

Stabilizing and Balancing of Linear and Rotary Inverted Pendulum System

Course No.: ME-400

Submitted By-

Nowab Md. Aminul Haq
Student ID: 1010130

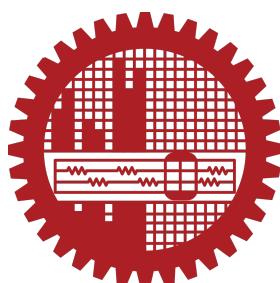
Ashik-E-Rasul
Student ID: 1010132

Submitted To-

Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of Bachelor of Science in
Mechanical Engineering.

Supervised by-

Dr. Md. Zahurul Haq
Professor
Department of Mechanical Engineering



Bangladesh University of Engineering and Technology (BUET)
Dhaka-1000, Bangladesh
February , 2016

CERTIFICATION

This is to certify that the work presented in the thesis is an outcome of the investigation carried out by the authors under the supervision of Professor Dr. Md. Zahurul haq, Department of Mechanical Engineering , Dhaka. It is declared that this thesis has been submitted only for the award of graduation.

Authors

Nowab Md. Aminul Haq

Ashik-E-Rasul

Signature of the Supervisor

Dr. Md. Zahurul Haq
Professor
Department of Mechanical Engineering
Bangladesh University of Engineering and Technology (BUET)

ACKNOWLEDGMENTS

We would like to express our deepest gratitude to our supervisor Dr. Md. Zahurul Haq for his guidance on this project showing us the path of conducting successful research and above all for always being there as our mentor. He shared his wisdom with us in analyzing subject matters and at the same time valued our thinking approach to synthesize those topics. His suggestions drove us towards better ways of thinking, his reviews enriched us in solving problems, and his support gave us strength at the time of our disappointment. We shall forever cherish the memories of working with him. We deeply thank our friends and families for always believing in us even at the moment when we were losing our confidence.

ABSTRACT

Balancing of an inverted pendulum is a classical problem in the field of Control Theory and Engineering. It's balancing is always challenging for the beginners in control engineering. This thesis deals with stabilization and control of both the linear and rotary inverted pendulum systems. Simultaneous approach is applied to system analysis as well as controller synthesis. Great similarity of both the systems is pointed out during the derivation of the equations of motion using the Euler-Lagrange equation. Linear inverted pendulum system is analyzed in two experimental setups , pendulum on cart and two wheel self balancing vehicle. PID controller is used to control the system by trial and error based tuning. Potentiometer (pot) and gyroscopic sensor is used as feedback sensor of the controlled system. Rotary inverted pendulum system is analyzed in a experimental setup developed by *QUANSER* and is balanced by Pole placement method. Rotary encoder is used as the feedback sensor of the controlled system. Later, Linear Quadratic Regulator (LQR) is designed for optimum control of the pendulum. System response and controller gains are simulated in *MATLAB* environment and after applying controller in experimental prototype, actual response of the system is reported.

Keywords: Inverted pendulum, PID controller, Pole placement method, LQR.

NOMENCLATURE

The following are the list of variables used in this thesis:

	For Rotary inverted pendulum-
For Linear inverted pendulum-	
M : Mass of the cart (kg)	m_p : Mass of the pendulum.(kg)
m : mass of the pendulum (kg)	L_p : Length of the pendulum.(m)
b : coefficient of friction for cart (N/m/sec)	J_p : Pendulum moment of inertia about center of mass.(kg.m ²)
l : length to pendulum center of mass (m)	B_p : Pendulum viscous damping coefficient
I : mass moment of inertia of the pendulum (kg.m ²)	m_r : Mass of rotary arm with two thumb screws.kg
F : force applied to the cart (N)	L_r :Rotary arm length from pivot to tip.(m)
x : cart position coordinate (m)	J_r : Rotary arm moment of inertia about pivot(kg.m ²)
θ : pendulum angle from vertical (down)	B_r : Arm viscous damping coefficient.
s :Laplace domain variable	θ : Arm angular displacement.(deg)
K_p : Proportional gain, a tuning parameter	α : Pendulum angular displacement from vertical plane.(deg)
K_i : Integral gain, a tuning parameter	τ : Torque induced by SRV02(N-m)
K_d : Derivative gain, a tuning parameter	η_g : Gearbox efficiency
e(t) :Error	K_g : High gear total gear ratio
t : Time domain variable or instantaneous time	η_m : Motor efficiency
T_i : Integral time	K_t : Motor current-torque constant(N-m/A)
T_d : Derivative time	K_m : Motor back emf constant
	R_m : Motor armature resistance(Ω)

TABLE OF CONTENTS

	Page
List of Tables	viii
List of Figures	ix
1 Introduction	1
2 Research Methodology	3
3 Linear Inverted Pendulum System	4
3.1 Inverted pendulum on cart	4
3.2 Two wheel self balancing vehicle	5
3.3 Mathematical Modeling	6
3.3.1 Inverted pendulum on cart	6
3.3.2 Self balancing vehicle	8
3.4 PID Controller	10
3.4.1 PID control of Plant	13
3.4.2 Ziegler Nichols Rules for Tuning PID Controllers	13
3.5 Design and simulation of PID controller	15
3.6 Prototype development	19
3.7 Experimental result: Prototype Response	22
4 Rotary Inverted Pendulum System	23
4.1 Mathematical Modeling	23
4.1.1 Model Conventions	23
4.1.2 Equation of motion	24
4.1.3 Linearizing	25
4.1.4 State Space Model of the system	26
4.2 Pole Placement Method	28
4.2.1 Controllability	29
4.2.2 Algorithm	29
4.2.3 Controller Design	29
4.2.4 Simulation Result	32
4.2.5 Prototype Response	33
4.3 Optimum Controller Design(LQR)	34
4.3.1 Why optimal control?	34

TABLE OF CONTENTS

4.3.2 Linear Quadratic Regulator(LQR)	35
4.3.3 Simulation Result	36
4.3.4 Advantage and disadvantages of LQR method	38
5 Conclusion	39
Bibliography	40

LIST OF TABLES

TABLE	Page
3.1 Ziegler Nichols Tuning rule based on Critical Gain K_{cr} and Critical Period P_{cr}	14
3.2 Arduino UNO development Board Specifications	21
4.1 Rotary Inverted pendulum specification[1]	24
4.2 State-Space Model Summery	28
4.3 Components of <i>QUANSER ROTPEN</i>	33

LIST OF FIGURES

FIGURE	Page
1.1 Inverted Pendulum on finger tip	1
3.1 (a) Cart-pendulum systems (b) Cart-pendulum experimental prototype	4
3.2 From Left To Right (a) Two wheel self balancing systems (b) Segway Human Transporter (c) Two wheel self balancing vehicle prototype	5
3.3 (a) Schematic diagram cart pendulum with system parameters (b) Free body diagram of cart pendulum system dynamics	6
3.4 Schematic diagram of two wheel self balancing systems	8
3.5 Feedback loop control system	10
3.6 Basic block diagram of PID controller	11
3.7 PID controller at a glance	11
3.8 System response curve	12
3.9 PID control of a plant	13
3.10 Closed loop system with a proportional controller	14
3.11 Sustained oscillation with period P_{cr} (P_{cr} is measured in seconds)	14
3.12 SIMULINK model of PID controlled cart pendulum system	15
3.13 System response without feedback and controller gain	16
3.14 System response with $K_p = 0, K_i = 0 \& K_d = 0$	16
3.15 System response with $K_p = 100, K_i = 0 \& K_d = 0$	17
3.16 System response with $K_p = 100, K_i = 0 \& K_d = 1$	17
3.17 System response with $K_p = 100, K_i = 1 \& K_d = 20$	18
3.18 (a) Two wheel self balancing vehicle experimental prototype (b) Cart-pendulum experimental prototype	19
3.19 Potentiometer(pot)	19
3.20 MPU 6050 gyroscopic sensor	20
3.21 Roll, Pitch and yaw angle in gyroscope	20
3.22 Arduino UNO development Board	21
3.23 Actual response of prototype for tuned PID values of $K_p = 110, K_i = 1.38 \& K_d = 0.33$	22
4.1 QUANSER Rotary Inverted Pendulum	23
4.2 Free body diagram of rotary inverted pendulum	24
4.3 Open Loop Poles and their position	29
4.4 Desired closed-loop pole locations	30
4.5 Control System in MATLAB with QUARC	32

4.6	Step response of rotary inverted pendulum system	32
4.7	Components of <i>QUANSER ROTPEN</i>	33
4.8	Flow Diagram of ROTPEN working principle	34
4.9	Step response of the prototype	34
4.10	Response and Control Effort	35
4.11	Starting value of LQR control	36
4.12	Step response within of ROTPEN within motor capacity	36
4.13	Pendulum and arm response for first controller	37
4.14	Applied motor voltage within the motor capacity	37
4.15	Pendulum angle within the control limit	37
4.16	Arm Angle within the control limit	38

C H A P T E R 1

INTRODUCTION

An inverted pendulum is a pendulum that has its center of mass above its pivot point. Whereas a normal pendulum is stable when hanging downwards, an inverted pendulum is inherently unstable, and must be actively balanced in order to remain upright; this can be done either by applying a torque at the pivot point, by moving the pivot point horizontally as part of a feedback system, changing the rate of rotation of a mass mounted on the pendulum on an axis parallel to the pivot axis and thereby generating a net torque on the pendulum, or by oscillating the pivot point vertically. A simple demonstration of moving the pivot point in a feedback system is achieved by balancing an upturned broomstick on the end of one's finger. [2]



Figure 1.1: Inverted Pendulum
on finger tip

The inverted pendulum is a classic problem in dynamics and control theory and is widely used as a benchmark for testing control algorithms (PID controllers, state space representation, neural networks, fuzzy control, genetic algorithms, etc.). It has some characteristics like; instability, multivariable and non-linearity. System has one input (force) and two output (angle and position) variable and angle and position of the pendulum have to be controlled with disturbance force only. Variations on this problem include multiple links, allowing the motion of the cart to be commanded while maintaining the pendulum, and balancing the cart-pendulum system on a see-saw. There are many instances of the inverted pendulum model both man made and found in

the natural world. Arguably the most prevalent example of an inverted pendulum is a human being. A person with an upright body needs to make adjustments constantly to maintain balance whether standing, walking, or running. The inverted pendulum is related to rocket or missile guidance, where the center of gravity is located behind the center of drag causing aerodynamic instability. The inverted pendulum has been employed in various devices and trying to balance an inverted pendulum presents a unique engineering problem for researchers. It's model has been used in some forms of personal transportation devices. Two-wheeled wheel chairs and other two wheeled motorized vehicles (Segway PT) can offer enhanced mobility for the driver.

There are different variations of the inverted pendulum on a cart ranging from a rod on a cart to a multiple segmented inverted pendulum on a cart. Another variation places the inverted pendulum's rod or segmented rod on the end of a rotating assembly. In both, (the cart and rotating system) the inverted pendulum can only fall in a plane. The inverted pendulums in these projects can either be required to only maintain balance after an equilibrium position is achieved or be able to achieve equilibrium by itself. Another platform is a two wheeled balancing inverted pendulum. The two wheeled platform has the ability to spin on the spot offering a great deal of maneuverability. Yet another variation balances on a single point. A spinning top, a unicycle, or an inverted pendulum atop a spherical ball all balance on a single point. As derived above the inverted pendulum can also be achieved by having a vertically oscillating base.

In this thesis balancing is done for both the Linear and rotary type inverted systems. The Literature review and research methodology of this thesis can be found accordingly in chapters 2 and 3 . Balancing of the inverted pendulum systems is discussed in chapter 4 and 5.

C H A P T E R 2

RESEARCH METHODOLOGY

The goal of this thesis was to study different control theories in balancing of Inverted Pendulum systems. We started the thesis work by studying the system dynamics and modeling of Inverted Pendulum in cart, which is the most popular research interest of inverted pendulum. Some simulation study has been done on *MATLAB*, to understand the system behavior. A *simulink* model has been developed in *MATLAB*, for PID control and the system is analyzed for different PID values. PID controller is selected as it provides satisfactory control without knowing the actual system specifications. Tuning of PID values has been done to stabilize the system response. After successful completion of simulation study, a prototype of cart pendulum system has been developed for experimental study and PID controlling has been applied in the prototype to make it balanced.

Next we used a rotary inverted pendulum system(ROTPEN) developed by *QUANSER*. The system specifications of ROTPEN is provided by the developer. So advanced control methods are implemented easily. We used pole placement method to stabilize the system. Later an optimal controller is designed and simulated in *simulink*. Advantage and disadvantages of the optimal controller is also studied.

C H A P T E R 3

LINEAR INVERTED PENDULUM SYSTEM

This chapter deals with the balancing of Linear Inverted Pendulum systems. The system prototypes are inverted pendulum on a cart and a self balancing two wheel vehicle. Firstly, the systems are briefly described, then the equations of motion are derived and mathematical modeling is done, then the design of the controller, simulation of the controlled model and finally prototype response is approached.

3.1 Inverted pendulum on cart

The simplest form of an inverted pendulum consists of a mass attached through a mass-less rod to a base mass. This is commonly known as a cart-pendulum system.

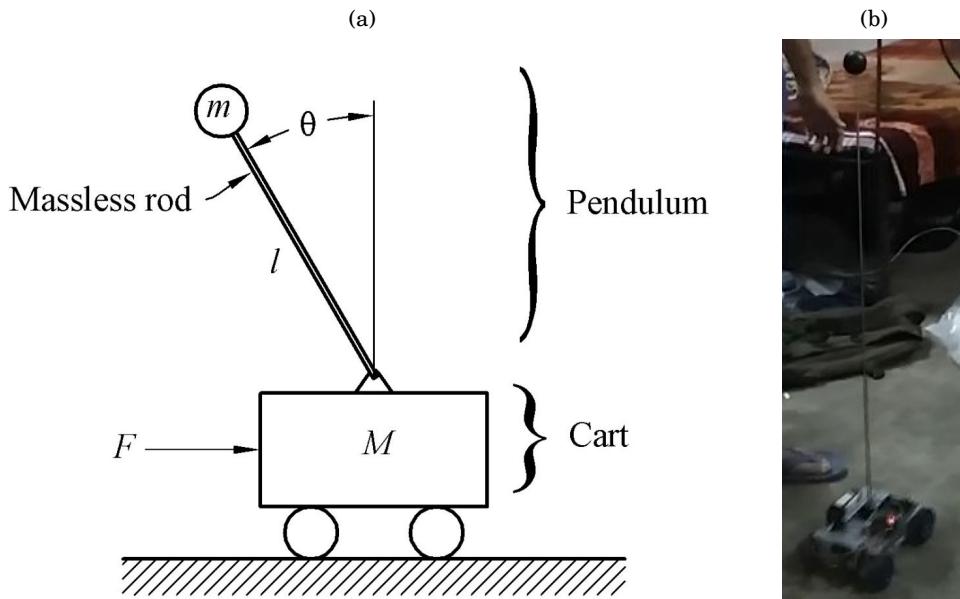


Figure 3.1: (a) Cart-pendulum systems (b) Cart-pendulum experimental prototype

This system is shown in Figure 3.1. The cart is free to move horizontally. The rod is connected to the cart through a rotational pin joint. This system is in unstable equilibrium when the rod is standing upright. Mathematically, this equilibrium can be maintained as long as there are no input forces whatsoever on the system. However, such conditions do not exist in real systems and

some means of stabilization is needed to maintain the pendulum in the upright position. A force F must be applied to the cart in order to move the cart pivot back and forth from one side of the pendulum mass center to the other side. The pendulum is always falling over, but the cart motion tries to keep the leaning angle, θ , at a small level.

3.2 Two wheel self balancing vehicle

A two wheel self balancing vehicle is popularly known as Segway Human Transporter. The two wheel self balancing vehicle system is shown in Figure 3.2(a), itself represents a linear inverted pendulum system , pivoted on the wheel axis.

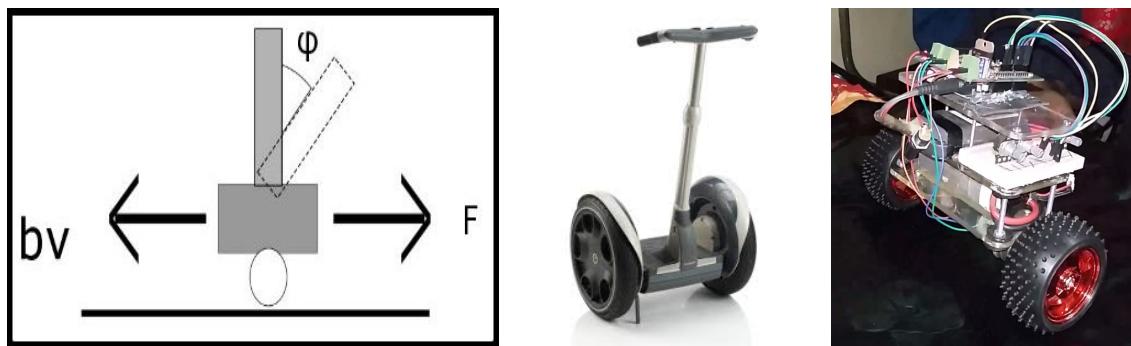


Figure 3.2: From Left To Right (a) Two wheel self balancing systems (b) Segway Human Transporter (c) Two wheel self balancing vehicle prototype

The Segway personal transporter, shown in Figure 3.2(b), is a device that transports one person at relatively low speeds. The low-speed (limited to approximately 12 mph) operation combined with its electric propulsion system makes the Segway a candidate for providing short-distance transportation on city streets, sidewalks, and inside buildings. When a Segway is in use, the device is driven by two wheels that are placed side-by-side, rather than the standard in-line configuration of a bicycle or a motorcycle. When the operator leans forward, the wheels turn in unison in the same direction to provide forward motion. In order to stop, the wheels must accelerate forward to get out in front of the system's center of mass and then apply a deceleration torque to slow the system down without causing the operator to fall forward off the device. These operating principles are reversed to allow the system to move backward. In order to turn, the wheels rotate at unequal speeds causing the system to travel in an arc. If the system is not translating forward or backward, then the wheels can rotate in opposite directions to turn the machine in place.

Two wheel self balancing vehicle prototype of our thesis work is shown in Figure 3.2(c). It is designed only to stabilize in vertically upright position, without any turning or directional control mechanism.

3.3 Mathematical Modeling

3.3.1 Inverted pendulum on cart

Force analysis and system equations

In this case a two-dimensional problem is considered, where the pendulum is constrained to move in the vertical plane shown in the figure 3.3(a) below. For this system, the control input is the force F that moves the cart horizontally and the outputs are the angular position of the pendulum θ and the horizontal position of the cart x .

This modeling is taken from control tutorials for MATLAB and SIMULINK available at [3] developed by Prof. Bill Messner at Carnegie Mellon and Prof. Dawn Tilbury at the University of Michigan.

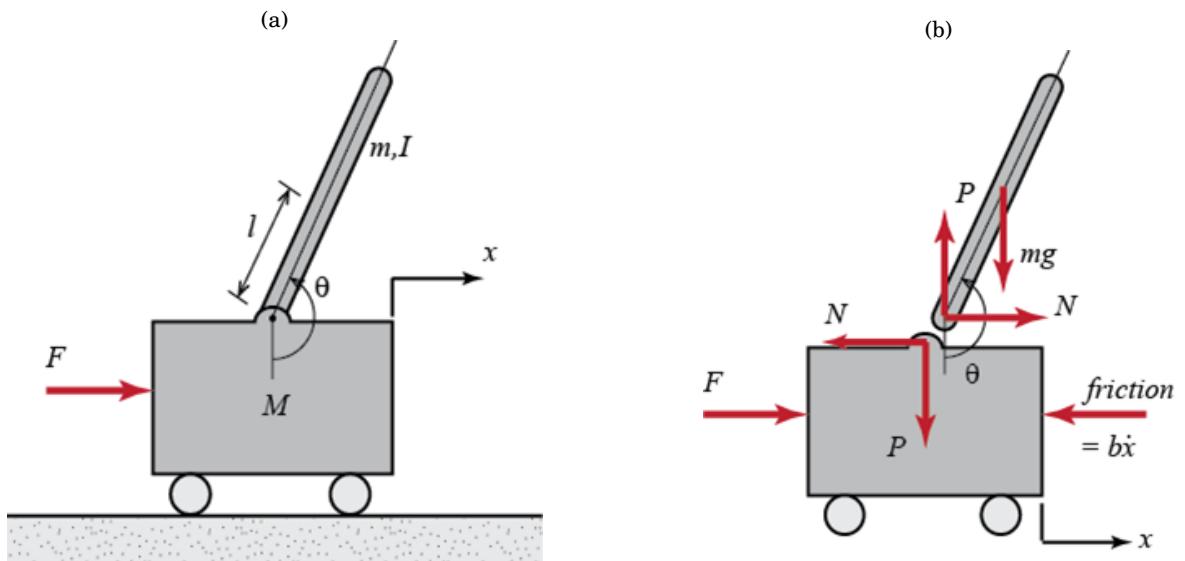


Figure 3.3: (a) Schematic diagram cart pendulum with system parameters (b) Free body diagram of cart pendulum system dynamics

Summing the forces in the free-body diagram of the cart in the horizontal direction, we get the following equation of motion.

$$(3.1) \quad M\ddot{x} + b\dot{x} + N = F$$

Summing the forces in the free-body diagram of the pendulum in the horizontal direction, we get the following expression for the reaction force N .

$$(3.2) \quad N = m\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta$$

substituting this equation into equation 4.1, we get one of the two governing equations for this system.

$$(3.3) \quad (M+m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F$$

Summing the forces perpendicular to the pendulum we get the second equation of motion for this system. Solving the system along this axis greatly simplifies the mathematics. We get the following equation-

$$(3.4) \quad Psin\theta + Ncos\theta - mgsin\theta = ml\ddot{\theta} + m\ddot{x}cos\theta$$

Summing the moments about the centroid of the pendulum we get the following equation-

$$(3.5) \quad -Plsin\theta - Nlcos\theta = I\ddot{\theta}$$

Combining these last two expressions, we get the second governing equation.

$$(3.6) \quad (I + ml^2)\ddot{\theta} + mglsin\theta = -ml\ddot{x}cos\theta$$

Since the analysis and control design techniques we will be employing , apply only to linear systems, this set of equations needs to be linearized. Specifically, we will linearize the equations about the vertically upward equilibrium position, $\theta = \pi$, and will assume that the system stays within a small neighborhood of this equilibrium. This assumption should be reasonably valid since under control we desire that the pendulum not deviate more than 20 degrees from the vertically upward position. Let ϕ represent the deviation of the pendulum's position from equilibrium, that is, $\theta = \pi + \phi$. Again presuming a small deviation (ϕ) from equilibrium, we can use the following small angle approximations of the nonlinear functions in our system equations:

$$(3.7) \quad cos\theta = cos(\pi + \phi) = -1$$

$$(3.8) \quad sin\theta = sin(\pi + \phi) = -\phi$$

$$(3.9) \quad \theta^2 = \phi^2 = 0$$

After substituting the above approximations into our nonlinear governing equations, we arrive at the two linearized equations of motion.

$$(3.10) \quad (I + ml^2)\ddot{\phi} - mg l \phi = ml\ddot{x}$$

$$(3.11) \quad (M + m)\ddot{x} + b\dot{x} - ml\phi = F$$

Transfer Function

To obtain the transfer functions of the linearized system equations, we must first take the Laplace transform of the system equations assuming zero initial conditions. The resulting Laplace transforms are shown below.

$$(3.12) \quad (I + ml^2)\phi(s)s^2 - mg l \phi(s) = mlX(s)s^2$$

$$(3.13) \quad (M + m)X(s)s^2 + bX(s) - ml\phi(s)s^2 = F(s)$$

To find our first transfer function for the output $\Phi(s)$ and an input of $U(s)$ we need to eliminate $X(s)$ from the above equations. Solve the first equation for $X(s)$.

$$(3.14) \quad X(s) = \left[\frac{I + ml^2}{ml} - \frac{g}{s^2} \right] \phi(s)$$

Then substitute the above into equation 4.13

$$(3.15) \quad (M + m) \left[\frac{I + ml^2}{ml} - \frac{g}{s^2} \right] \phi(s) s^2 + bX(s) - ml\phi(s)s^2 = F(s)$$

Rearranging, the transfer function is then the following

$$(3.16) \quad \frac{\phi(s)}{F(s)} = \frac{\frac{ml}{q}s^2}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mgl}{q}s^2 - \frac{bmgsl}{q}s}$$

where,

$$(3.17) \quad q = [(M + m)(I + ml^2) - (ml)^2]$$

From the transfer function above it can be seen that there is both a pole and a zero at the origin. These can be canceled and the transfer function becomes the following.

$$(3.18) \quad \frac{\phi(s)}{F(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b(I+ml^2)}{q}s^2 - \frac{(M+m)mgl}{q}s - \frac{bmgsl}{q}}$$

3.3.2 Self balancing vehicle

The Basic Transfer Function for two wheel self balancing vehicle is same as pendulum on cart, but the system dynamics and some of the system parameters are different. Figure 3.4 shows a schematic diagram of two wheel self balancing systems.

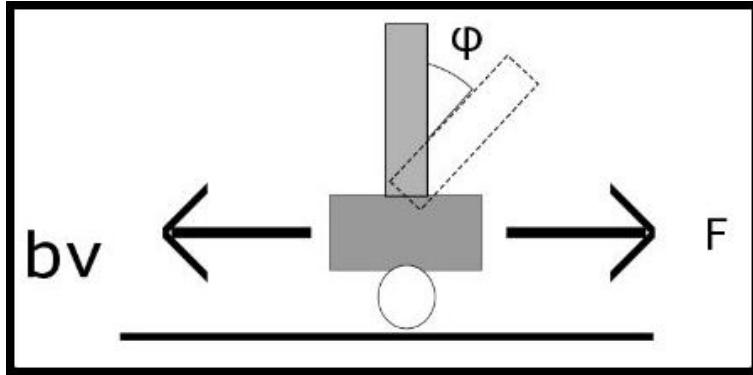


Figure 3.4: Schematic diagram of two wheel self balancing systems

The Transfer Function is,

$$(3.19) \quad \frac{\phi(s)}{F(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b(I+ml^2)}{q}s^2 - \frac{(M+m)mgl}{q}s - \frac{bmgsl}{q}}$$

where , ϕ = the yaw angle, M = mass of Lower Base, m = mass of Upper Part, I = moment of inertia , b = damping coefficient and $q = [(M + m)(I + ml^2) - (ml)^2]$. Here $F(s)$ is the Laplace

transformation of the Force required $F(t)$. This Force is directly connected to the torque of the motors τ .

$$(3.20) \quad F = 2\tau r$$

Again we can show , torque is dependent on the input voltage of a motor.

$$(3.21) \quad I = mr^2$$

$$(3.22) \quad \tau = I\alpha$$

From [?] we know the relation between Input Voltage and Angular Displacement of the motor.

$$(3.23) \quad \frac{\theta(s)}{E(s)} = \frac{\frac{K_t}{R_a J_m}}{s[s + \frac{1}{J_m}(D_m + \frac{K_t K_b}{R_a})]}$$

Here the value of $K_t R_a$ and K_b can be obtained from the relation ship between the Torque of a motor and the angular speed [?]

$$(3.24) \quad T_m = -\frac{K_b K_m}{R_a} \omega_m + \frac{K_t}{R_a} e_a$$

We know,

$$(3.25) \quad \alpha(s) = s^2 \theta(s)$$

So, the equation (5) will become

$$(3.26) \quad \frac{\alpha(s)}{E(s)} = s \frac{\frac{K_t}{R_a J_m}}{[s + \frac{1}{J_m}(D_m + \frac{K_t K_b}{R_a})]}$$

Now, from equations (1),(3),(4) and (8) we get the final equation

$$(3.27) \quad \phi(s) = AE(s)$$

where

$$(3.28) \quad A = \frac{ml}{s^3 + \frac{b(I+ml^2)}{q}s^2 - \frac{(M+m)mgl}{q}s - \frac{bmgsl}{q}} \frac{1}{2mr^3} \frac{\frac{K_t}{R_a J_m}}{[s + \frac{1}{J_m}(D_m + \frac{K_t K_b}{R_a})]}$$

This is the Required relation between Voltage of the Motor and ϕ .

3.4 PID Controller

The Proportional-Integral-Derivative (PID) controller is the most popular form of controller utilized in the control industry today. It has a simple controller structure, robustness to constant disturbances and availability of many tuning methods. It allows engineers to operate them in a simple, straightforward manner.

A PID controller continuously calculates an "error value" as the difference between a measured process variable and a desired set point. The controller attempts to minimize the error over time by adjustment of a control variable. The basic idea behind a PID controller is to read a sensor, then compute the desired actuator output by calculating proportional, integral, and derivative responses and summing those three components to compute the output.

Before we start to define the parameters of a PID controller, a closed loop/feedback loop system with some of the terminologies associated with it need to be discussed. A feedback loop is a common and powerful tool when designing a control system. Feedback loops take the system output into consideration, which enables the system to adjust its performance to meet a desired output response. In a typical control system, the process variable is the system parameter that needs to be controlled, such as temperature (°C), pressure (psi), or flow rate (liters/minute). A sensor is used to measure the process variable and provide feedback to the control system. The set point is the desired or command value for the process variable, such as 100 degree angle in a position servo motor control system. At any given moment, the difference between the process variable and the set point is used by the control system algorithm , to determine the desired actuator output to drive the system (plant). For instance, if the measured angle process variable is 90 degree and the desired angle set point is 100 degree, then the output specified by the control algorithm might be to set PWM to drive the motor. Driving the motor to turn the motor head causes the difference between set angle and desired angle decrease. This is an example of closed loop control system, because the process of reading sensors to provide constant feedback and calculating the desired actuator output is repeated continuously and at a fixed loop rate.

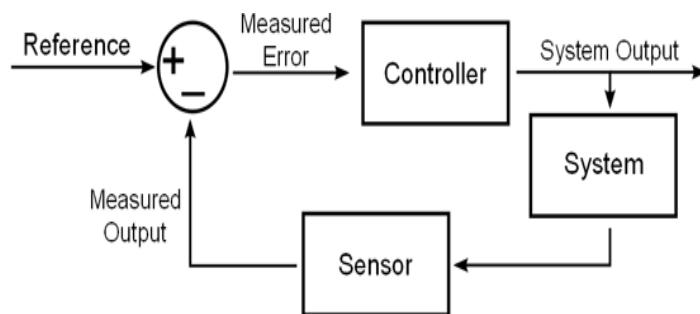


Figure 3.5: Feedback loop control system

The concept of the feedback loop to control the dynamic behavior of the system is illustrated in figure 3.5. This is negative feedback, because the sensed value is subtracted from the desired value to create the error signal, which is amplified by the controller.

The PID controller is named after its three correcting terms, the proportional, integral, and derivative terms and then their summation constitutes the controller output. If $u(t)$ is defined as the controller output, the final form of PID output is:

$$(3.29) \quad u(t) = K_p e + K_i \int e dt + K_d \frac{de}{dt}$$

In laplace Domain:

$$(3.30) \quad U(s) = K_p + \frac{K_i}{s} + K_d s$$

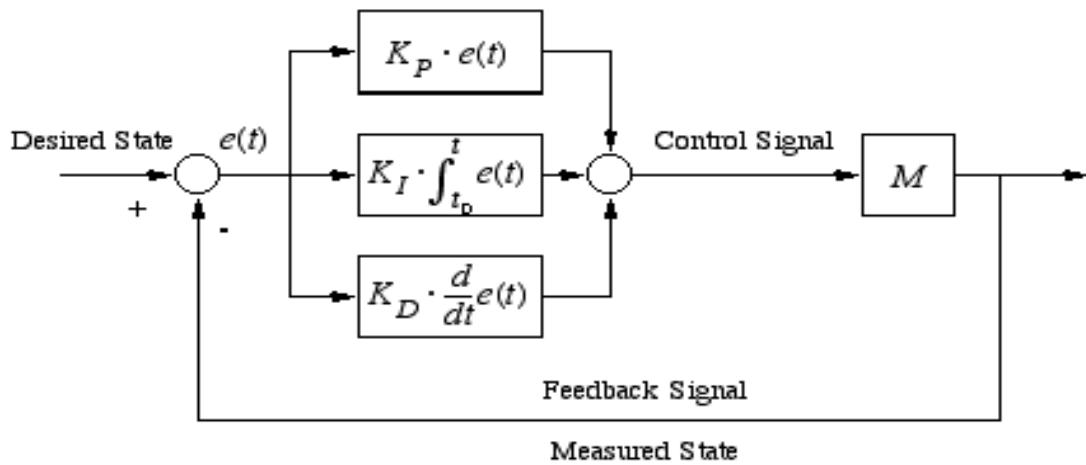


Figure 3.6: Basic block diagram of PID controller

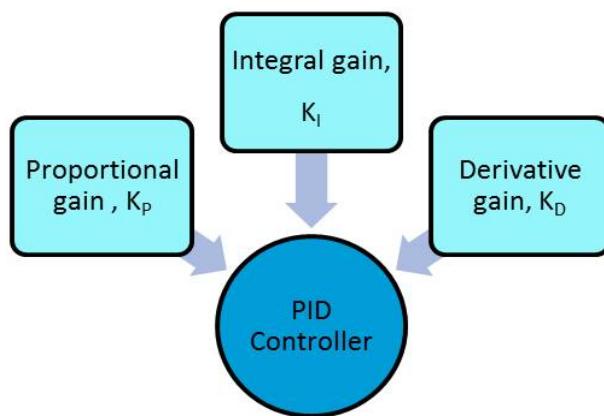


Figure 3.7: PID controller at a glance

In Figure 3.6, the basic block diagram of a PID controller system, Figure 3.7 illustrates a PID controller at a glance and in Figure 3.8 a system response curve is shown.

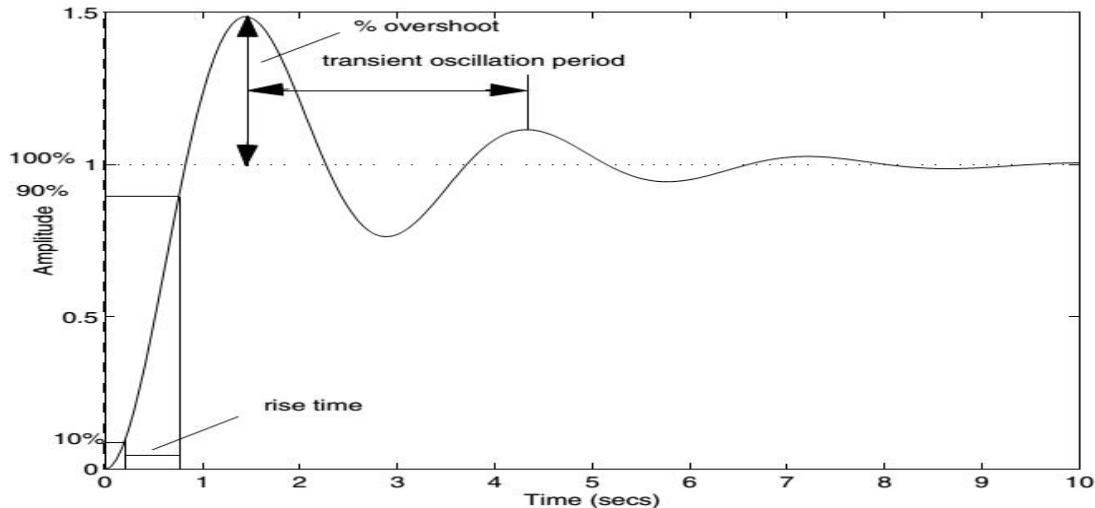


Figure 3.8: System response curve

Proportional Response (K_p)

The proportional component depends only on the difference between the set point and the process variable. This difference is referred to as the Error term. The proportional gain (K_p) determines the ratio of output response to the error signal. For instance, if the error term has a magnitude of 10, a proportional gain of 5 would produce a proportional response of 50. In general, increasing the proportional gain will increase the speed of the control system response. However, if the proportional gain is too large, the process variable will begin to oscillate. If K_p is increased further, the oscillations will become larger and the system will become unstable and may even oscillate out of control.

Integral Response (K_i)

The integral component sums the error term over time. The result is that even a small error term will cause the integral component to increase slowly. The integral response will continually increase over time unless the error is zero, so the effect is to drive the Steady-State error to zero. Steady-State error is the final difference between the process variable and set point. A phenomenon called integral windup results when integral action saturates a controller without the controller driving the error signal toward zero.

Derivative Response (K_d)

The derivative component causes the output to decrease if the process variable is increasing rapidly. The derivative response is proportional to the rate of change of the process variable. Increasing the derivative gain K_d parameter, will cause the control system to react more strongly to changes in the error term and will increase the speed of the overall control system response. Most practical control systems use very small derivative gain (K_d), because the Derivative Response

is highly sensitive to noise in the process variable signal. If the sensor feedback signal is noisy or if the control loop rate is too slow, the derivative response can make the control system unstable.

3.4.1 PID control of Plant

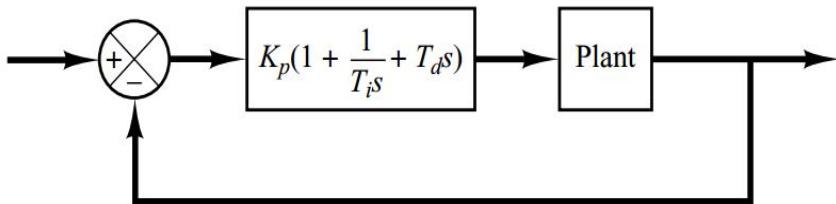


Figure 3.9: PID control of a plant

Figure 3.9 shows a PID control of a plant. If a mathematical model of the plant can be derived, then it is possible to apply various design techniques for determining parameters of the controller that will meet the transient and steady-state specifications of the closed-loop system. However, if the plant is so complicated that its mathematical model cannot be easily obtained, then an analytical or computational approach to the design of a PID controller is not possible. Then we must resort to experimental approaches to the tuning of PID controllers.

The process of selecting the controller parameters to meet given performance specifications is known as controller tuning. Ziegler and Nichols suggested rules for tuning PID controllers (meaning to set values and) based on experimental step responses or based on the value of that results in marginal stability when only proportional control action is used. Ziegler Nichols rules, which are briefly presented in the following, are useful when mathematical models of plants are not known. (These rules can, of course, be applied to the design of systems with known mathematical models.) Such rules suggest a set of values of and that will give a stable operation of the system. However, the resulting system may exhibit a large maximum overshoot in the step response, which is unacceptable. In such a case we need series of fine tunings until an acceptable result is obtained. In fact, the Ziegler Nichols tuning rules give an educated guess for the parameter values and provide a starting point for fine tuning, rather than giving the final settings for and in a single shot.

3.4.2 Ziegler Nichols Rules for Tuning PID Controllers

Ziegler and Nichols proposed rules for determining values of the proportional gain, K_p integral time, T_i and derivative time T_d based on the transient response characteristics of a given plant. Such determination of the parameters of PID controllers or tuning of PID controllers can be made by engineers on site by experiments on the plant.

There are two methods called Ziegler Nichols tuning rules: the first method and the second

method.[4] A brief presentation of the 2nd method is given here.

In the 2nd method first the $T_i=\infty$ and $T_d=0$. has been set.

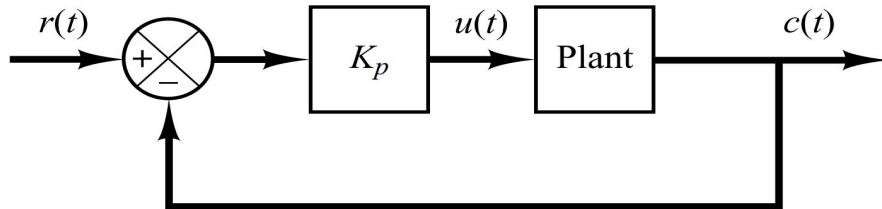


Figure 3.10: Closed loop system with a proportional controller

Using the proportional control action only (Figure 3.10) , increase K_p from 0 to a critical value K_{cr} at which the output first exhibits sustained oscillations. (If the output does not exhibit sustained oscillations for whatever value K_p may take , then this method does not apply.) Thus, the critical gain K_{cr} and the corresponding period of oscillation P_{cr} are experimentally determined. (Figure: 3.11)

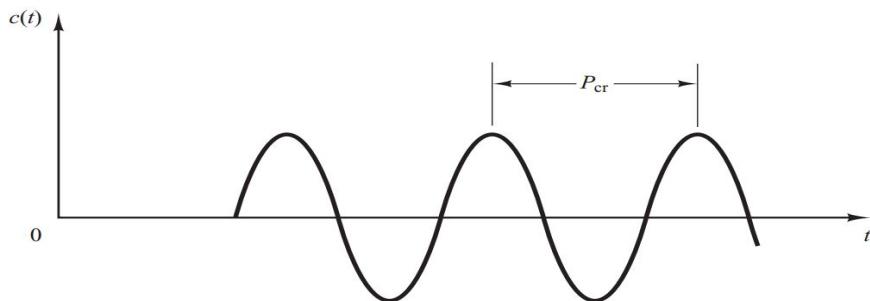


Figure 3.11: Sustained oscillation with period P_{cr} (P_{cr} is measured in seconds)

Ziegler and Nichols suggested that we set the values of the parameters K_p , T_i and T_d and according to the formula shown in Table 4.1

Type of Controller	K_p	T_i	T_d
P	$0.5K_{cr}$	∞	0
PI	$0.45K_{cr}$	$\frac{1}{1.2}P_{cr}$	0
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

Table 3.1: Ziegler Nichols Tuning rule based on Critical Gain K_{cr} and Critical Period P_{cr}

3.5 Design and simulation of PID controller

First of all the behavior of the system is simulated in MATLAB, using the system transfer function. After that the PID controller is designed and simulated in MATLAB to visualize the controlled system response with the changing value of gains. The assumed value of the system parameters are-

M : mass of the cart 0.5 kg

m : mass of the pendulum 0.2 kg

b : coefficient of friction for cart 0.1 N/m/sec

l : length to pendulum center of mass 0.3 m

I : mass moment of inertia of the pendulum 0.006 kg.m²

By putting the values of the system parameters in equation 4.18, the transfer function of pendulum cart system is -

$$(3.31) \quad \frac{\phi(s)}{F(s)} = \frac{1.405 \times 10^{-5} s}{2.3 \times 10^{-6} s^3 + 4.182 \times 10^{-7} s^2 - 7.172 \times 10^{-5} s - 1.025 \times 10^{-5}}$$

Figure 3.12 illustrates the SIMULINK model of our MATLAB simulation . The system transfer

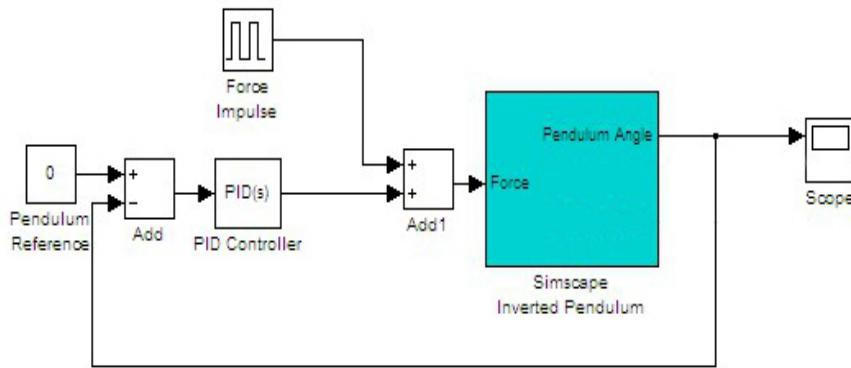


Figure 3.12: SIMULINK model of PID controlled cart pendulum system

function is used to define the system plant. A set value 0 is used for pendulum reference which indicates the vertically upright position of the pendulum.

At first the system behavior is observed without applying and controller gain.The behavior is illustrated in figure 3.13 , that represents a totally unstable system.

Applying feedback The PID tuning has been started with $K_p = 0$, $K_i = 0$ & $K_d = 0$

Figure 3.14 illustrates the response , which also represents an unstable system after few seconds.

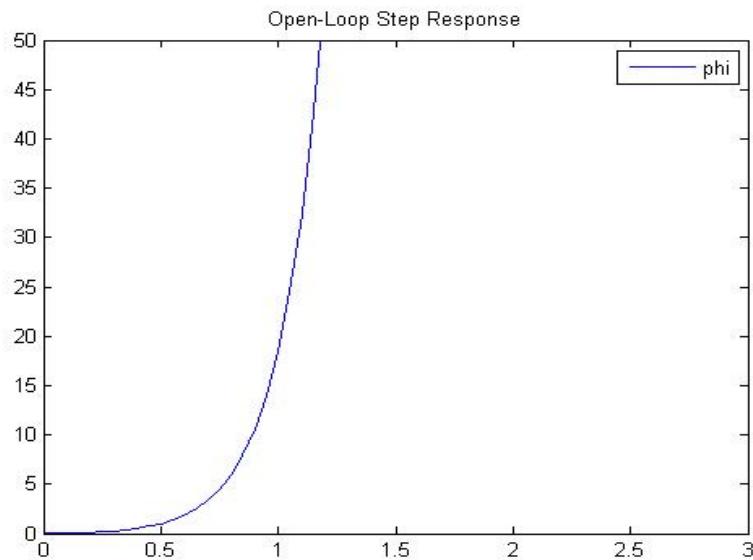


Figure 3.13: System response without feedback and controller gain

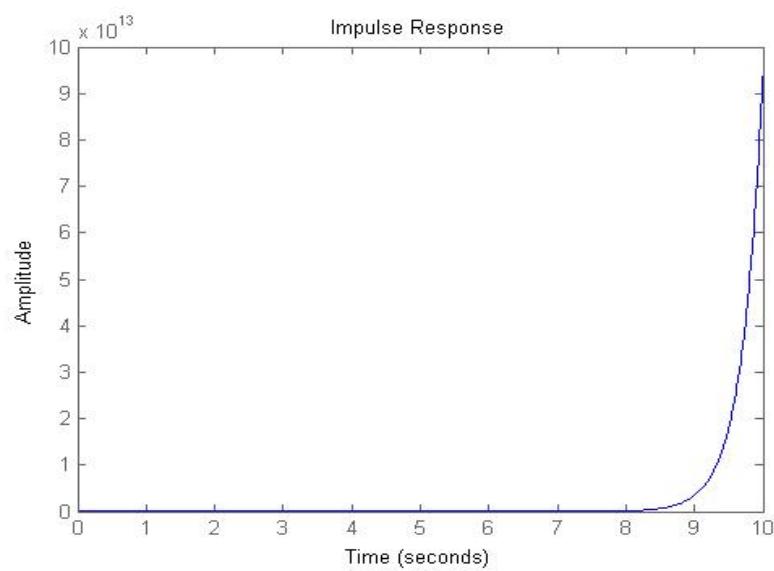


Figure 3.14: System response with $K_p = 0$, $K_i = 0$ & $K_d = 0$

Keeping $K_i = 0$ & $K_d = 0$, first K_p is tuned. The system response for $K_p = 100$, $K_i = 0$ & $K_d = 0$ is shown in figure 3.15, which represents a oscillatory system with with faster response. Further increase of K_p value makes the systems response more oscillatory. So, the K_p value is tuned at 100.

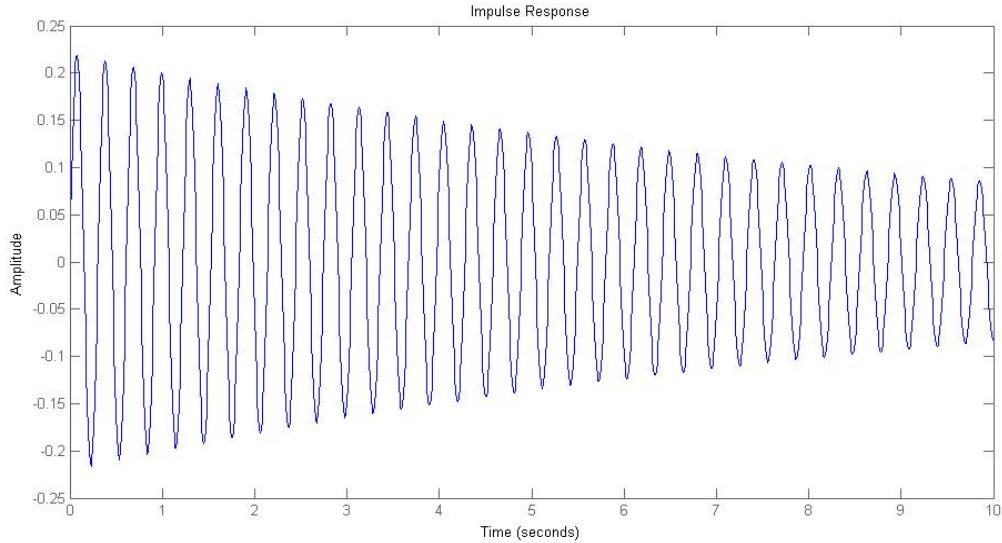


Figure 3.15: System response with $K_p = 100$, $K_i = 0$ & $K_d = 0$

After tuning the value of K_p , derivative gain K_d is tuned to reduce overshoot and settling time. Figure 3.16 illustrates the system response for $K_p = 100$, $K_i = 0$ & $K_d = 1$.

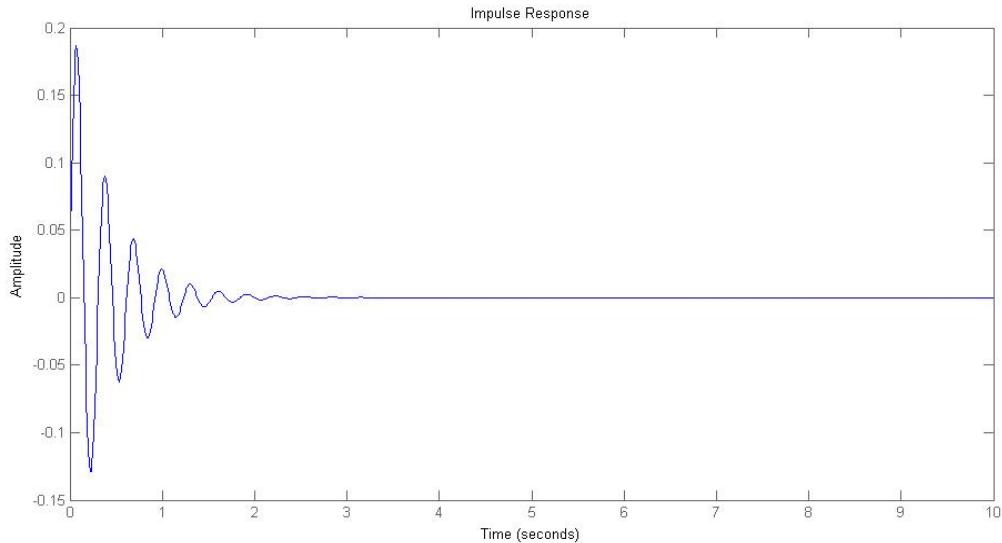


Figure 3.16: System response with $K_p = 100$, $K_i = 0$ & $K_d = 1$

After some more increase of K_d value, the value is set at 20. Finally applying $K_i=1$, the

complete PID tuning is $K_p = 100$, $K_i = 1$ & $K_d = 20$. Figure 3.17 illustrates the response.

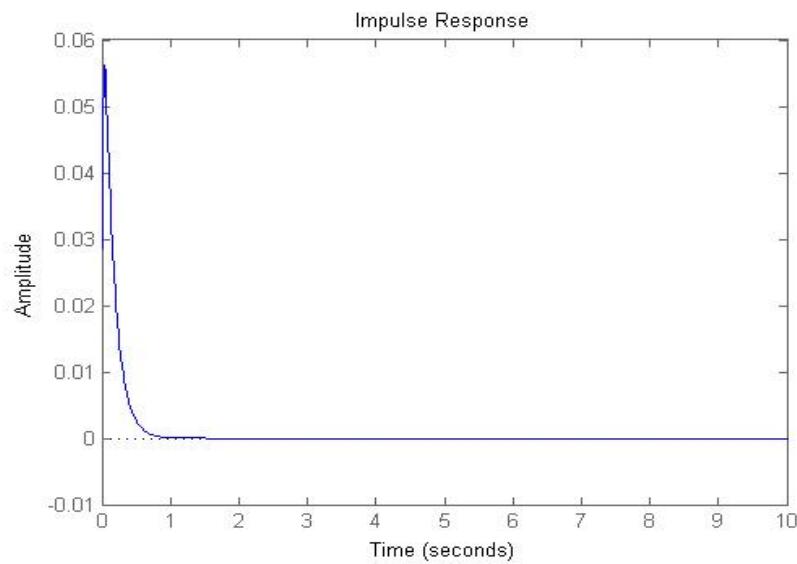


Figure 3.17: System response with $K_p = 100$, $K_i = 1$ & $K_d = 20$

The system response , for $K_p = 100$, $K_i = 1$ & $K_d = 20$ represents a quiet stable system. So, the tuning process is ended with these values.

3.6 Prototype development

Two prototypes has been developed for the experimental study of linear inverted Pendulum, one is pendulum on a cart(Figure: 3.18(b)) and the other is two wheel self balancing vehicle(Figure: 3.18(a)).

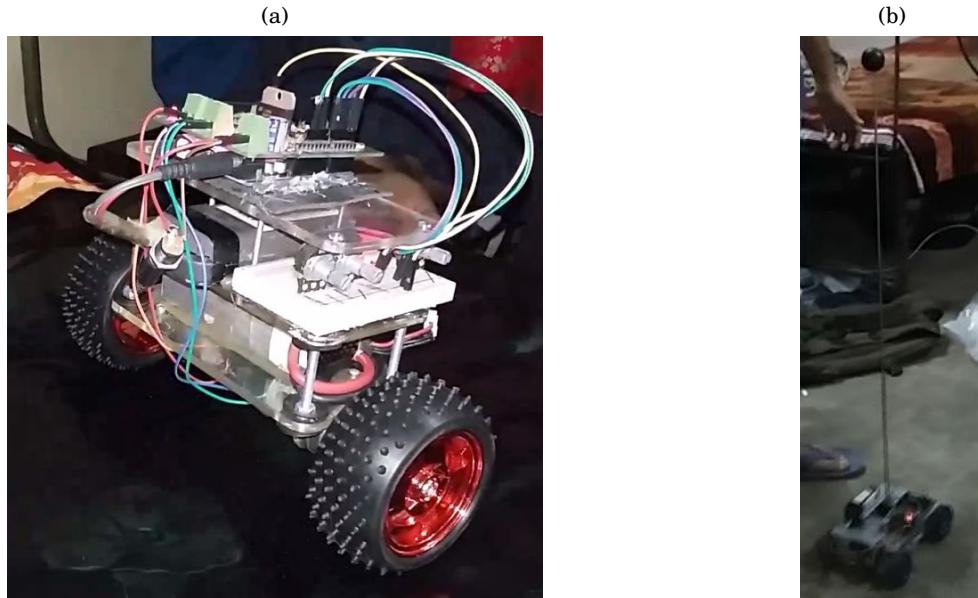


Figure 3.18: (a) Two wheel self balancing vehicle experimental prototype (b) Cart-pendulum experimental prototype

Pendulum on cart consists of a thin vertical rod attached at the bottom, referred to as pivot point mounted on a mobile toy car. The car, depending upon the direction of the deflection of the pendulum moves horizontally in order to bring the pendulum to absolute rest. A potentiometer is used as feedback sensor. The variation in its resistance causes change in voltage and afterward, it was compared with the reference voltage to generate the appropriate control signal. The comparison or difference between the reference and the potentiometer generates control signal is used to drive the system. Potentiometer used in this setup is illustrated in figure 3.19.



Figure 3.19: Potentiometer(pot)

In two wheel self balancing vehicle, MPU6050 Gyroscopic sensor is used to give feedback. The



Figure 3.20: MPU 6050 gyroscopic sensor

sensor continuously gives roll, pitch and yaw angles. For our experiment we used only roll angle to measure the angle produced by the vehicle from vertical plane. The angle is fed back to the controller to generate control signal to drive the motor at appropriate direction and speed to keep the vehicle in upright position.

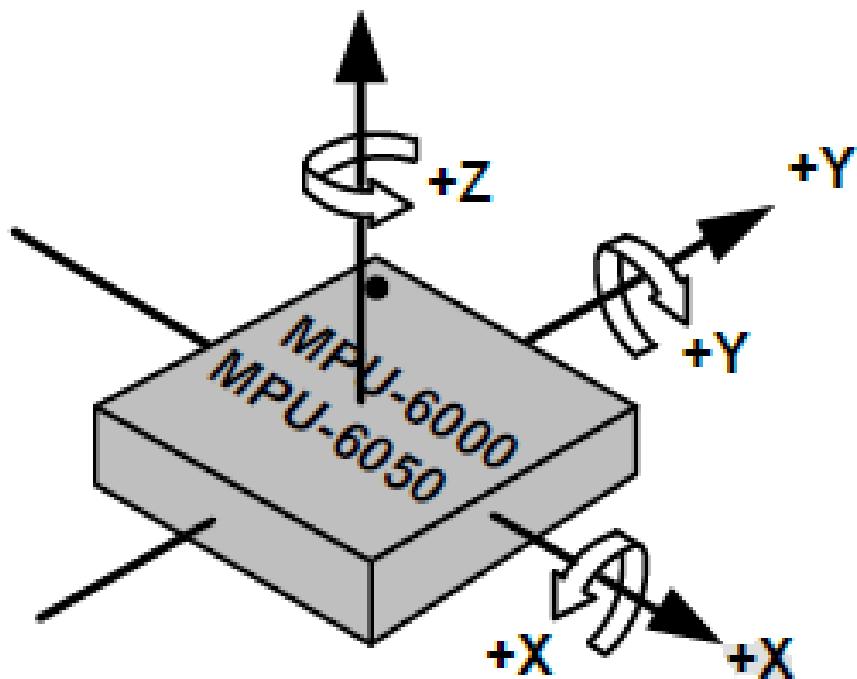


Figure 3.21: Roll, Pitch and yaw angle in gyroscope

Arduino UNO development board was used (Figure 3.21) as the brain of the controller. The specifications are given in Table 3.2

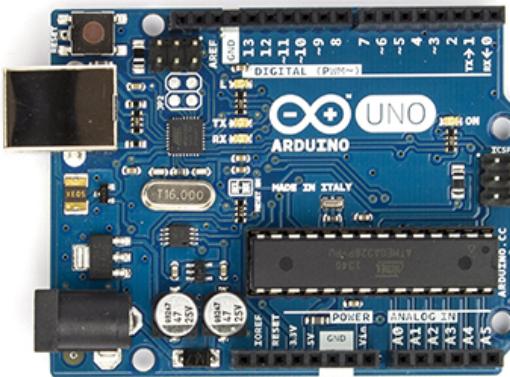


Figure 3.22: Arduino UNO development Board

Content	Specification
Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage	7 – 12V
Digital I/O pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC current Per I/O Pin	40 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

Table 3.2: Arduino UNO development Board Specifications

Other components are Motor Driver for Arduino UNO, Potentiometer for on board PID tuning, LiPo Battery, Bread board, Connecting wires etc.

To provide sufficient damping we used wheels with rubber spikes.

3.7 Experimental result: Prototype Response

Applying PID control to our experimental prototype of two wheel self balancing vehicle, the actual response is shown in figure 3.2

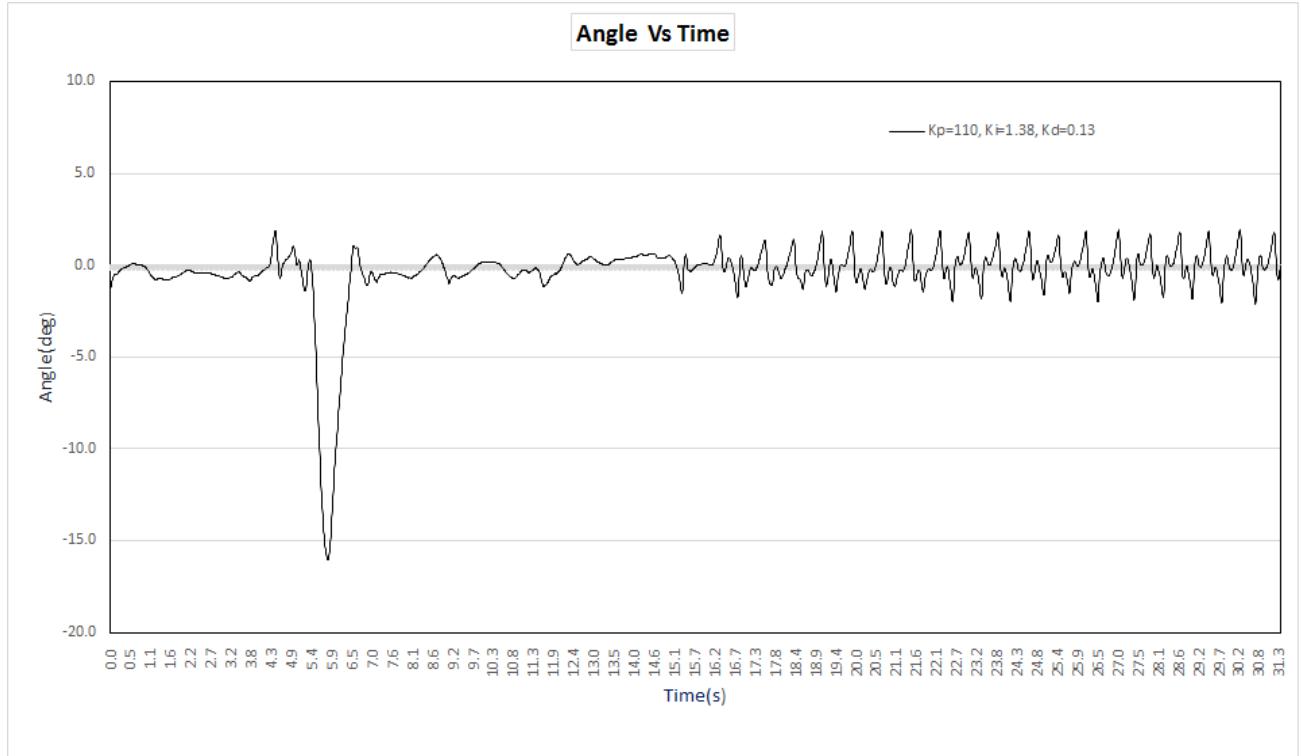


Figure 3.23: Actual response of prototype for tuned PID values of $K_p = 110$, $K_i = 1.38$ & $K_d = 0.33$

In our *simulink* model, we didn't consider the disturbance. But disturbance do exists in the real model. Hence some deviations from the simulation plot is observed.

C H A P T E R 4

ROTARY INVERTED PENDULUM SYSTEM

4.1 Mathematical Modeling



Figure 4.1: *QUANSER* Rotary Inverted Pendulum

Rotary inverted pendulum has a pendulum which has it's one end pivoted to a rotated disc or bar. The disc or bar needs to be rotated to provide the torque required to keep the pendulum in upright position. For this part of the thesis a rotary type inverted pendulum (ROTPEN) is used. It's made by *QUANSER*. The experiment was carried out in different steps. First a mathematical model was established for the system. A controller was designed to balance the ROTPEN system. The designed controller was verified in a simulation environment. And finally the controller is implemented in the model.

4.1.1 Model Conventions

The free body Diagram is shown in the following figure:

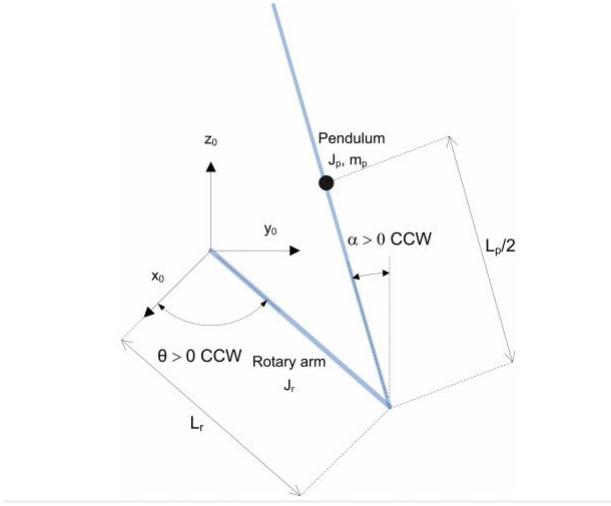


Figure 4.2: Free body diagram of rotary inverted pendulum

The rotary arm is pivoted with a servo motor(SRV02) system and is actuated to balance the pendulum. Various pendulum specifications are shown in table

Variable	Value	Unit
m_p	0.127	kg
L_p	0.337	m
J_p	0.0012	kg.m ²
B_p	0.0024	N.m.s/rad
m_r	0.257	kg
L_r	0.2159	m
J_r	9.98×10^{-4}	kg.m ²
R_m	2.6	Ω
K_g	70	
k_m	0.0077	V/(rad/s)
k_t	0.0077	N-m/A
g	9.8	kg.m/s ⁻²
η_g	0.9	
η_m	0.69	

Table 4.1: Rotary Inverted pendulum specification[1]

4.1.2 Equation of motion

Lagrange method[5] is used to find the equations of motion of the system. The motions of the rotary arm and the pendulum with respect to servo motor voltage is described by these EOMs.

$$(4.1) \quad \frac{\partial^2 L}{\partial t \partial q_i} - \frac{\partial L}{\partial \dot{q}_i} = \dot{Q}_i$$

The variables q_i are called the generalized coordinates. For this system let

$$q(t)^T = [\theta(t) \alpha(t)]$$

With the generalized coordinates defined, the Euler-Lagrange equations for the rotary inverted pendulum system are

$$(4.2) \quad \frac{\partial^2 L}{\partial t \partial \dot{\theta}} - \frac{\partial L}{\partial \theta} = Q_1$$

$$(4.3) \quad \frac{\partial^2 L}{\partial t \partial \dot{\alpha}} - \frac{\partial L}{\partial \alpha} = Q_2$$

The Lagrangian of the system is described as

$$L = T - V$$

Where T is the total kinetic energy and V is the total potential energy of the system. The generalized forces Q_i are used to describe non-conservative forces(e.g. friction) applied to the system with respect to generalized co-ordinates. for this case non-conservative force acting on the rotary arm is

$$Q_1 = \tau - B_r \dot{\theta}$$

and acting on the pendulum is

$$Q_2 = -B_p \dot{\alpha}$$

After the Lagrangian of the system is found and various derivatives are calculated to find the following EOMs from equation (4.2) and (4.3).

$$(4.4) \quad \left(m_p L_r^2 + \frac{1}{4} m_p L_p^2 - \frac{1}{4} m_p L_p^2 \cos(\alpha)^2 + J_r \right) \ddot{\theta} - \left(\frac{1}{2} m_p L_p L_r \cos(\alpha) \right) \ddot{\alpha} + \left(\frac{1}{2} m_p L_p^2 \sin(\alpha) \cos(\alpha) \right) \dot{\theta} \dot{\alpha} + \left(\frac{1}{2} m_p L_p L_r \sin(\alpha) \right) \dot{\alpha}^2 = \tau - B_t(\theta)$$

$$(4.5) \quad -\frac{1}{2} m_p L_p L_r \cos(\alpha) \ddot{\theta} + \left(J_p + \frac{1}{4} m_p L_p^2 \right) \ddot{\alpha} - \frac{1}{4} m_p L_p^2 \cos(\alpha) \sin(\alpha) \dot{\theta}^2 - \frac{1}{2} m_p L_p g \sin(\alpha) = -B_p \alpha$$

The torque applied at the base of the rotary arm is generated by a servo motor which is described by the equation

$$(4.6) \quad \tau = \frac{\eta_g K_g \eta_m k_t (V_m - K_g K - m \dot{\theta})}{R_m}$$

4.1.3 Linearizing

Initial condition for all the variables are taken zero. $\theta = 0^\circ, \alpha = 0^\circ, \dot{\theta} = 0, \dot{\alpha} = 0$ for a multivariable system control variable z is defined

$$z^T = [z_1, z_2]$$

and if $f(z)$ is to be linearized about the operating point

$$z_0^T = [a, b]$$

the linearized function can be written as

$$(4.7) \quad f_{lin} = f(z_0) + \left(\frac{\partial f(z)}{\partial z_1} \right) \Big|_{z=z_0} (z_1 - a) + \left(\frac{\partial f(z)}{\partial z_2} \right) \Big|_{z=z_0} (z_2 - b)$$

Applying this linearization method and taking the left hand side of 4.4 and 4.5 as

$$f(z)$$

we get the linearized form of the EOMs.

$$(4.8) \quad (m_p L_r^2 + J_r) \ddot{\theta} - \frac{1}{2} m_p L_p L_r \ddot{\alpha} = \tau - B_r \dot{\theta}$$

$$(4.9) \quad -\frac{1}{2} m_p L_p L_r \ddot{\theta} + \left(J_p + \frac{1}{4} m_p L_p^2 \right) \ddot{\alpha} - \frac{1}{2} m_p L_p g \alpha = -B_p \dot{\alpha}$$

4.8 and 4.9 can be written in matrix form as follows

$$(4.10) \quad \begin{vmatrix} m_p L_r^2 + J_r & -\frac{1}{2} m_p L_p L_r \\ -\frac{1}{2} m_p L_p L_r & J_p + \frac{1}{4} m_p L_p^2 \end{vmatrix} \begin{vmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{vmatrix} + \begin{vmatrix} B_r & 0 \\ 0 & B_p \end{vmatrix} \begin{vmatrix} \dot{\theta} \\ \dot{\alpha} \end{vmatrix} = \begin{vmatrix} \tau \\ -\frac{1}{2} m_p L_p g \alpha \end{vmatrix}$$

4.1.4 State Space Model of the system

The linear state space equations are

$$(4.11) \quad \begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned}$$

here x is the state and u is the control input. A, B, C and D are the state-space matrices. For the rotary inverted pendulum system the state and the output are defined

$$(4.12) \quad x^T = [\theta \quad \alpha \quad \dot{\theta} \quad \dot{\alpha}]$$

and

$$(4.13) \quad y^T = [x_1 \quad x_2]$$

In the output equation only the position of the servo and link angles are being measured. Based on this C and D matrices of the output equations are

$$(4.14) \quad \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{vmatrix}$$

and

$$(4.15) \quad D = \begin{matrix} 0 \\ 0 \end{matrix}$$

From equation (4.10)

$$\begin{vmatrix} m_p L_r^2 + J_r & -\frac{1}{2} m_p L_p L_r \\ -\frac{1}{2} m_p L_p L_r & J_p + \frac{1}{4} m_p L_p^2 \end{vmatrix} \begin{vmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{vmatrix} = \begin{vmatrix} \tau - B_r \dot{\theta} \\ \frac{1}{2} m_p L_p g \alpha - B_p \dot{\alpha} \end{vmatrix}$$

solving for the acceleration terms-

$$\begin{vmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{vmatrix} = \frac{1}{J_T} \begin{vmatrix} J_p + \frac{1}{4} m_p L_p^2 & \frac{1}{2} m_p L_p L_r \\ \frac{1}{2} m_p L_p L_r & J_r + m_p L_r^2 \end{vmatrix}$$

here, $J_T = (J_p m_p L_r^2 + J_r J_p + \frac{1}{4} J_r m_p L_p^2)$ we find the following acceleration expressions form the above

$$(4.16) \quad \ddot{\theta} = \frac{1}{J_T} \left(-\left(J_p + \frac{1}{4} m_p L_p^2 \right) B_r \dot{\theta} - \frac{1}{2} m_p L_p L_r B_p \dot{\alpha} + \frac{1}{4} m_p^2 L_p^2 L_r g \alpha + \left(J_p + \frac{1}{4} m_p L_p^2 \right) \tau \right)$$

$$(4.17) \quad \ddot{\alpha} = \frac{1}{J_T} \left(\frac{1}{2} m_p L_p L_r B_r \dot{\theta} - (J_r + m_p L_r^2) B_p \dot{\alpha} + \frac{1}{2} m_p L_p g (J_r + m_p L_r^2) \alpha + m_p L_p L_r \tau \right)$$

form (reftwelve) $\dot{x}_1 = x_3$ and $\dot{x}_2 = x_4$ substituting x in the acceleration terms the space matrices A and B can be calculated

$$(4.18) \quad A = \frac{1}{J_T} \begin{vmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{1}{4} m_p^2 L_p^2 L_r g & -(J_p + \frac{1}{4} m_p L_p^2) B_r & -\frac{1}{2} m_p L_p L_r B_p \\ 0 & \frac{1}{2} m_p L_p g (J_r + m_p L_r^2) & \frac{1}{2} m_p L_p L_r B_r & -(J_r + m_p L_r) B_p \end{vmatrix}$$

$$(4.19) \quad B = \frac{1}{J_T} \begin{vmatrix} 0 \\ 0 \\ J_p + \frac{1}{4} m_p L_p^2 \\ \frac{1}{2} m_p L_p L_r \end{vmatrix}$$

As the A, B, C and D are known now we know the complete state space model of the system. putting the values of from fig:4.1.1 we get the exact values of state space matrices. Eigen value of matrix A gives the open loop poles of the system. The obtained results are shown in figure 5.3

Description	Symbol	Value	Units
State-Space Matrix	A	$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 80.3 & -45.8 & -0.930 \\ 0 & 122 & -44.1 & -1.40 \end{bmatrix}$	
State-Space Matrix	B	$\begin{bmatrix} 0 \\ 0 \\ 83.4 \\ 80.3 \end{bmatrix}$	
State-Space Matrix	C	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$	
State-Space Matrix	D	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$	
Open-loop poles	OL	{-48.42, 7.06, -5.86, and 0}	

Table 4.2: State-Space Model Summary

4.2 Pole Placement Method

From table: 4.2 and figure: 4.3 it is seen that there is a positive pole of the system that represents the inherent instability of the system.

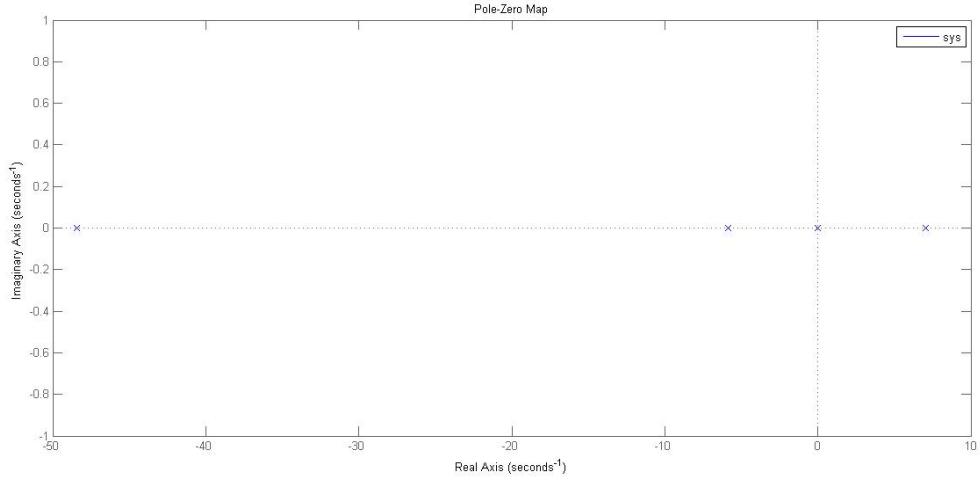


Figure 4.3: Open Loop Poles and their position

4.2.1 Controllability

The system is controllable if the rank of the controllability matrix
 $T = \begin{bmatrix} B & AB & A^2B & A^3B & A^4B \end{bmatrix}$
 equals the number of states in the system (e.g. 4) The system is checked to be controllable.

4.2.2 Algorithm

Following algorithm is followed to calculate the control gain.[6]

step 01: The companion matrices \tilde{A} and \tilde{B} are calculated to find $W = T\tilde{T}^{-1}$

Step 02: \tilde{K} is calculated to assign the poles of $\tilde{A} - \tilde{B}\tilde{K}$ to the desired locations.

Step 03: $K = \tilde{K}W^{-1}$ is the desired controller gain.

4.2.3 Controller Design

The first controller is designed to meet the following specifications.

Specification 1: Damping ratio: $\zeta = 0.7$

Specification 2: Natural frequency: $\omega_n = 4$ rad/s.

Specification 3: Maximum pendulum angle deflection: $|\alpha| < 15$ deg.

Specification 4: Maximum control effort / voltage: $|V_m| < 10$ V.

from the assumed specification two dominant poles calculated are

$$p_1 = -\sigma + j\omega_d$$

and

$$p_2 = -\sigma - j\omega_d$$

Here, $\sigma = \zeta\omega$ and $\omega_d = \omega_n\sqrt{1-\zeta^2}$. Two dominant poles are calculated as $-2.80 \pm j2.86$ and two other poles are taken arbitrarily $p_3 = -30$ and $p_4 = -40$. Positions of the poles are shown in Figure: 4.4

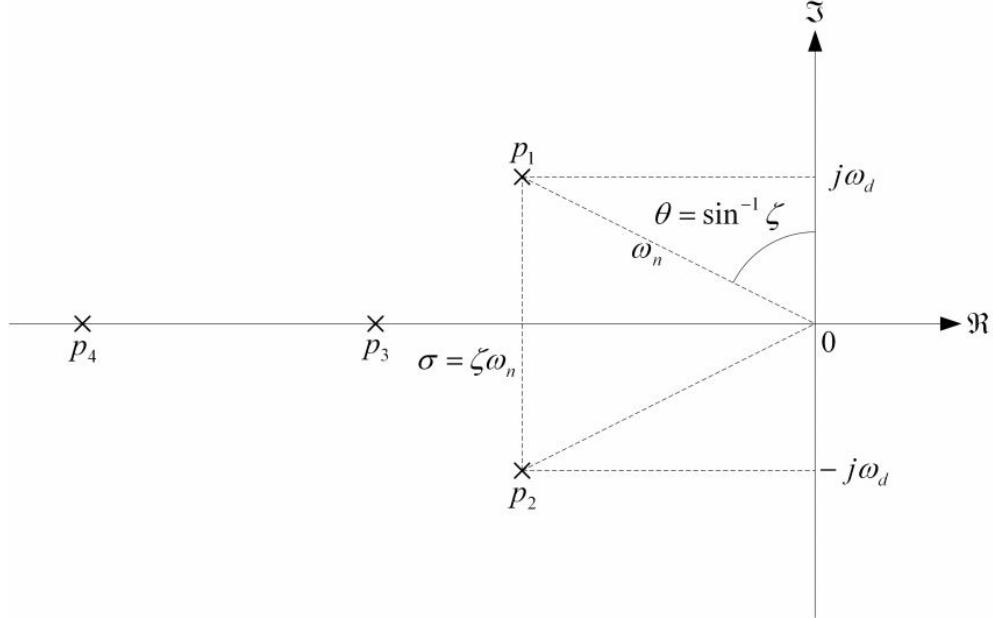


Figure 4.4: Desired closed-loop pole locations

From table:4.2 open loop polynomial is

$$s(s + 48.42)(s - 7.06)(s + 5.86) = s^4 + 47.22s^3 - 99.48s^2 - 2003.21s$$

Fitting the coefficients in matrix format

$$\tilde{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 2003.21 & 99.48 & -47.22 \end{bmatrix}$$

and

$$\tilde{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

The characteristic polynomial for desired closed loop poles is

$$(4.20) \quad (s + 2.80 - j2.86)(s + 2.80 + j2.86)(s + 30)(s + 40) = s^4 + 75.6s^3 + 1608s^2 + 7840s + 19200$$

when applying the control $u = -\tilde{K}x$ to the companion form it changes (\tilde{A}, \tilde{B}) to $(\tilde{A} - \tilde{B}\tilde{K}, \tilde{B})$.

$$\tilde{A} - \tilde{B}\tilde{K} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -k_1 & 2003.21 - k_2 & 99.48 - k_3 & -47.22 - k_4 \end{bmatrix}$$

The characteristic equation for this system is

$$s^4 + (47.22 + k_4)s^3 + (k_3 - 99.48)s^2 + (k_2 - 2003.21)s + k_1$$

Equating the coefficients of this polynomial and 4.20 gives

$$47.22 + k_4 = 75.6$$

$$k_3 - 99.48 = 1608$$

$$k_2 - 2003.21 = 7840$$

$$k_1 = 19200$$

so

$$\tilde{K} = \begin{bmatrix} 19200 & 9843 & 1707 & 29 \end{bmatrix}$$

Calculation of $W = T\tilde{T}^{-1}$ is done in *MATLAB*. Finally the control gain is found from $K = \tilde{K}W^{-1}$

$$K = \begin{bmatrix} -5.26 & 28.16 & -2.76 & 3.22 \end{bmatrix}$$

4.2.4 Simulation Result

MATLAB is used for numerical calculations and *simulink* is used as the simulation environment of the response of rotary inverted pendulum. The calculation process is similar as described in 5.2.3

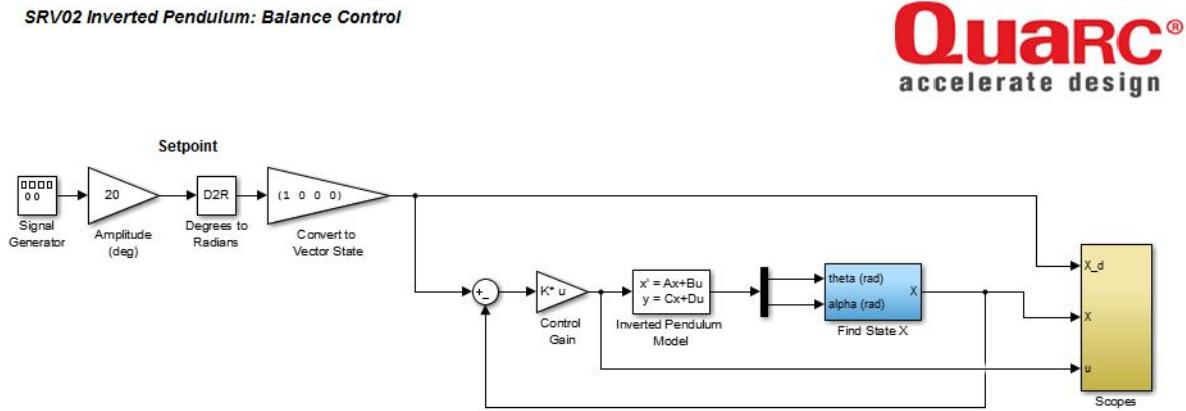


Figure 4.5: Control System in *MATLAB* with *QUARC*

Simulation result for pendulum angle(α), rotary arm angle(θ) and motor voltage(V_m) is shown in the following figure.

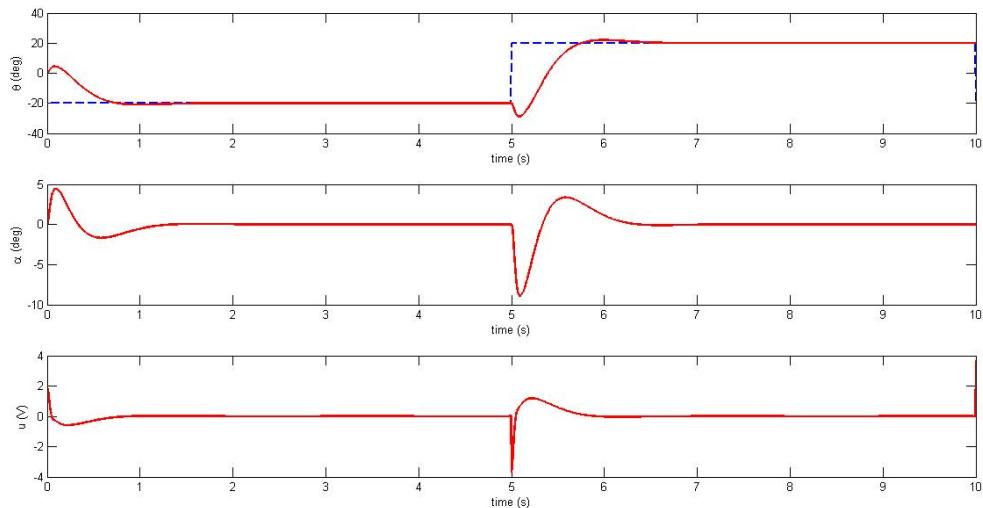


Figure 4.6: Step response of rotary inverted pendulum system

4.2.5 Prototype Response

QUANSER rotary inverted pendulum consists of a flat arm, or hub, with a pivot at one end and a metal shaft on the other end. The pivot-end can be mounted on top of the SRV02 load gear shaft and fastened with screws. The actual pendulum link is fastened onto the metal shaft and the shaft is instrumented with a sensor to measure its angle. The result is a horizontally rotating arm with a pendulum at the end. The ROTPEN is instrumented with an encoder(1024 line) to obtain a digital measurement of the pendulum and is free to rotate 360 degrees.

Different component of ROTPEN is shown in Figure: 4.7 and table 4.3

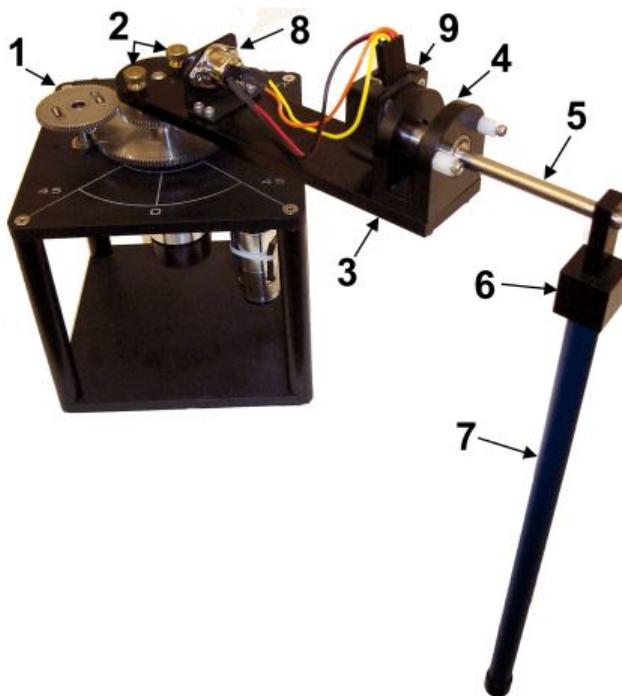


Figure 4.7: Components of *QUANSER* ROTPEN

ID	Component
1	SRV02
2	Thumbscrews
3	Rotary Arm
4	Shaft Housing
5	Shaft
6	Pendulum T-fitting
7	Pendulum Link
8	Pendulum Encoder Connector
9	Pendulum Encoder

Table 4.3: Components of *QUANSER* ROTPEN

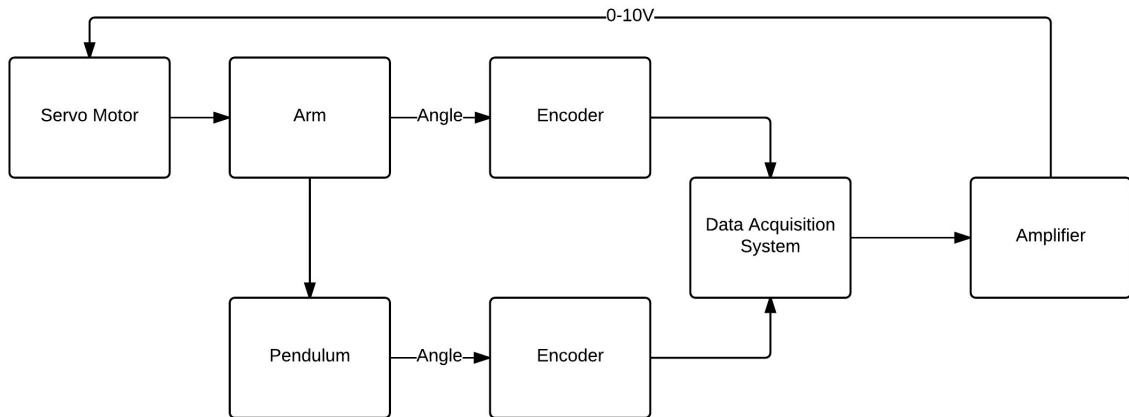


Figure 4.8: Flow Diagram of ROTPEN working principle

The ROTPEN is interfaced with *MATLAB* through a *QUANSER* made add-on called *QUARC*. It automatically uses the Control gain found in the process described in 5.2.3. Manual control gain input is also available. The response of the system is observed using scope block in *simulink*.

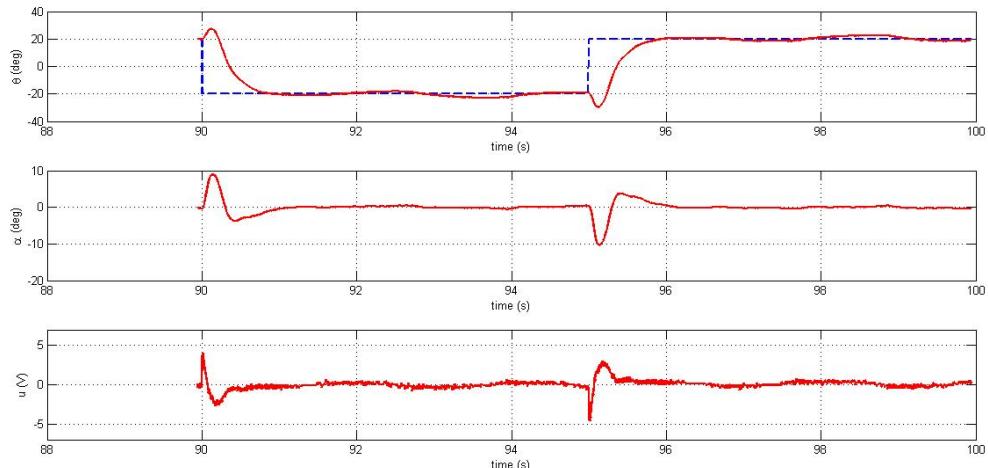


Figure 4.9: Step response of the prototype

Similarity with Simulation result from figure:4.6 can easily be observed.

4.3 Optimum Controller Design(LQR)

4.3.1 Why optimal control?

In the pole placement method 2 poles were chosen arbitrarily. It may be questioned that why poles are not taken with as negative real part as possible. Because it is known that poles with more negative real part means better response[7]. Here the limiting factor is the control effort. As the motor is of limited power it can provide force/torque within a certain range of value.

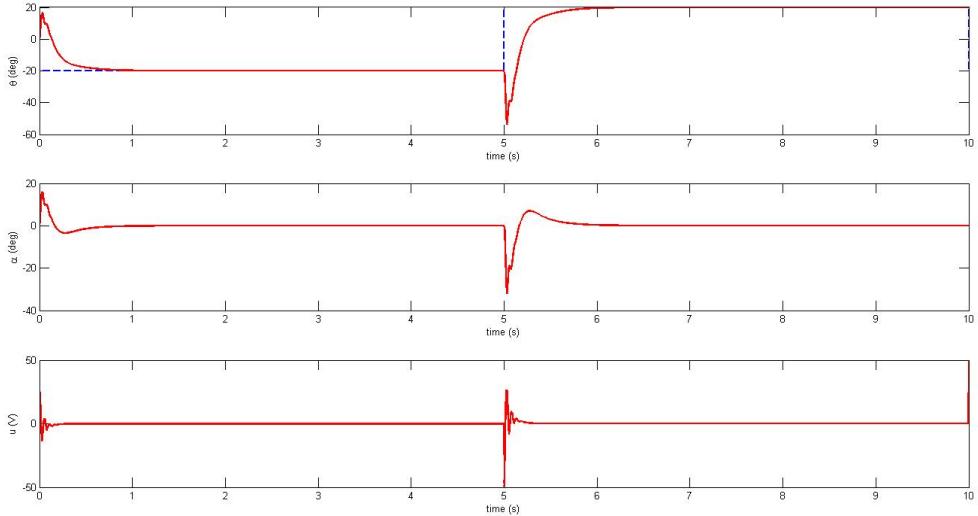


Figure 4.10: Response and Control Effort

Figure 4.10 shows fast response of the system. But to achieve this response around 100V motor is required. Which is not available in our case. Therefore Optimal control is necessary to minimize the control effort to obtain desired output[8]. For an optimal control method Linear Quadratic Regulator(LQR) Control is chosen. As LQR has shown better control than pole placement method.[9]

4.3.2 Linear Quadratic Regulator(LQR)

In LQR method a control effort u is calculated such that the performance criterion or cost function

$$(4.21) \quad J = \int_0^{\infty} [(x_{ref} - x(t))^T Q (x_{ref} - x(t)) + u(T)^T R u(t)] dt$$

is minimized. The design matrices Q and R hold the penalties on the deviations of state variables from their setpoint and the control actions, respectively.

Here Q is an $n \times n$ symmetric positive semidefinite matrix and R is an $m \times m$ symmetric positive definite matrix. Where n is the number of state variable and m is the number of control effort. When an element of Q is increased, therefore, the cost function increases the penalty associated with any deviations from the desired setpoint of that state variable, and thus the specific control gain will be larger. When the values of the R matrix are increased, a larger penalty is applied to the aggressiveness of the control action, and the control gains are uniformly decreased.[10]

The desired control gain can be found from $K_{lqr} = R^{-1}B^TP$ where P is the solution of Matrix Algebraic Riccati Equation(MARE)[11]

$$(4.22) \quad PA + A^T P + Q - PBR^{-1}B^T P = 0$$

4.3.3 Simulation Result

In this this experiment *MATLAB* fuction *lqr* is used to obtain design matrices(e.g. Q,R) directly.

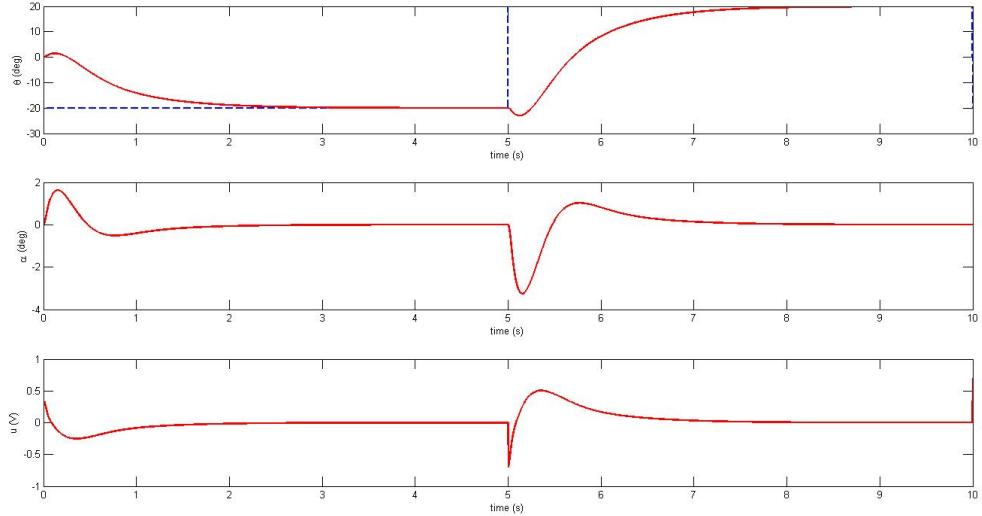


Figure 4.11: Starting value of LQR control

Figure 4.11 shows very poor step response. But only 1V of motor voltage is used. So staring with this values Q and R are recalibrated to make the step response as fast as possible within the control effort limit (10V).

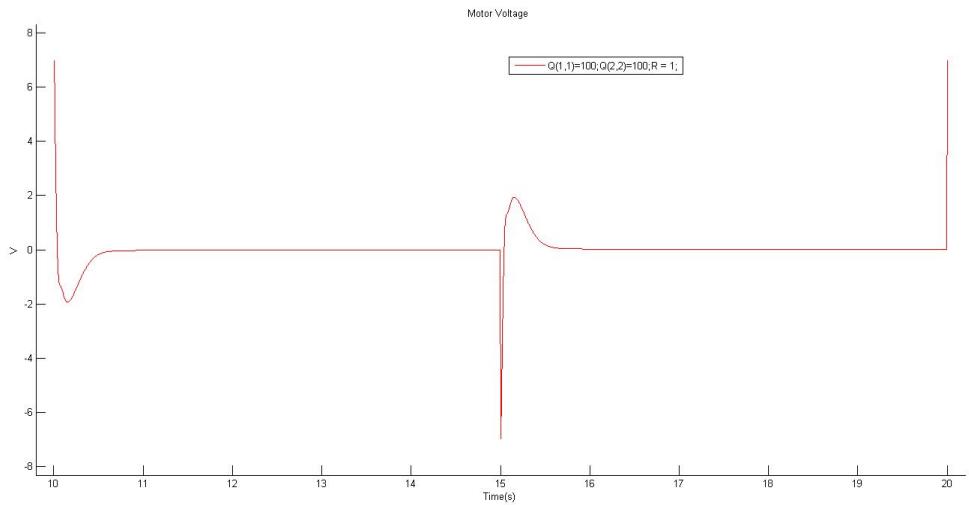


Figure 4.12: Step response within of ROTPEN within motor capacity

Pendulum and rotary arm response for this value of Design matrices are seen better than than pole placement method as expected.

Further calibration gives a wide range of response. The best design can be chosen as far specification.

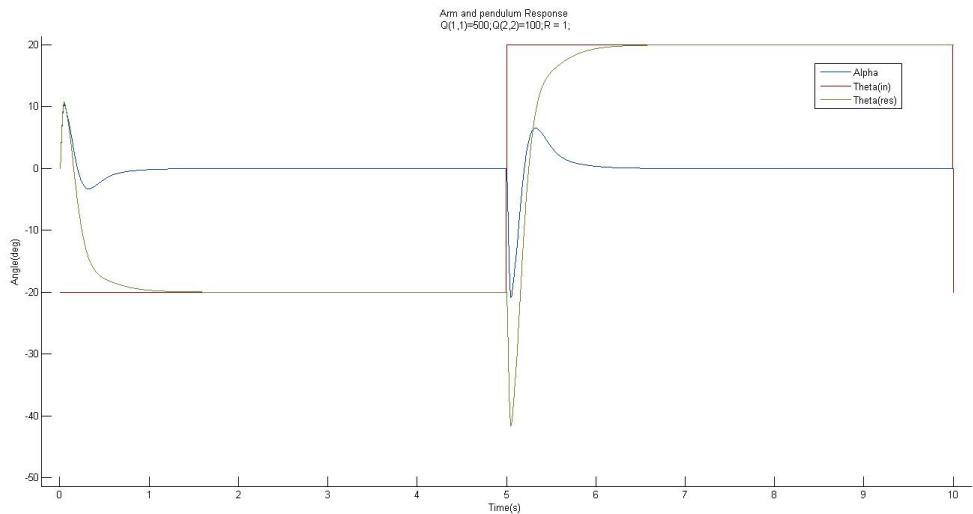


Figure 4.13: Pendulum and arm response for first controller

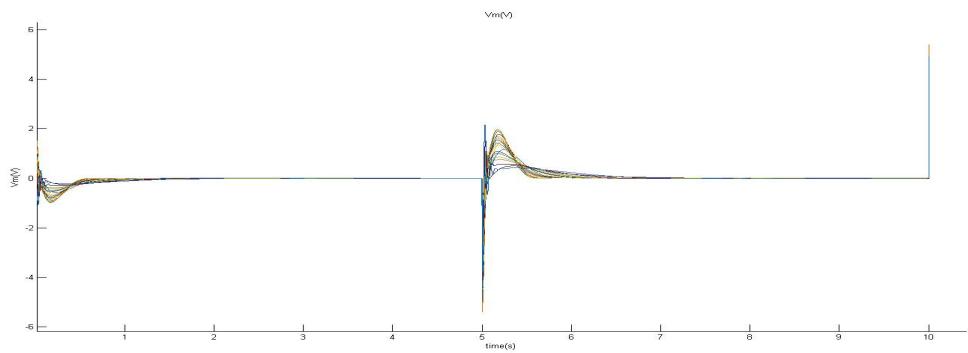


Figure 4.14: Applied motor voltage within the motor capacity

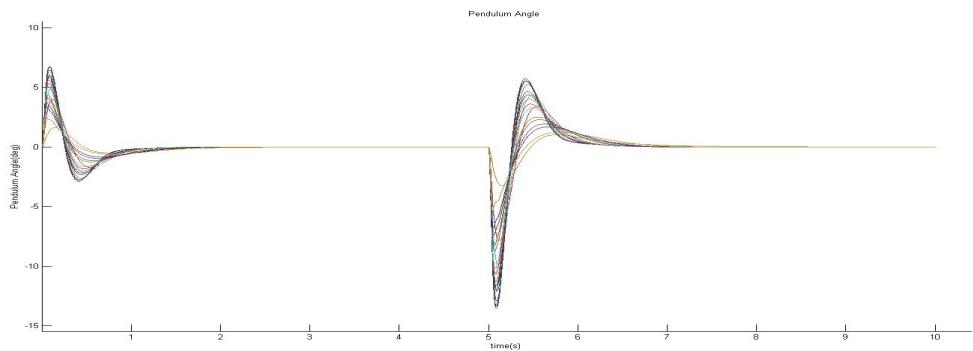


Figure 4.15: Pendulum angle within the control limit

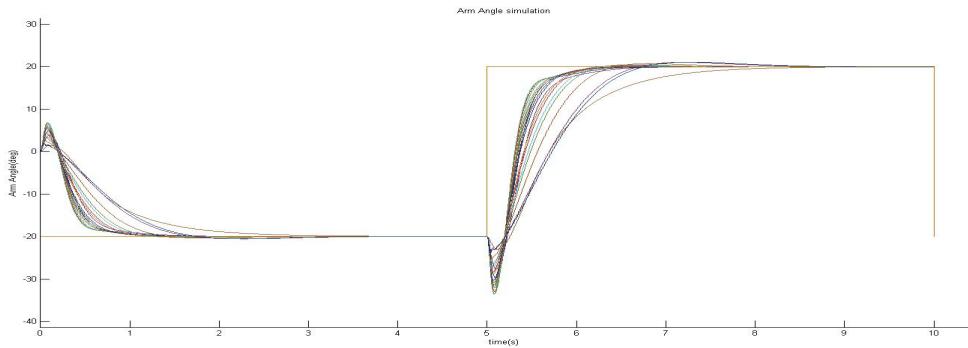


Figure 4.16: Arm Angle within the control limit

4.3.4 Advantage and disadvantages of LQR method

The advantages of LQR

- 1) Stability is guaranteed if
 - a) all of the states in the system available for feedback and
 - b) a really good model of the system is available. In fact, not only is stability guaranteed, but the stability margins are guaranteed.
- 2) The controller is automatically generated by simply selecting a couple of parameters.
However there are some cases when it is difficult to achieve desired control using LQR
 - 1) If the knowledge of system is primarily experimental. Then it is required to derive a model from the experimental data.
 - 2) If all of the states of the system can not be measured,
 - 3) If the system is incomplete (unmodeled dynamics), it may be difficult to get a controller that works the desired way.
 - 4) The parameters that are used to generate the controller are generally not directly related in an intuitive way to the requirements, but are rather "knobs" that you turn to get the desired effect.

C H A P T E R 5

CONCLUSION

Our goal was to study and implement different control methods to stabilize and balance the inverted pendulums systems. But we were able to implement the PID control and Pole placement method only. The LQR method is studied and simulated in MATLAB environment , but we could not implement it in our prototypes during our thesis period. So, there are lot of things can be done in future work.

The linear inverted pendulum prototypes are stabilized and balanced considering the Pendulum angle only. A brief study of Linear inverted pendulum can be done , considering both the Pendulum angle and cart / vehicle position. We applied Integer order PID controller to stabilize the pendulum. Fractional order PID can be used to achieve better result.[12]

In Rotary inverted pendulum Bryson's method can be used for a better LQR tuning.[13] we only stabilized the pendulum in vertically upright position. A swing up control experiment can be done with this experimental setup.[14]

Balancing can be studied with other modern controllers , ex. Fuzzy Controller, Neural Network etc. A comparative study of different controllers can also be done , to analyze which controller provides the best balancing.

BIBLIOGRAPHY

- [1] Quanser.
Rotary Inverted Pendulum-User Manual.
- [2] Inverted pendulum; from wikipedia, the free encyclopedia.
- [3] Inverted pendulum: System modeling; from control tutorials for matlab & simulink.
- [4] Katsuhiko Ogata.
Modern Control Engineering.
Pearson, 5th edition edition, 12 2010.
- [5] Olav Egeland and Jan Tommy Gravdahl.
Modeling and simulation for automatic control, volume 76.
Marine Cybernetics Trondheim, Norway, 2002.
- [6] Quanser.
Rotary Inverted Pendulum student workbook.
- [7] Norman S. Nise.
Control Systems Engineering.
Wiley, 6th edition binder ready version edition, 12 2010.
- [8] Shuang CONG, Dong-jun ZHANG, and Heng-hua WEI.
Comparative study on three control methods of the single inverted-pendulum system [j].
Systems Engineering and Electronics, 11:016, 2001.
- [9] Navid Razmjoooy, Ali Madadi, Hamid-Reza Alikhani, and Mahmood Mohseni.
Comparison of lqr and pole placement design controllers for controlling the inverted pendulum.
Journal of World, Äôs Electrical Engineering and Technology, 2322:5114, 2014.
- [10] Qube-servo lqr control workbook (student) published by-quanser inc.
- [11] Iwase Kobayashi and Furuta Suzuki.
Swinging-up and balancing control of double fututa pendulum by using state-dependent riccati equation.
Graduate School of Systems Engineering. Tokyo Denki University, 2001.
- [12] Majid Zamani, Masoud Karimi-Ghartemani, Nasser Sadati, and Mostafa Parniani.
Design of a fractional order pid controller for an avr using particle swarm optimization.
Control Engineering Practice, 17(12):1380–1387, 2009.

- [13] Arthur Earl Bryson.
Applied optimal control: optimization, estimation and control.
CRC Press, 1975.
- [14] Karl Johan Åström and Katsuhisa Furuta.
Swinging up a pendulum by energy control.
Automatica, 36(2):287–295, 2000.