT." EE-371 CONTROL SYSTEMS LABORATORY. Web. 12 May 2016.