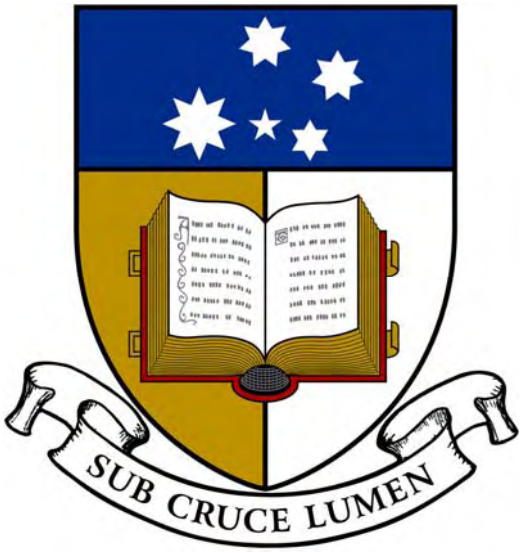

Control of a Ball and Beam System

Wei Wang

School of Mechanical Engineering

The University of Adelaide

South Australia 5005

AUSTRALIA

Submitted for the degree of Advanced Master on the 5rd June, 2007

Abstract

The ball and beam system can usually be found in most university control labs since it is relatively easy to build, model and control theoretically. The system includes a ball, a beam, a motor and several sensors. The basic idea is to use the torque generated from motor to the control the position of the ball on the beam. The ball rolls on the beam freely. By employing linear sensing techniques, the information from the sensor can be taken and compared with desired positions values. The difference can be fed back into the controller, and then in to the motor in order to gain the desired position. The mathematical model for this system is inherently nonlinear but may be linearized around the horizontal region. This simplified linearised model, however, still represents many typical real systems, such as horizontally stabilizing an airplane during landing and in turbulent airflow. By considering real plant problems such as the sensor noise and actuator saturation, the controllers of the system become more efficient and robust.

In this thesis, a theoretical analysis of the ball and beam system is conducted by employing physical laws and linear approximation. A performance comparison among the different control techniques is presented in both simulation and experimental conditions. A system identification (ID) process is essential in control design

processes. Different ID techniques have been tested in this system, and an accuracy comparison is presented.

Statement of Originality

To the best of my knowledge and belief, all the material presented in this thesis, except where otherwise referenced, is my own original work, and has not been published previously for the award of any other degree and diploma in any university. If accepted for the award of the degree of advanced master, I consent that this thesis be made available for loan and photocopying

Wei Wang

Acknowledgements

I would like to thank all the people who have helped towards the work in this thesis. I am especially grateful to Associate Professor Ben Cazzolato for his supervision and continued support. Without his help, this thesis would never be possible. I would also like to give thanks to staff in the electronic lab and mechanical workshop, Silvio De Ieso, Joel walker, and George Osborne for their help with building the rig, sensors and interfaces for the system. Many thanks are given to Dorothy Missingham for helping me write skillfully.

I would like give most thanks to my parents, YaoPing Wang and JianPing Jiang who give me so much support, both in emotional and financial.

Finally, I would like to express my grateful to Lili Wang for her selfless support all the time.

Contents

Abstract ..... **i**

Statement of Originality ..... **iii**

Acknowledgements ..... **iv**

Contents ..... **v**

1 Introduction **1**

1.1 Introduction 1

1.2 Literature review ..... **Error! Bookmark not defined.**

1.3 Aims 7

1.4 Assumptions 8

1.5 Conclusion 8

2 Potential Design Solutions **10**

2.1 Configuration 10

2.2 Transmission mechanism 12

2.2.1 Reducing Gearbox 12

2.2.2 Transmission belts 13

2.2.3 No transmission 14

2.3 Materials selection 14

2.4 System Overview 15

2.5 Sensors 18

2.5.1 Position sensor 18

2.5.2 Angle sensor 20

2.6 Motor 21

2.7 Control hardware and software 22
2.8 Conclusion 23
3 Theoretical Analysis 24
3.1 Mathematical model 25
3.1.1 Analysis of the force balance of the ball 26
3.1.2 Analysis of the torque balance of the motor and beam 27
3.1.3 DC motor equation 28
3.1.4 Summary of dynamics 29
3.2 State space model 29
3.3 Model for simulation in 'Matlab' 32
3.4 Simplified model with four states 35
3.5 Conclusion 38
4 Simulation & Experiment 40
4.1 Hardware and software setup 40
4.2 Introduction to the control structures 41
4.2.1 Full state feedback control 42
4.2.2 Full state estimator control 42
4.2.3 Reduced order estimator control 43
4.2.4 Command tracking 45
4.3 Introduction to gain calculation methods 51
4.3.1 Pole placement method 51
4.3.2 LQR method 51
4.3.3 LQG method 52
4.4 Full state feedback control 53
4.4.1 Pole placement method 55
4.4.1.1 Simulation 56
4.4.1.2 Experiment 58
4.4.2 LQR method 61
4.4.2.1 Simulation 62
4.4.2.2 Experiment 64

4.5 Full state estimate control 66
4.5.1 Pole placement method 67
4.5.1.1 Simulation 68
4.5.1.2 Experiment 69
4.5.2 LQR and LQG methods 71
4.5.2.1 Simulation 72
4.5.2.2 Experiment 74
4.6 Reduced order system (LQR and LQG methods) 76
4.6.1 Simulation 78
4.6.2 Experiment 80
4.7 Conclusion 82
5 System Identification 83
5.1 Introduction to system identification (ID) 84
5.1.1 'Free structured parameterizations' approach 85
5.1.2 'Grey box' approach 86
5.1.3 'Black box' approach 87
5.2 System identification of the open-loop plant (simulation) 88
5.3 System identification of the closed-loop system 92
5.3.1 Simulation 93
5.3.2 Experiment 97
5.4 ID from full state estimator plant 100
5.4.1 Simulation 101
5.4.2 Experiment 105
5.5 ID from reduced order system 107
5.5.1 Simulation 109
5.5.2 Experiment 113
5.6 Conclusion 115
6 Considerations in the Experiment 117
6.1 Sensor calibration 117
6.1.1 Position sensor calibration 118

6.1.2 Encoder calibration 120
6.2 System parameter measurement 121
6.2.1 Moment of inertia of the beam 121
6.2.2 System damping 123
6.3 Motor compensation 125
6.4 Conclusion 130
7 Conclusions 132
7.1 Conclusions 132
7.2 Recommendations for further work 134
7.2.1 Model the friction between the ball and resistive wires 134
7.2.2 Other advanced control techniques 134
References 135
Appendix 138
A Matlab M-files for simulation and experiment 138
A.1 Full state feedback & estimator control by the pole placement method 138
A.2 Full state feedback & estimator control by the LQR method 141
A.3 Reduced order observers by the LQR method 144
B Matlab M-files for system identification 147
B.1 ID the open loop plant 147
B.1.1 Main program 147
B.1.2 M-file for ‘grey box’ approach 151
B.2 ID full state feedback 152
B.2.1 Main program 152
B.2.2 M-file for ‘grey box’ approach 157
B.3 ID full state observers 158
B.4 ID Reduced observers 163
C DC motor (RH-11D-3001-E100-AO) 169
C.1 DC motor (RH-11D-3001) 169
C.2 Digital encoder (E100) for the DC motor 171
C.3 Mechanical drawing of the DC motor with Encoder **Error! Bookmark not defined.**

D Mechanical drawings **173**
D.1 Solid model of the entire ball and beam system 173
D.2 Base 174
D.3 Support 175
D.4 Link 176
D.5 Beam 178
E Circuit Diagrams **181**
F The ball and beam system Manual **183**
F.1 Introduction 183
F.2 Hardware equipments 183
F.2.1 DC motor 184
F.2.2 Sensors 184
F.2.3 Power Amplifier 184
F.2.4 Data acquisition boards (Quanser Q4 board) 185
F.3 Software tools 188
F.3.1 Simulink 188
F.3.2 Wincon 190
F.4 Operation of instrumentation 192
F.4.1 Connections 193
F.4.2 Caring for the system 194
G System identification results **196**
G.1 Nominal plant 196
G.2 Simulation results 197
G.2.1 ID from Open-loop plant 197
G.2.2 ID from closed-loop plant 198
G.2.3 ID from full state observer plant 199
G.2.4 ID from reduced order observer plant 199
G.3 Experiment results 200
G.3.1 ID from closed loop plant 200
G.3.2 ID from full state observer plant 201

G.3.3 ID from reduced order observer plant.....202

Chapter 1

Introduction

1.1 Introduction

The ball and beam system is also called ‘balancing a ball on a beam’. It can usually be found in most university control labs. It is generally linked to real control problems such as horizontally stabilizing an airplane during landing and in turbulent airflow. There are two degrees of freedom in this system. One is the ball rolling up and down the beam, and the other is beam rotating through its central axis. The aim of the system is to control the position of the ball to a desired reference point, and reject disturbances such as a push from a finger. The control signal can be derived by feeding back the position information of the ball. The control voltage signal goes to the DC motor via a power amplifier ,then the torque generated from the motor drives the beam to rotate to the desired angle. Thus, the ball can be located at the desired position.

It is important to point out that the open loop of the system is unstable and nonlinear.

The problem of ‘instability’ can be overcome by closing the open loop with a feedback controller. The modern state-space method can be employed to stabilize the system. The nonlinear property is not significant when the beam only deflects a small angle from the horizontal position. In this case, it is possible to linearize the system. However, the non-linearities become significant when the angle of the beam from the horizontal is larger than 30 degrees, or smaller than -30 degrees. In this case, a simple linear approximation is not accurate. Thus a more advanced control technique such as nonlinear control should work better.

1.2 Literature review

The ball and beam system has been built previously by many organizations. In the following, a brief literature review is presented.

Arroyo (2005) built the system named the ‘Ball on Balancing Beam’ in 2005, as seen in Figure 1.1. The system employed the resistive wire sensor to measure the position of the ball. The resistive position sensor acted as a wiper similar to a potentiometer resulting in the position of the ball. (Quanser 2006) The signal from the sensor was processed in a DSP. A DC motor with reducing gear was used. The system was controlled by PD controller.

This system was easy to build, and the simple PD controller was easy to design. The

negative aspects of the ‘Ball on Balancing Beam’ system includes that the beam was made of acrylic, which may be too brittle for a sudden impact. Additionally, although the position of the ball was controlled by the PD controller, the tilt angle of the beam was not measured and controlled. Therefore, the system may be not very robust. .

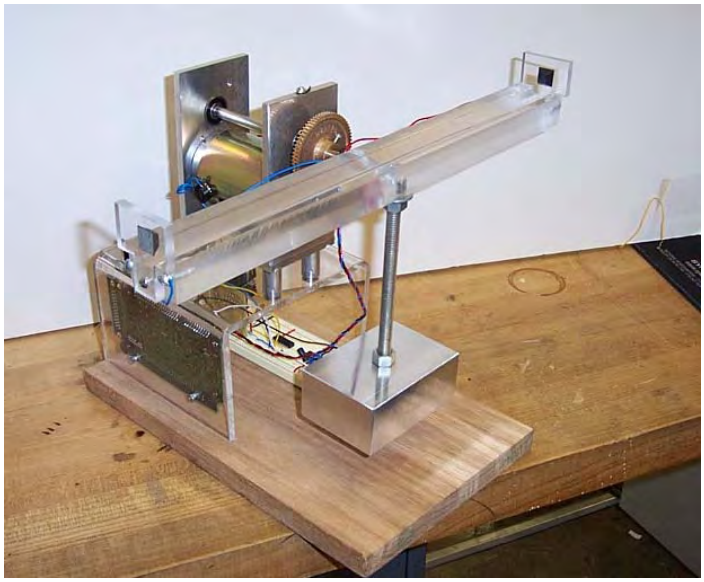

Figure 1.1 ‘Ball on balancing beam’ built by Berkeley Robotics Laboratory (Arroyo 2005)

The Department of Electrical Engineering at Lakehead University built a system named the ‘Ball and Beam Balancer’ (Ambalavanar, Moinuddin & Malyshev 2006), as shown in Figure 1.2. The system employed a DC motor with an integrated gearbox, a resistive wire position sensor, and a digital encoder. The system was controlled by LQR controller.

The ‘Ball and Beam Balancer’ system had one input (voltage input of the motor), and two outputs (the position of the ball and the tilt angle of the beam). The system may be very robust because the state space method with the LQR controller is good

at controlling MIMO (Multiple Input, Multiple Output) system. The beam of the 'Ball and Beam Balancer' system was heavy because it was made of aluminum, which has a high density compared to acrylic.

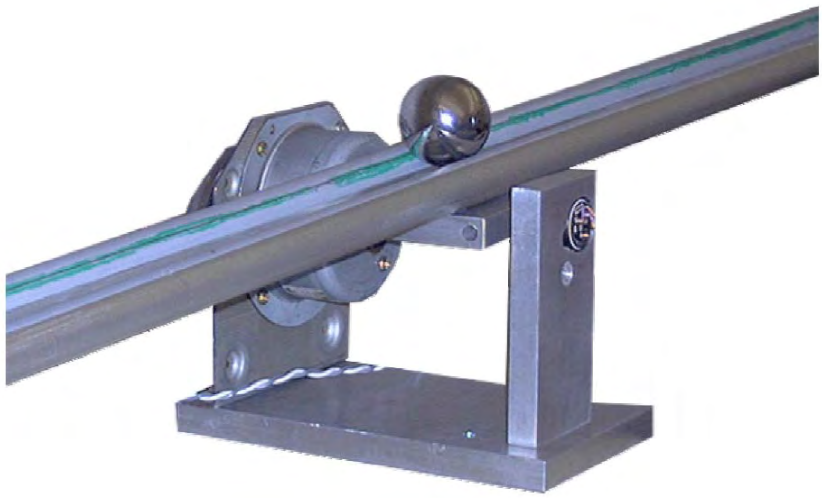

Figure 1.2 'Ball and Beam Balancer' built by The University of Lakehead (Ambalavanar, Moinuddin & Malyshev 2006)

Quanser (2006) present its commercial product named 'ball and beam module', shown in Figure 1.3. The ball and beam module consisted of the position sensor made by resistive wires and a DC servo motor with reducing gearbox. The system could be controlled by a PID controller or a state space controller.

A relative small motor could be used for the system due to the leverage effect. The configuration of the 'ball and beam module' is more complicated than 'Ball and Beam Balancer' shown in Figure 1.2.

Figure 1.3: Ball and Beam module presented by 'Quanser' (Quanser 2006)

Hirsch (1999) built his 'Ball on Beam System' in 1999. A photograph of the system is shown in Figure 1.4. The system employed an ultrasonic sensor to measure the position of the ball. The angle of the beam was measured though the use of a potentiometer. The motor with a gearbox was driven with a high power op-amp circuit. The system is controlled by a PD controller.

Hirsch's system was easy to build due to the simple mechanical configuration. However, the shaft, which supported the weight the beam, was too long for the motor bearing, as shown in Figure 1.5. Therefore the motor bearing would experience a large moment from the beam.

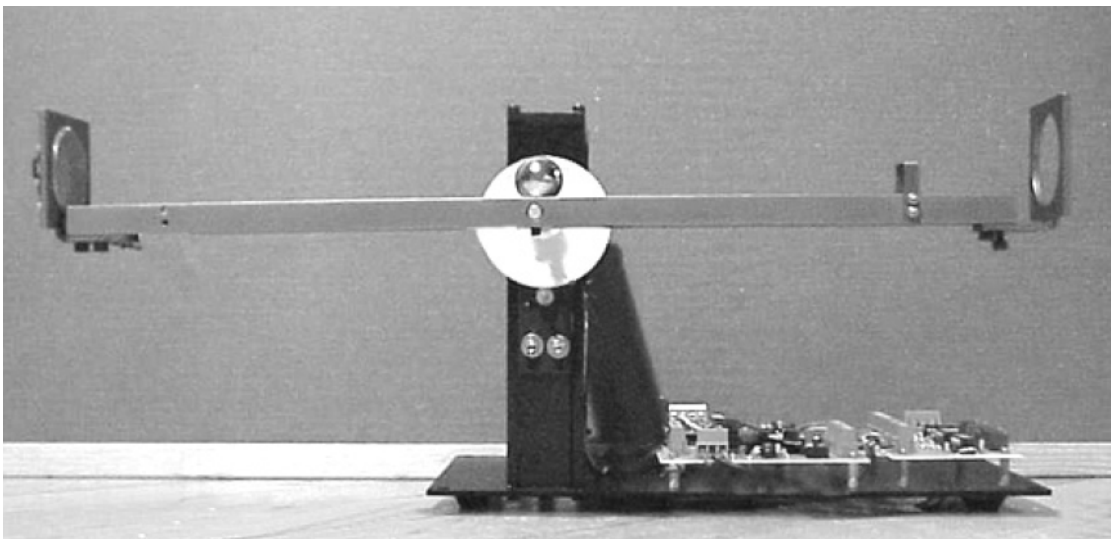

Figure 1.4: Ball on Beam System (Hirsch 1999)

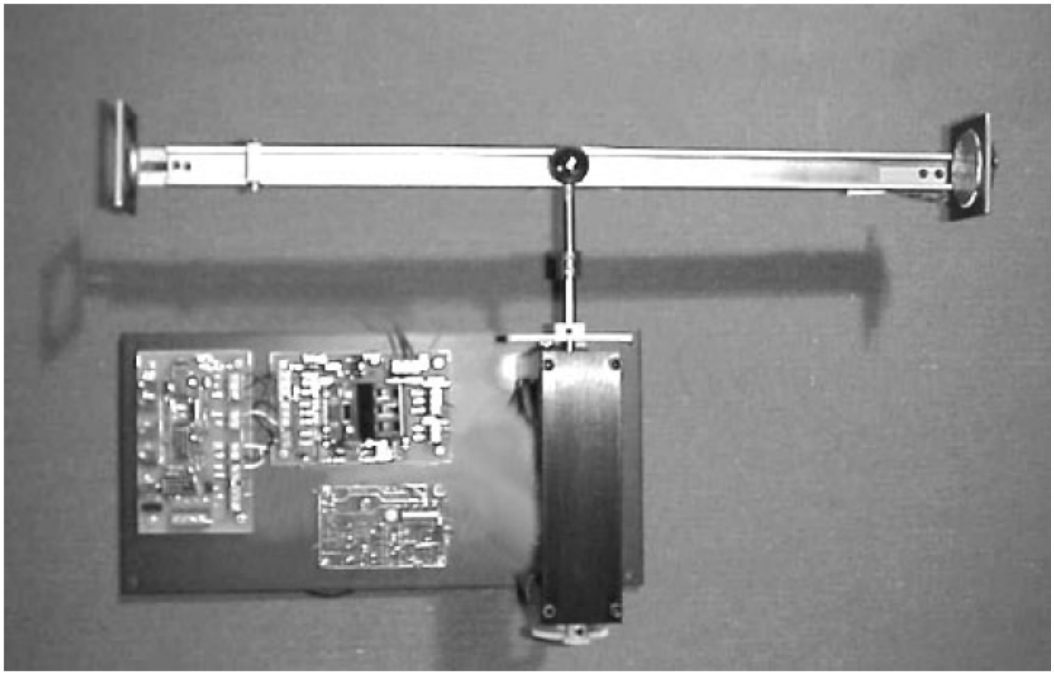

Figure 1.5: Top view of Ball on Beam System (Hirsch 1999)

Rosales (2004) built the ball and beam system, which was similar with the 'Ball and Beam Balancer' system (Ambalavanar, Moinuddin & Malyshev 2006). Rosales's system was made of acrylic, but the 'Ball and Beam Balancer' system was made of aluminum.

Lieberman (2004) build a system named ‘A Robotic Ball Balancing Beam’, shown in Figure 1.6. The system is similar with ‘Ball on Beam System’ (Hirsch 1999). The difference between the two systems is that Lieberman’s system used a resistive wire position sensor, and the Hirsch’s system used an ultrasonic position sensor.

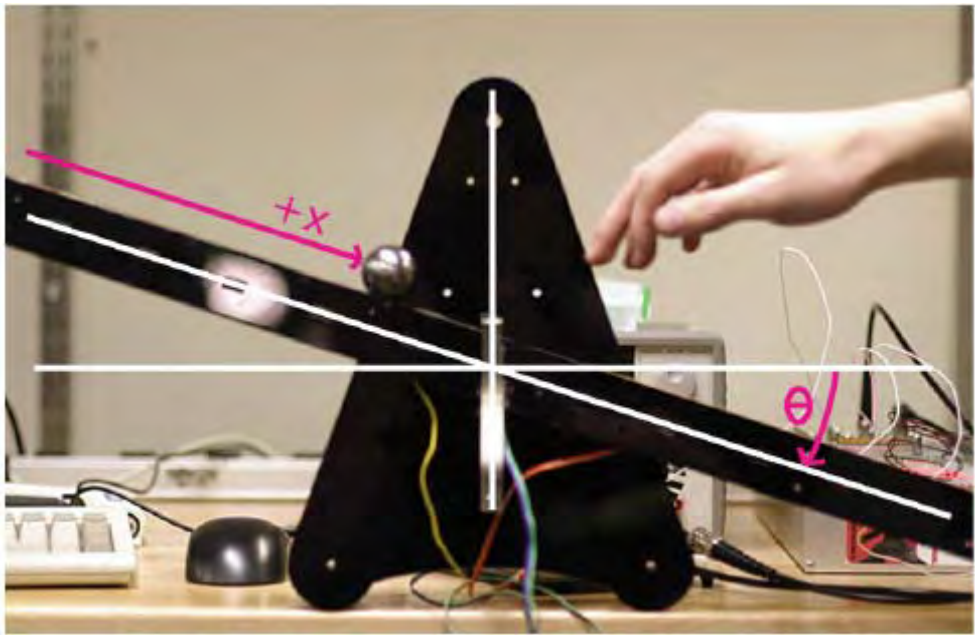

Figure 1.6: A Robotic Ball Balancing Beam (Lieberman 2004)

1.3 Aims

The major aim of the project is to build and control a real ball and balancing rig. Specifically, the aims are listed in the following:

- Build a simulation model for this system (including state model and VR model)
- Build the real plant for this system
- System ID & determination of system parameters

- Design several linear controllers for this system, and compare the result using the different control techniques
- Use the designed the controllers to control the real plant, and compare the different performances
- If time permits, try and apply more advanced control techniques such as non-linear control and robustness control

1.4 Assumptions

In order to theoretically model the ball and beam system, the assumptions are listed in the following:

- The ball rolls on the beam without slip
- The gearbox embedded in the motor exhibit no backlash effect
- The ball and linear sensor (resistive wire) contact with each other very well
- The base of the system is static with respect to the ground
- The beam is able to rotate between -30 degrees to 30 degrees relative to the horizontal

1.5 Conclusion

In this chapter, a detailed introduction and the final outcomes of this project was presented. A belief literature review was given. In order to theoretically model the

ball and beam system, some assumptions were detailed in the end of this chapter.

In Chapter 2, the project rigs designed and built for this project will be discussed. Some potential design solutions related to major aspects of the ball and beam system, including configuration, transmission mechanism, materials for the system, sensors and motors, will be presented.

Chapter 2

Potential Design Solutions

To build the ball and beam system, it is necessary to discuss what components are involved in this project. Like many other typical control projects, an actuator, some sensors, mechanical parts, a computer and a control board are essential components. This chapter will discuss those components, and presents an evaluation of each component among numerous options which are compared.

2.1 Configuration

A number of configurations related to this project are possible. However, it is essential to understand all the possible configurations in order to devise the final.

There are two configurations to support the beam. One configuration is shown in the Figure 2.1, which illustrates that the beam is supported in the middle, and rotates against its central axis. Most ball and beam systems use this type of configuration. The advantage of this form is that it is easy to build, and the mathematical model is

relatively simple. The other configuration is shown in Figure 2.2. The beam is supported on the both sides by two level arms. One of level arms is pined, and the other is coupled to output gear. The disadvantage of this configuration is that more consideration of the mechanical parts is required, and this may add some difficulties in deriving a mathematical model. The 'Quanser' ball and beam system uses this configuration for its product (see Figure 1.3) (Quanser 2006). The advantage of this system is that relatively small motor can be used due to the leverage effect.

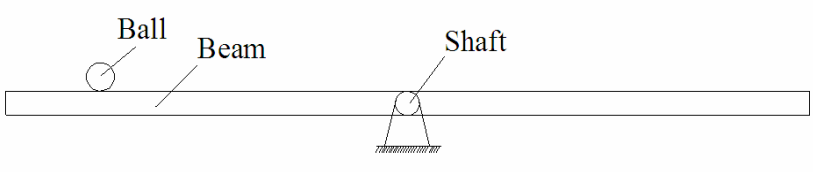

Figure 2.1: Beam supported in the centre

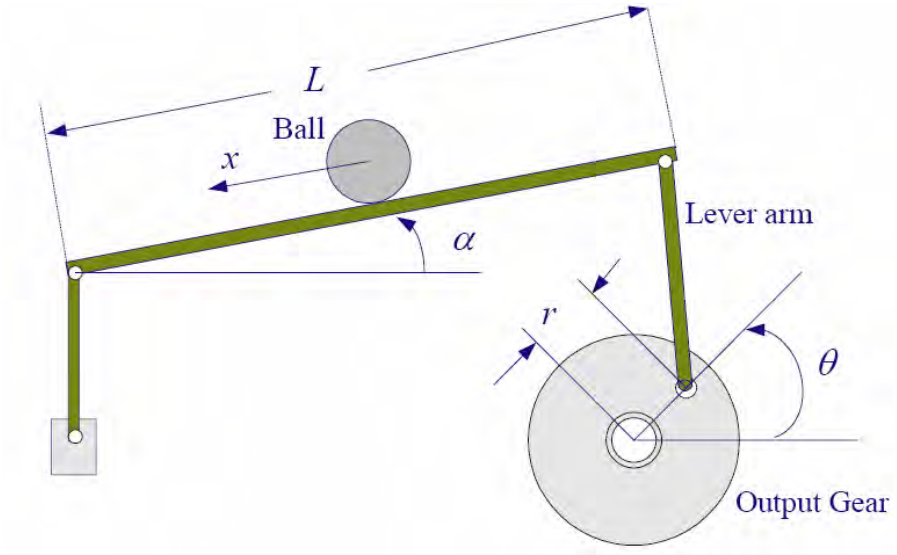

Figure 2.2: Beam supported on both sides（University of Alberta 2006）

Based on the shaft supporting the beam in the middle, as seen in Figure 2.1, there are two further configurations. The first one, shown in Figure 2.3, positions the beam in the middle of two supports. The advantage of this system is that the beam is supported does not experience a moment. The second configuration, shown in Figure

2.4, cantilevers the beam off the shaft. This configuration has been employed in this project because of the simplified mechanical design.

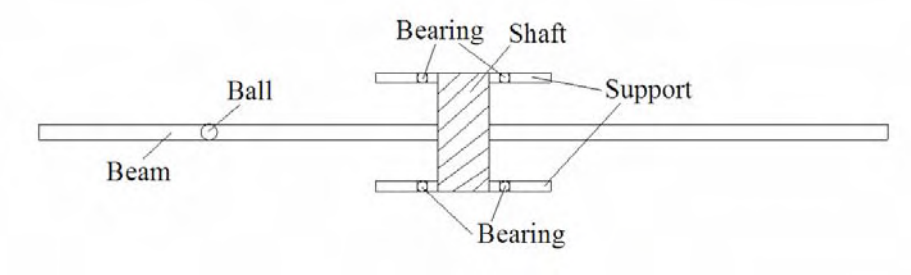

Figure 2.3: Beam sits in the middle of supports

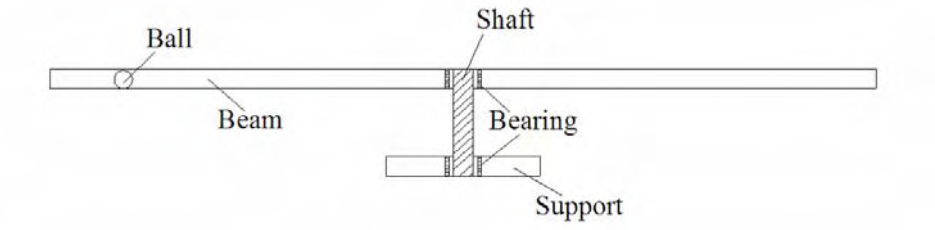

Figure 2.4: Beam is cantilevered off the motor shaft

2.2 Transmission mechanism

The function of the transmission mechanism is to transfer the force or torque from an actuator to the beam. In this project, the transmission mechanism is used to reduce the rotational speed of the motor. Generally, a reducing gearbox, transmission belt, and chain are commonly involved in this type of project.

2.2.1 Reducing Gearbox

A reducing gearbox is a compact unit made by several gears and a box. Gears can be set in a way to reduce the rotational speed from the motor to the beam. The

advantage of the gearbox is that it has good torsional stiffness and produces large torque. The use of the gearbox results in further advantages in control including high bandwidth and high cross over frequency (Rosales 2004). The disadvantage of the gearbox is that when employed with closed loop control oscillations may occur due to the backlash. In this project, the gearbox and the motor are fabricated as a single assembly.

2.2.2 Transmission belts

Transmission belts have many applications in the industry, for example, the car engine uses the transmission belt to transfer the torque from the crankshaft to the gearbox. The principle of the transmission belt is to employ friction between the belt and pulleys to transfer the motion from one mechanism to the other. The transmission belt is very attractive for the ball and beam project since it has following advantages (Rosales 2004):

- it can transfer a large torque with a low price
- there is no backlash if it is fixed properly
- overload protection (the belt will slip on the pulleys when overload happens)

The disadvantages of transmission belts are that they have less stiffness than gearbox, and can not work without pulleys. Those disadvantages bring difficulties in fabrication.

2.2.3 No transmission

No transmission means the motor and beam are directly connected. Without transmission, the system has high torsional stiffness and is easy to build. Unfortunately the motor must generate much larger torques to meet the same speed dynamic response as other transmission methods. Therefore, the expense of the motor would be higher.

2.3 Materials selection

Selection of appropriate material for a particular application is an essential element of all engineering projects. In this project, the main parts of the system are the base, support, beam, ball and sensors. The motor is not discussed here because it is commercially supplied. Clearly, cheap, light and stiff materials should have advantages in this project. Hence, following materials are considered:

- polycarbonate (Acrylic)
- aluminum alloy
- balsawood
- stainless steel or carbon steel

The beam can be made of balsawood, polycarbonate or aluminum alloy. These materials are light and stiff. In this application, a combination of polycarbonate beam and aluminum channel is used to make the beam. This combination overcomes the

disadvantage of brittle polycarbonate, and optimizes the mass of the system. Figure 2.5 shows the aluminum channel with polycarbonate beam is used for this project.

To minimize the mass of the system, the frame and base can be made of, for example polycarbonate or aluminum. In this project, aluminum was chosen because it is cheap and easy to fabricate. A stainless steel ball (Figure 2.5) with good conductivity and corrosion resistance ability is also used for this application.

2.4 System Overview

The photographs of the ball and beam system built in this project are shown in this section. Detailed information about each component is presented.

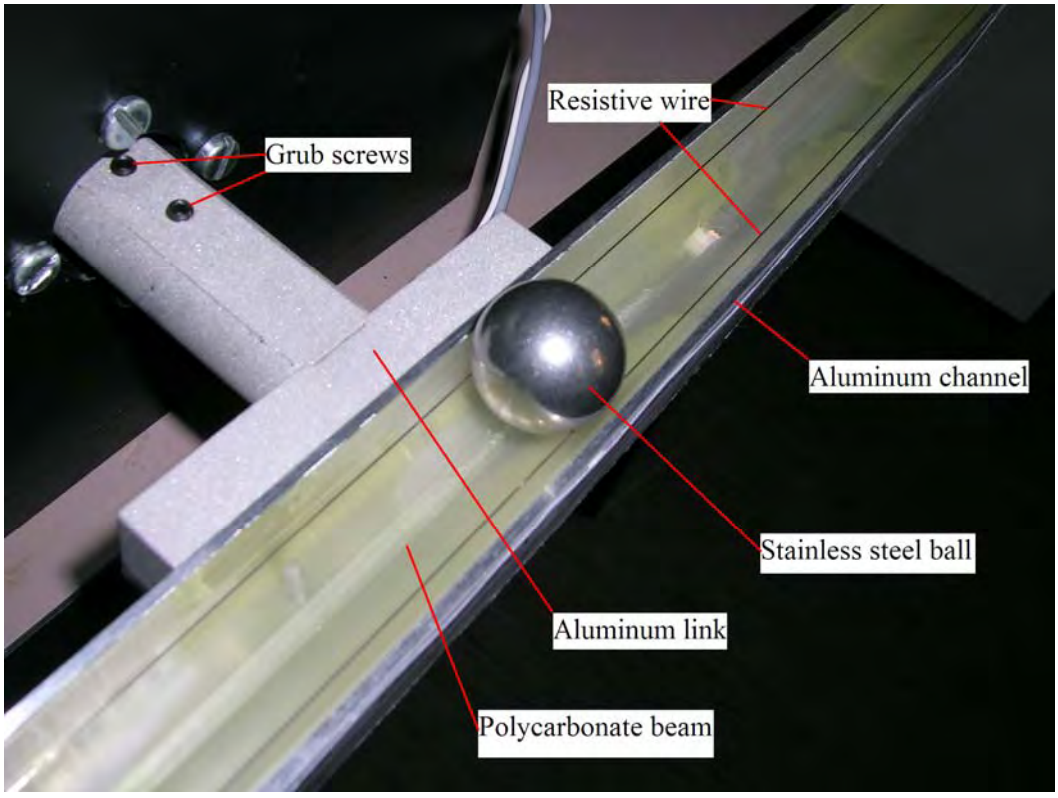

Figure 2.5: The stainless steel ball on the polycarbonate beam

- Aluminum channel, shown in Figure 2.5, is made of the standard Aluminum channel, which has the width of 25.8mm, and the thickness 1.4mm. The detail information can be found in Appendix D, Figure D.12.
- Stainless steel ball (Figure 2.5) has the diameter 20mm with the grade 100.
- Aluminum link, shown in Figure 2.5, is used to link the motor shaft with the beam. The drawing of the link is shown in Appendix D (Figure D.7, Figure D.8, Figure D.9, and Figure D.10)
- Polycarbonate beam in the Aluminum channel has the length of 700mm. The drawing the beam can be found in Appendix D.
- Resistive wires made of nickel-chromium 80 with a diameter 0.01 inches (about $16\,\Omega$/meter)
- Aluminum support and base, shown in Figure 2.6, have the thickness 12mm. The drawing of them is shown in Appendix D.
- DC motor (RH-11D-3001-E100-AO) in Figure 2.6 with the gearbox and the digital encoder is commercial available from Harmonic Drive. (Harmonic Drive 2006) The technical date of the DC motor is presented in Appendix C.
- Position sensor amplifier, shown in Figure 2.7, is used to amplify and filter the high frequencies of the ball's position signal. The circuit of the position sensor amplifier is shown in Appendix E.
- Power amplifier is commercially available from Quanser (Quanser 2006). The type of the amplifier is UPM 2405, which is discussed in Appendix F.2.2.

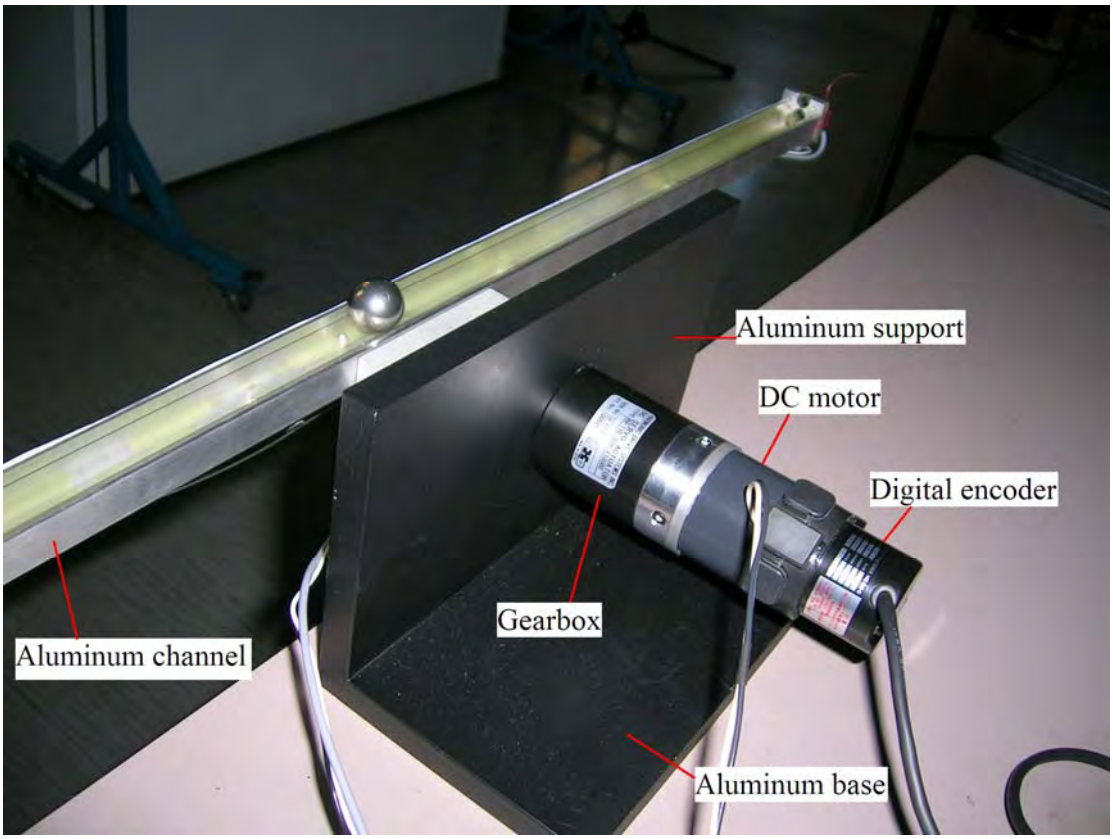

Figure 2.6: Back view of the ball and beam system

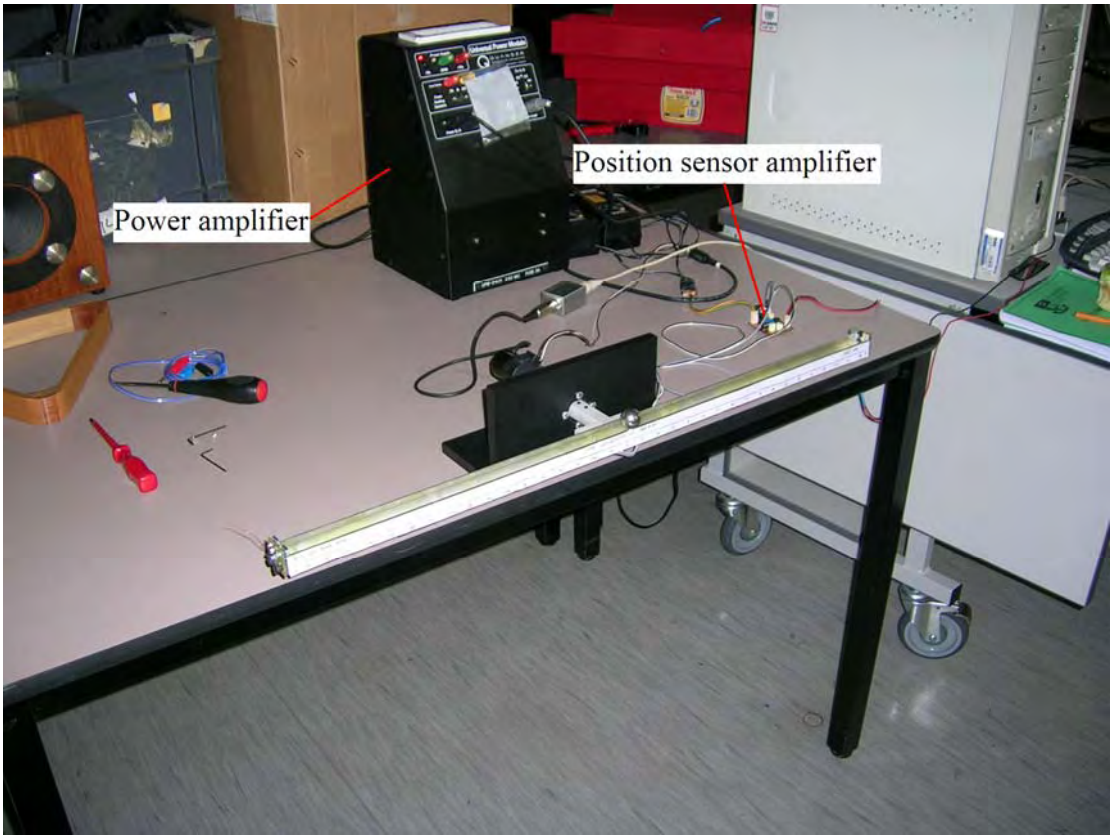

Figure 2.7: Overview of the ball and beam system

2.5 Sensors

The sensor is a device which is used to record the presence of something, or changes in something. Two types of sensors are involved in this application, the position sensor which measures the position of the ball on the beam and the angle sensor which measures the rotational position of the beam. Cheap, reliable, and good resolution sensors are required in this application.

2.5.1 Position sensor

There are several position sensors which are suitable for this project. They are conductive plastic, resistive wire (nickel-chromium), ultrasonic range transducer and vision sensor.

The position sensors made of the conductive plastic or the resistive wires forming the track on which the ball is free to roll. The position of the ball is obtained by measuring the voltage at the voltage out wire, shown in Figure 2.8. "*When the ball rolls along the track, it acts as a wiper similar to a potentiometer resulting in position of the ball.*" (Quanser 2006)

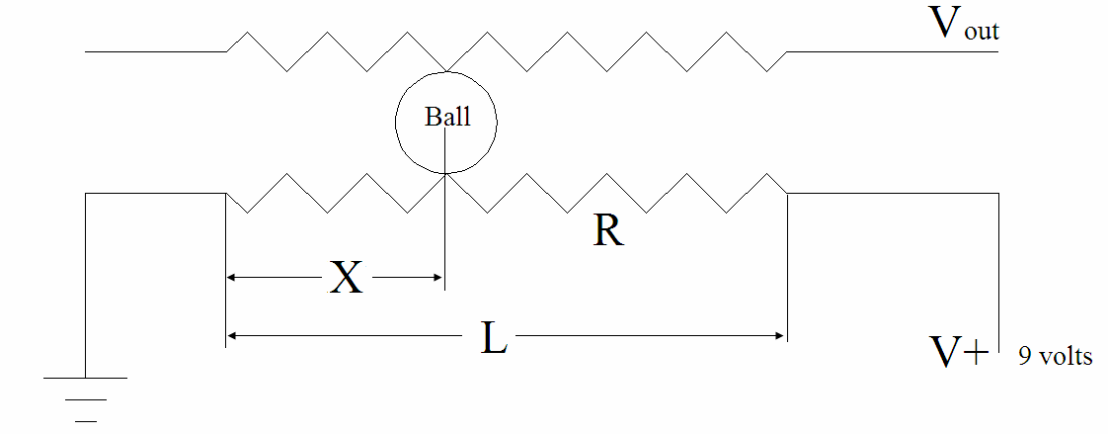

Figure 2.8: Sketch of the resistive wire position sensor

For conductive plastic, the advantage is that it has large resistance of about 3000 ohm/cm. The disadvantage is that the conductivity is poor, and conductive plastic is susceptible to scratches. (Rosales 2004) Thus it is unreliable for this project.

Resistive wire is a good choice since it has a relative large resistance of about 15ohm/m. An additional amplifying circuit is necessary for this application since the voltage signal (around several ‘mv’) is too small for sampling. Hence the amplified signal with a high noise level can be expected. This disadvantage can be overcome by adding a low pass filter.

The vision system is too expensive for this simple project. Moreover, much effort is required to put it in the application. The ultrasonic ranger transducer is unreliable for this project since the signal from the transducer contains high level noise. In this project, a cheap nickel-chromium 80-resistance wire alloy with a diameter 0.011 inches was chosen (see Figure 2.5). The signal from the resistive wire position

sensor is processed by the position sensor amplifier, shown in Figure 2.7.

2.5.2 Angle sensor

The angle sensor for this project can be a DC coupled accelerometer, a rotary potentiometer or a digital encoder. A low price rotary potentiometer is easy to put in an application, but the analog output signal must be transferred to a digital signal by an A/D convertor for the actual application. A digital encoder (see Figure 2.6) is the best sensor for the control of the ball and beam system because it is accurate and reliable, and has good resolution. Additionally the digital encoder presents a digital signal.

2.6 Motor

A motor is a device that changes electricity into movement. It is the most important part of the control of the ball and beam system. A cheap, linear motor with good controllability is necessary for this application.

A DC motor with an integrated gearbox, a stepper motor, or an AC motor may be used for this application. In many project applications, a DC motor is chosen because it is easy to drive and control. For this project, the motor made by ‘Harmonic Drive’ (Harmonic drive 2006) is used. From Figure 2.9, it is easy to

identify that the gearbox, motor and encoder are assembled as a unit. A photograph of the DC motor used in this project is presented in Figure 2.6.

The stepper motor is a perfect actuator for this application since the output is in angle step increments, which directly meets the beam's requirements. However, one disadvantage is that the stepper motor is hard to drive. Besides, the stepper motor presents the discrete positions, which are not continuous and non-linear. The non-continuous and non-linear characteristics are undesirable for the ball and beam system.

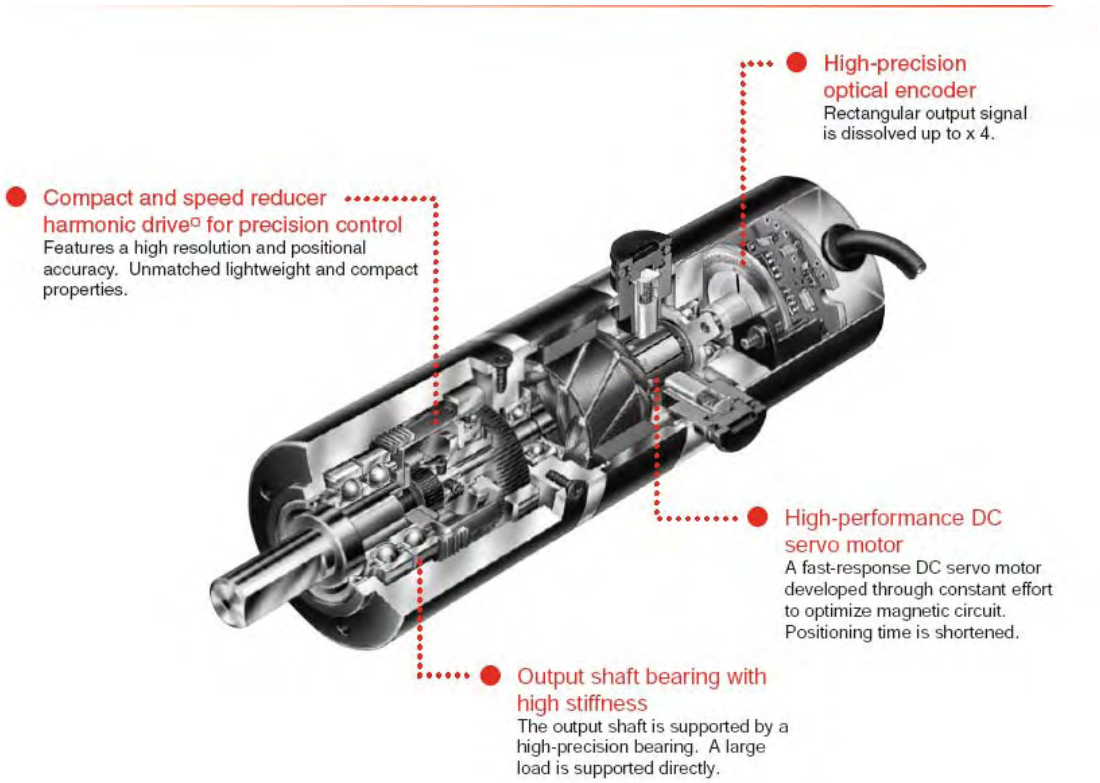

Figure 2.9: DC motor (Harmonic drive 2006)

2.7 Control hardware and software

The control hardware of the ball and beam system, including the power amplifier for DC motor, the data acquisition boards, and the position sensor amplifier, and the control software of the system, including the Matlab/Simulink (Matlab 7.01 2006), Wincon (Wincon 5.0 2006), and Visual C++ (Visual C++ 6.0 2006) are presented in Appendix F.

2.8 Conclusion

It is essential to pay attention to the limitations of the components in this project since those limitations would affect the final outcome of the system. Through comparing the different options for particular applications of the ball and beam system, the most appropriate option has been chosen for this control project. This option suggests a configuration of the beam supported in the middle, a transmission mechanism of a reducing gearbox, the material of a combination of acrylic and aluminum, sensors of a resistance wire sensor and a digital encoder, and a DC motor for the ball and beam system.

In the next chapter, a theoretical analysis of the ball and beam system will be

presented. The mathematical model of the system is derived to model the system. The model will be further simplified to achieve a better controllability.

Chapter 3

Theoretical Analysis

The ball and beam system should be fully understood before attempting to control it. A theoretical analysis is the first step to approach this ‘black box’ system. Usually, analytical processes require engineers to investigate a system based on universal laws of physics and their own experience.

In the ball and beam system, the ball rolls on the beam which is driven by the motor. In order to position the ball on the desired value, the beam must rotate correctly against its central axis. This further requires that the motor is electrically driven correctly. Therefore it is essential to build a mathematical model of the system to express the relationships between all components. Usually, there are several techniques used to derive the mathematical model: the transfer function between input and output. The simplest way to deliver the mathematical model is to employ physics and electronic laws to express the system. In this case, the system is very simple, thus this method of delivering is most efficient to derive a mathematic model. Some other methods, such as a system identification method or experimental method,

are applied in more complex systems, which are impossible to derive an accurate model by simple laws. It is worthwhile to note that the model derived in the following is only an ideal model regardless which kind of method is used. In other words, it is impossible to build a perfect model.

3.1 Mathematical model

According to physics laws, mathematical equations are derived to model a system. Therefore, a detailed derivation of those equations is presented.

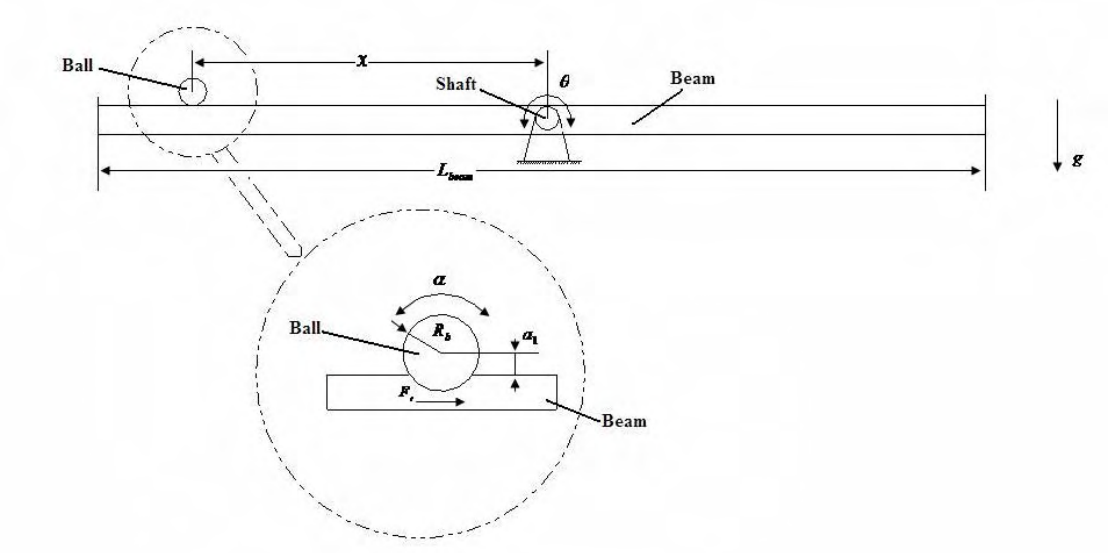

Figure 3.1: Sketch map of the system

Figure 3.1, presented above, describes the basic system model from which the mathematical model is derived. This derivation consists of the force balance of the ball and the torque balance of the beam.

3.1.1 Analysis of the force balance of the ball

The following equation can be derived from analysis of the force balance, F_b of the ball employing Newton's law.

$$\Sigma F_b = M_{ball} g \sin\theta - F_r = M_{ball} \ddot{x} + b_1 \dot{x} \tag{3.1}$$

where M_{ball} is the mass of the ball, g is acceleration of gravity, x is the vertical distance between the center of ball and center of the shaft, b_1 is the friction constant while the ball rolls on the channel of the beam, θ is the beam's tilt angle from the horizontal position, F_r represents the externally applied force.

The position of the ball is equal to the rotational angle the ball rotates through, α, multiplied by the rotational radius of the ball, a_1:

$$x = \alpha \times a_1 \tag{3.2}$$

where a_1 is the vertical distance between the centre of the ball and the contact point between the ball and the beam.

The torque balance of the ball is expressed as:

$$\Sigma \tau_b = F_r a_1 = J_{ball} \ddot{\alpha} \tag{3.3}$$

where $\ddot{\alpha}$ is the angular acceleration of the ball, $\Sigma \tau_b$ is the sum of the torque on the ball, J_{ball} is the moment of inertia of the ball, given by:

$$J_{ball} = \frac{2}{5} M_{ball} R_b^2 \tag{3.4}$$

where R_b is the radius of the ball.

From Equations (3.1) to (3.4), the following result can be derived.

$$\left(1+\frac{2}{5}\times\left(\frac{R_b}{a_1}\right)^2\right)\ddot{x}+\frac{b_1\dot{x}}{M_{ball}}=g\sin\theta \tag{3.5}$$

Equation (3.5) relates ball's position with the angle of the beam, and indicates that the large tilt beam results in large acceleration and high speed of the beam.

3.1.2 Analysis of the torque balance of the motor and beam

The following equation is derived from analysis of the torque balance of the motor:

$$T_{motor}=KI-J_{motor}\ddot{\theta}-b\dot{\theta} \tag{3.6}$$

where K is electromotive force constant, I is current flow into the motor, J_{motor} is the moment of inertia of DC motor, b is the damping constant of the rotational system, T_{motor} is the torque generated by the DC motor, $\ddot{\theta}$ is the angular acceleration of the beam, $\dot{\theta}$ is angular velocity of the beam.

The torque generated by the ball can be expresses in Equation (3.7), and the torque balance of the beam is shown in Equation (3.8):

$$T_{ball}=-xM_{ball}g\cos\theta \tag{3.7}$$

$$T_{beam}=T_{motor}+T_{ball} \tag{3.8}$$

where T_{beam} is the torque on the beam.

The moment of inertia of the beam and motor is expressed as follows:

$$J_{bm}=J_{beam}+J_{motor} \tag{3.9}$$

where J_{beam} is the moment of inertia of the beam, J_{bm} is the sum of the moment

of inertia of the beam and motor,

$$J_{beam} = \frac{1}{12} M_{beam} L_{beam}^2 \tag{3.10}$$

where M_{beam} is the mass of the beam, and L_{beam} is the length of the beam assuming the beam is a rectangular solid.

From Equations (3.6) to (3.10), the following result can be derived.

$$\ddot{\theta} = \frac{KI - xM_{ball} g \cos\theta - b\dot{\theta}}{J_{bm}} \tag{3.11}$$

which describes the angular acceleration of the shaft.

3.1.3 DC motor equation

Since the DC motor is armature-controlled, and based on the Newton's law combined with the Kirchhoff's law, Equation 3.12 can be derived:

$$L\frac{dI}{dt} + RI = V - K_e \dot{\theta} \tag{3.12}$$

where L is the armature induction, I is the electrical current flow into motor, V is input voltage applied to the armature, K_e is motor constant related to the back electromotive force and R is the resistance of armature.

This equation relates the supplied electrical current to voltage. Therefore, rearranging Equation (3.12), the following equation can be derived:

$$\dot{I} = \frac{V - RI - K_e \dot{\theta}}{L} \tag{3.13}$$

3.1.4 Summary of dynamics

A summary of the above equations is as follows:

$$\left(1+\frac{2}{5}\times\left(\frac{R_b}{a_1}\right)^2\right)\ddot{x}+\frac{b_1\dot{x}}{M_{ball}}=g\sin\theta$$

$$\ddot{\theta}=\frac{KI-xM_{ball}g\cos\theta-b\dot{\theta}}{J_{bm}}$$

$$\dot{I}=\frac{V-RI-K_e\dot{\theta}}{L}$$

3.2 State space model

The mathematical model may be expressed in state space form. The state space model is expressed as matrix form, which is essential for the state space control method.

Using Equations (3.5), (3.11), and (3.13), the state space form of the system is:

$$\dot{X}=\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{\theta}\\ \ddot{\theta}\\ \dot{I}\end{bmatrix} \tag{3.14}$$

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{\theta}\\ \ddot{\theta}\\ \dot{I}\end{bmatrix}=\begin{bmatrix}0 & 1 & 0 & 0 & 0\\ 0 & \dfrac{b_1}{\left(1+\frac{2}{5}(\frac{R_b}{a'})^2\right)M_{ball}} & \dfrac{g}{1+\frac{2}{5}(\frac{R_b}{a'})^2} & 0 & 0\\ 0 & 0 & 0 & 1 & 0\\ \dfrac{-M_{ball}g}{J_{bm}} & 0 & 0 & \dfrac{-b}{J_{bm}} & \dfrac{K}{J_{bm}}\\ 0 & 0 & 0 & \dfrac{-K_e}{L} & \dfrac{-R}{L}\end{bmatrix}\begin{bmatrix}x\cos\theta\\ \dot{x}\\ \sin\theta\\ \dot{\theta}\\ I\end{bmatrix}+\begin{bmatrix}0\\ 0\\ 0\\ 0\\ \dfrac{1}{L}\end{bmatrix}V \tag{3.15}$$

where X is state space vector, x is the displacement of the ball, θ is the beam angle, and I is the current flow into the motor. The dot symbol represents the temporal differential.

Clearly, the above state space model is nonlinear. However since θ is relatively small under most circumstances, the above model can be linearized by employing following equations:

$$\theta \approx \sin\theta \ldots\ldots\ldots\ldots \text{ where } (\theta << 1) \tag{3.16}$$

$$\cos\theta = 1 \ldots\ldots\ldots\ldots\ldots\ldots \text{where } (\theta << 1) \tag{3.17}$$

Placing Equations (3.16), (3.17) into the state space matrix and the state space vector, the following linear state space model can be derived:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{I} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & \dfrac{b_1}{\left(1+\frac{2}{5}(\frac{R_b}{a'})^2\right)M_{ball}} & \dfrac{g}{1+\frac{2}{5}(\frac{R_b}{a'})^2} & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ \dfrac{-M_{ball}g}{J_{bm}} & 0 & 0 & \dfrac{-b}{J_{bm}} & \dfrac{K}{J_{bm}} \\ 0 & 0 & 0 & \dfrac{-K_e}{L} & \dfrac{-R}{L} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \\ I \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \dfrac{1}{L} \end{bmatrix} V \tag{3.18}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} X \tag{3.19}$$

The output of the system y is the position of the ball x and the beam tilt angle θ. The control signal is the voltage supplied to the motor.

The system parameters of the ball and beam system are presented in Table 3.1. Many of these parameters were determined through direct measurements or

experimentations as presented in Section 6.2. Some were obtained from the technical data of the DC motor as shown in Table C.1. Others are defined for the ball and beam system. For example, the limits of the beam tilt angle θ are defined as shown in Table 3.1 to meet the assumption for the linear ball and beam system. The limits of the position of ball x are defined because the beam is 0.7m, and the original point of the system is at the middle of the beam. The limits of the control voltage V are defined to protect the motor from voltage overload.

Table 3.1: System parameters

Number	Parameter	Symbol	Unit	value	Source of value
1	Beam tilt angle	θ	rad	$(-\frac{\pi}{6}, \frac{\pi}{6})$	Defined
2	Mass of ball	M_{ball}	kg	0.0327	Directly Measured see Table 6.1
3	DC motor electric resistance	R	ohms	4.7	Appendix C see Table C.1
4	Electromotive force constant	K	Nm/A	4.91	Appendix C see Table C.1
5	Ball radius	R_b	m	0.01	Directly Measured see Table 6.1
6	Electrical inductance	L	H	0.0016	Appendix C see Table C.1
7	Control voltage	V	volts	(-20, 20)	Defined
8	Position of ball	x	m	(-0.35, 0.35)	Defined
9	Moment of inertia	J_{bm}	$kg.m^2$	0.062	Sum of J_{beam} and J_{motor}

10	Damping ratio of mechanical system	b	Nm/(rad/s)	1.5279	Experimentally determined see Section 6.2.2
11	Acceleration of gravity	g	m/s^2	9.81	Referred
12	Length of the beam	L_{beam}	m	0.7	Directly Measured see Table 6.1
13	Mass of the beam	M_{beam}	kg	0.381	Directly Measured see Table 6.1
14	Friction constant	b_1	Ns/m	Very small	Not measured
15	Moment of inertia of the beam	J_{beam}	$kg.m^2$	0.019	Experimentally determined see Section 6.2.2
16	Moment of inertia of the motor at output shaft	J_{motor}	$kg.m^2$	0.043	Appendix C see Table C.1
17	Rotational radius of the ball	a_1	m	0.005	Calculated from Figure D.12
18	Motor constant related to the back electromotive force	K_e	volts/(rad/s)	4.77	Appendix C see Table C.1

3.3 Model for simulation in ‘Matlab’

Once the state space model has been derived, it is necessary to create a numerical model for simulation. This model is very useful in aiding further control design.

With Equations (3.5), (3.11), (3.13), it is easy to build a ‘Simulink’ model in ‘Matlab’. This model is used in the numerical simulation of the system with designed controllers. This step is important because the response of this model is

closely related with the real system. Thus, initially this ideal model can be used to design the controller to achieve the best performance of the ball and beam system.

Figure 3.2 shows the ‘Simulink’ model in full state feedback and estimation structure. Figure 3.3 shows that the ball and beam system has one input and five outputs. The input is the voltage supplied to the motor. The outputs include the motor current, the beam tilt angle, the beam tilt speed, the ball’s position, and the speed of ball. The friction constant b_1 is so small that it has been assumed to be zero.

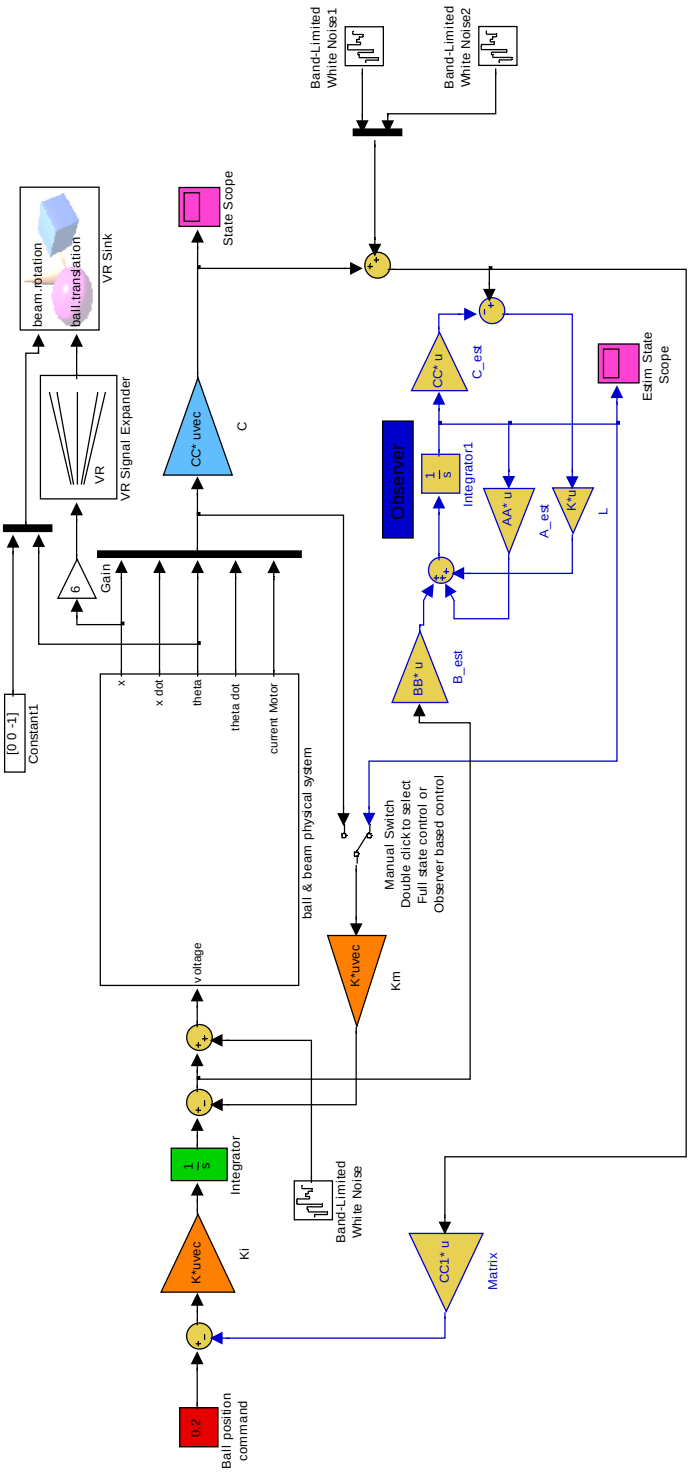

Figure 3.2: Overall look of the simulink model in 'Matlab'

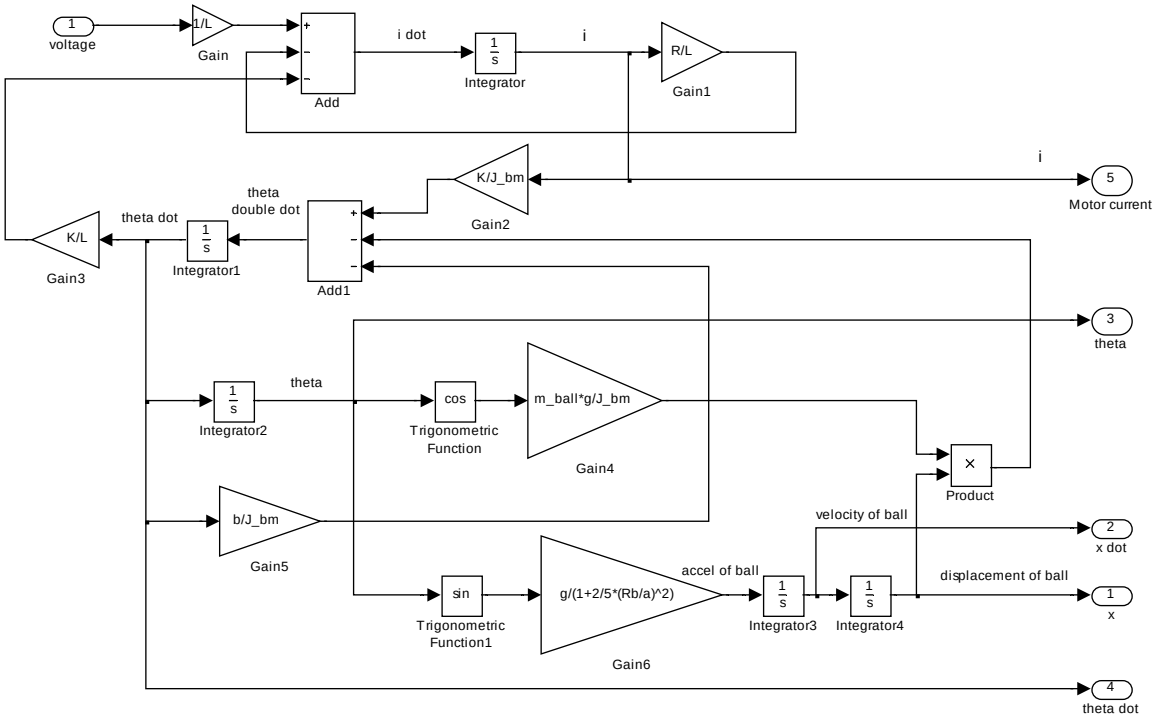

Figure 3.3: Inner structure of the 'ball & beam physical system' in the 'Simulink' model

3.4 Simplified model with four states

Some system parameters are so small that they can be neglected, thereby improving the control conditioning of the ball and beam system. Discarding these parameters is known as the simplification process. The simplification process makes the model less accurate but better conditioned for practical control.

The previous five-state model, shown in Figure 3.2 and Figure 3.3, is uncontrollable when all the system parameters are substituted into the state space equation. This

lack of control is because the armature induction in this application is so small that makes control condition poor. Considering that armature inductance is very small L=1.6mH, Equation (3.13) can be approximated as follows:

$$V = RI + K_e \dot{\theta} \tag{3.20}$$

Therefore with Equation (3.20) and Equation (3.11), the following equation can be derived:

$$\ddot{\theta} = \frac{\frac{K}{R}V - (\frac{KK_e}{R} + b)\dot{\theta} - XM_{ball} g \cos\theta}{J_{bm}} \tag{3.21}$$

With Equations (3.21) and Equation (3.5) considering friction constant b_1 assumed to be zero, the following state-space equation can be derived:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{g}{1+\frac{2}{5}\left(\frac{R_b}{a_1}\right)^2} & 0 \\ 0 & 0 & 0 & 1 \\ -\dfrac{M_{ball}}{J_{bm}} & 0 & 0 & -\dfrac{\left(\frac{KK_e}{R} + b\right)}{J_{bm}} \end{bmatrix} \begin{bmatrix} x\cos\theta \\ \dot{x} \\ \sin\theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{K}{RJ_{bm}} \end{bmatrix} V \tag{3.22}$$

Again using the small angle the approximation with Equations (3.16) and Equation (3.17), the following linear state space equation can be derived:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{g}{1+\dfrac{2}{5}\left(\dfrac{R_b}{a_1}\right)^2} & 0 \\ 0 & 0 & 0 & 1 \\ -\dfrac{M_{ball}}{J_{bm}} & 0 & 0 & -\dfrac{\left(\dfrac{KK_e}{R}+b\right)}{J_{bm}} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{K}{RJ_{bm}} \end{bmatrix} V \tag{3.23}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} X; \tag{3.24}$$

The physical definition of the ball and beam system parameters is the same as the previous definitions for state space model in Table 3.1.

Figure 3.4, represents an inner structure of the simplified four states state space model, which has one input and four outputs. The input again is the voltage flow into the motor, and the outputs include position and velocity of the ball, and tilt angle and rational speed of the beam.

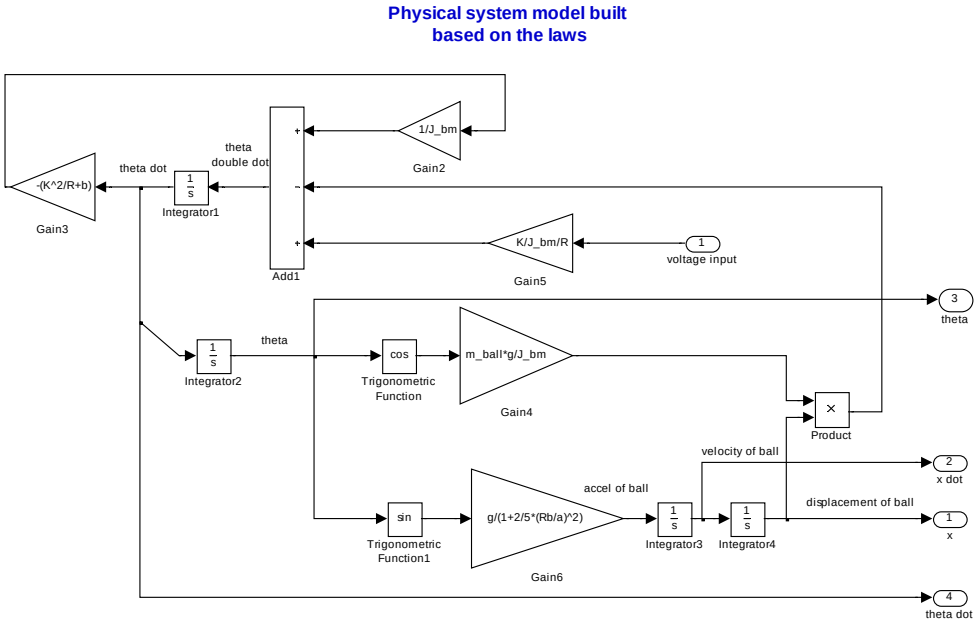

Figure 3.4: Inner structure of the 'ball & beam physical system' in the simplified 'Simulink' model (four states)

3.5 Conclusion

The mathematical model of the ball and beam system is derived by employing the physical and electrical laws. The model process is conducted based on the assumptions which are presented in the introduction section: the ball rolls on the beam without any slip, the gearbox embedded in the motor has no backlash, the ball and linear sensor (resistive wire) contact with each other very well, the base of the system is static with respect to the ground, and the beam rotates between -30 degrees to 30 degrees relative to the horizontal.

A numerical simulation model is suggested in this chapter according to the

mathematical model. In addition, the further simplification of the five state mathematical model results in a simplified four state model, which has a much better controllability. The simplified model is represented herewith:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{g}{1+\frac{2}{5}\left(\frac{R_b}{a_1}\right)^2} & 0 \\ 0 & 0 & 0 & 1 \\ -\dfrac{M_{ball}}{J_{bm}} & 0 & 0 & -\dfrac{\left(\frac{KK_e}{R}+b\right)}{J_{bm}} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{K}{RJ_{bm}} \end{bmatrix} V$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} X$$

Substituting values of the system parameters, shown in Table 3.1, into Equation (3.23), following Equation can be derived:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 3.7731 & 0 \\ 0 & 0 & 0 & 1 \\ -5.170 & 0 & 0 & -105.1 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 16.85 \end{bmatrix} V \qquad (3.25)$$

The following chapter presents the simulation and experiment results of the ball and beam system. Several control structures, full state feedback, full state estimate, and reduced order estimate will be investigated.

Chapter 4

Simulation & Experiment

In this chapter, several different control structures, which are ‘full state feedback control’, ‘full state estimate control’, and ‘reduced ordered system control’, will be investigated. When the control structure involves gains (Feedback gains & Estimate gains) calculation, pole placement, LQR method, and LQG method are employed. The comparative results from simulation and experiment are presented.

Simulation plays a very important role in the control process as the final performance of the system can be predicted through simulation. Thus, the results from simulation can be a guide in ‘real’ design processes. In many cases, final budget and time can be efficiently reduced through the simulation process.

4.1 Hardware and software setup

It is necessary to discuss the hardware and software setup before conducting the simulations and experimentations for the ball and beam system. A detailed discussion

about the control hardware and software setup is presented in Appendix F.

The sensors, including the resistive wire position sensor and the digital encoder, are fully calibrated before the experiments. The sensors calibration is presented in Section 6.1. The DC motor used in the ball and beam system is nonlinear. The DC motor tuning is presented in Section 6.3.

4.2 Introduction to the control structures

The control structures employed in this chapter include the full state feedback control, the full state estimator control, and the reduced order estimator control. In the following, a brief discussion about these control structures is presented.

A state space model of a typical plant is given by:

$$\dot{x} = Ax + Bu \tag{4.1}$$

$$y = Cx + Du \tag{4.2}$$

where Equation (4.1) is the state equation, Equation (4.2) is the output equation, A is the state matrix, B is the input matrix, C is the output matrix, D is the direct transmission matrix, x is the plant states, y is the outputs of the plant, and u is the vector of input signals (Cazzolato 2006).

4.2.1 Full state feedback control

The idea of the feedback in state space control is similar to that in the classical control system. The control signal in the full state feedback control is defined as follows:

$$u = kx \tag{4.3}$$

where k is the proportional control gains.

Using Equations (4.1), (4.2) and (4.3), following new state matrix can be derived:

$$\dot{x} = (A-Bk)x + Bu \tag{4.4}$$

$$y = Cx + Du \tag{4.5}$$

The closed-loop system dynamic is governed by Equation (4.4). The control signal of u is a linear combination of plant states, which are fully accessible. The essential problem of the full state feedback control is calculating control gains k to achieve desired performance.

The full state feedback control structure of the ball and beam system for simulation is shown in Figure 4.1. That for experiment is shown in Figure 4.2.

4.2.2 Full state estimator control

The full state estimator control is a control structure which employs state feedback using full state observers. '*An observer is used to calculate state variables that are not accessible from the plant.'*(Dorf & Bishop 2005). The full-state observer for the system defined in Equation (4.1) and (4.2), assuming D is zero, is:

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x}) \tag{4.8}$$

where $\hat{x}$ represents the estimate of the state x, The L is the observer gain matrix (Dorf & Bishop 2005). The plant controlled by full state feedback with the observer is given by:

$$\dot{\hat{x}} = (A - Bk - LC)\hat{x} + Ly \tag{4.9}$$

$$u = -k\hat{x} \tag{4.10}$$

The full state estimator system dynamic is governed by Equation (4.9). The control signal of u is a linear combination of estimated plant states, which are observed from the measured states y. The essential problem of the full state estimate control is calculating observer gains L and control gains k to achieve desired control performance. The observer gains L should be chosen much faster than control gains k in order to predict the system states quick enough to pick up system dynamic.

The full state estimator control structure of the ball and beam system for simulation is shown in Figure 4.1 by double clicking the manual switch. That for experiment is shown in Figure 4.2.

4.2.3 Reduced order estimator control

The reduced order estimator control structure integrates the full state feedback with reduced order observers, which estimate only what is required to estimate. The states of the system, which is controlled by reduced order observers, need to be divided into

what can/cannot be measured as follows:

$$x = \begin{bmatrix} x_a \\ x_b \end{bmatrix} \tag{4.11}$$

where x_a is the states which are measurable, and x_b is the states which are estimated.

The state matrix A, the input matrix B, and the output matrix C for reduced order system should be redefined as follows (Cazzolato 2006):

$$A = \begin{bmatrix} A_{aa} & A_{ab} \\ A_{ba} & A_{bb} \end{bmatrix} \tag{4.12}$$

$$B = \begin{bmatrix} B_a \\ B_b \end{bmatrix} \tag{4.13}$$

$$C = \begin{bmatrix} I & 0 \end{bmatrix} \tag{4.14}$$

where A_{aa} is the state matrix for measurable states x_a, A_{bb} is the state matrix for estimated states x_b, B_a is the output matrix for measurable states x_a, and B_b is the output matrix for estimated states x_b.

The state equations for the reduced order system are given by (Cazzolato 2006):

$$\begin{bmatrix} \dot{x}_a \\ \dot{x}_b \\ \dot{x}_c \end{bmatrix} = \begin{bmatrix} A_{aa} + B_a D_r & A_{ab} & B_a C_r \\ A_{ba} + B_b D_r & A_{bb} & B_b C_r \\ B_r & 0 & A_r \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} + \begin{bmatrix} B_a \\ B_b \\ 0 \end{bmatrix} u \tag{4.15}$$

where $A_r = A_{bb} - LA_{ab} - (B_b - LB_a)k_b$

$B_r = A_r L + A_{ba} - LA_{aa} - (B_b - LB_a)k_a$

$C_r = -k_b$

$D_r = -k_a - k_b L$

x_c is the pseudo state estimates, which is defined as follows:

$$x_c = \hat{x}_b - Ly \tag{4.16}$$

where $\hat{x}_b$ represents the estimate of the state x_b, k_a is control gains for measurable states, k_b is control gains for estimated states.

The reduced order estimate system dynamic is governed by Equation (4.15). The control signal of u is a linear combination of estimated states with measured states. The essential problem of the reduced order estimate control is calculating observer gains L and control gains k_a, k_b to achieve desired control performance. The observer gains L should be chosen much faster than control gains k_b in order to predict the system states quick enough to pick up system dynamic.

The reduced order estimate control structure of the ball and beam system for simulation is shown in Figure 4.3. That for experiment is shown in Figure 4.4.

4.2.4 Command tracking

Strictly speaking, the full state feedback control, full state estimate control, and reduced order control only concern with regulator design. In order to achieve good command tracking, it is necessary to introduce a reference signal into the state controller, and add the integral controller into the closed loop (State Augmention) (Cazzolato 2006).

$$\dot{x}_I = Cx + Bu - r \tag{4.17}$$

where x_I is the augmented states, r is the reference input.

Control law for full state feedback, using integral controller:

$$u = \begin{bmatrix} k_0 & k_I \end{bmatrix} \begin{bmatrix} x \\ x_I \end{bmatrix} \tag{4.18}$$

where k_0 are the control gains from the plant, and k_I are the control gains for the error integral (augmented states)

Control law for full state estimation, using integral controller:

$$u = \begin{bmatrix} 0 & k_I & k_0 \end{bmatrix} \begin{bmatrix} x \\ x_I \\ \hat{x} \end{bmatrix} \tag{4.19}$$

Control law for reduced order estimate, using integral controller:

$$u = \begin{bmatrix} k_a & k_b & k_I \end{bmatrix} \begin{bmatrix} x_a \\ \hat{x}_b \\ x_I \end{bmatrix} \tag{4.20}$$

By adding augmented states to the original system, the bias between the reference input and the output of the system can be eventually removed.

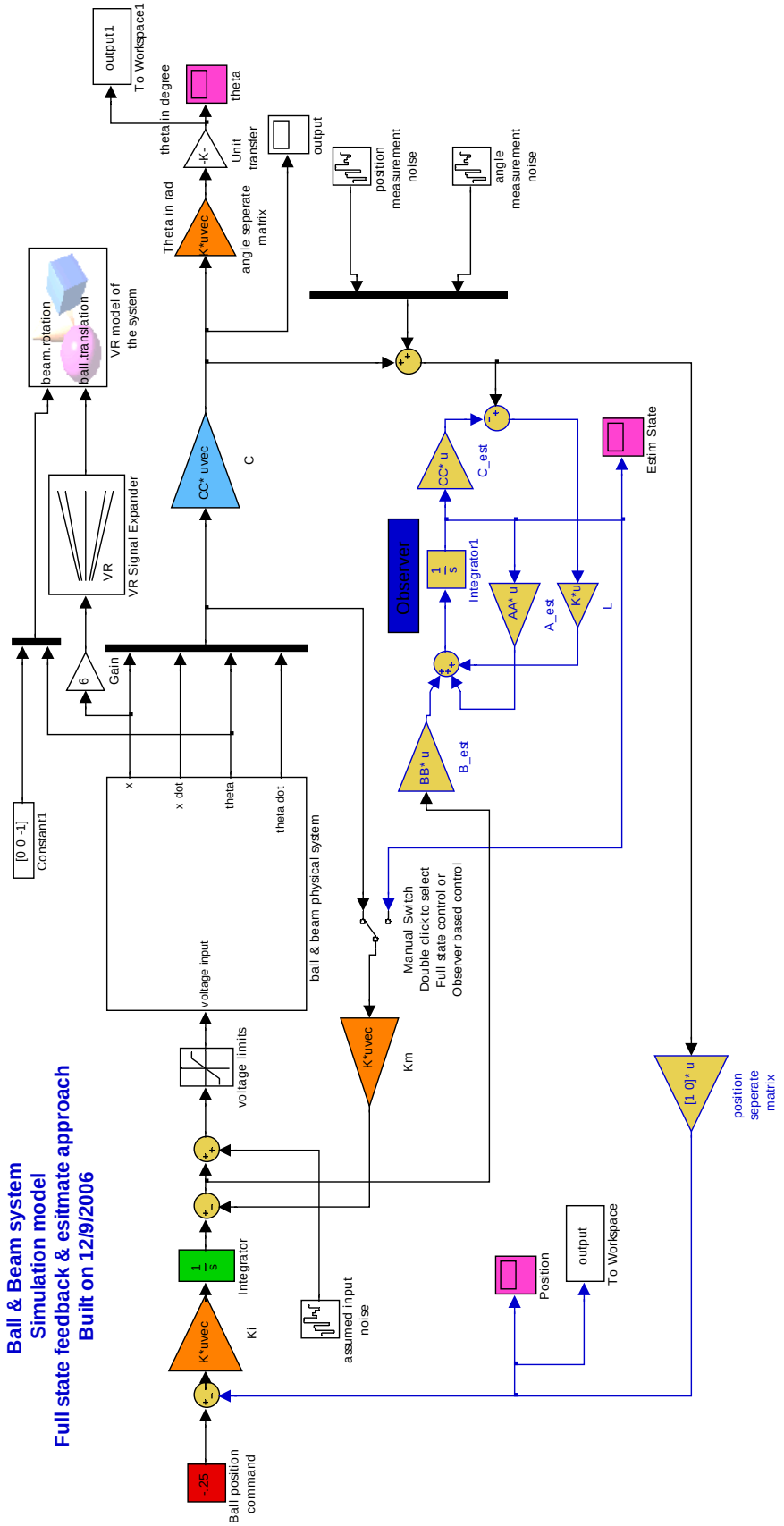

Figure 4.1: Full state feedback and estimate (simulation model) (Cazzolato 2006)

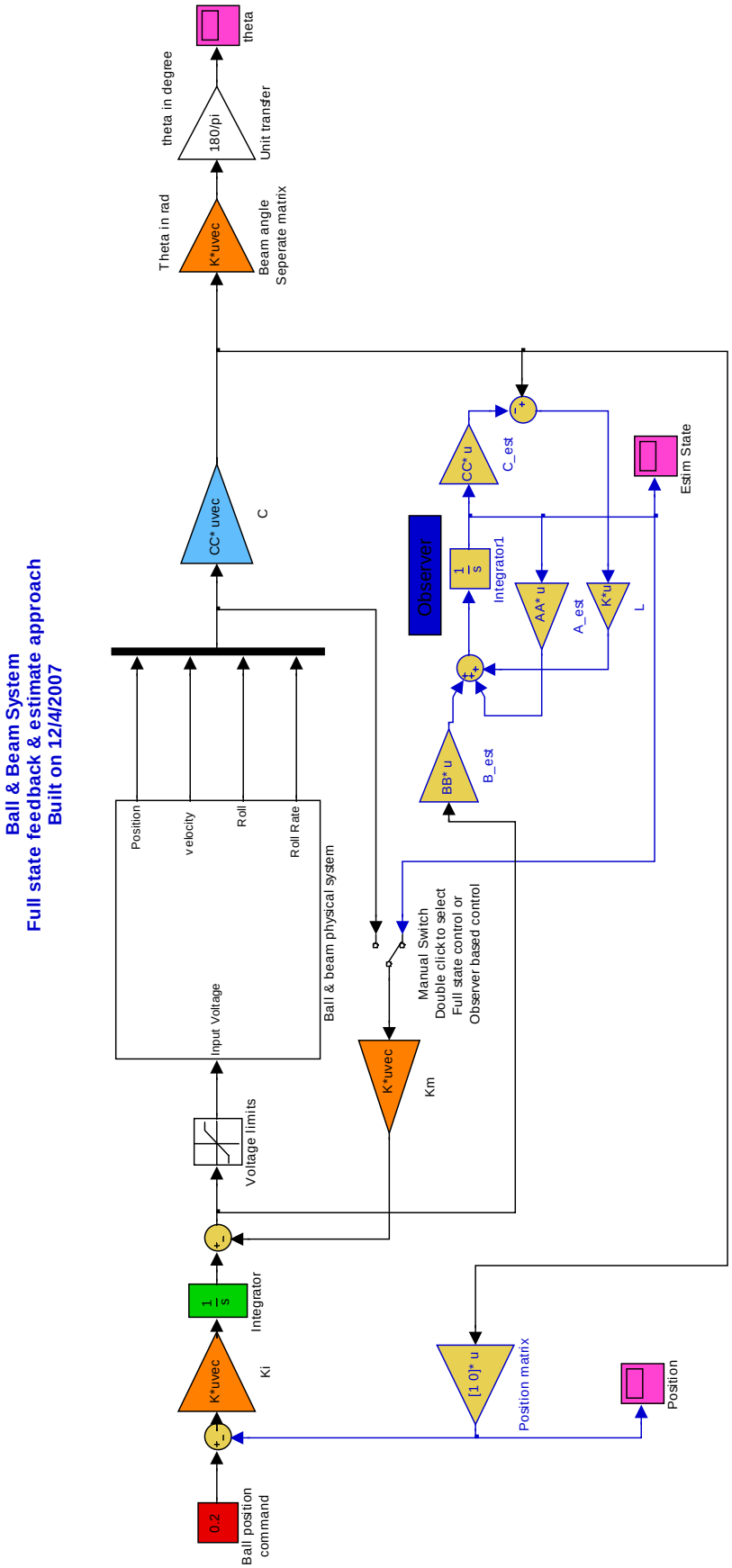

Figure 4.2: Full state feedback and estimate (experimental model) (Cazzolato 2006)

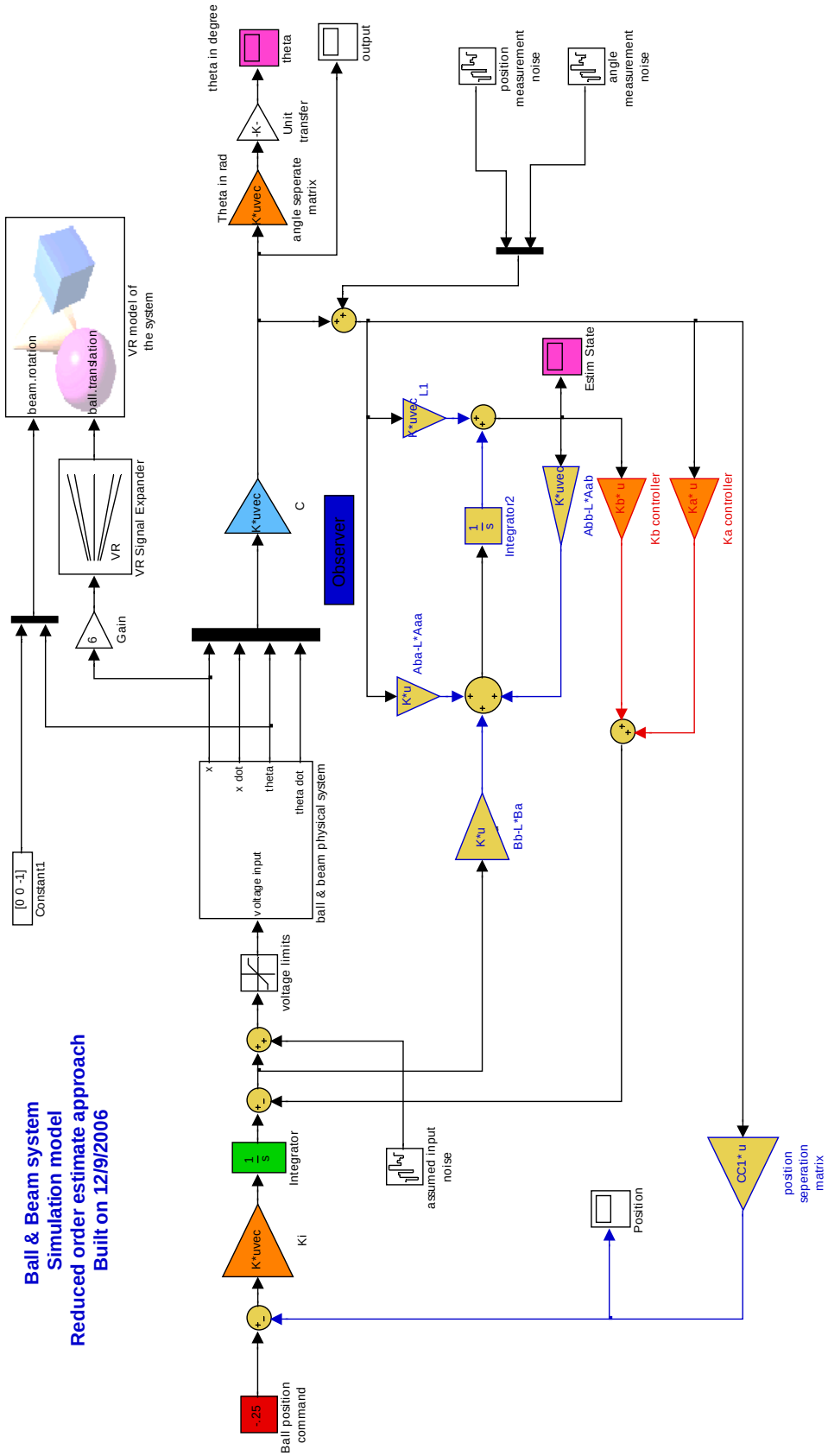

Figure 4.20: Reduced order system (simulation model) (Cazzolato 2006)

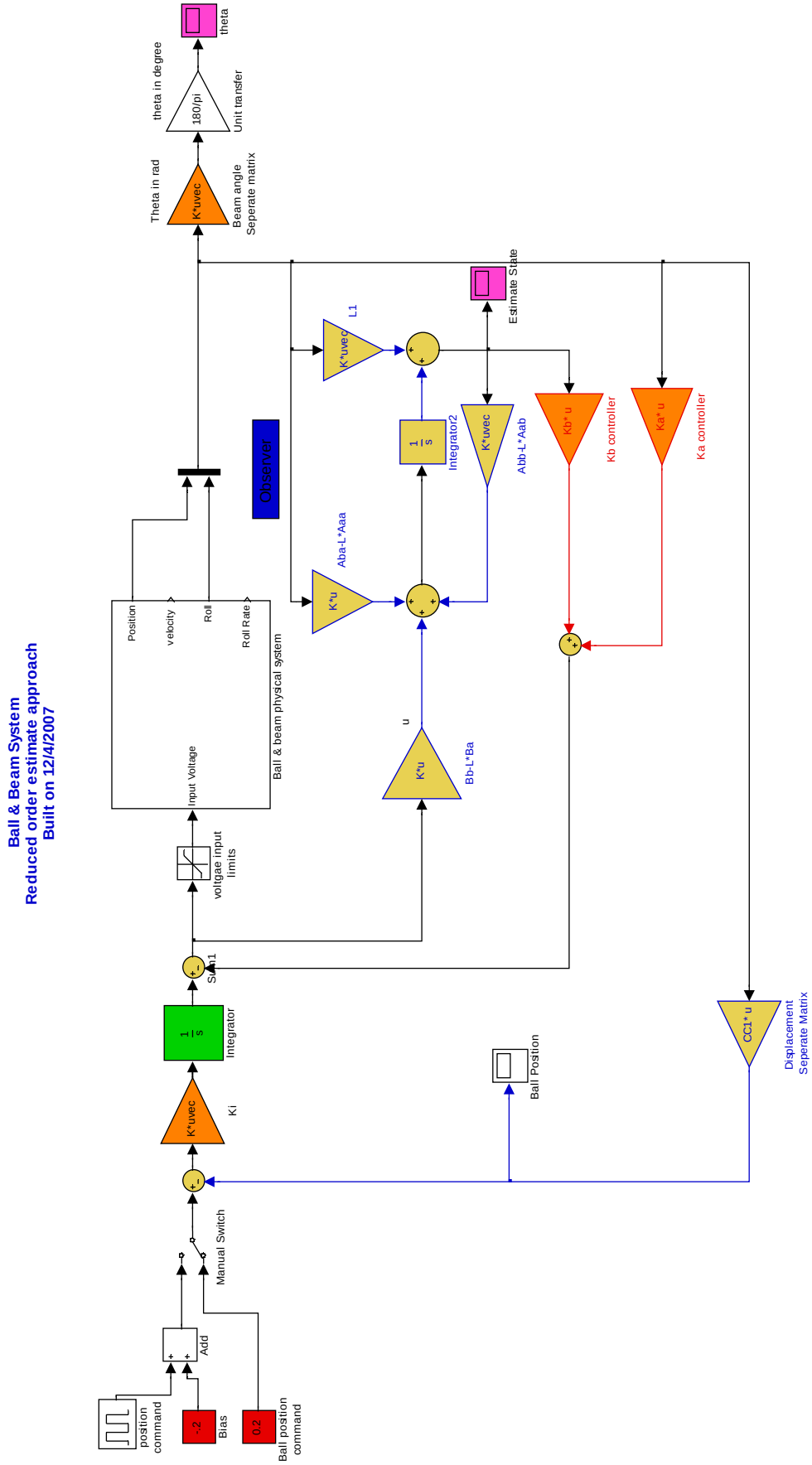

Figure 4.23: Reduced order system (experimental model) (Cazzolato 2006)

4.3 Introduction to gain calculation methods

4.3.1 Pole placement method

The pole placement is widely used in control engineering. The concept of the pole placement is that the closed loop poles of the system can be placed desired locations in order to satisfy some performance criteria, given that the system is completely controllable and observable. When involved in gain matrix calculation by pole placement, Ackermann's formula is the strong tool. The control gain matrix of SISO (signal-input, signal-output) systems derived by Ackermann's formula is given by (Dorf & Bishop 2005):

$$\mathrm{k} = \begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix} \mathrm{P_c^{-1}} q(\mathrm{A}) \tag{4.21}$$

where

$$q(\mathrm{A}) = \mathrm{A}^{\mathrm{n}} + \alpha_1 \mathrm{A}^{\mathrm{n-1}} + \cdots + \alpha_{n-1}\mathrm{A} + \alpha_n \mathrm{I} \tag{4.22}$$

$$\mathrm{P_c} = \begin{bmatrix} \mathrm{B} & \mathrm{AB} & \mathrm{A}^2\mathrm{B} & \cdots & \mathrm{A}^{\mathrm{n-1}}\mathrm{B} \end{bmatrix} \tag{4.23}$$

(α_1 α_2 ... α_n) in Equation (4.22) are the coefficients of the desired characteristic equation:

$$q(\lambda) = \lambda^{\mathrm{n}} + \alpha_1 \lambda^{\mathrm{n-1}} + \cdots + \alpha_{n-1}\lambda + \alpha_n \tag{4.24}$$

4.3.2 LQR method

The LQR stands for Linear Quadratic Regulator, which seeks a gain matrix that minimises some performance index J (Cazzolato 2006):

$$J = \int_{t}^{T} \left[x^{T}(\tau) Q x(\tau) + u^{T}(\tau) R u(\tau) \right] d\tau \tag{4.25}$$

where Q is the state weighting matrix, and R is the control weighting matrix. The optimal gain matrix is calculated by following Equations:

$$k = R^{-1} B^{T} P \tag{4.26}$$

where P is determined by solving the algebraic Riccati Equation, shown in follwing:

$$A^{T} P + PA - PBR^{-1} B^{T} P + Q = 0 \tag{4.27}$$

It is important to note that '*a unique, positive definite solution P, which minimises the performance index J* ' (Cazzolato 2006), can be determined by Equation (4.27), given that the system is controllable.

4.3.3 LQG method

The LQG stands for Linear Quadratic Gaussian, which is used to optimize the observers' performance by considering the process noise and measurement noise in the observer design process. The state equation with noises is given by (Cazzolato 2006):

$$\dot{x} = Ax + Bu + Gw \tag{4.28}$$

$$y = Cx + Du + v \tag{4.29}$$

where w is the process noise, v is the sensor noise, G is the process noise input matrix.

The observer gain matrix L is derived to make '*a compromise between good disturbance rejection (high observer gain) and filtering sensor noise (low observer*

gain') (Cazzolato 2006). The following Equation is used to determine the observer gain matrix L:

$$L = PC^T R^{-1} \tag{4.30}$$

where R is defined as: $E\{v\,v^T\} = R$, and P is determined by the following Ricccati Equation:

$$AP + PA^T - PC^T R^{-1} CP + GQG^T = 0 \tag{4.31}$$

where Q is defined as: $E\{w\,w^T\} = Q$

It is important to note that a unique solution for p by solving Equation (4.31) can be ensured, given that the system is completely controllable and observable.

4.4 Full state feedback control

A full state feedback control structure is the simplest control technique, as full state feedback control requires less calculation and has an efficient control structure. In the early design stage, those advantages are extremely useful.

A basic requirement use of this structure is that all the states in the system can be measured. Otherwise, full state feedback control structure cannot be used. In the ball and beam system, two states of the system, which are ball's position on the beam and beam's angle, are directly measured. The other two states, which are the speed of the ball and the angular speed of the beam, are derived from the differential of ball's position and beam's angle respectively. The direct differential is very noisy and hence,

may influence the system's performance.

The simulation model built is shown in Figure 4.1. By double clicking the Manual switch, the full state feedback control structure will be transferred to the full state estimate structure. Three noise sources are added in order to predict input and output noise of the actual system. An augmented state is presented for command following. The frequency of the simulation is set to 1000Hz. The saturation in Figure 4.1 limits the voltages into the motor in order to protect motor from voltage overload. The upper limit is 4.5 volts, and lower limit is -4.5 volts.

A 'VR' model was built to provide a visual illustration of the simulation process as shown in Figure 4.5.

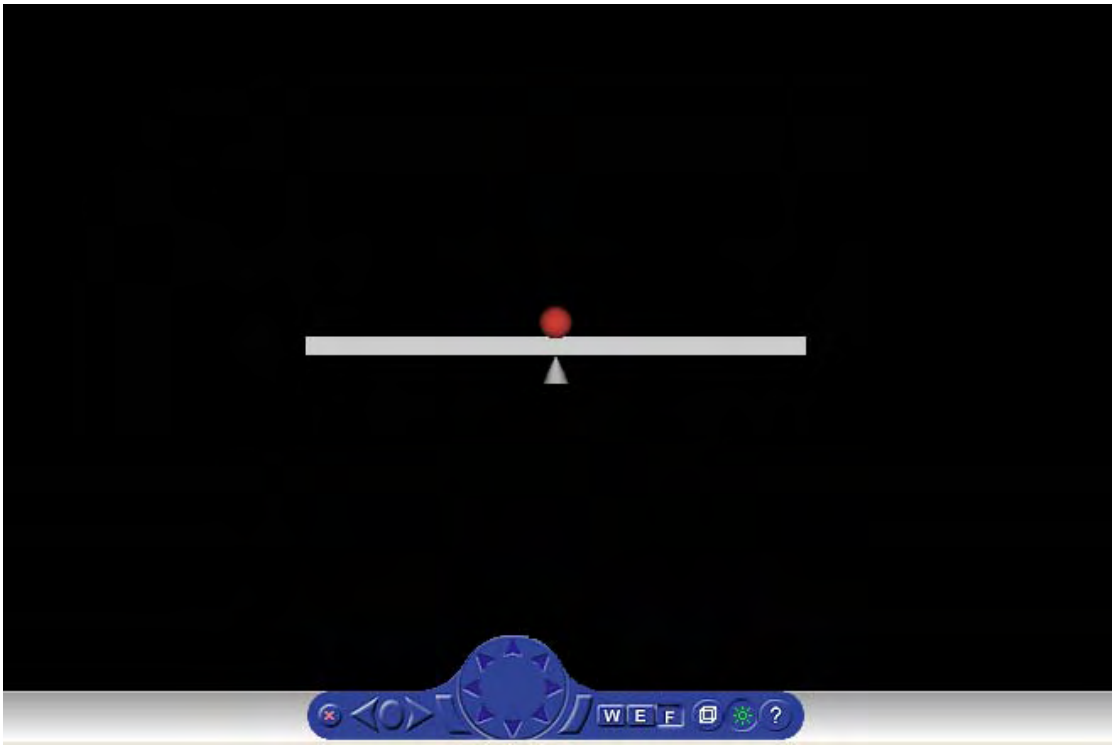

Figure 4.5: VR model of the ball and beam system for simulation

Here the figure represents the interior view of the 'VR' model for the ball and beam

system, by which the simulation can be visually demonstrated from the model.

An introduction to the full state feedback is presented in Section 4.2.1.

4.4.1 Pole placement method

In many cases, a gain of the system can be derived by pole placement method. Pole placement method requires users to provide desired close-loop poles for the gain matrix calculation. Before calculation, a check of controllability is necessary as if the system cannot be controlled, the gain cannot be calculated by the pole placement method.

The theoretical background of the pole placement method can be found in Section 4.3.1.

Trough iteration, the following closed loop poles were chosen for the simulation and experimentation.

Desired closed loop poles: [-2 -2 -5 -1 -10] rad/s

Those poles may not be the best, but they give a relatively good system performance. The control gain calculated using those poles by Ackermann's formula Equation (4.21) is presented as follows:

k_0 = [6.928volt/m 5.882volt/(m/s) 7.893volt/rad -5.050volt/(rad/s)] (4.32)

$$k_I = 3.146 \text{volt/m} \quad (4.33)$$

4.4.1.1 Simulation

The simulation model is presented in Figure 4.1. In the simulation process, the noise power is related to the real situation. For the simulation input noise power of 0.01walt/Hz is chosen. For measurement of noise, the position measurement noise power chosen is 0.00001walt/Hz, and the angle measurement noise power chosen is 0.0000001walt/Hz. Input noise power is higher than measurement noise due to the nonlinearity of the motor which generates high level process noise. However, the position sensor is not as accurate as an angle sensor (high resolution encoder), thus the position sensor would be expected to have a high level noise. Those noises reflect actual noise levels of the ball and beam system in the experiment.

The results from the simulation using the above noise values are presented in Figure 4.6 and Figure 4.7. The variation of the output signals is due to the input and output noise. The result of the tilt angle (approximately -4 to 4 degree) of the beam is presented in Figure 4.6.

The output of the ball's position is shown in Figure 4.7. The final output tracks the reference command '0.2m'. The setting time is approximately 5 seconds which is reasonable for the ball and beam system. The variation of the output is due to the

noise.

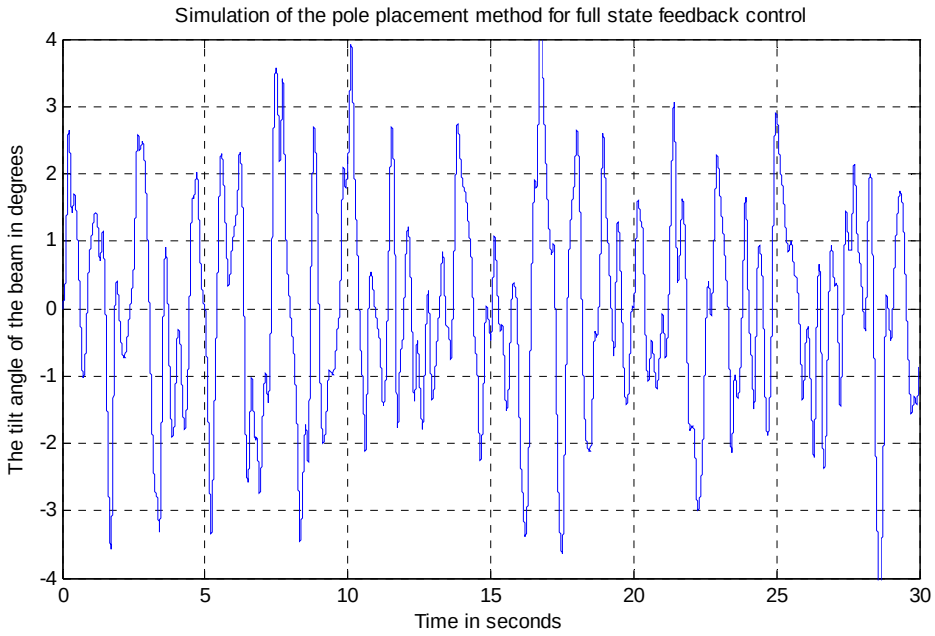

Figure 4.6: The pole placement simulation result of the beam tilt angle (in degrees)

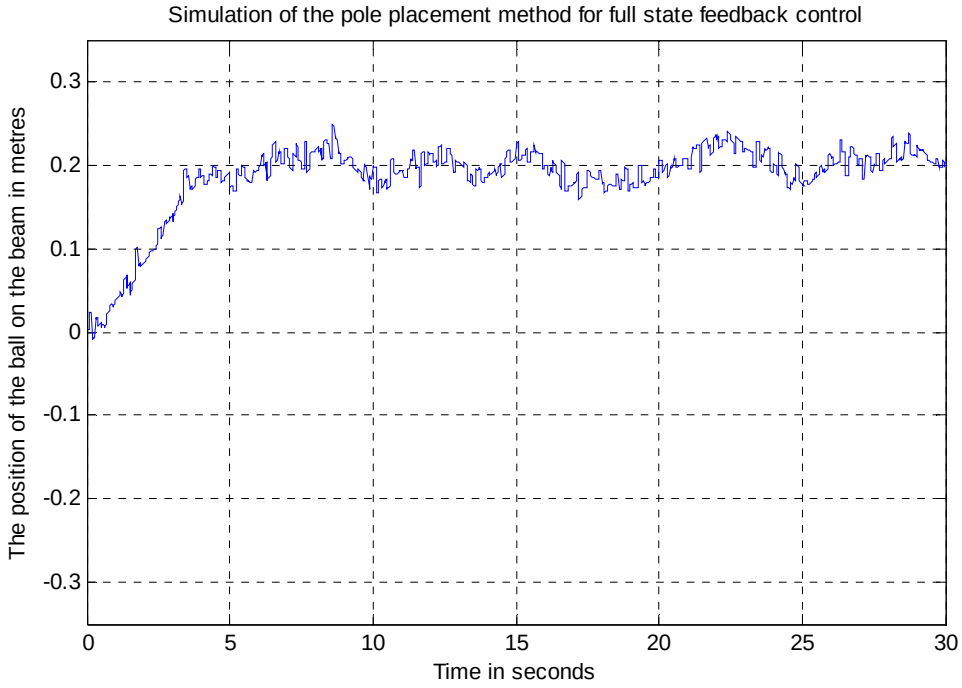

Figure 4.7: The pole placement simulation result of the position of the ball (in metres)

The discrete result of the ball position in Figure 4.7 above is due to the process and measurement noises being added in the control loop.

4.4.1.2 Experiment

The information about control hardware and software for the ball and beam system experiment can be found in Appendix F. The experimental model for full state feedback and estimation is shown in Figure 4.2. The reference point for the ball position in this experiment is 0.2, which means that the ball needs to be stopped at the position of ‘0.2m’. Figure 4.8, shown below, describes the interface between the physical system and the Quanser Q4 board (Quanser 2006). The interface includes the voltage input from the controller to the motor, the signal read from the digital encoder, and the signal measured from the resistive position sensor. The sensors calibration is discussed in detail in Chapter 6.

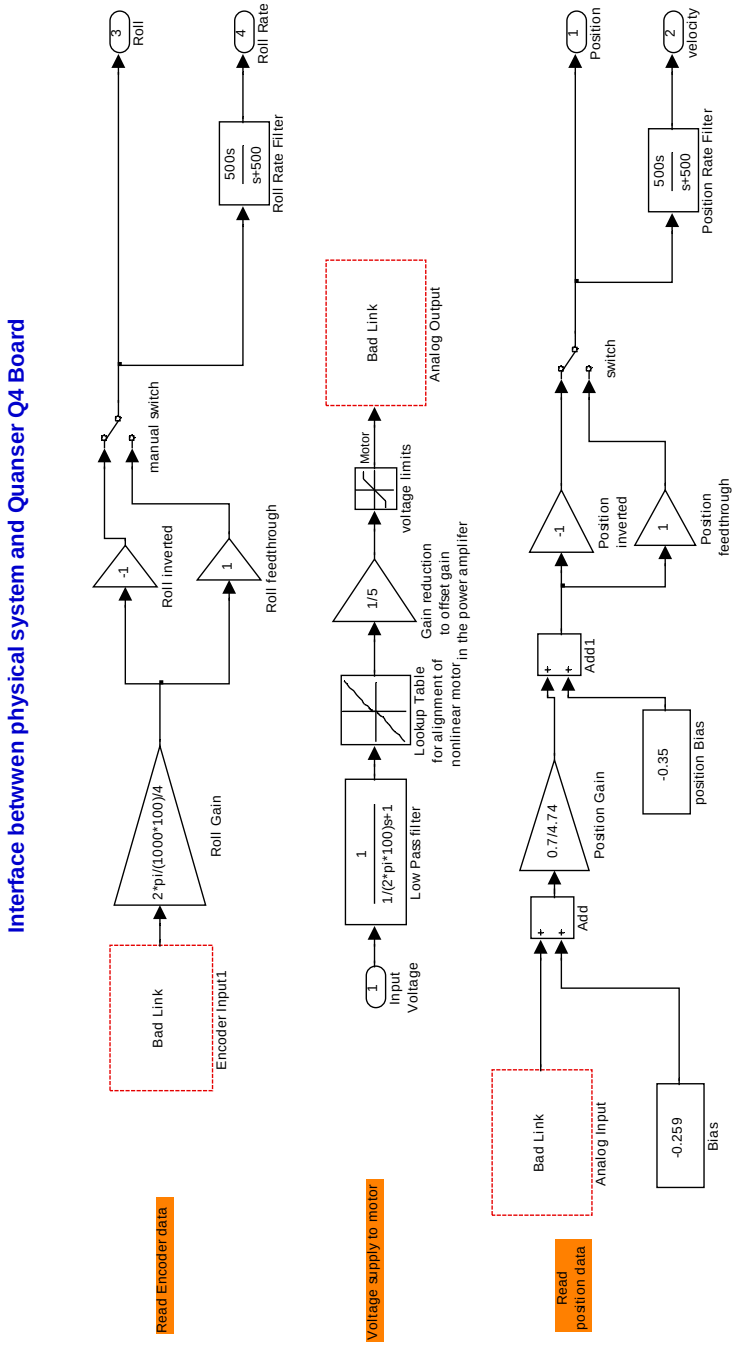

Figure 4.8: Interface between the physical system and Quanser Q4 board (Quanser 2006) ('Ball & Beam physical system' in Figure 4.2)

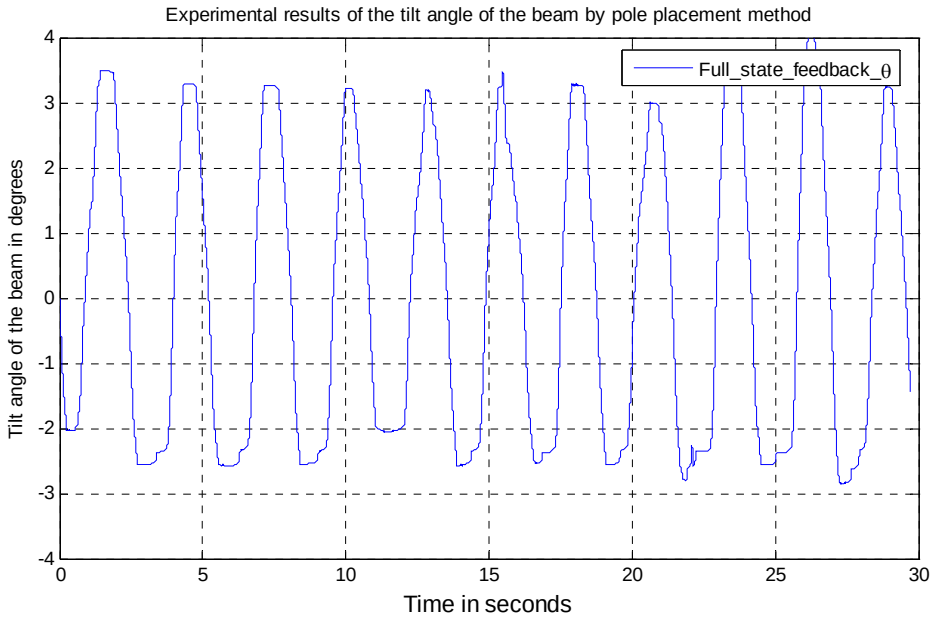

Figure 4.9: Experimental results of the tilt angle of the beam

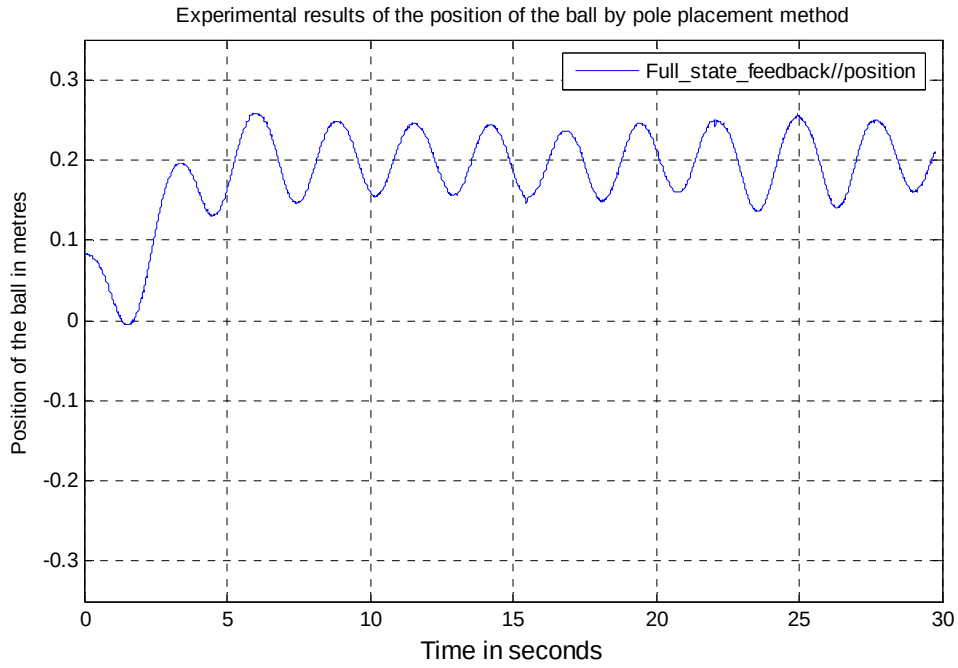

Figure 4.10: Experimental results of the position of the ball

Figure 4.10, shown above, displays the experimental result of the position of the ball by using the pole placement method. The graph shows the setting time is about 5 seconds, and the ball rolls around the reference point 0.2. Figure 4.9 shows the

experimental result of the tilt angle the beam, which is between -3 and 4 degrees. As a result, motor must rotate forward and backward frequently. The results from the above experiment indicate that the ball and beam system has been controlled; however, the performance of the ball and beam system is not efficient because the system requires high level of control effort but achieves an unsatisfactory output. The oscillations of the ball position may be due to high level noise of the speed of the ball and the angular speed of the beam, since they are differential from the position of the ball and the beam tilt angle.

4.4.2 LQR method

The LQR method employs different arithmetic to calculate computer gain matrix compared with the pole placement method. The gain matrix calculated by the LQR method generally performs to a higher quality than the pole placement method. This increase in performance quality is because the LQR method takes into account actuator effort. The LQR method requires the users to designate the state weighting matrix ‘Q’ and the control weighting matrix ‘R’. The performance of the system can be adjusted appropriately by adjusting the weighting matrix.

An introduction to the LQR method is presented in Section 4.3.2. The noise level chosen in this section for the simulation process is the same as for the pole placement method. Additionally, the LQR system also includes an augment state for the

command following (see Section 4.2.4).

The following Q and R weighting matrices are chosen for the ball and beam system:

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 10 \end{bmatrix}; \qquad R = [0.05]$$

The values of the second and fourth diagonal elements are 1 instead of 10 because the velocity of the ball and the rotational speed of the beam, which are obtained by differentiating resistive sensor output and the digital encoder output, which induces noise. Therefore, the velocity of the ball and the rotational speed of the beam need to be penalized. The control weighting matrix R is chosen to limit the control signal size.

The control gain calculated using above Q and R is presented as follows:

$$k_0 = [29.67\text{volt/m} \quad 24.72\text{volt/(m/s)} \quad 40.94\text{volt/rad} \quad 1.748\ \text{volt/(rad/s)}] \tag{4.34}$$

$$k_I = 14.14\ \text{volt/m} \tag{4.35}$$

4.4.2.1 Simulation

The LQR simulation results are described as follows. Figure 4.11 shows the tilt angle of the beam is approximately -1 to 1 degree, which is more accurate than the -4 to 4 degrees shown in Figure 4.6. Figure 4.12 shows that the ball stops at the desired positions within 4 seconds, which is slightly better than the setting time of 5 seconds

presented in Figure 4.7.

From the simulation process, it is evident that LQR method results in a better control performance than the pole placement method. The improvement in control performance is due to the control gain (see Equations 4.34 and 4.35) calculated by LQR method is much larger (approximately 3 times) than that (see Equations 4.32 and 4.33) calculated by pole placement method. The large gains should perform better, given that the system is stable.

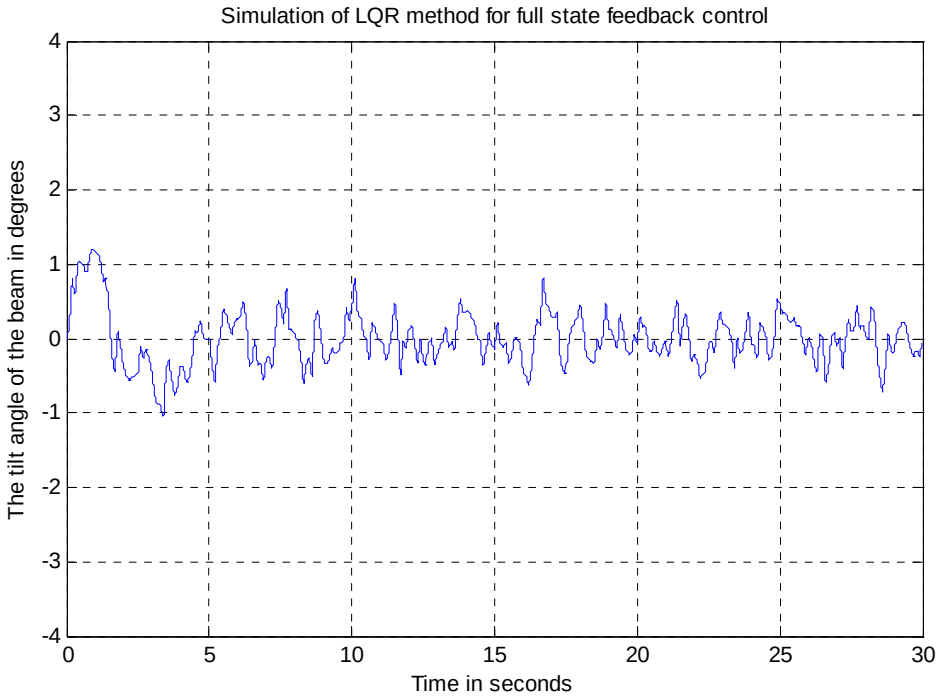

Figure 4.11: LQR simulation result of the tilt angle of the beam (in degrees)

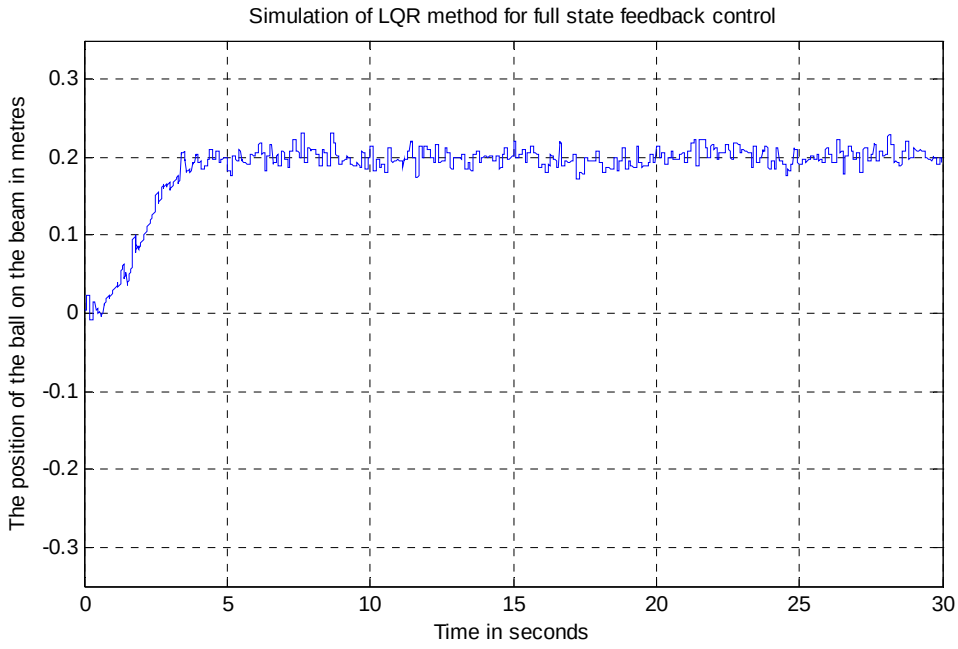

Figure 4.12: LQR Simulation result of ball's position (in metres)

4.4.2.2 Experiment

The experiment, shown below, further evidences the LQR method results achieve a more accurate performance. The reference of this experiment is '0.2m'. Figure 4.14 indicates that the ball rotates closely around the desired reference point. The tilt angle of the beam is between -1 to 1 degree, which is smaller than that in Figure 4.9 by the pole placement method. Some spikes can be detected in both Figure 4.13 and Figure 4.14, which may be caused by defects of the resistive wire, the ball that is not perfect round, and the nonlinearity of the motor.

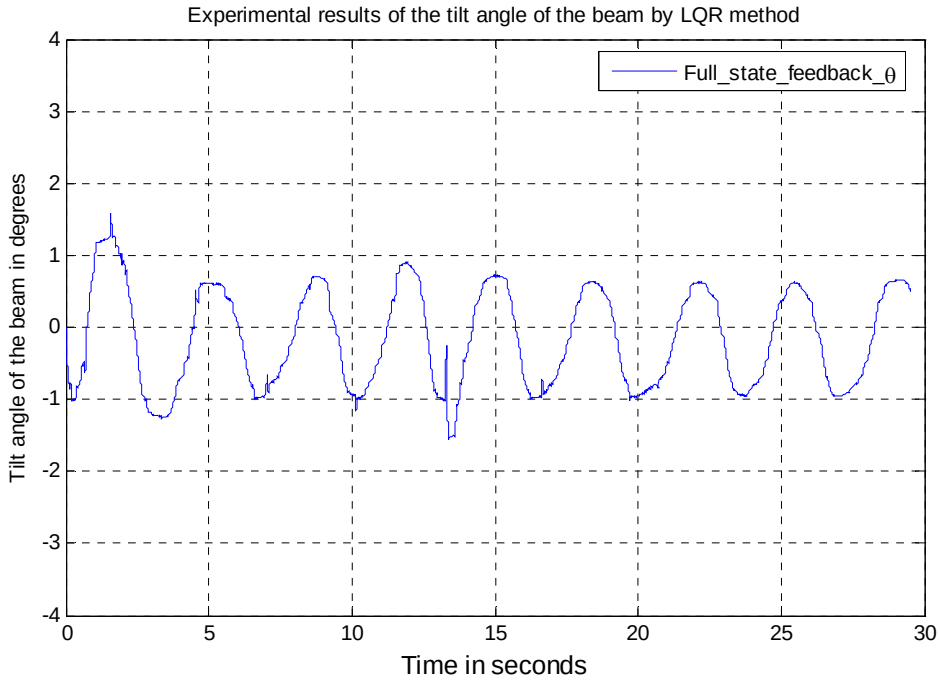

Figure 4.13: Experiment results of the tilt angle of the beam by LQR method

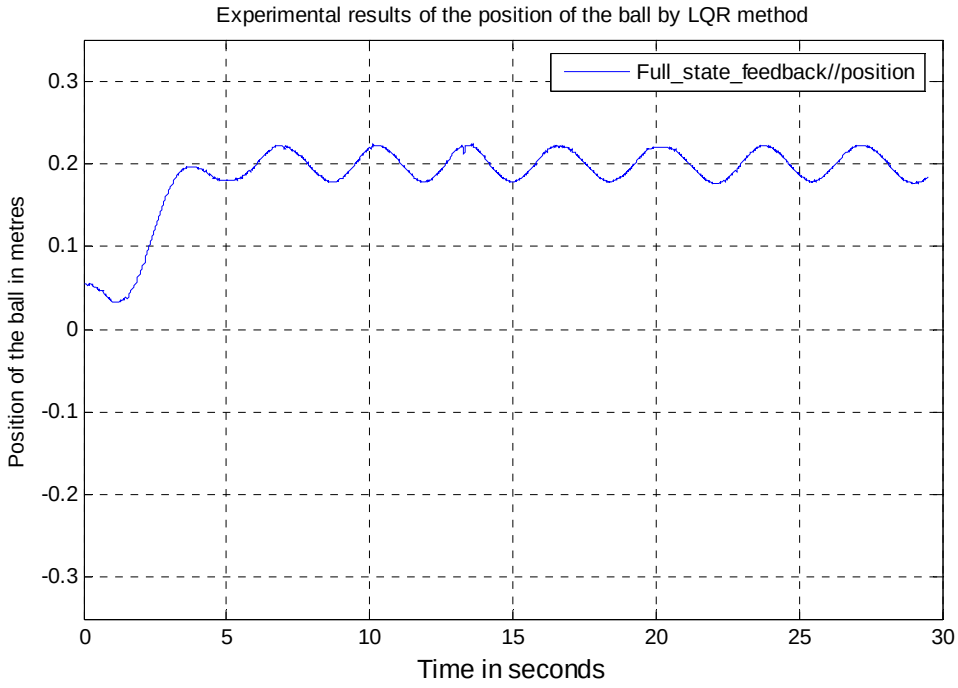

Figure 4.14: Experiment results of the position of the ball by LQR method

4.5 Full state estimate control

The full state estimate control structure can be used when some states of the ball and beam system cannot be measured which occurs as there are not as many sensors as states. In some situations, it is impossible to use sensors to measure the states due to the limitation of the sensor technique. Additionally the more sensors involved in the system the greater the cost.

All the states can be predicted from measured states through the full state estimate control structure (See Section 4.2.2 & Figure 4.1), given that the system is completely controllable and observable. Those predicted states are then fed back to the controller in order to derive the control signal. This structure has the advantage of less cost than the full state feedback for the ball and beam system since less sensors are required. The full state estimate control structure is highly related to the accuracy of the mathematical model because the estimated states are predicted based on this model.

The full state estimated control structure is shown in Figure 4.1. The manual switch needs to be double clicked to transfer from the full state feedback control structure to full state estimate control structure.

4.5.1 Pole placement method

Controller design:

The pole placement method is used to calculate the control gains and estimator gains for the ball and beam system. The poles chosen for calculation of the control gain is the same as in Section 4.4.1.

Desired closed loop poles: [-2 -2 -5 -1 -10] rad/s

The control gain is the same with that in Section 4.4.1 since the same desired closed loop poles are used. The control gains are represented as follows:

k_0 = [6.928volt/m 5.882volt/(m/s) 7.893volt/rad -5.050volt/(rad/s)]

k_I = 3.146volt/m

Observer design:

The estimator poles are chosen twenty-two times faster than the feedback poles through iteration. The estimated poles are present as follows:

Desired estimated loop poles: [-44 -44 -110 -22]rad/s

The estimation gain calculated by pole placement using Ackermann's formula Equation (4.21) is presented as follows:

$$L = \begin{bmatrix} 66.00(1/s) & 0 \\ 968.0(1/s^2) & 3.773(m/(rad \times s^2)) \\ 0 & 48.91(1/s) \\ -5.170(rad/(ms^2)) & -299.7(1/s^2) \end{bmatrix} \quad (4.36)$$

4.5.1.1 Simulation

The input and output noise for the simulation process is chosen as the same as that in Section 4.4.1. The reference point in this experiment is 0.1m. The simulation results of the ball and beam system with full state estimate control by pole placement method are presented in Figure 4.15 and 4.16. Figure 4.15 shows the tilt angle of the beam varies between -4 to 4 degrees. The system performance is poor regarding the ball rotation back and forward around the desired position, shown in Figure 4.16. The poor performance is because the full state estimate structure is very sensitive to the uncertainty of the model and the process and measurement noise of the system.

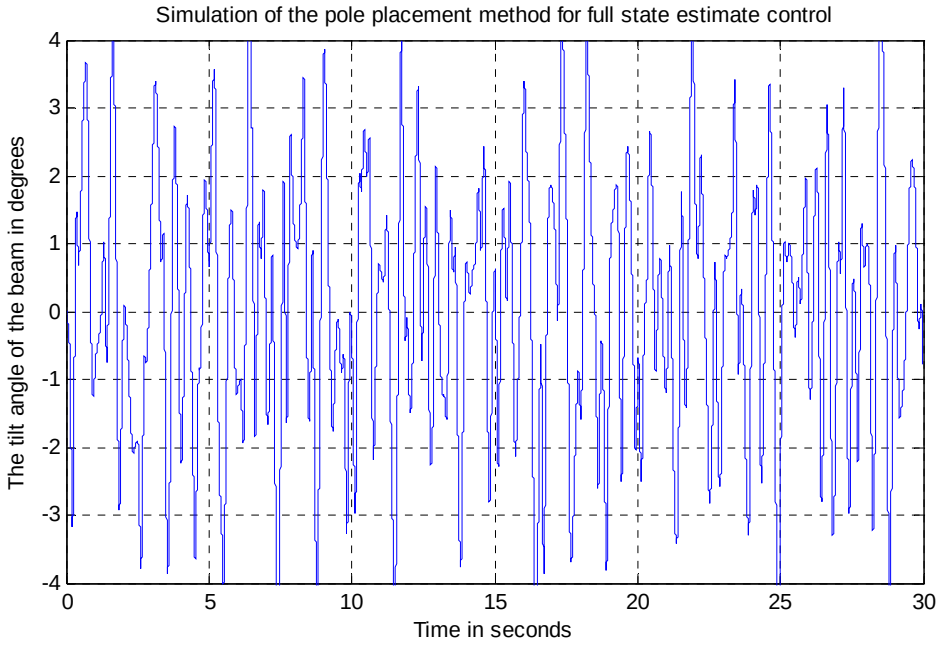

Figure 4.15: The pole placement simulation result of the tilt angle of the beam (in degrees) using a full state estimator

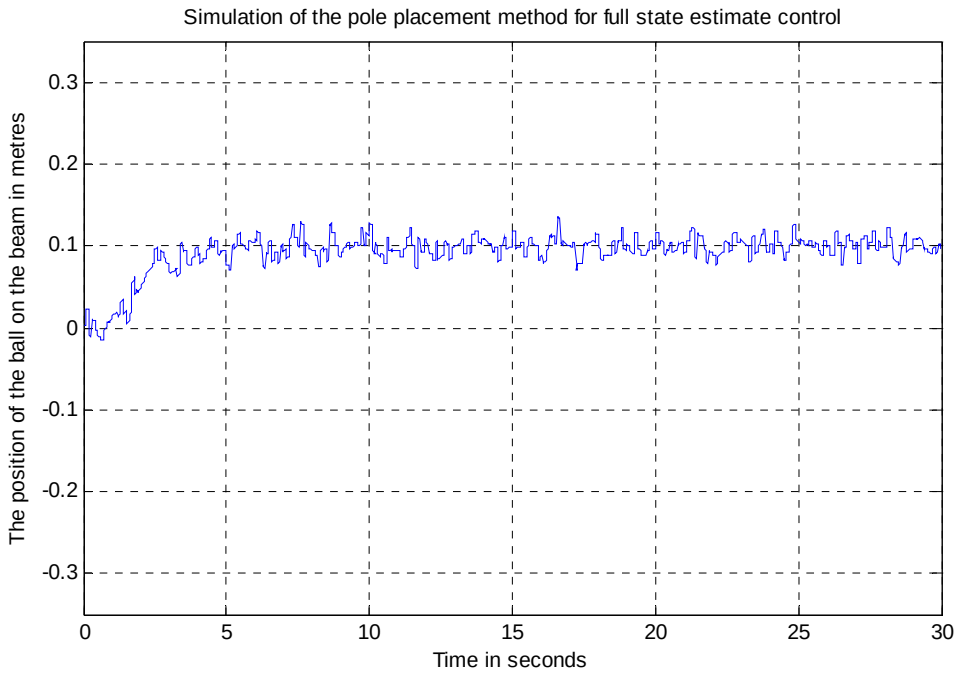

Figure 4.16: The pole placement simulation result of ball's position (in metres) using a full state estimator

4.5.1.2 Experiment

The experimental results of the full state estimate structure with pole placement method are presented in Figure 4.17 and 4.18. The reference point in this experiment is 0.1m.

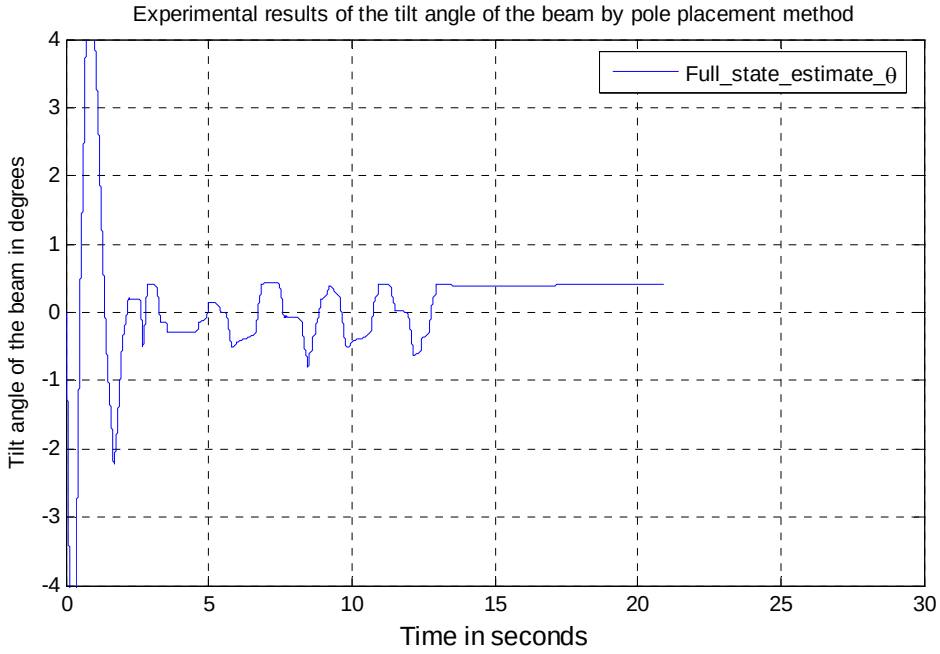

Figure 4.17: Experimental results of the tilt angle of the beam by pole placement method using a full state estimator

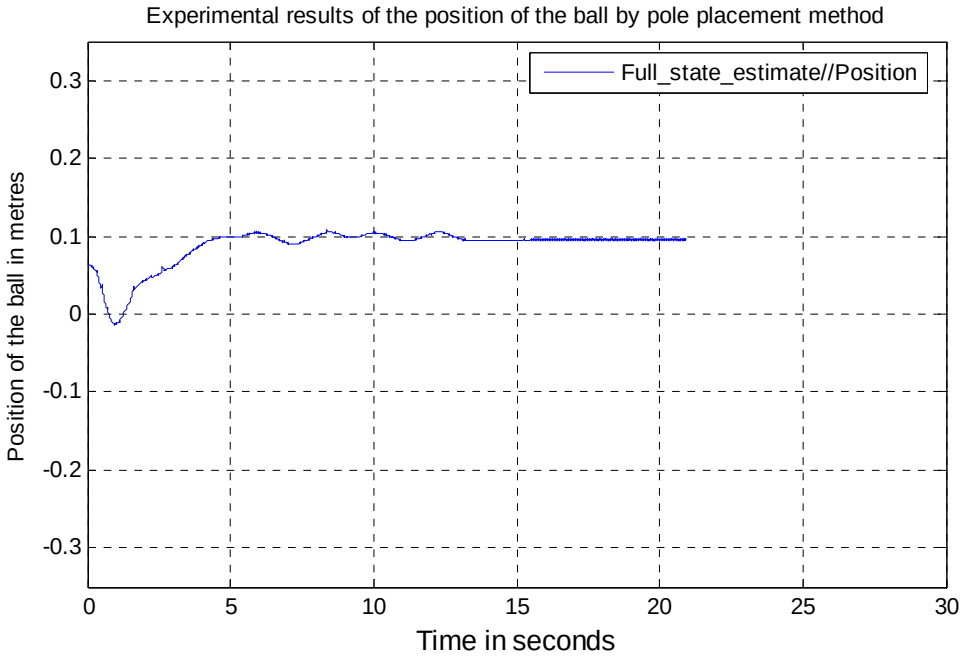

Figure 4.18: Experimental results of the position of the ball by pole placement method using a full state estimator

Figure 4.17 indicates the tilt angle of the beam is reasonable for the ball and beam system. Figure 4.18 represents the setting time of the ball and beam system at about 5 seconds, and the ball rotates to the desired value with small offset error. This error is due to static friction between the ball and resistive wire, which has not been considered in deriving the mathematical model of the ball and beam system. Overall, the system performance is acceptable.

The experimental results presented in Figures 4.17 and 4.18 is slightly better than the simulation results shown in Figures 4.15 and 4.16 may be because the power of the noises (see Section 4.4.1.1) chosen for the simulation are larger than that in the experiment.

4.5.2 LQR and LQG methods

Controller design:

The LQR method (see Section 4.3.1) was used to calculate the control gain for the ball and beam system when using a full state observer. The weighting matrix Q and R are chosen the same with that in Section 4.4.2 for control gain calculation, which are:

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 10 \end{bmatrix}; \qquad R = \begin{bmatrix} 0.05 \end{bmatrix}$$

The control gain calculated using above Q and R by LQR method is same with that in

Section 4.4.2, which is represented as follows:

k_0 = [29.67volt/m 24.72volt/(m/s) 40.94volt/rad 1.748 volt/(rad/s)]

k_I = 14.14 volt/m

Observer design:

The Q and R for the estimator gain calculation by LQG method (see Section 4.3.2) is presented as follows:

$$Q = 1000; \qquad R = 0.001 \times \begin{bmatrix} 5 & 0 \\ 0 & 0.01 \end{bmatrix}$$

The following estimator gains are derived by LQG method:

$$L = \begin{bmatrix} 0.5809(1/s) & 0 \\ 0.1687(1/s^2) & 3.772(m/(rad \times s^2)) \\ 0 & 932.5(1/s) \\ -0.011(rad/(m \times s^2) & 434830(1/s^2) \end{bmatrix} \tag{4.37}$$

Q is chosen much larger than R due to the process noise of the ball and beam system being significantly larger than the measurement noise. This is because the nonlinear motor has dead zones when the input voltage is small. Thus the large Q is selected. The second diagonal element in R is larger than the first one since the angle senor (digital encoder) is much more accurate than the position sensor (resistive wire).

4.5.2.1 Simulation

The input and output noise for the simulation is selected the same as that in Section

4.4.1.1. The results of the ball and beam system using LQR method with full state estimator control structure are presented as follows. Figure 4.19 shows that the beam tilts between -1 to 1 degree, which is small, compared that in Section 4.5.1.1. The ball moves to the desired location as required, as shown in Figure 4.20. The performance of the ball and beam system is improved due to the LQG method taking account of the noise into the design process.

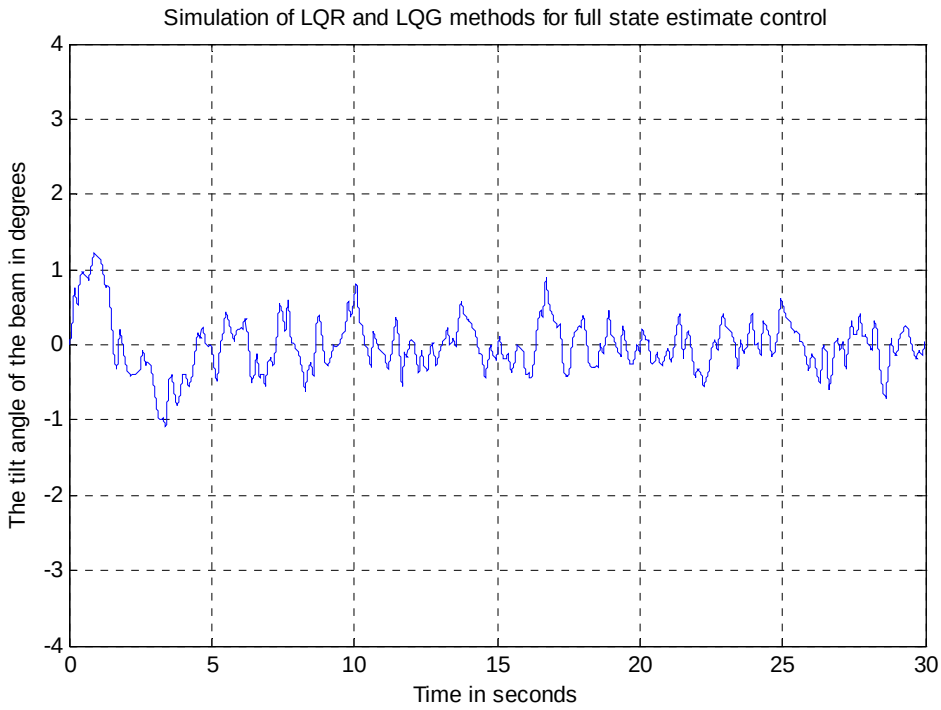

Figure 4.19: LQR and LQG simulation result of the tilt angle of the beam (in degrees) using a full-state estimator

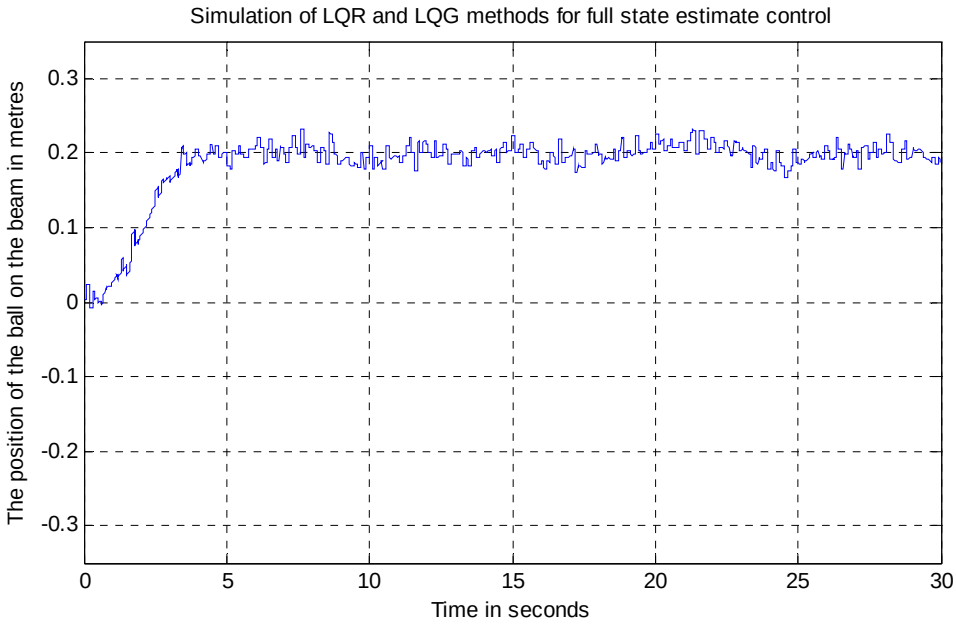

Figure 4.20: LQR and LQG simulation result of ball's position (in meters) using a full-state estimator

4.5.2.2 Experiment

The experimental results of the LQR estimator are presented in Figure 4.21 and Figure 4.22. Figure 4.21 presents the beam tilt angle of the ball and beam system. The beam tilt angle is only -0.5 to 0.5 degrees, which indicates the control effort of the system is reasonable. Figure 4.22 shows the desired position, which is 0.2m, for the ball is achieved. The 80% setting time of the system is at about 8 seconds, which is acceptable. Overall, the control performance of the LQR estimator is good.

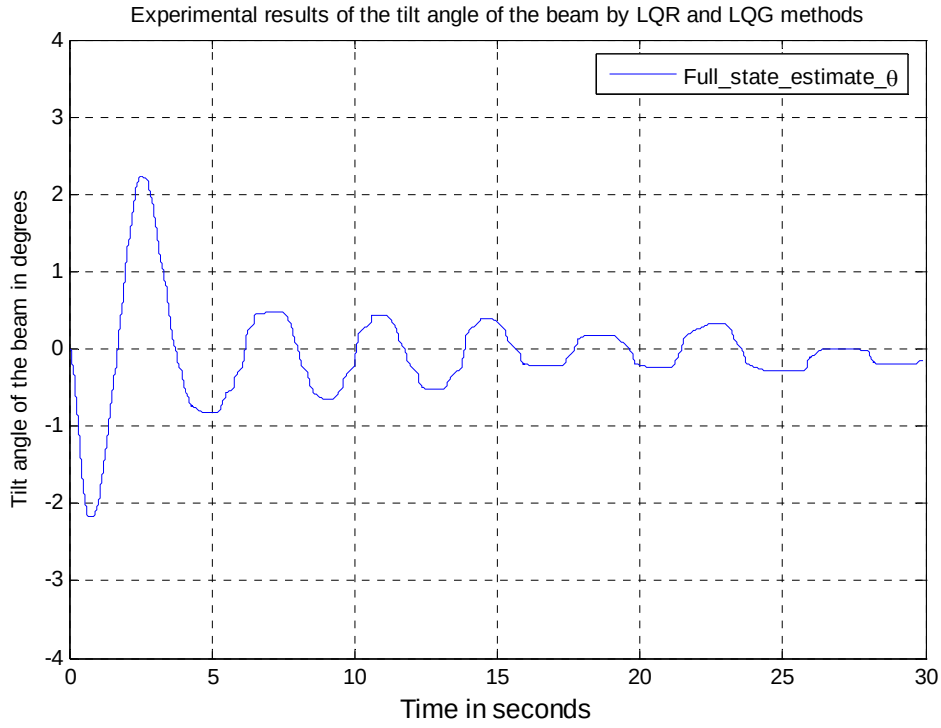

Figure 4.21: Experimental results of the tilt angle of the beam by LQR and LQG method using a full-state estimator

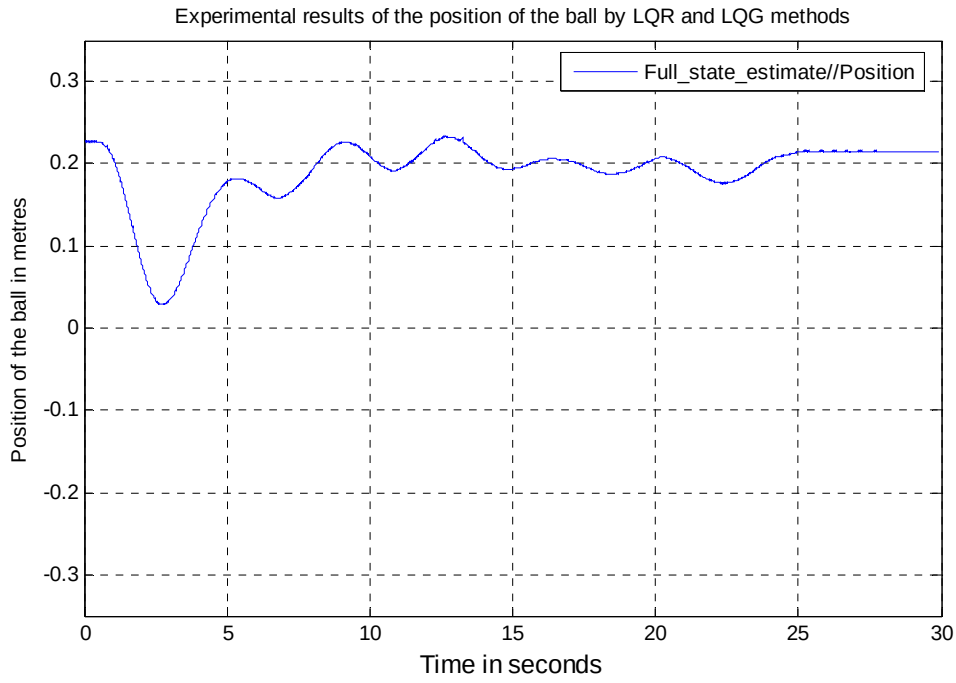

Figure 4.22: Experimental results of the position of the ball by LQR and LQG method using a full-state estimator

The experiment results of the LQR and LQG methods, presented in the Figures 4.21 and 4.22, are not as good as those in Figures 4.17 and 4.18 by pole placement method. This may be because the mathematical model of the ball and beam present in Chapter 3 is very accurate. Additionally, Q and R used in LQG method are guessed not measured (out of this project scope), therefore, they may contain large level error.

4.6 Reduced order system (LQR and LQG methods)

Some states of the system can be measured by sensors. In the ball and beam system, the position of the ball and tilt angle of the beam can be directly measured. However, some other states which cannot be measured need to be estimated from the measured states. Therefore, the reduced order system is used in this case to estimate only what unknown states are required to be estimated.

In order to construct a reduced order system, the states should be rearranged into measurable states and estimated ones (see Section 4.2.3). Therefore, Equations (3.23) and (3.24) are rearranged as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \dfrac{g}{1+\dfrac{2}{5}\left(\dfrac{R_b}{a_1}\right)^2} & 0 & 0 \\ \dfrac{-M_{ball}g}{J_{bm}} & 0 & 0 & \dfrac{-\left(\dfrac{KK_e}{R}+b\right)}{J_{bm}} \end{bmatrix} \begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{K}{RJ_{bm}} \end{bmatrix} V \qquad (4.38)$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} X \tag{4.39}$$

An introduction to the reduced order system was presented in Section 4.2.3.

Controller design

LQR method is used to calculate the feedback and estimate gain matrix, the weighting matrix Q and R for feedback gain calculation are presented as follows:

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}; \qquad R = \begin{bmatrix} 0.05 \end{bmatrix}$$

The fifth diagonal element which represents the augmented state in the matrix Q is chosen to 2 in order to decrease the setting time of the ball and beam system (fast response). The matrix R is chosen to limit the control voltage supplied to the motor (control effort penalty).

The control gain is presented as follows:

$$k_a = [23.96 \text{ volt/m} \quad 61.98 \text{ volt/rad}] \tag{4.40}$$

$$k_b = [30.73 \text{ volt/(m/s)} \quad 9.374 \text{ volt/(rad/s)}] \tag{4.41}$$

$$k_I = 6.325 \text{ volt/m} \tag{4.42}$$

where k_a, k_b, and k_I are defined in Section 4.2.3.

Observer design

The matrix Q and R for estimate gain calculation are presented as follows:

$$Q = [10]; \qquad R = \begin{bmatrix} 1 & 0 \\ 0 & 0.01 \end{bmatrix}$$

The estimator gain calculated by LQG method is presented as follows:

$$\mathrm{L} = \begin{bmatrix} 11.93\left(1/\mathrm{s}^2\right) & 0 \\ 0 & 88.21(1/\mathrm{s}^2) \end{bmatrix} \tag{4.43}$$

The process noise (disturbance) due to the nonlinearity of the motor is much larger than the measurement noise. Therefore, Q is chosen larger than R. The second diagonal element which represents the measurement noise of the encoder in the matrix R is chosen smaller than the first diagonal element because the signal from the digital encoder contains less noise than the linear position sensor.

4.6.1 Simulation

Figure 4.3 shows the structure of the reduced order system for the ball and beam system. It can be identified that the feedback gain matrix has been separated into two control gains, ‘Ka’ and ‘Kb’, where ‘Ka’ is the feedback gain matrix for measured states and ‘Kb’ is for the estimated states. An augmented state is added into the system for command following (see section 4.2.4). The noise sources which are chosen to reflect the real experimental noise level, and they have the same value with Section 4.4.1.1.

The simulation results of the ball and beam system are presented in Figure 4.23 and

Figure 4.24. Figure 4.23 shows the tilt angle of the beam is about -2 to 2 degree. Figure 4.24 shows the setting time of the system is about 6 seconds. The setting time is slightly larger than the full state feedback control due to the control effort penalty. The result is the best for all the simulation process, since there is no overshoot, and the ball only deflects a little from the desired position. Overall the performance of the system is improved through estimation of the states required.

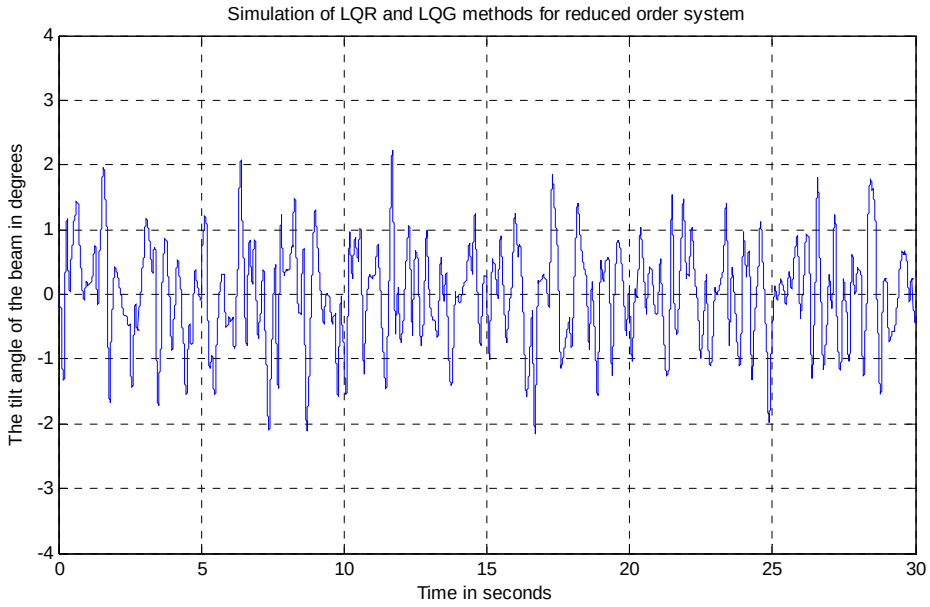

Figure 4.23: LQR simulation result of the tilt angle of the beam (in degrees) for reduced order observer

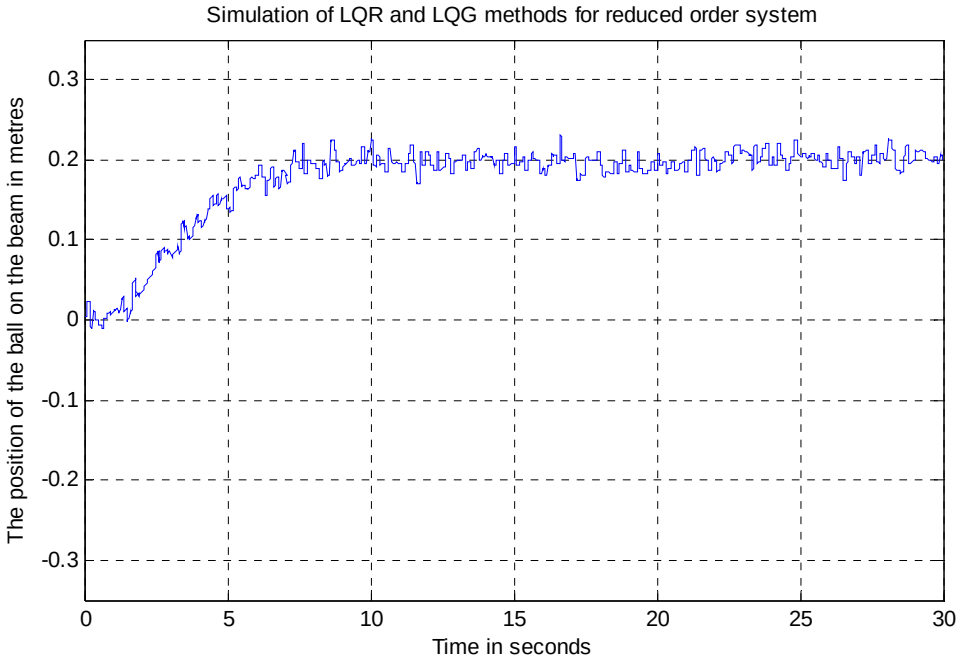

Figure 4.24: LQR Simulation result of ball's position (in metres) for reduced order observer

4.6.2 Experiment

The experiment of the reduced order system by LQR method is presented as follows. The model used in the experiment model is shown in Figure 4.4. The reference input of this experiment is 0.2m. The inner structure of the system is shown in Figure 4.8. The final results of this experiment are displayed in Figure 4.25 and Figure 4.26.

Figure 4.25 illustrates the tilt angle of the beam during the experiment. The tilt angle of the beam is a good outcome due to less control effort being required.

Figure 4.26 shows the desired system performance is achieved. The setting time of the system is less than 8 seconds. The ball stops at desired position with a slight steady stead error due to the static friction between the ball and beam. Overall the system

controlled by reduced order system is pleasant.

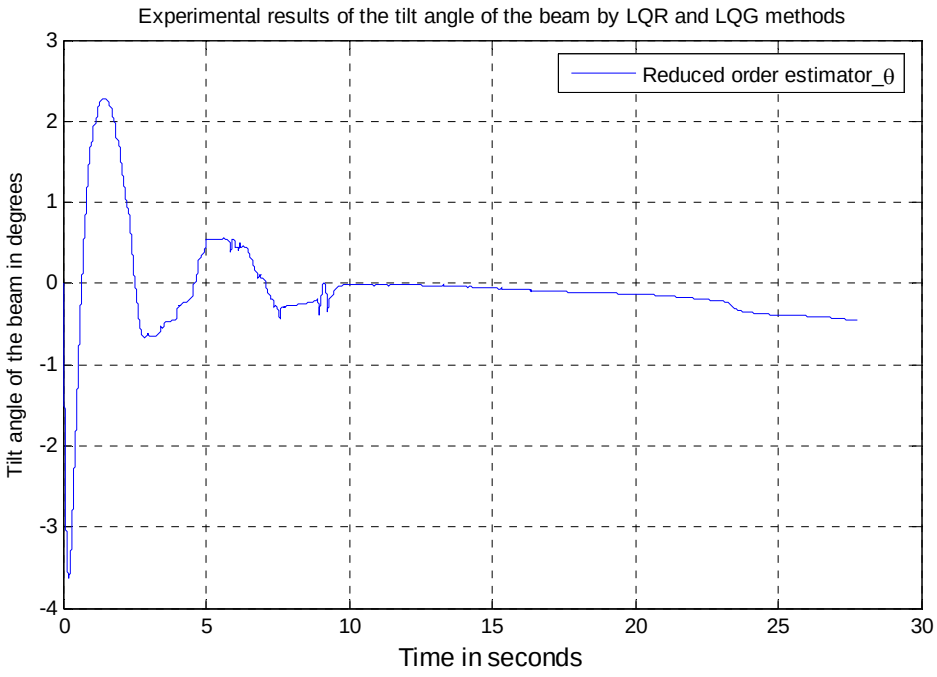

Figure 4.25: Experiment results of the tilt angle of the beam by LQR method

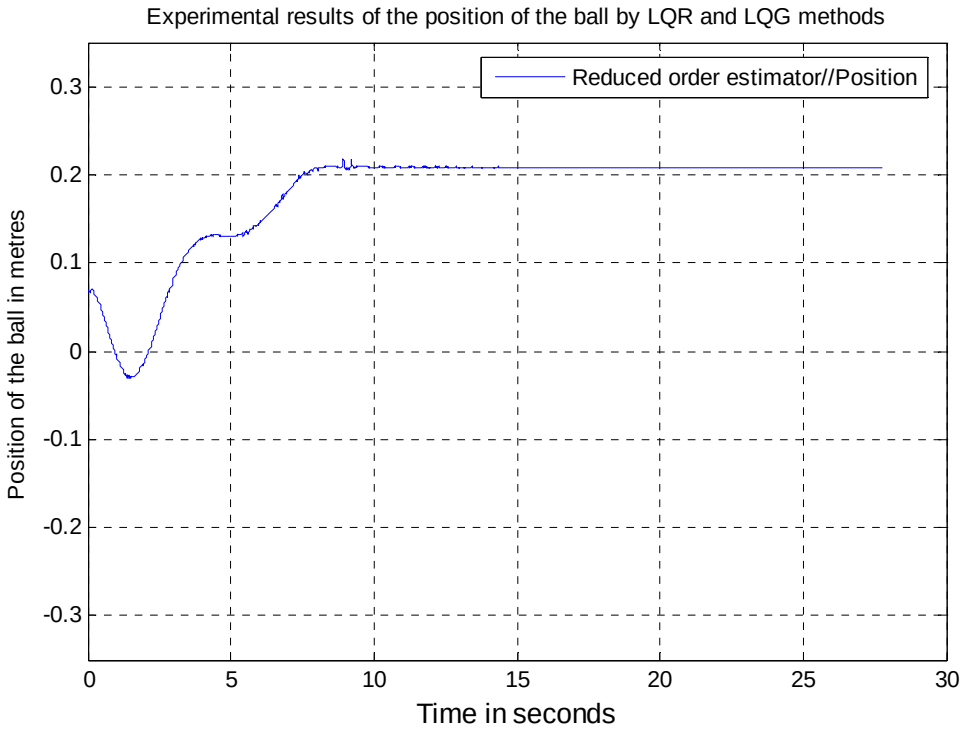

Figure 4.26: Experiment results of the position of the ball by LQR method

4.7 Conclusion

This chapter presented several different control structures, which are full state feedback control, full state estimate control, and reduced order estimator control, for the ball and beam system. For full state feedback and full state estimate control, pole placement method, LQR method, and LQG method, are involved in calculation of the feedback and estimator gain. For reduced order estimator control, LQR and LQG methods are used to calculate the control and estimator gains.

Both simulated and experimental results are presented in a comparative way. From those results, the conclusion is that the LQR method for calculating the gains usually performs better than the pole placement method, and the reduced order system works better than the full state feedback and full-state estimator control structure. Overall, the reduced order system with the LQR and LQG methods achieves the best performance in the ball and beam system.

The next chapter investigates the system identification technique applied to the ball and beam system. Three different control structures, which are full state feedback, full state estimate, and reduced order estimate, are selected to perform the identification process.

Chapter 5

System Identification

In some cases, plants or systems are so complicated that the dynamics cannot be derived using simple physical laws. Therefore, it is necessary to use other methods to identify mathematical models of those systems. The technique of system identification is therefore used in complex modeling situations. Inputs and outputs of the ball and beam system have been measured through experiments. With this data, a system identification technique can be used to estimate both structures and parameters of the system, given that system is stable. In the case of an unstable open-loop system, it is hard to directly estimate parameters. Instead a feedback technique can be used to stabilize the system, then inputs and outputs of the closed loop system can be measured to estimate the system parameters. The inputs of the system should be chosen very carefully. They need to cover the whole bandwidth of the system, and definitely should have enough energy to drive the system; otherwise some characteristics of the system could not be displaced (Cazzolato, 2006).

In the ‘Matlab’, there is a ‘system identification toolbox’. This toolbox provides

many commands to identify an unknown plant. The following design is based on this toolbox.

The experimental determination of some system parameters of the ball and beam system is presented in Section 6.2. The parameters of the DC motor are presented in Appendix C.

5.1 Introduction to system identification (ID)

Zadeh (1962, p75) claims that *'Identification is the determination, on the basis of input and output, of a system within a specified class of the systems, to which the system under test is equivalent'*. The system identification is the technique to model the unknown system dynamic based on the experimental data. The system identification has following steps:

- Generate the input of the system
- Measure the system output
- Process the input and output data
- Select model structure and estimate system parameters
- Validate the estimated model

'Free structured parameterizations' approach, 'grey box' approach, and 'black box' approach are used to identify the ball and beam system in state space form.

5.1.1 'Free structured parameterizations' approach

'Free structured parameterizations' approach is to estimate system parameters in a user-specified state space model without requiring the user to write a special M-file (Matlab) as 'grey box' approach does. '*Certain elements in the state space matrices are free to be adjusted, while others are fixed'* (Matlab 7.01 2007). The approach is useful when the state-space model is derived from physical laws and initial parameter values based on physical insight are provided (Ljung 2007). The model defined in 'free structure parameterizations' approach can be express as follows (Matlab 7.01 2007):

$$\begin{aligned} \dot{x} &= A(\lambda)x + B(\lambda)u + K(\lambda)e \\ x(0) &= x_0(\lambda) \\ y &= C(\lambda)x + D(\lambda)u + e \end{aligned} \tag{5.1}$$

where λ represents for the parameters need to be estimated, K is the Kalman gain, e is the disturbance of the model, $X(0)$ is the initial state, and other symbols are defined in Chapter 4.

Equation (5.1) relates the nominal matrices, A, B, C, and D, and the initial matrix $X(0)$ with the free parameters λ, which are estimated. In the Matlab, the state-space model structures with various parameterizations are constructed by function 'idss'. A 'NaN' in any position in matrices A, B, C, and D denotes '*a freely adjustable parameter, and a numeric value denotes a fixed and nonadjustable*

parameter' (Matlab 7.01 2007).

By using the input and output data collected in simulation or experiment, the unknown parameters λ can be ultimately determined. The function 'pem' in the Matlab can be used to estimate the unknown parameters.

5.1.2 'Grey box' approach

Generally, if the physical system can be expressed in a differential equation, then the grey-box approach can be used. *"A grey-box model is a flexible model structure that you can specify the mathematical structure of the model explicitly, including coupling between parameters and known parameter values"* (Matlab 7.01 2007). Grey-box modelling is useful when the relationships between variables, constraints on model behavior, or explicit equations representing system dynamics are known. (Matlab 7.01 2007) The advantages of the "grey box model" are:

- The model characteristics, such as system parameters, can be adjusted.
- The model has less order and parameter to be estimate, compared to the 'black box model'
- It is possible to specify the dynamics and coupling between the parameters. (Matlab 7.01 2007)

The 'Grey box' approach discussed in this section is more structured approach than

'free structured parameterizations' approach. A detailed mathematical expression of the system is required to construct the estimated model. In the matlab, the linear grey-box model structure is constructed by using 'idgrey' command. This command further requires user to write an 'M-file' to express the grey-box model, and to define the system parameters need to be estimated. The process of estimation is similar with that of 'free structured parameterizations' approach.

5.1.3 'Black box' approach

When there is no information about the system, or the system is so hard to express by physical laws, the black-box approach can be employed. "*A black-box model is a flexible structure that is capable of describing many different systems and its parameters might not have any physical interpretation.*" (Matlab 7.01 2006) The significant advantage of the system is that the detail information about system is not necessary, and hence the 'black box' is much easier to apply.

'Black box' approach allows all the state-space A, B, C, D matrices to be freely adjusted. This approach only requires users to specify the order of the state-space model.

5.2 System identification of the open-loop plant (simulation)

An open loop plant can be identified given that it is stable. Figure 5.1 shows the simulation model for system identification of the open-loop ball and beam system. The input and output of the system from the simulation can be transferred to the Matlab workspace through 'D_input' and 'D_output' blocks. The data can be further used to identify the system parameters. The input of the system is white noise with the power 0.1walt/Hz. In state space form the dynamic of an open-loop plant is given by (Cazzolato 2006):

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \tag{5.2}$$

The state-space matrices A, B, C, D can be directly estimated by 'free structured parameterizations' approach, 'grey box' approach, and 'black box' approach.

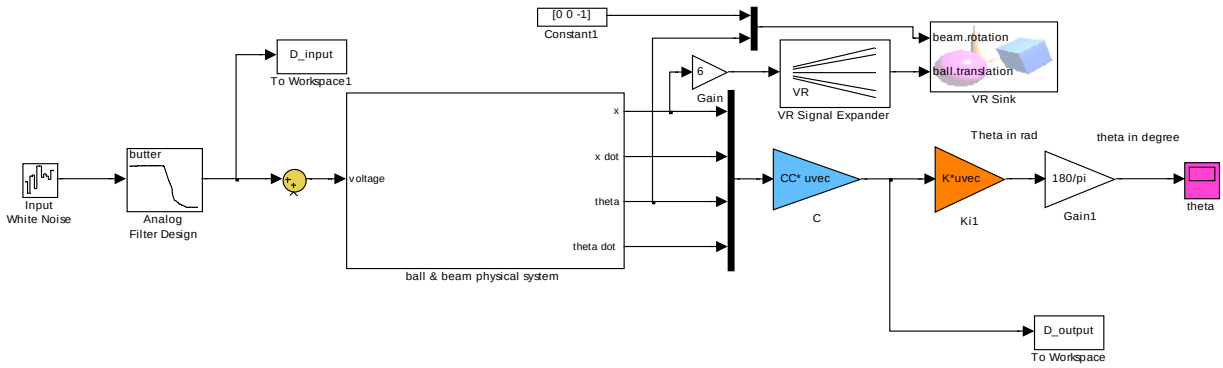

Figure 5.1: This simulation model is used to generate the input and output data for open-loop ID

A low-pass filter is added in the open loop to filter the input signal, which is larger than 100Hz, since the bandwidth of the open loop system is approximately 107rad/s that is approximate 17Hz. 100Hz that is six times of the open loop bandwidth therefore is enough to pick up the system dynamic. The analog low-pass filter here is designed by 'butterworth' method with 36^{th} order.

Figure 5.2 shows the input voltage of the ball and beam system is between -20 to 20 volts, which meets the requirement of voltage supply of the motor. However, since the open loop ball and beam system is unstable, the output of the ball's position exceeds the physical band limits, which is between -0.35m to 0.35m.

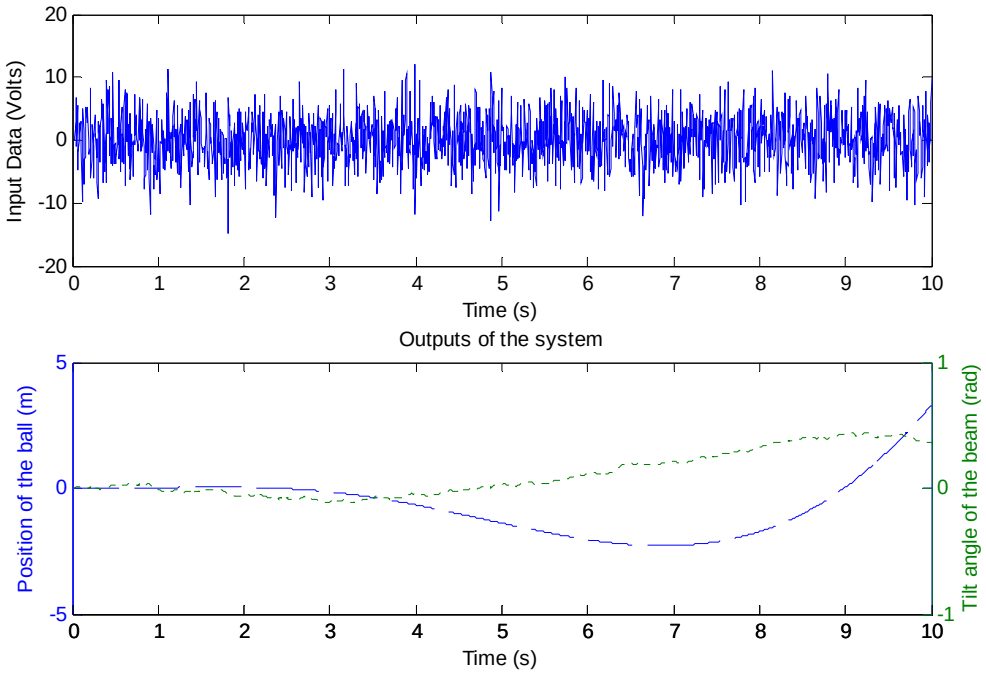

Figure 5.2: Input (white noise) and output data (ball's position and beam tilt angle) for the open loop system

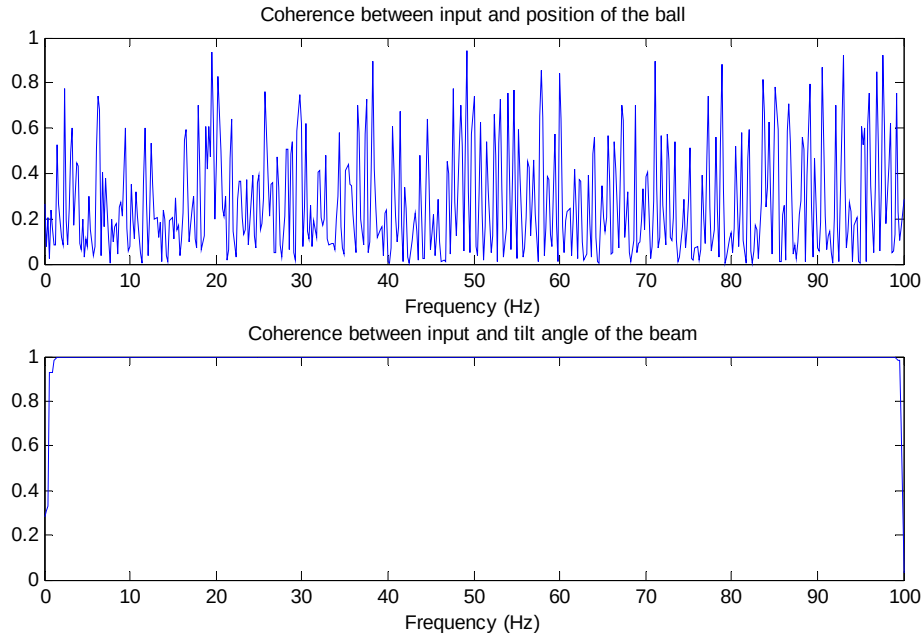

Figure 5.3: Coherence between the input and the outputs for the open loop system

Figure 5.3 shows the coherence between the input and the outputs of the system. The coherence is calculated using following parameters:

- Sample frequency: 200Hz for simulations, 1000Hz for experiments
- Average number is 1024
- Window: 'Hanning'
- Overlap: 75% overlap

All coherences presented in this chapter, are calculated using above parameters.

It can be identified that the coherence between the input and tilt angle of the beam is nearly unity. However, the coherence between the input and ball's position is poor over all the frequency bands. This is because the system is linear given that the tilt angle is small. For the unstable open-loop ball and beam system, this linear

assumption does not hold. Therefore the highly nonlinear character results in poor coherence between the voltage input into the motor and the position of the ball.

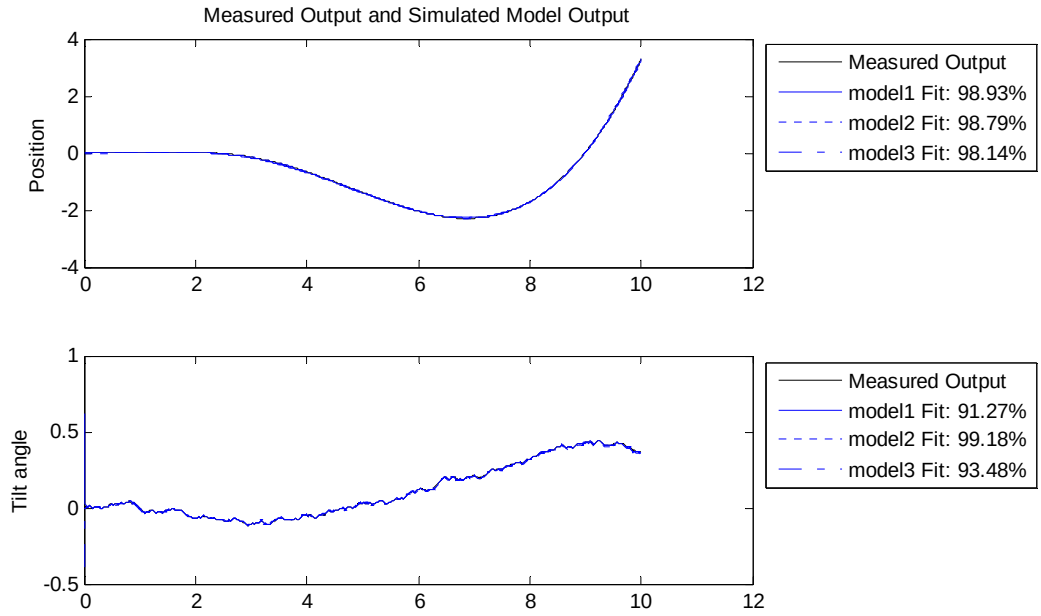

Figure 5.4: Final results of the open loop system identification. model 1: the structure of the system is known ('free structured parameterizations' approach). model 2: both the structure and inner expression of the system are known ('grey' box approach). model 3: no information about the system ('black box' approach)

Figure 5.4 compares the identification results of the ball and beam system by three different methods, the 'free structured parameterizations' approach, 'grey box' approach, and 'black box' approach. The 'free structured parameterizations' approach presents an estimated model of the 4th order. The 'grey box' approach presents a model of the 4th order, which is the same with the nominal model. The 'black box' presents an estimated model of the 4th order. The percentage in Figure 5.4 is defined as follows (Matlab 7.01 2007):

$$100\% \times \frac{\left(1 - \sqrt{\sum\left(\hat{y} - y\right)^2}\right)}{\sqrt{\sum\left(y - \bar{y}\right)^2}}$$

where y represent the measured output of the system, $\hat{y}$ is the output from the estimated model, and $\bar{y}$ is the mean of measured output y.

All the percentages in this chapter are defined as above.

In this simulation, the ID results indicate that the 'grey box' approach is the best since the detailed expression of system is known. Other two approaches 'free structured parameterizations' and 'black box' are accurate too. The conclusion is if there is more information about the system, then a better system identification result may be achieved.

The final ID results (state-space matrices A, B, C, D) by three different approaches: 'free structured parameterizations', 'grey box', 'black box', are presented in Appendix G. The Matlab programs for this section can be found in Appendix B.1.

5.3 System identification of the closed-loop system

The system identification of the ball and beam system in previous section is good, but the identification process can only be conducted in simulation because the open-loop ball and beam system is unstable. ID of unstable systems does not work

since the estimator model of the plant does not converge. Thus it is necessary to close the open loop in order to stabilize the system. Once the system is stabilized, the input and output of the ball and beam system can be measured through the experimental method.

5.3.1 Simulation

Figure 5.5 shows the closed-loop ball and beam system to generate the input and output data from the simulation. The power of the input noise of the simulation is 0.01walts/Hz. The feedback gain for the system has been calculated before the system identification process. In other words, the feedback gain is known, and the ball and beam system parameters need to be identified. The pole placement method by Ackermann's formula (see Section 4.3.1) is used in calculation of the feedback gain. The following closed-loop poles are selected for feedback gain calculation:

Closed loop poles: [-5 -5 -5 -8] rad/s

The feedback control gain calculated by pole placement is presented as follows:

k = [15.42volt/m 11.40volt/(m/s) 11.57volt/rad -5.008volt/(rad/s)] (5.3)

In state-space form closed-loop-system dynamic are given by (Cazzolato 2006):

$$\begin{bmatrix} \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A - Bk & B \\ C & D \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \tag{5.4}$$

where A, B, C, D, and k are defined in Section 4.2.

Using the system identification technique, it is possible estimate state-space matrices A-Bk, B, C, D directly. Control gains k is known. Then A matrix can be calculated indirectly as follows (Cazzolato 2006):

$$A = (A-Bk) + Bk \tag{5.5}$$

A low-pass filter is added in the control loop to filter the input signal, which is larger than 12Hz, since the bandwidth of the closed loop system is 8rad/s that is approximate 1.2Hz. 12Hz that is ten times of the closed loop bandwidth therefore is enough to pick up the system dynamic. The analog low-pass filter here is designed by 'butterworth' method with 8^{th} order.

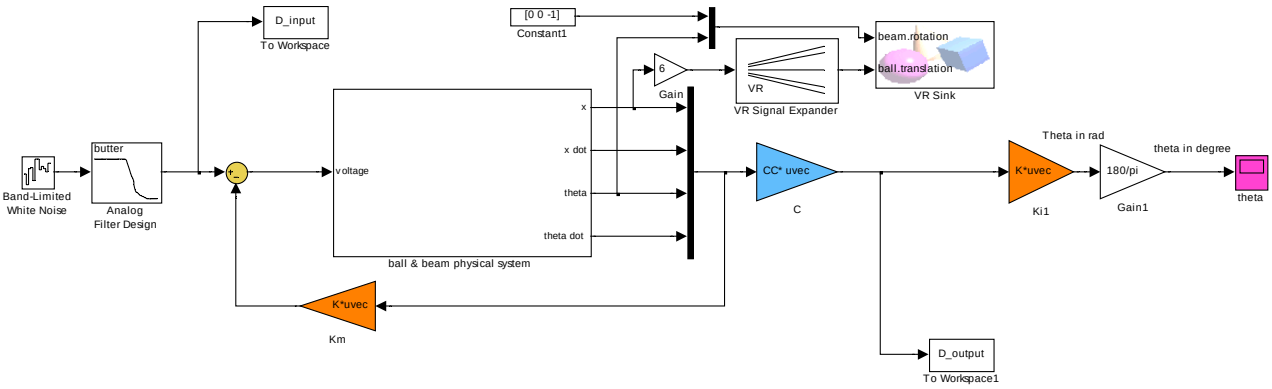

Figure 5.5: Closed loop system with full state feedback for system ID

Figure 5.6 shows the input and output from the closed-loop system. The input voltage of the system is within motor voltage supply requirements. The outputs of the ball and beam systems are within acceptable bonds.

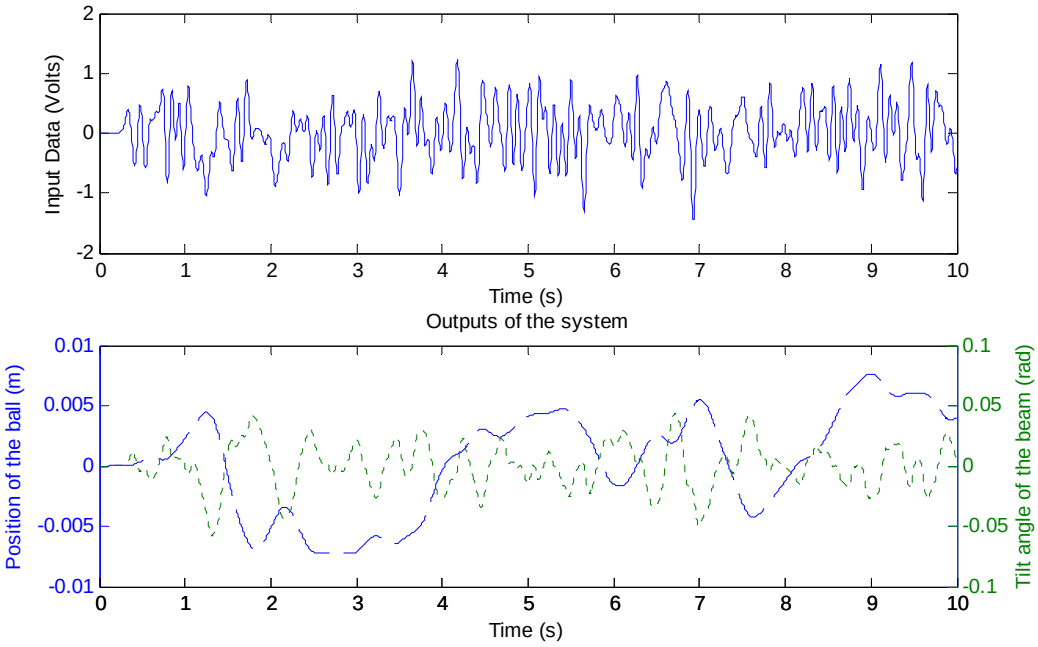

Figure 5.6: Input (white noise) and output data (ball's position and tilt angle) for the closed-loop system ID

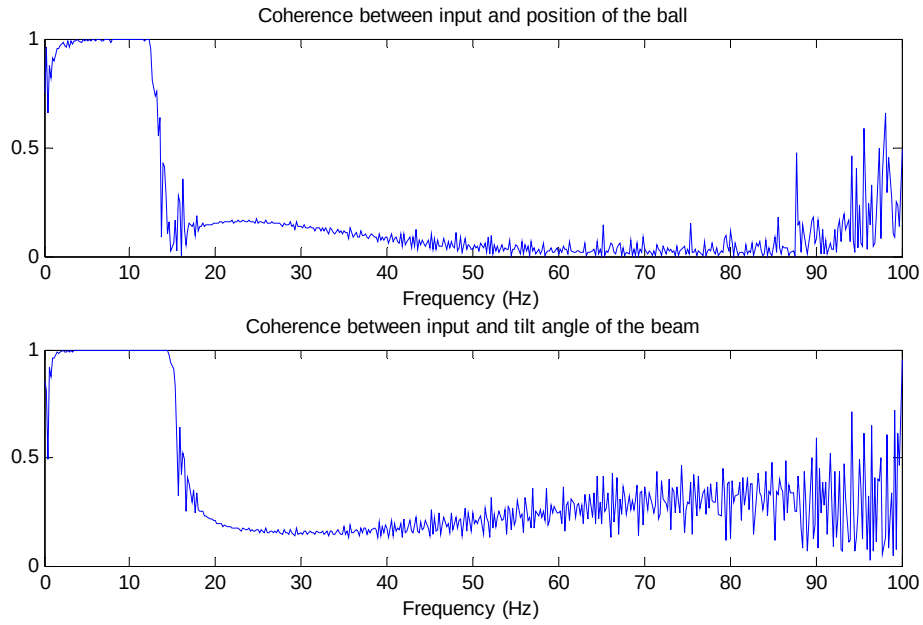

Figure 5.7: Coherence between the input and outputs of the closed-loop system

Figure 5.7 shows the coherence between the input and outputs of the closed loop ball and beam system. The parameters for calculation of the coherence are the same with Section 5.2. The coherences are nearly unity within the interested frequency band

(0-12Hz), which is much better than that of the open loop system in Figure 5.3. This phenomenon can be explained by considering that linear approximation is true for the majority of the time when operating in the closed loop. Therefore, it is evident that good coherence means the system is linear. Therefore, better identification results can be predicted.

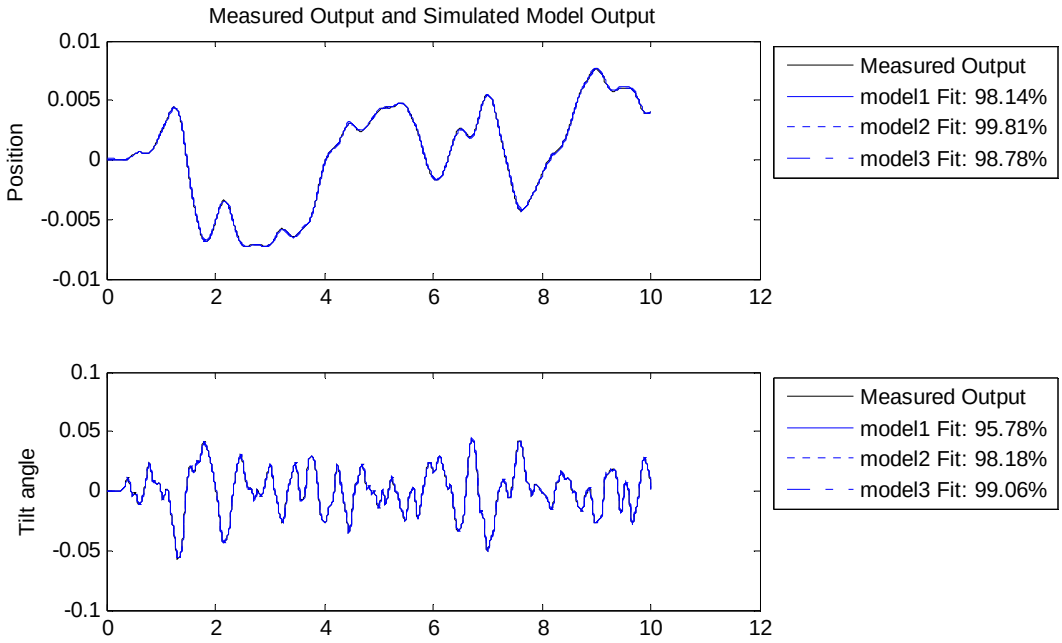

Figure 5.8: Final results of the closed loop system identification. model 1: the structure of the system is known ('free structured parameterizations' approach). model 2: both the structure and inner expression of the system are known ('grey box' approach). model 3: no information about the system ('black box' approach).

Figure 5.8 above shows the final ID results of the closed loop system. It is clear that all the estimated models are very accurate due to the good coherence in Figure 5.7.

From the identified closed loop model, the estimated plant state matrix A, from the model 2 using 'grey box' approach is presented as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 3.81 & 0 \\ 0 & 0 & 0 & 1 \\ -28.39 & \underline{\underline{-17.35}} & \underline{\underline{-17.30}} & -105.39 \end{bmatrix};$$

The nominal open-loop plant is shown as follows:

$$A\_nominal = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 3.773 & 0 \\ 0 & 0 & 0 & 1 \\ -5.169 & \underline{\underline{0}} & \underline{\underline{0}} & -107.4 \end{bmatrix};$$

Comparing the nominal open-loop plant ‘A_nominal’ to the open-loop state matrix ‘A’, most elements in the state matrix are the same, and the other states have a slight difference. It is worth noting that the underlined elements in the matrix ‘A’ are non-zero values, which are different with that in matrix ‘A_nominal’, because A is obtained indirectly from (A-Bk)+Bk as stated in Equation 5.5.

The ID results from other approaches are presented in Appendix G. The Matlab programs for this section can be found in Appendix B.2.

5.3.2 Experiment

The input and output data from the experiment is shown in Figure 5.9. In this experiment the input white noise power is 0.05walts/Hz. The ball rolls on the beam

within the physical limits (-0.35m, 0.35m).

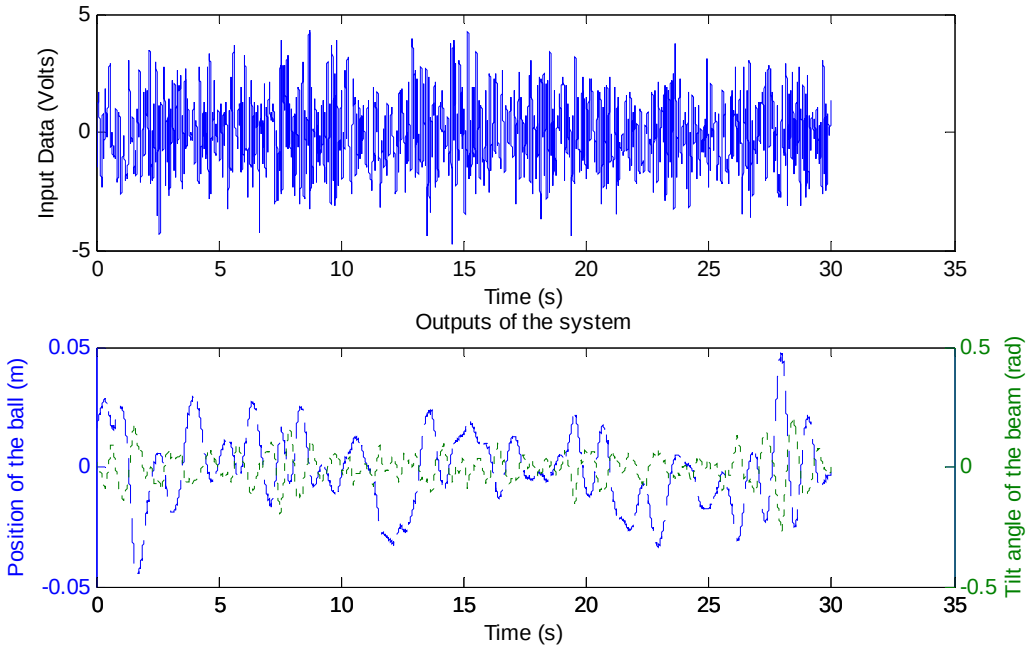

Figure 5.9: Input (white noise) and output data (ball's position and tilt angle) for closed-loop system ID

The coherence between input and output data is presented in Figure 5.10. The coherence between the input and the position of ball is poor due to the non-linearity of the ball and beam system, and the un-modeled friction between the ball and resistive wire. The coherence between input and the tilt angle of the beam in the interested frequency band (0-12Hz) is good.

Overall the coherence between the input and output is not very good, which indicates the high nonlinearity of the ball and beam system. However, as the ball and beam system is a non-linear system, the poor coherence between the input and the position of the ball is acceptable.

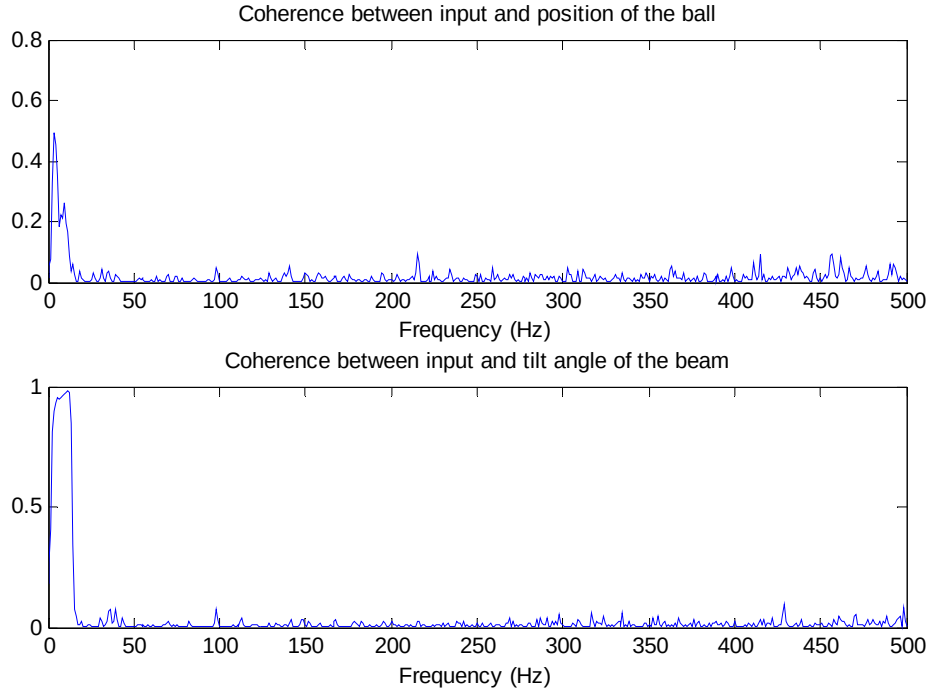

Figure 5.10: Coherence between the input and outputs of the closed-loop system

The ID results from the three methods, 'free structured parameterizations' with 4th order, 'grey box' with 4th order, and 'black box' with 4th order, are presented in Figure 5.11.The ID results from the 'grey box' approach with 4th order have the best accuracy. The ID results from other approaches are accurate too. The experimental results again show that the 'grey box' approach presents the best results. In a conclusion, more accurate ID results could be achieved by providing more detailed information in the design process.

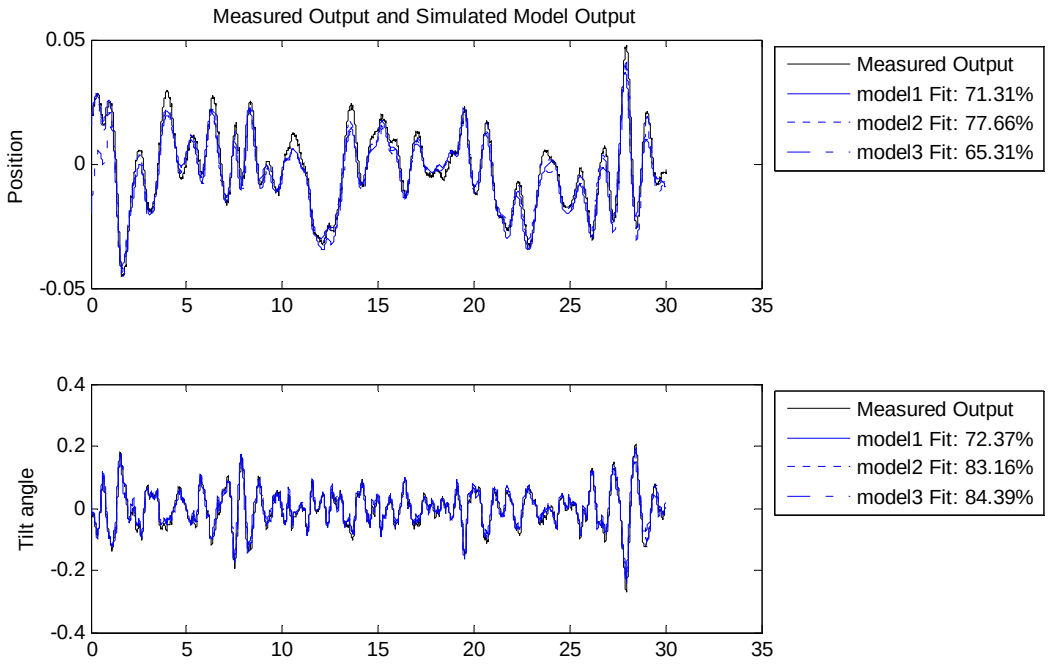

Figure 5.11: Final results of the closed loop system identification. model 1: the structure of the system is known ('free structured parameterizations' approach). model 2: both the structure and inner expression of the system are known ('grey box' approach). model 3: no information about the system ('black box' approach).

The ID results of the closed loop ball and beam system from experiment is presented in Appendix G.

5.4 ID from full state estimator plant

In some case, there are not as many sensors as system states such as the ball and beam system, some states therefore have to be estimated though the observer. The ID process of the full-state-estimator plant is similar to that of the open loop plant. The system controlled by the full state feedback with a full state observer is shown in Equation (5.6) (Cazzolato 2006),

$$\begin{bmatrix} \dot{x} \\ \dot{\hat{x}} \\ \hline y \end{bmatrix} = \left[\begin{array}{cc|c} A & -Bk & B \\ LC & \hat{A}-\hat{B}k-L\hat{C} & \hat{B} \\ \hline C & 0 & D \end{array}\right] \begin{bmatrix} x \\ \hat{x} \\ \hline u \end{bmatrix} \tag{5.6}$$

where the symbols in Equation has been defined in Section 4.2.

Matrices A, B, C, and D can be directly obtained from system identification. k, L, $\hat{A}$, $\hat{B}$, $\hat{C}$, and $\hat{D}$ are known since they are predefined (Cazzolato 2006).

5.4.1 Simulation

Simulation modeling is very important in the system identification. Figure 5.12 shows the full state estimate model used to generate the input and output of the system for the system identification process. In this model, all the states are estimated from two measured states, which are the position of ball and the tilt angle of beam. The power of the input noise is 0.1walts/Hz. The feedback gain and estimator gain has been calculated before the ID process. In other words, the feedback and estimator gain are known, but the ball and beam system parameters need to be identified.

Both The feedback gain k and the estimator gain L calculated by pole placement method are presented as follows:

Closed loop poles: [-3, -3, -5, -8]rad/s

$$k = \begin{bmatrix} 5.356\text{volt/m} & 5.615\text{volt/(m/s)} & 7.537\text{volt/rad} & -5.245\text{volt/(rad/s)} \end{bmatrix} \tag{5.7}$$

Estimator poles: [-60, -60, -100, -160]rad/s

$$L = \begin{bmatrix} 220(1/s) & 1.1(m/(rad \times s)) \\ 9612(1/s^2) & 71.2(m/(rad \times s^2)) \\ 11.9(rad/(ms)) & 52.6(m/(rad \times s^2)) \\ -92.3(rad/(ms^2)) & 349.4(1/s^2) \end{bmatrix} \quad (5.8)$$

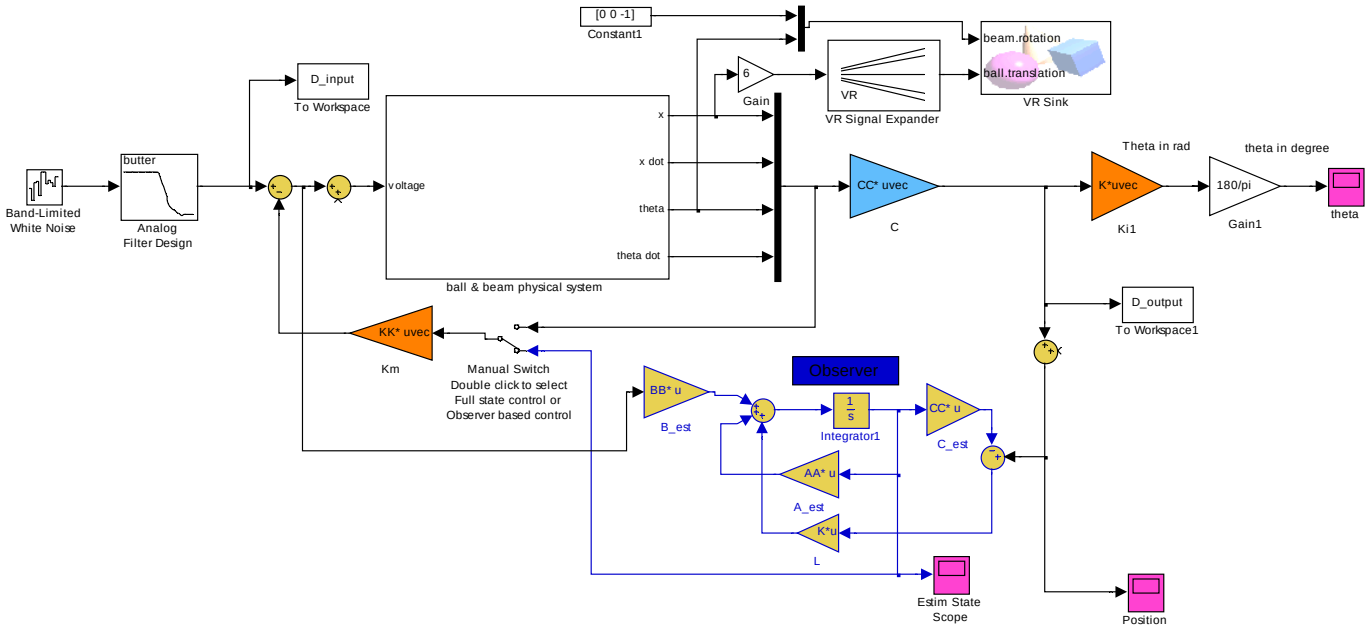

Figure 5.12: Full state estimator model for the system identification

The highest estimator pole of the system is 160rad/s that is approximately 25Hz. A 30Hz low-pass filter is placed after the input source in Figure 5.12 in order to filter unnecessary input noise. The analog low-pass filter here is designed by 'butterworth' method with 8^{th} order.

The input and output data from the full state estimate model are represented below.

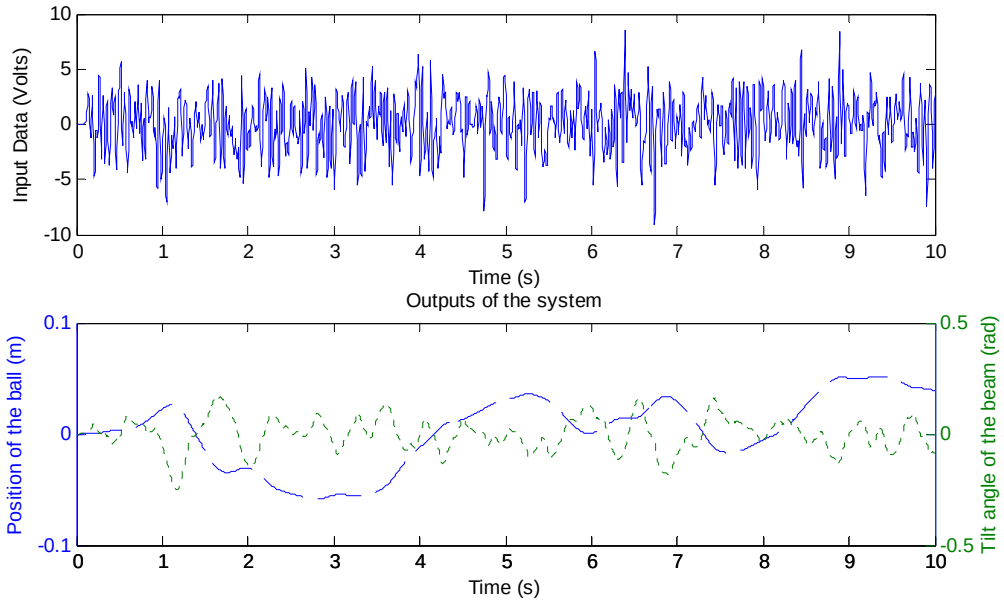

Figure 5.13: Input (white noise) and output data (ball's position and tilt angle) for full state estimation

The range of the input and output data of the full state estimate system is presented indicates that the results fall within acceptable physical limits. Good coherences within the interested bands (0-30Hz) between input and outputs are presented in Figure 5.14. The good coherences are due to the tilt angle of the beam is within linear bands.

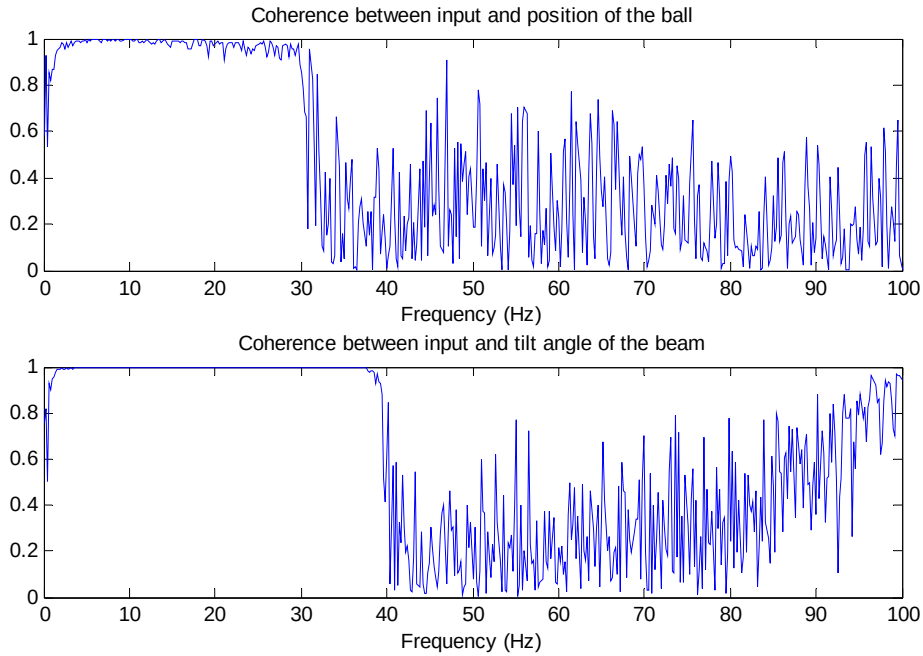

Figure 5.14: Coherence between the input and outputs when employing an estimator

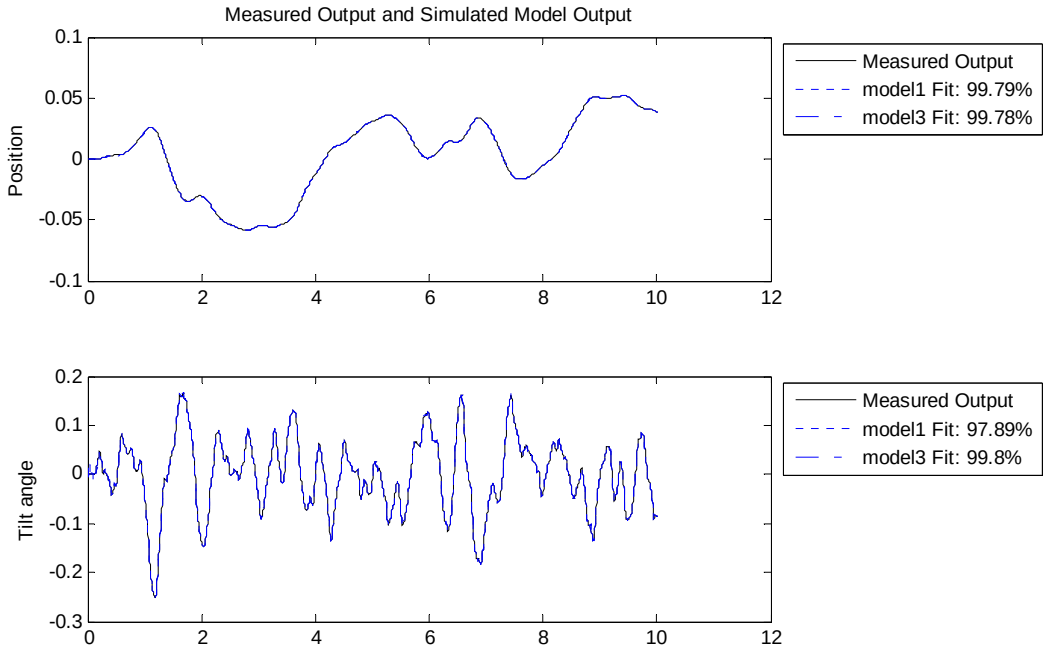

Figure 5.15: Final results of the system identification for full state estimation: model1: use 'free structured parameterizations' approach, model3: use 'black box' approach

Figure 5.15 shows the results from the system identification by the 'free structured

parameterizations' approach with 8^{th} order and 'black box' approach with 8^{th} order. Both 'model1' and 'model3' have a good accuracy due to the good coherence between the input voltage and output (ball's position and beam's tilt angle). The 'grey box' approach is not used here since the inner structure of the full-state-estimator system is too hard to derive. The result from the estimated model accurately matches the nominal model. The state matrix A, B, C, and D can be obtained directly from the ID results.

The ID results of the full-state-estimator ball and beam system are presented in Appendix G. The 'Matlab' program for this section is presented in Appendix B.3.

5.4.2 Experiment

In this experiment the input noise power is 0.05walts/Hz. The feedback gain and estimate gain is the same as the previous section of the simulation. Figure 5.16 shows the input and output data from the full state estimate model in the experiment. The poor coherences between the input and output are presented in Figure 5.17, which indicates the high non-linearity of the ball and beam system.

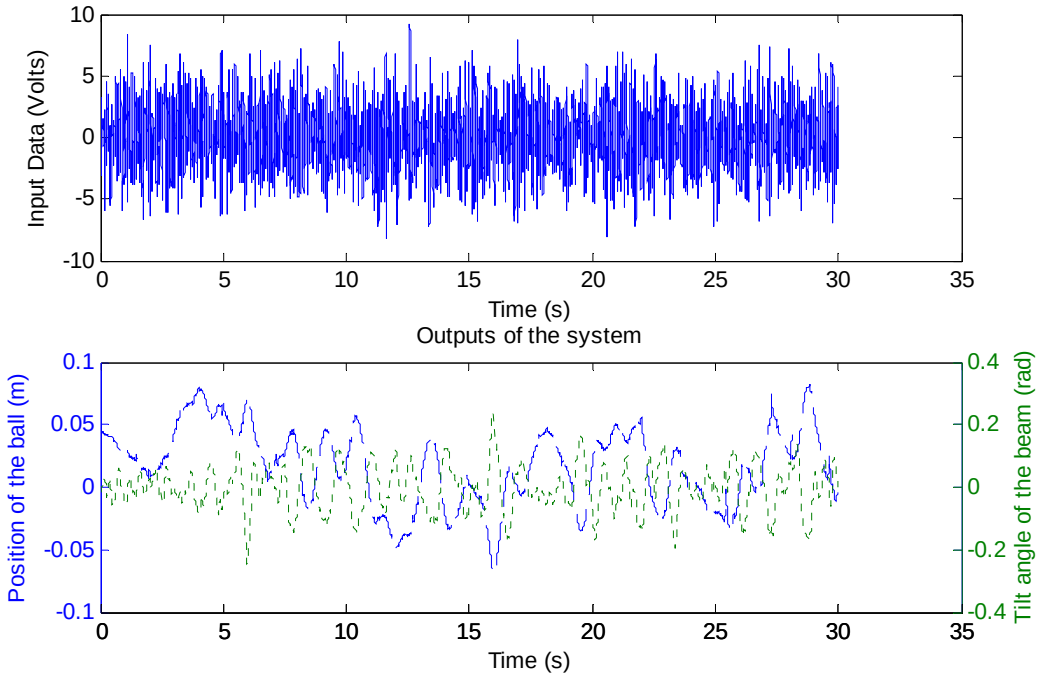

Figure 5.16: Experimental results of input (white noise) and output data (ball's position and tilt angle) for full state estimation

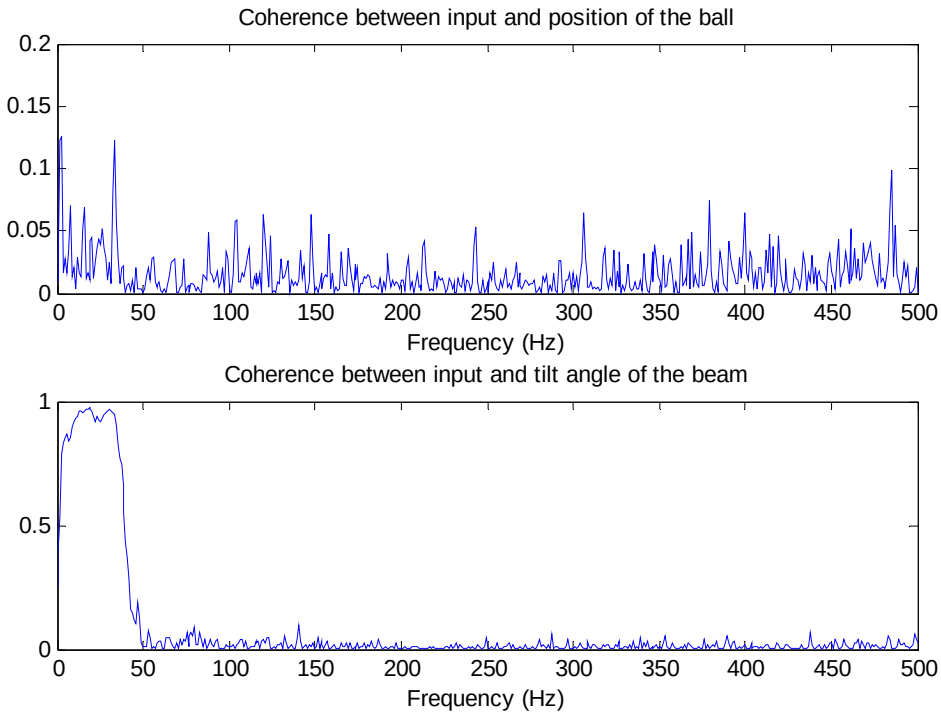

Figure 5.17: Experimental results of Coherence between the input and outputs when employing an estimator

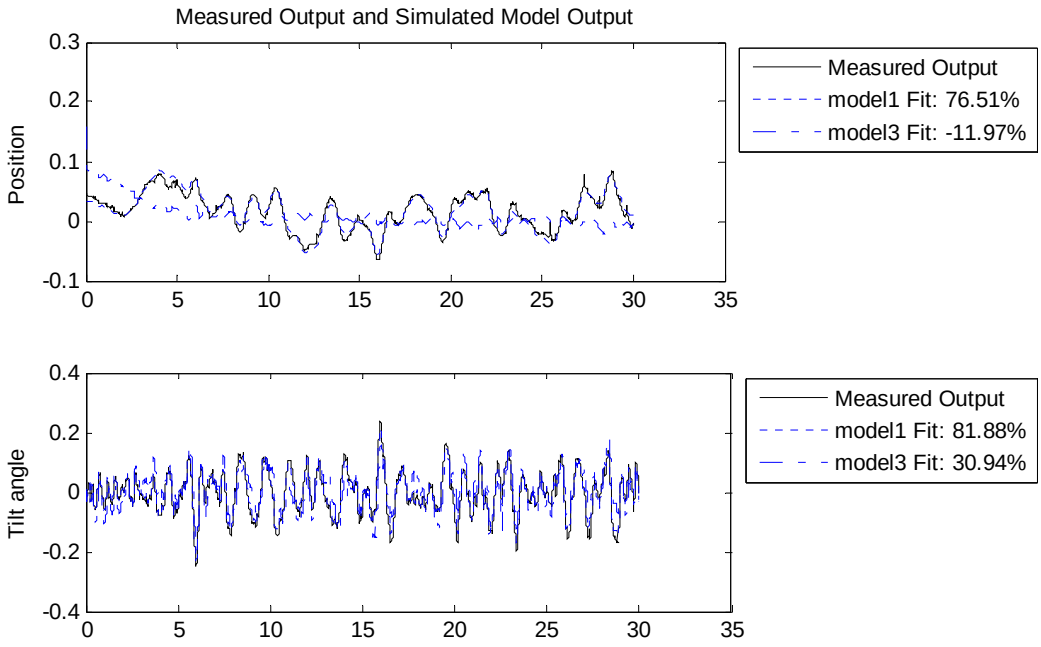

Figure 5.18: Experimental results of the system identification using 'free parameterisation' approach for full state estimation

The ID final results are presented in Figure 5.18, which shows the estimated model using the 'free structured parameterizations' approach with the 8th order fits the nominal model accurately. The high level of accuracy may be resulted from the good controllability of the full state estimate model. The 'black box' with 8th order used here is not accurate due to the high nonlinearity of the ball and beam system.

The final model from system identification is presented in Appendix G.

5.5 ID from reduced order system

The ball and beam system can be identified from its reduced order system, which improves the system performance by observing the states required. The ID process is

similar to that of the open-loop plant. However, the states of the ball and beam system need to be expressed as Equation (4.38) and (4.39).

In state form the dynamics of the system controlled using state feedback with a reduced-order observer is given by Equation (5.2) (Cazzolato 2006):

$$\begin{bmatrix} \dot{x}_a \\ \dot{x}_b \\ \dot{x}_c \\ y \end{bmatrix} = \begin{bmatrix} A_{aa} + B_a D_r & A_{ab} & B_a C_r & B_a \\ A_{ba} + B_b D_r & A_{bb} & B_b C_r & B_b \\ B_r & 0 & A_r & 0 \\ C & 0 & 0 & D \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \\ u \end{bmatrix} \tag{5.9}$$

where the symbols of Equation (5.9) are defined as follows:

$$A_r = A_{bb} - L\hat{A}_{ab} - \left(\hat{B}_b - L\hat{B}_a\right)k_b \tag{5.10}$$

$$B_r = A_r\, L + \hat{A}_{ba} - L\hat{A}_{aa} - \left(\hat{B}_b - L\hat{B}_a\right)k_a \tag{5.11}$$

$$C_r = -k_b \tag{5.12}$$

$$D_r = -k_a - k_b L \tag{5.13}$$

where the symbols here have been defined in Section 4.2.3, hat symbol $\hat{}$ represents the estimates of the plant dynamics used in the observer (Cazzolato, 2006).

In this form the matrices, A_r, B_r, C_r, D_r k_a, k_b, and L are predefined. The matrices, A_{aa}, A_{ba}, A_{ab}, A_{bb}, B_a, and B_b need to be identified. A_{ab}, A_{bb}, B_a, and B_b can be obtained directly from ID results. However, A_{ab}, A_{bb} are obtained indirectly by considering following equations (Cazzolato, 2006):

$$A_{aa} = \left(A_{aa} + B_a D_r\right) - B_a D_r \tag{5.14}$$

$$A_{ba} = \left(A_{ba} + B_b D_r\right) - B_b D_r \tag{5.15}$$

5.5.1 Simulation

The simulation model of the reduced order ball and beam system is presented in Figure 5.19. In this model, the input and output data are exported to Matlab workspace. The input noise power is 0.1walts/Hz. The unknown states, the speed of the ball and the angular speed of the beam are estimated through the measured states, the position of the ball and the tilt angle of the beam. The feedback gains, k_a and k_b are determined by the pole placement method. The observer gains L are also calculated by pole placement method.

Closed loop poles: [-5 -5 -5 -8]rad/s

$$k_a = [15.42 \text{ volt/m} \quad 11.57\text{volt/rad}] \tag{5.16}$$

$$k_b = [11.40\text{volt/(m/s)} \quad -5.007\text{volt/(rad/s)}] \tag{5.17}$$

Estimator poles: [-150 -240]rad/s

$$L = \begin{bmatrix} 150\left(1/s^2\right) & 0 \\ 0 & 132.6\left(1/s^2\right) \end{bmatrix} \tag{5.18}$$

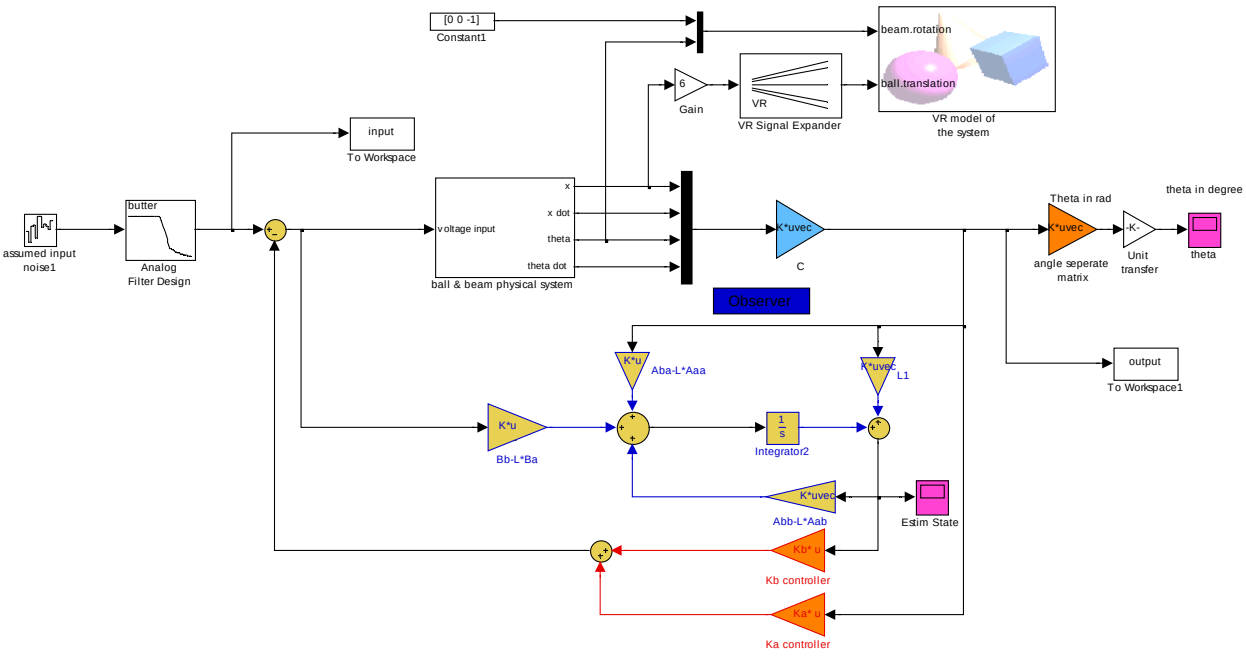

Figure 5.19: System identification process of reduced order estimator

The highest estimator pole of the system is 240rad/s that is approximately 38Hz. A 50Hz low-pass filter is placed after the input source in Figure 5.19 in order to filter unnecessary input noise. The analog low-pass filter here is designed by 'butterworth' method with 8^{th} order.

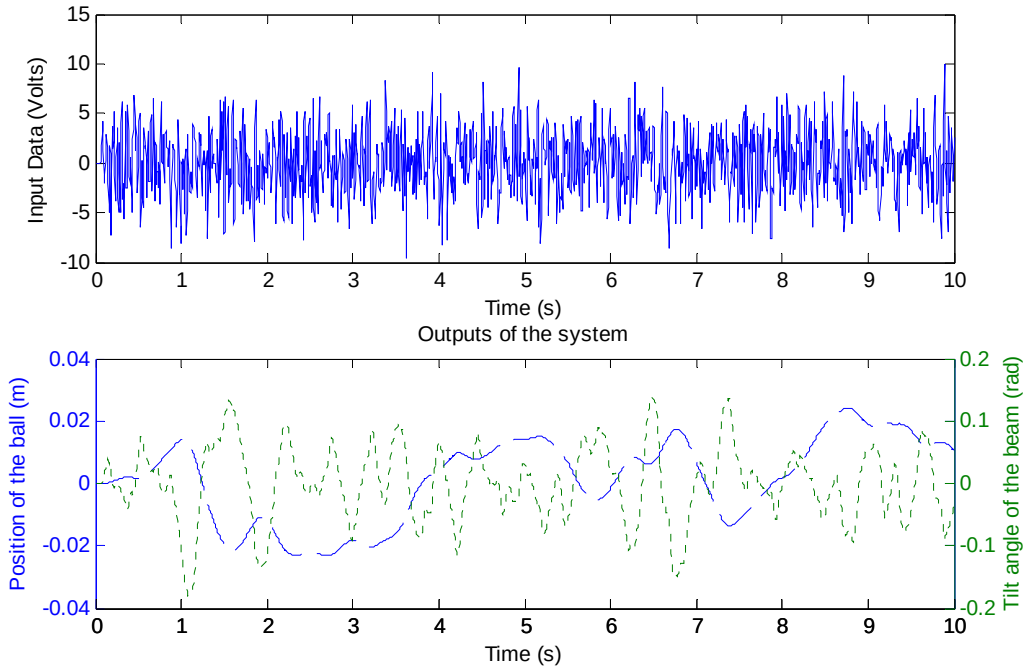

Figure 5.20: The input and outputs of the reduced order ball and beam system

The input and output of the reduced order ball and beam system from the simulation are presented in Figure 5.20. The input, which is the voltage supplied to the motor, is within the band limits (-20 volts, 20 volts). The outputs, which are the position of the ball and the tilt angle of the beam, are reasonable for the system identification.

Figure 5.21 shows the coherence between the input and output of the ball and beam system. The coherences are near unity in interested frequency band (0-50Hz). The coherences indicate the high linearity of the reduced order ball and beam system.

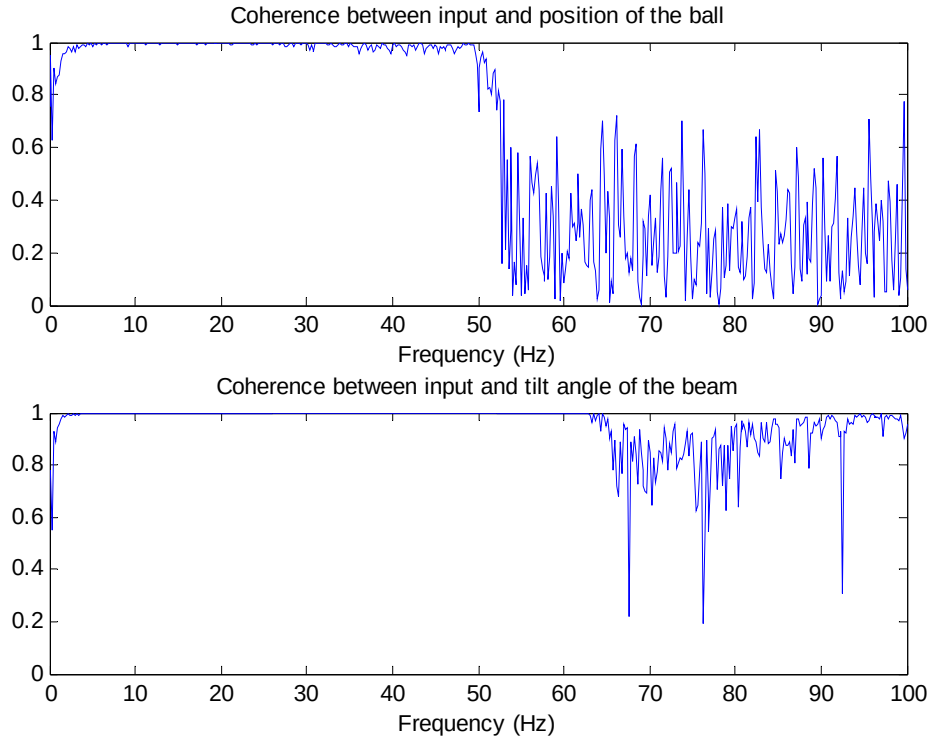

Figure 5.21: Coherence between the input and output of the ball and beam system for reduced order system ID

Figure 5.22 shows the results from the system identification by using the 'free structured parameterizations' approach of the 6th order and the 'black box' approach of the 6th order. It is clear that the estimated results of the 'model1' match the measured result accurately. The 'model3' is not as accurate as model1. This may be because the noise level of the input source is high, or the 6th order estimation is not enough.

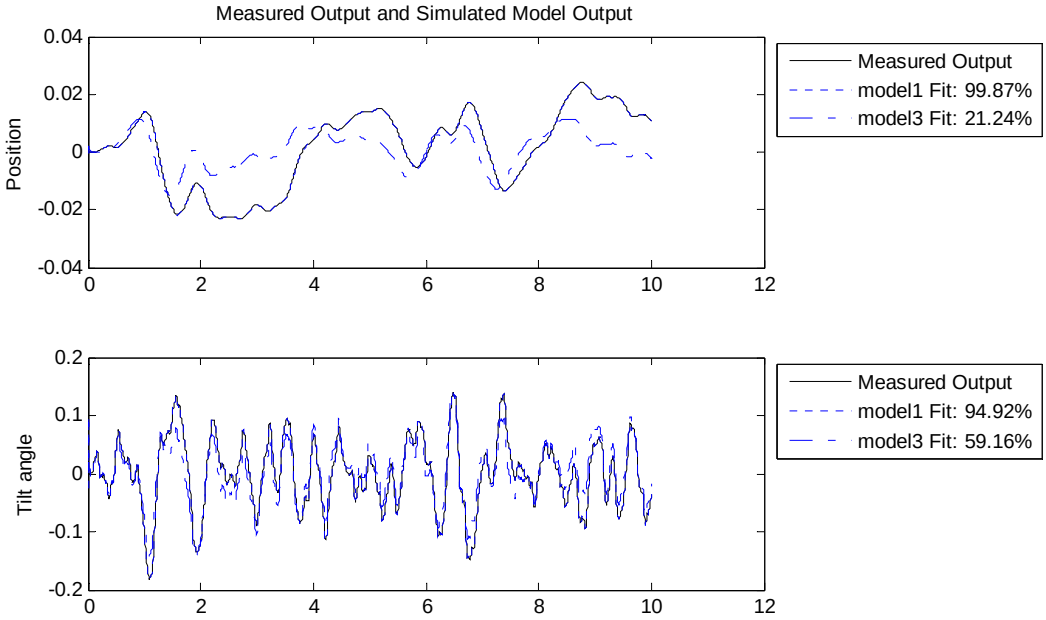

Figure 5.22: Results from the system identification of the reduced order ball and beam system, model1: use the 'free structured parameterization' approach, and model3: use the 'black box' approach'

The detail ID results are presented in Appendix G. The 'Matlab' program for this section is presented in Appendix B.4.

5.5.2 Experiment

The input and outputs data of this experiment is presented in Figure 5.23. Both input and outputs of the ball and beam system are within the physical limits. The experimental ID results of the reduced order ball and beam system are presented in Figure 5.25, which shows a relatively good accuracy of ID results by using the 'free parameterisation' approach of the 6^{th} order. The results from 'black box' approach with 6^{th} order are poor due to the poor coherence. The coherence, shown in Figure 5.24, between input and output in the experiment is not good due to the high

non-linearity of the ball and beam system.

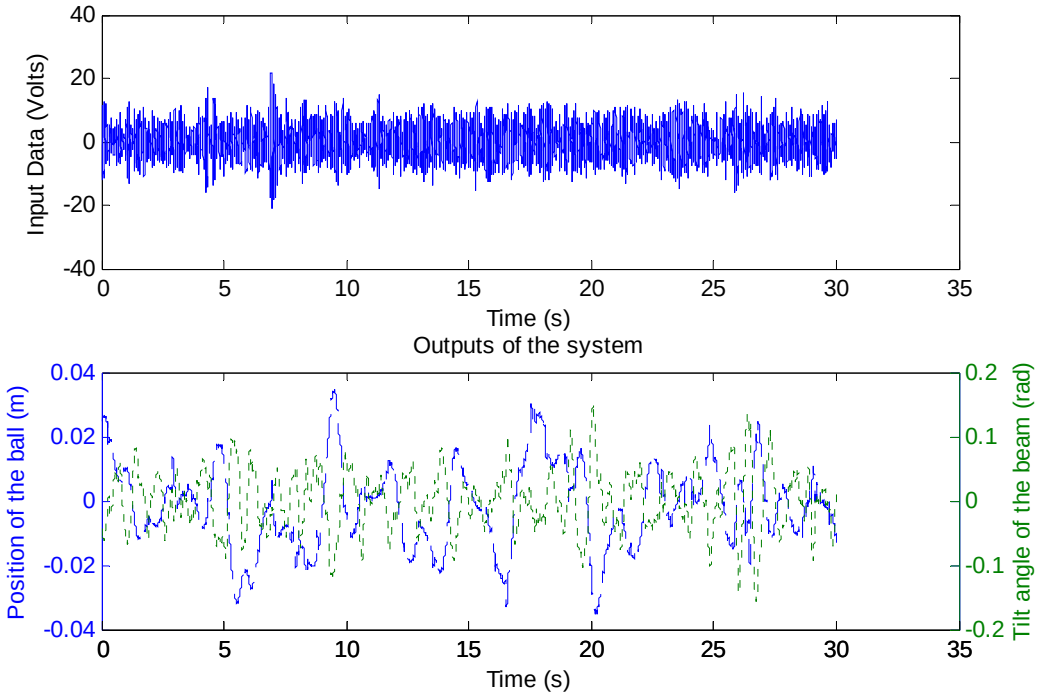

Figure 5.23: Experimental input and outputs of the reduced order ball and beam system

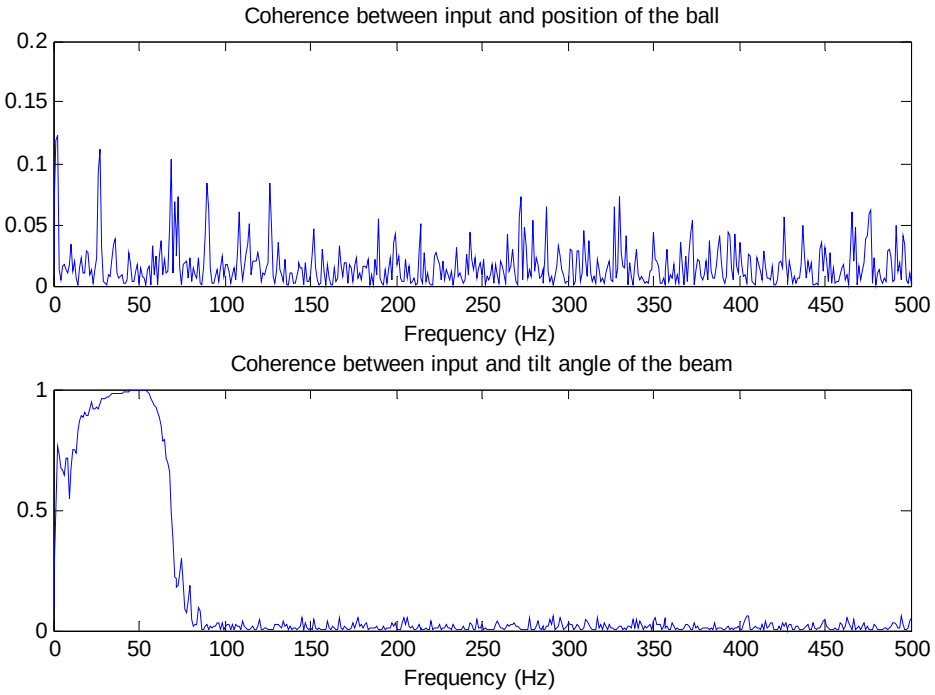

Figure 5.24: Coherence between the input and output of the reduced order ball and beam system in experiment

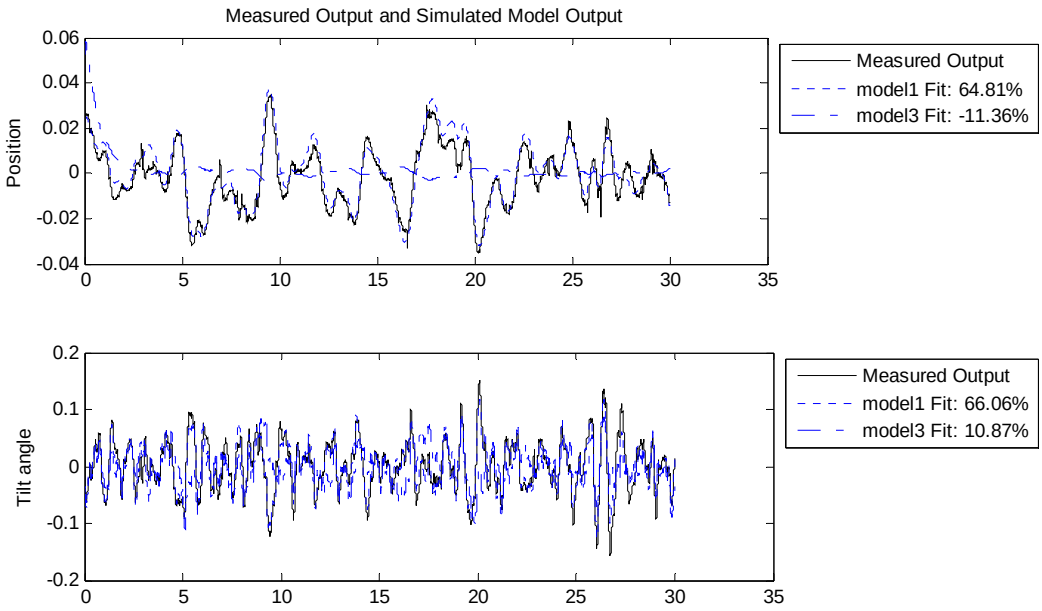

Figure 5.25: Experimental results from the system identification of the reduced order ball and beam system

The detail ID results are presented in Appendix G.

5.6 Conclusion

In this chapter, several system identification approaches, the 'free structured parameterizations', 'grey box', and 'black box', are applied in the ball and beam system. It is evident that the accurate ID results can be derived from the full state feedback structure, full state estimate structure, and reduced order estimate structure. ID of the open-loop system can be conducted, given that the open-loop system is stable. The accuracy of the ID results relies highly to the linearity of the system. The 'grey box' approach generally presents the most accurate results, compared to the other ID approaches, due to more detailed information being applied in the

identification process.

ID results in this chapter, which are presented in Appendix G, prove that the model of the ball and beam obtained from direct measurement is very accurate. The Matlab programs using for system identification for this chapter are presented in Appendix B.

In the next chapter, some considerations in the experiment, including the sensor calibration, system parameters determination, and DC motor calibration, will be presented.

Chapter 6

Considerations in the Experiment

For the ball and beam system, there are many practical problems, such as the nonlinearity of the motor and sensor calibration. The considerations in the experiment, including sensor calibration, interface between physical system and the control board, motor tuning, and system parameters measurement, are presented in this chapter.

6.1 Sensor calibration

The sensors involved in the ball and beam system include a position sensor made by the resistive wire and a digital encoder. The signals from the sensors usually contain some biases. Therefore, it is essential to calibrate the sensors correctly before being applied to the control application.

6.1.1 Position sensor calibration

The position senor for the ball and beam system is the resistive wires as shown in Figure 2.5. The resistive wire on one side of the beam is supplied with 9 volts shown in Figure 6.1. The resistive wire on the opposite side is the voltage signal return wire. The signal from the return wire is further filtered and amplified shown in Figure 6.2, which details the configuration and electronic components such as amplifiers (TL072ACN) and adjustable regulator (LM317BT).

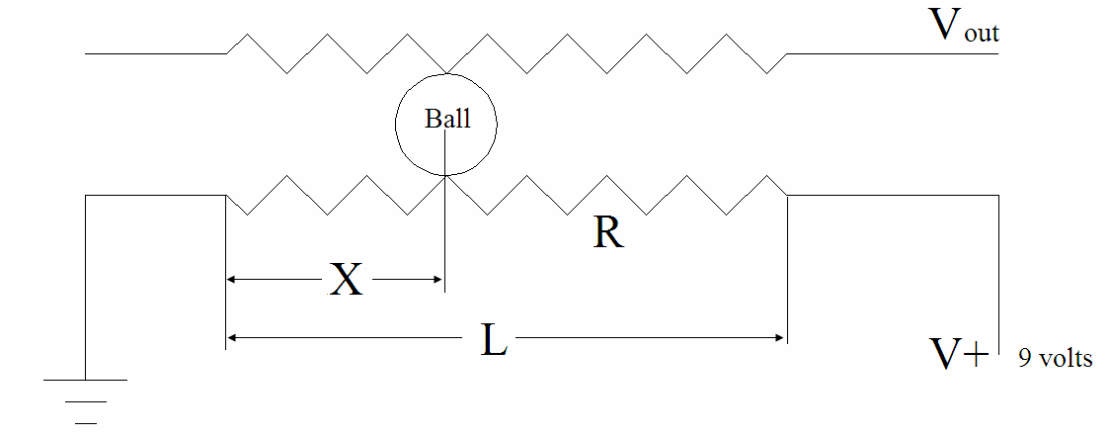

Figure 6.1: Position sensor made by the resistive wire (X is distance of the center of the ball to the voltage ground side, L is length of the resistive wire, R is the resistance of the resistive wire)

Equation 6.1 presents the relationship between the voltage supplier and voltage output of the resistive wire position sensor.

$$V_{out} = \frac{L - X}{L} \times V \qquad (6.1)$$

The position of the ball can be derived by rearranging Equation 6.1. By measurement of the output voltage through the amplifier shown in Figure 6.2, the position of the ball can be determined.

$$X = \left(1 - \frac{Vout}{V}\right) \times L \qquad (6.2)$$

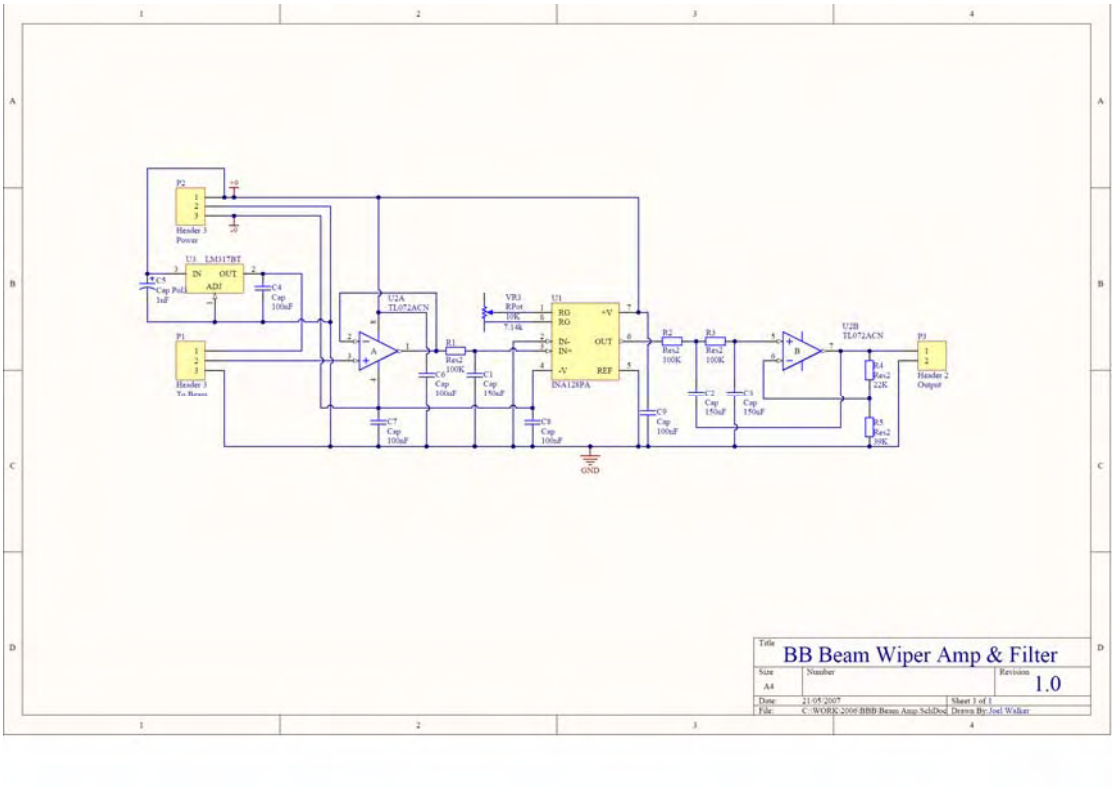

Figure 6.2: Resistive wire position sensor amplifier and filter for the ball and beam system (Walker, 2007)

Figure 6.3 shows the calibration of the position sensor made by resistive wire in the Matlab. The middle of the beam is defined as zero position for the system. After calibration, the output from the resistive sensor is within $\pm 0.35\,\mathrm{m}$.

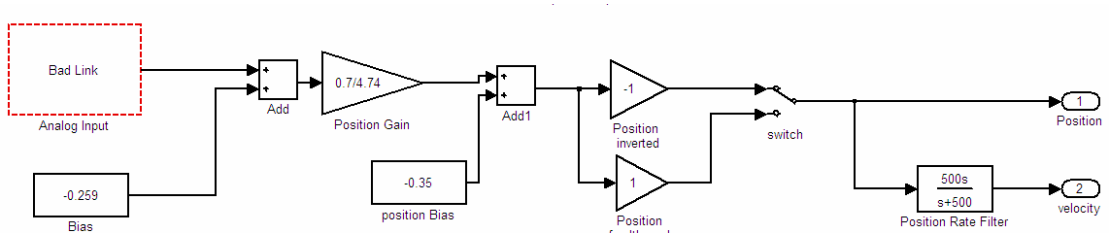

Figure 6.3: The resistive wire position sensor calibrated in the Matlab

6.1.2 Encoder calibration

The digital encoder is directly connected to the DC motor. The resolution of the encoder itself is 1000. The resolution at the output shaft is $1000 \times 100 = 100000$. (where 100 is the reduction ratio of the gearbox) Decoding transitions of signal A and signal B shown in Figure 6.4 by using sequential logic circuits increase the resolution by 4 times. The final resolution of the encoder for the ball and beam system is 400000 pulses per revolution of output shaft. In order to determine the tilt angle of the beam, the information should be calibrated by a factor of $\frac{2 \times \pi}{1000 \times 100 \times 4}$. After calibration, the unit of signal is in rad.

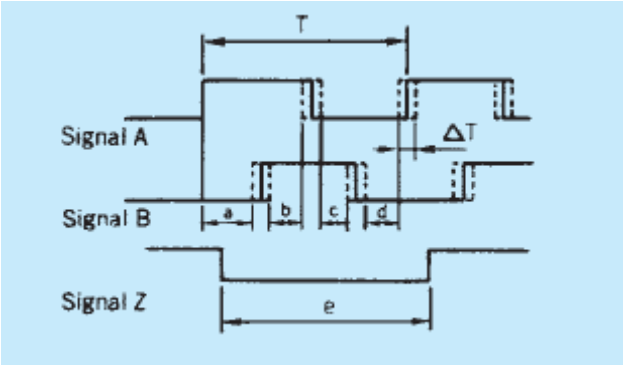

Figure 6.4: Encoder output wave form (Harmonic driver 2006)

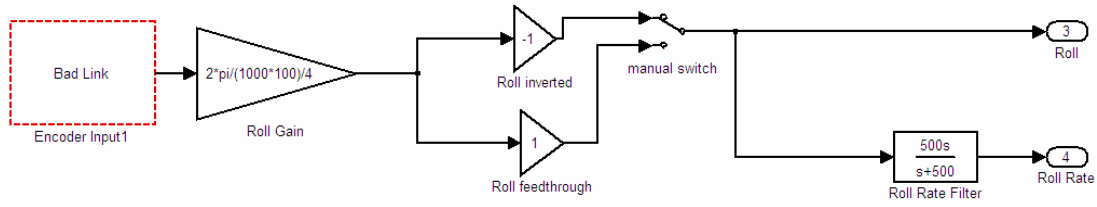

Figure 6.5: Encoder calibration by the 'Roll Gain'

Figure 6.5 shows the calibration of the digital encoder in the 'Matlab'. It is worthwhile to point out that initial value of the encoder is zero. Therefore, it is necessary to balance the ball on the beam manually before to do experiment.

6.2 System parameter measurement

Most system parameters can be directly measured by the equipment shown in Table 6.1. Some are indirectly measured such as the moment inertia of the beam. The system parameters which are directly measured are presented as follows:

Table 6.1: Parameters which are directly measured

Parameters	Measurement Equipment	Value
Ball radius R_b	micrometer	19.97mm
Length of the beam L_{beam}	Vernier calipers	0.7m
Mass of the beam M_{beam}	balance	381g
Mass of the ball M_{ball}	balance	33g
Vertical distance from the centre of the ball to the beam surface a_1	ruler	5mm

6.2.1 Moment of inertia of the beam

The moment of inertia of the beam can be measured through a physical pendulum shown in Figure 6.6. The relationship between the period and the moment of inertia of the pendulum is expressed as follows:

$$T = 2\pi\sqrt{\frac{I_{pivot}}{mgL}} \tag{6.1}$$

Where T is the period, m is the mass of the beam, g is acceleration of gravity,

L is the distance between the pivot point and the centre mass of the beam, and I_{pivot} is the moment of inertia of the beam at the pivot point.

The parallel axis theorem is used to transfer I_{pivot} to I_{centre_mass}, using

$$I_{pivot} = I_{centre\_mass} + mL^2 \tag{6.2}$$

where I_{centre_mass} is the moment of inertia of the beam at the centre of mass point.

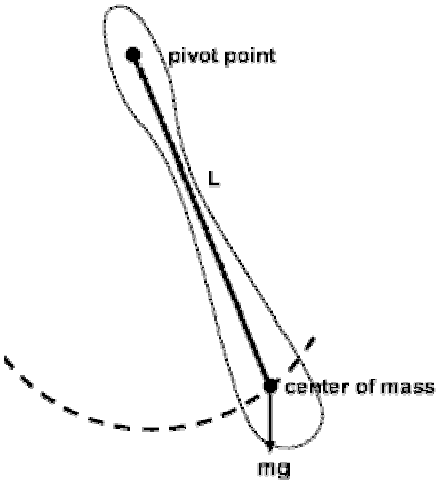

Figure 6.6: A schematic physical pendulum (Colwell 2007)

From the Rearrangement the Equation (6.1) and Equation (6.2), following expression can be derived:

$$I_{centre\_mass} = I_{pivot} - mL^2 = \frac{T^2 mgL}{4\pi^2} - mL^2 \tag{6.3}$$

By measurement, T equals 14.2 seconds, L equals 0.364m, m equals M_{beam}, and g equals $9.81\,\mathrm{m^2/s}$. Substituting those values into the equation (6.3), I_{cental_mass} can be determined , namely:

I_{cental_mass} =$0.019\,\mathrm{kgm^2}$

6.2.2 System damping

System damping is measured through experiment as shown in Figure 6.7. By matching the outputs, which are tilt angle of the beam and rotational speed of the beam, from the nominal plant and the real plant, the damping of the system can be determined. The damping of the system is 1.528 Nm/(rad/s). Figures 6.8 and 6.9 shows that the tilt angle of beam from the nominal plant matches that from the real plant, and rotational speed from the nominal plant matches that from the real plant, by using the damping shown as above. The rotation speed of the beam from real plant is noisy since it is directly differential from the tilt angle of the beam.

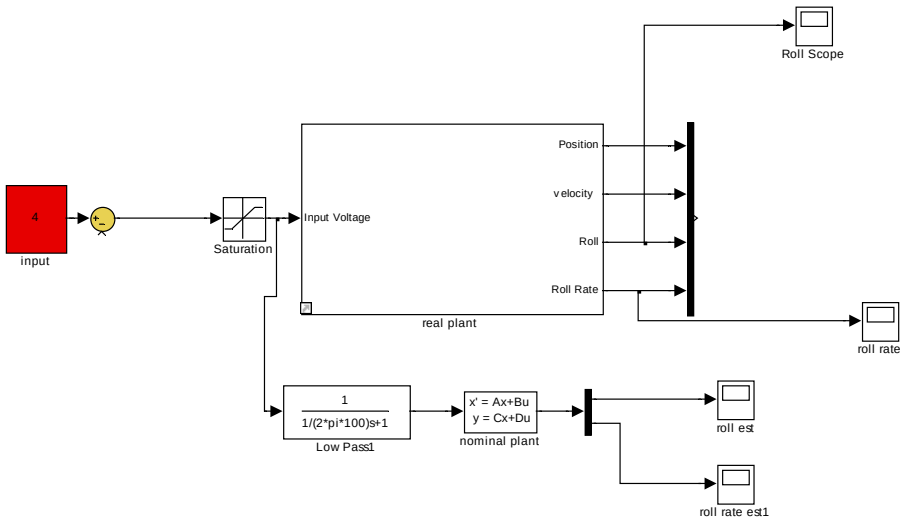

Figure 6.7: Experimentally determine the damping of the ball and beam system

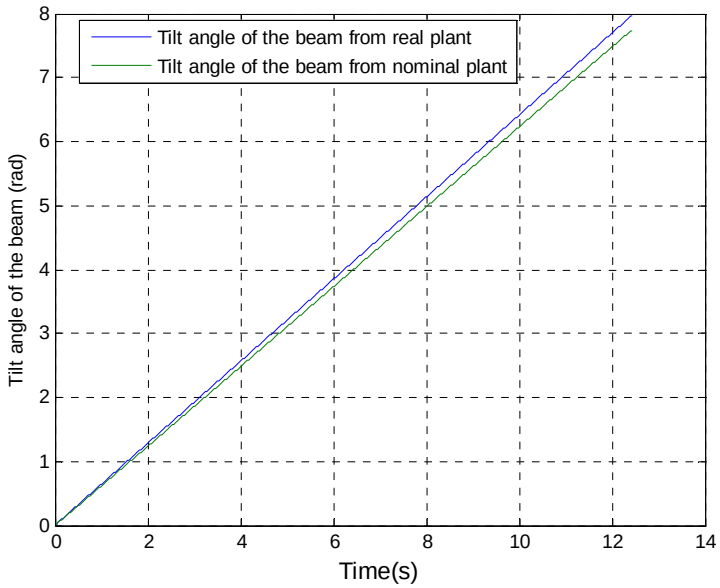

Figure 6.8: Match the tilt angle of the beam from real plant and nominal plant

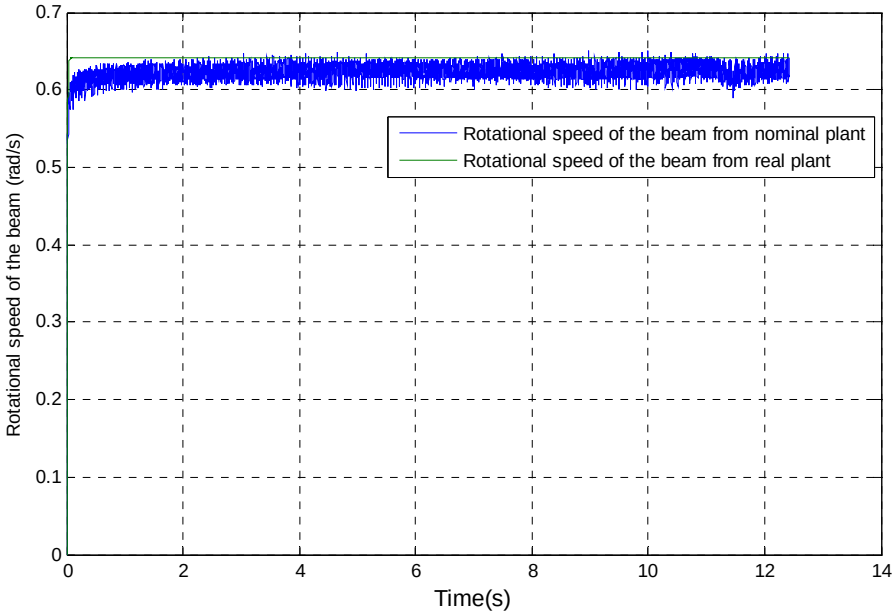

Figure 6.9: Match the rotational speed of the beam from real plant and nominal plant

6.3 Motor compensation

The response of the DC motor used for the ball and beam system in this project is nonlinear. The nonlinearity of the DC motor degrades the controllability of the ball and beam system due to the system being modelled linearly. It is therefore necessary to build a linear compensator to turn the nonlinear motor into a linear one for the actual application.

Figure 6.10 shows the voltage input from -20 volts to 20 volts of the DC motor versus rotational speed of the DC motor. When the voltage input to the DC motor is within about -2 volts to 2 volts, it stops. The zone, where the voltage input is nonzero but the rotational speed output of the DC motor is zero, is called the dead zone. The dead zone results in a discontinuous rotational speed output of the nonlinear DC motor.

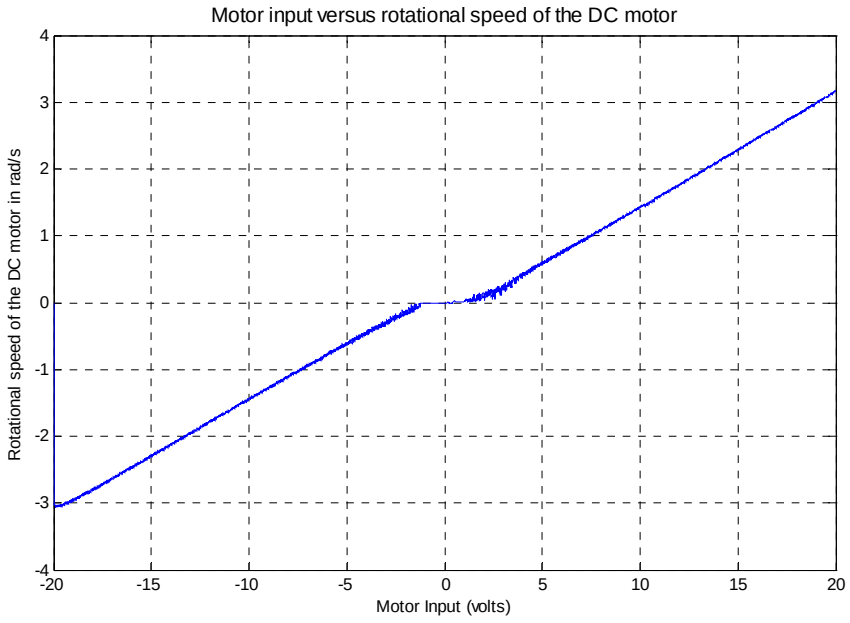

Figure 6.10: Voltage input versus rotational speed of the DC motor

Table 6.2 is the lookup table for compensating the nonlinear motor into the linear one. The lookup table computes an approximation to some function. The rotational speed of the DC motor for a given voltage input can be derived by measuring the angle position of the motor shaft and recording the time used, or alternatively by directly reading angle position from the digital encoder and recording the time used. Using Equation (6.4) and Equation (6.5), the desired slope and desired input voltage can be derived.

$$\text{Desired slope} = \frac{\text{Voltage input in 20volts}}{\text{Measured rotational speed at -3.199rad/s}} = -6.251\text{volt}/(\text{rad/s}) \quad (6.4)$$

$$\text{Desired input voltage} = \text{Desired slope} \times \text{Measured rational speed} \quad (6.5)$$

Table 6.2: Lookup table for motor deadzone compensation

Voltage input in volts	Measured rotational speed in rad/s	Desired input voltage in volts
20	-3.199	20
18	-2.841	17.760
16	-2.492	15.580
14	-2.148	13.434
12	-1.818	11.366
10	-1.454	9.093
8	-1.124	7.027
7	-0.938	5.870
6	-0.771	4.823
5	-0.609	3.812
4	-0.450	2.818
3.5	-0.371	2.324
3	-0.293	1.831
2.5	-0.215	1.346
2.2	-0.166	1.043
2	-0.133	0.837
1.8	-0.103	0.646
1.6	-0.066	0.416
1.4	-0.015	0.096
1.2	0	0

1	0	0
0.8	0	0
0.6	0	0
0.5	0	0
0.4	0	0
0.3	0	0
0.2	0	0
0.1	0	0
0	0	0
-0.1	0	0
-0.2	0	0
-0.3	0	0
-0.4	0	0
-0.5	0	0
-0.6	0	0
-0.8	0	0
-1	0	0
-1.2	0	0
-1.4	0	0
-1.6	0.041	-0.254
-1.8	0.084	-0.531
-2	0.119	-0.746
-2.2	0.153	-0.956
-2.5	0.204	-1.275
-3	0.284	-1.779
-3.5	0.363	-2.271
-4	0.444	-2.781
-5	0.595	-3.721
-6	0.757	-4.736
-7	0.942	-5.889
-8	1.111	-6.946
-10	1.464	-9.154
-12	1.818	-11.36
-14	2.180	-13.63
-16	2.485	-15.54
-18	2.887	-18.05
-20	3.168	-19.80

The lookup table has been derived in Table 6.2, with 2 dimensions and the Interpolation-Extrapolation lookup method. In the following, a deduction is given to

prove that by using the lookup table, the nonlinear motor has been linearly compensated. The function of the lookup table is defined:

$$\text{Voltage input} = F\ (\text{Desired input voltage}) \quad (6.6)$$

where 'Voltage input' is the voltage which is direct supplied to the DC motor, 'Desired input voltage' is the input voltage to the lookup table, both 'Voltage input' and 'Desired input voltage' are presented in Table 6.2, F is the function which represents the lookup table.

The relationship between the DC motor input and output can be expressed as follows:

$$\text{Measured rotational speed} = f\ (\text{Voltage input}) \quad (6.7)$$

where 'Measured rotational speed' (Table 6.2) is the speed measured at given voltage input.

Using the Equation (6.6) and Equation (6.7), following Equation can be derived:

$$\text{Measured rotational speed} = f\ [F\ (\text{Desired input voltage})] \quad (6.8)$$

Using the Equation 6.5 and Equation 6.8, following Equation can be derived:

$$\text{Measured rotational speed} = f\ [F\ (\text{Desired slope} \times \text{Measured rational speed})] \quad (6.9)$$

Rearrange the Equation (6.9), following equation can be derived:

$$\text{Measured rotational speed} = \text{Desired slope} \times f\ [F\ (\text{Measured rational speed})] \quad (6.10)$$

Since ‘Measured rotational speed’ is a variable, using a free variable ‘x’ to substitute ‘Measured rotational speed’ in Equation (6.9), following Equation can be derived:

$$f[F(x)]= \frac{1}{\text{Desired slope}}(x) \tag{6.11}$$

Equation (6.11) shows that function ‘f [F()]’ represents the ‘$\frac{1}{\text{Desired slope}}$’, which is the constant shown in Equation (6.4). This means that the nonlinear open loop DC motor function ‘f’ has been linearly compensated by the lookup table function ‘F’.

Figure 6.11 illustrates that by using the lookup table, the input to the lookup table is linearly related to the output of the DC motor by the factor of ‘$\frac{1}{\text{Desired slope}}$’.

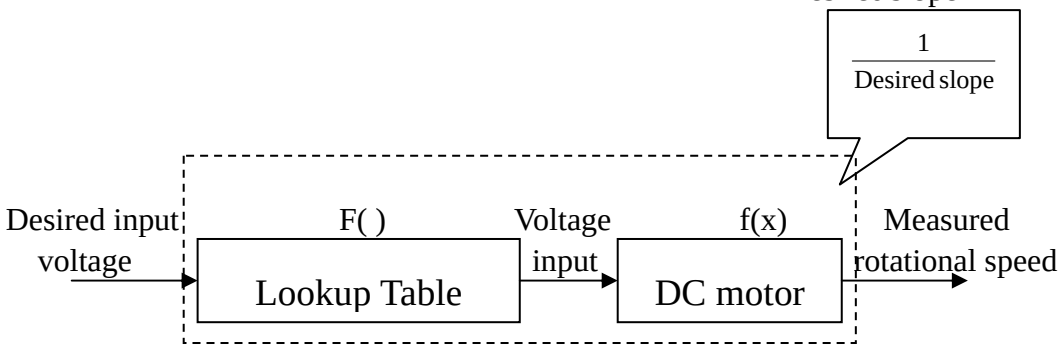

Figure 6.11: Illustration the use of the lookup table to compensate the DC motor

Figure 6.12 shows the rotational speed of the motor after passing through the input signal through the lookup table. By using the lookup table, the deadzone of the DC motor is almost completely offset.

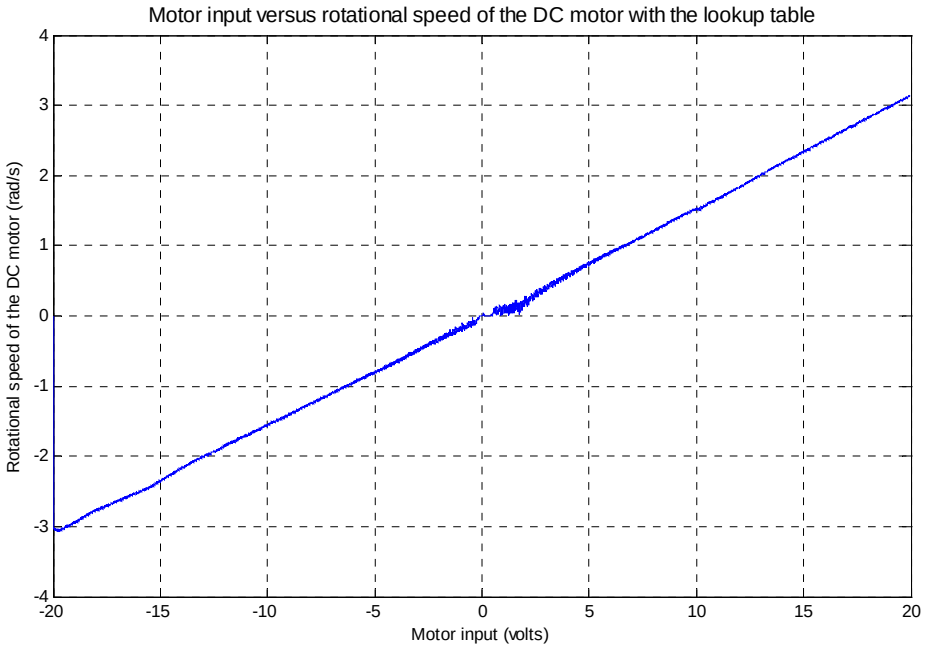

Figure 6.12: The rotational speed of the DC motor by using the lookup table

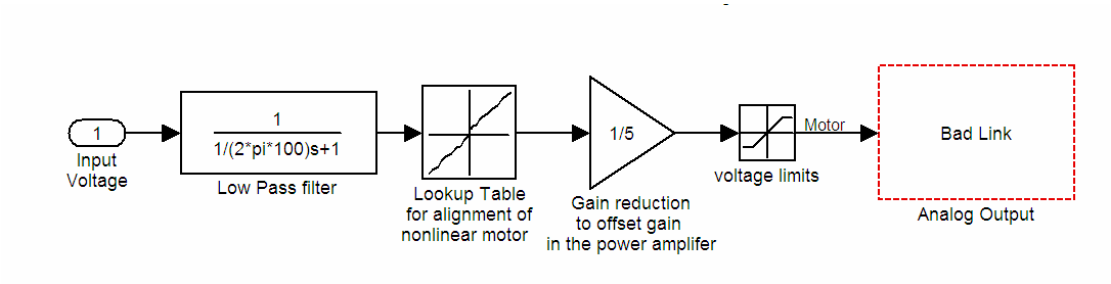

Figure 6.13: Interface between the input voltage of the motor and Quanser Q4 board

Figure 6.13 shows the use of the lookup table in the interface between the input voltage and the DC motor.

6.4 Conclusion

Chapter 6 deals with the experimental issues, such as the calibration of the sensors, measurement of the system' s parameters, and compensator for the nonlinear DC motor. By calibration of the sensors, meaningful output of the system, the position of

the ball and the tilt angle of the beam, can be derived. The parameters of the system were measured directly, except the moment of inertia of the beam, which is determined through the pendulum experiment. Finally, a lookup table was built in order to offset the nonlinearity in the motor.

The next chapter presents the conclusions of this thesis and suggests recommendations for future work.

Chapter 7

Conclusions

7.1 Conclusions

A mathematical model of the ball and beam system was developed by using the physical laws and electrical laws. A simplified state space mathematical model was derived by considering some system parameters, such as the friction between the ball and resistive wire and inductance of the DC motor, were so small that can be neglected. The controllability of the ball and beam system was thus improved by the simplification.

Simulations and experiments of the ball and beam system were conducted to investigate several control structures, for example full state feedback control, full state estimate control, and reduced order estimate control, and two gain calculation methods, which are the pole placement method and the LQR method. It is evident that the reduced order estimate control technique has the best performance since this

technique only estimates the states required. From the simulation and experimentation, the control gains and estimate gains by the LQR method present better performances than the pole placement method.

The system parameters were identified through direct measurement and experimentation. The advanced system identification methods, including the ‘free structured parameterizations’ approach, the ‘grey box’ approach, and the ‘black box’ approach, were used to estimate the system parameters from the open loop system and full state feedback control structure. The system parameters can also be estimated from the full state estimate control and reduced order estimate control structure by ‘free structured parameterizations’ approach. It is evident that the estimated parameters fit the measured parameters accurately, which further indicates the accuracy of the system parameters measurements.

The ball and beam system is fully controlled by full state feedback control, full state estimate control, and reduced order estimate control techniques with the pole placement and the LQR methods. The advanced system identification techniques prove that the system parameters are measured and modeled accurately.

A user manual has been written to guide users on the physical setup and operation of the rig. This is included in Appendix F.

7.2 Recommendations for further work

7.2.1 Model the friction between the ball and resistive wires

The ball rolling on the resistive wires suffers from the friction created. When the tilt angle of beam is small and the velocity of the ball is small too, the friction between the ball and resistive wire stops the ball rolling. The friction is not modeled in this ball and beam project. The ball and beam system could be modelled more accurately if the friction is considered in the modelling process.

7.2.2 Other advanced control techniques

The ball and beam system is controlled by the modern control method: state space control. It is likely that the system can be controlled by more advanced control techniques, such as nonlinear control, robust control, and fuzzy logical control, however, they are out of the scope of this project. The nonlinear control techniques should have advantages in the ball and beam system, since the ball and beam system is a nonlinear system.

References

Ambalavanar, S., Moinuddin, H. M. & Malyshev, A. (2006), Ball and Beam Balancer, BA Thesis, Lakehead University.

Arroyo, S. (2005), *Ball on Balancing Beam*, viewed 12 December 2006, http://bleex.me.berkeley.edu/ME102/proj_archive/F03/Proj9/contactinfo.htm

Cazzolato, B. (2006), Lecture notes distributed in the unit, 3028 Dynamics & Control II, School of Mechanical Engineering, The University of Adelaide.

Cazzolato, B. (2006), Tutorial handout distributed in the unit, System ID using the Matlab System Identification Toolbox, School of Mechanical Engineering, the University of Adelaide.

Colwell, C. H. (2007), *A physical pendulum, the parallel axis theorem and a bit of calculus*, viewed 3 February 2007, http://dev.physicslab.org/Document.aspx?doctype=2&filename=RotaryMotion_PhysicalPendulum.xml

Dorf, R. C. & Bishop, R. H. (2005), *Modern Control Systems*, 10nd edn, Prentice Hall, Upper Saddle River.

Hirsch, R. (1999), Ball on Beam Instructional system, Shamdor Motion systems.

Harmonic drive (2006), *DC servo systems*, viewed 10 August 2006, http://www.harmonicdrive.net/media/support/catalogs/pdf/dc_servo_catalog.pdf

Lieberman, J. (2004), *A robotic ball balancing beam*, viewed 12 September 2006, http://bea.st/sight/rbbb/rbbb.pdf

Ljung, L. (2007), *System Identification Toolbox 7 User's Guide*, Mathworks.

Matlab 7.01 (2007), *Matlab user guide*, 7.01 edn, Mathworks.

Matlab 7.01 (2007), *Getting started with system identification toolbox 7*, 7.01 edn, Mathworks.

Quanser (2006), Q4 hardware in the loop board, viewed 10 September, 2006, http://www.quanser.com/english/html/solutions/fs_Q4.html

Quanser (2006), *The ball and beam module*, viewed 3 September 2006, www.quanser.com/english/downloads/products/Rotary/Quanser_SRV02_BallBeam_PIS.pdf

Quanser (2006), *Universal power module - user manual*, viewed 13 September 2006, www.quanser.com/english/downloads/solutions/UPM_UserManual.pdf

Rosales, R. A. (2004), A ball-on-beam project kit, BA Thesis, Massachusetts Institute of Technology.

Solid Edge V19 (2006), UGS, http://www.ugs.com/

University of Alberta (2006), '*EE357 Control System I: Lab Manual*', viewed 4 December 2006, http://www.uofaweb.ualberta.ca/ece/

Visual C++ 6.0 (2006), Microsoft, http://www.microsoft.com/

Wellstead, P., Ball and beam 1: Basics, viewed 10 December 2006, http://www.control-systems-principles.co.uk/whitepapers/ball-and-beam1.pdf

Wincon 5.0 (2006), *WinCon User's guide*, 5.0 edn, Quanser.

Zadeh, L (1962), 'A critical view of our research in automatic control', *Automatic control*, vol.7, pp.74-75.

Appendix A

Matlab M-files for simulation and experiment

A.1 Full state feedback & estimator control by the pole placement method

```
%*****************************************************************%
%*************       Project: Ball & beam system      ************%
%*************       Supervisor: Dr. Ben Cazzolato        ********%
%*************                Wei Wang                  ***********%
%**********             The University of Adelaide         *******%
%*********             School of Mechanical Engineering          ***********%
%*****************************************************************%
clear all
close all
clc

%---- System parameters
Rb= 0.01                                    % ball's radius (m)
m_ball=4/3*pi*(Rb)^3*7800                   % mass of the ball (Kg)
m_beam= 0.381                               % mass of the beam (Kg)
L_beam= 0.7                                 % length of the beam (m)
```

```
R=    4.7                                   % Dc motor armuature resistance (ohm)
K=    4.91                                  % Motor torque constant (Nm/A)
Ke=0.5/(2*pi/60)                  % Motor back electromotive-force constant (Volt/(rad/s))
L= 1.6e-3                                   % Motor electrical inductance (H)
J_beam1= 1/12*(m_beam)*L_beam^2             % moment of inertia of the beam (Kg.m^2)
J_beam=0.019                                % moment of inertia of the beam by
measurement (Kg.m^2)
J_motor= 43e-3                              % moment of inertia of the motor (Kg.m^2)
J_bm=J_beam+J_motor               % moment of inertia of the beam & motor (Kg.m^2)
b=1.6e-1/(2*pi/60)                          % DC motor damping constant    (Nm/(rad/s))
g=9.81                                      % Acceleration of gravity (m/s^2)
a=0.005                                     % length from the center of the ball to the
beam surface (m)
n=100                                       % gear box reduction ratio


% Simplified state space model of the system (has been linearized)
AA=[0 1 0 0;
      0 0 g/(1+2/5*(Rb/a)^2) 0;
      0 0 0 1 ;
      -m_ball*g/J_bm 0 0 -(b+K*Ke/R)/J_bm];
BB=[0;0;0;K/(R*J_bm)];
CC=[1 0 0 0; 0 0 1 0];
DD=0


G=ss(AA,BB,CC,DD,'statename',{'position' 'ball velo' 'beam angle' 'beam rational velo'},...
'inputname',{'voltage'},'outputname',{'position','beam angle',},'notes','Created 5/8/2006' )


% Control system design by pole placement
% Check controllability and the condition number
Cm=ctrb(AA,BB)
Rank_c=rank(Cm)
cond(Cm)

% Check observability and condition number
Om=obsv(AA,CC)
Rank_o=rank(Om)
cond(Om)

% Assign closed loop poles
damp_c= 1
wn=2
```

```
[num,den]=ord2(wn,damp_c)
r_c=roots(den)
pole_c=[r_c(1),r_c(2),-5,-1]

% Calculate the feedback gain matrix via Ackermann's formula
KK=acker(AA,BB, pole_c)

% Calculate the estimate gain matrix via Ackermann's formula
pole_o=22*pole_c
LL=place(AA',CC',pole_o)'


% Add an augment state for command following
AA_a=[[AA;[1 0 0 0]],zeros(5,1)]
BB_a=[BB;zeros(1,1)]
Cm_a=ctrb(AA_a,BB_a)
Rank_c_a=rank(Cm_a)

% Calculate the feedback gain matrix via Ackermann's formula for the
% Augment ball and beam system
pole_c_a=[pole_c,-10]
KK_a=acker(AA_a,BB_a,pole_c_a)

% Separate feedback control gain for the augment state and the original
% states
KK_111=KK_a(5)
KK_222=KK_a(1:4)


% Plot results for full state feedback control
T=[0:10/length(output):10]

figure
plot(T(1:length(output)), output)
title('Simulation of the pole placement method for full state feedback control')
xlabel('Time in seconds')
ylabel('The position of the ball on the beam in metres')
grid


figure
plot(T(1:length(output)), output1)
title('Simulation of the pole placement method for full state feedback control')
xlabel('Time in seconds')
```

```
ylabel('The tilt angle of the beam in degrees')
grid


% Plot results for full state estimate control (double click the manual
% switch in 'full_state_estimate_simulation.mdl' and run simulation again)
clc
figure

plot(T(1:length(output)), output)
title('Simulation of the pole placement method for full state estimate control')
xlabel('Time in seconds')
ylabel('The position of the ball on the beam in metres')
grid

figure
plot(T(1:length(output)), output1)
title('Simulation of the pole placement method for full state estimate control')
xlabel('Time in seconds')
ylabel('The tilt angle of the beam in degrees')
grid
```

A.2 Full state feedback & estimator control by the LQR method

```
%*****************************************************************%
%*************       Project: Ball & beam system      ************%
%*************       Supervisor: Dr. Ben Cazzolato        *********%
%*************              Wei Wang              *****************%
%***********            The University of Adelaide        **********%
%**********           School of Mechanical Engineering          *******%
%*****************************************************************%
clear all
close all
clc

%---- System parameters
Rb= 0.01                                    % ball's radius (m)
m_ball=4/3*pi*(Rb)^3*7800                   % mass of the ball (Kg)
m_beam= 0.381                               % mass of the beam (Kg)
```

```
L_beam= 0.7                                  % length of the beam (m)
R=    4.7                                    % Dc motor armuature resistance (ohm)
K=    4.91                                   % Motor torque constant (Nm/A)
Ke=0.5/(2*pi/60)                             % Motor back electromotive-force constant
(Volt/(rad/s))
L= 1.6e-3                                    % Motor electrical inductance (H)
J_beam1= 1/12*(m_beam)*L_beam^2              % moment of inertia of the beam (Kg.m^2)
J_beam=0.019                                 % moment of inertia of the beam by
measurement (Kg.m^2)
J_motor= 43e-3                               % moment of inertia of the motor (Kg.m^2)
J_bm=J_beam+J_motor                          % moment of inertia of the beam & motor
(Kg.m^2)
b=1.6e-1/(2*pi/60)                           % DC motor damping constant
(Nm/(rad/s))
g=9.81                                       % Acceleration of gravity (m/s^2)
a=0.005                                      % length from the center of the ball to the
beam surface (m)
n=100                                        % gear box reduction ratio

% Simplified state space model of the system (has been linearized)
AA=[0 1 0 0;
    0 0 g/(1+2/5*(Rb/a)^2) 0;
    0 0 0 1 ;
    -m_ball*g/J_bm 0 0 -(b+K*Ke/R)/J_bm];
BB=[0;0;0;K/(R*J_bm)];
CC=[1 0 0 0; 0 0 1 0];
DD=0



CC1=[1 0]
G=ss(AA,BB,CC,DD,'statename',{'position' 'ball velo' 'beam angle' 'beam rotational velo'},...
'inputname',{'voltage'},'outputname',{'position','beam rotational angle'},'notes','Created
5/8/2006' )


% Add an augment state for command following
AA_a=[[AA;[1 0 0 0]],zeros(5,1)]
BB_a=[BB;zeros(1,1)]
Cm_a=ctrb(AA_a,BB_a)
Rank_c_a=rank(Cm_a)

% Design based on LQR method
% Assign control weighting matrix 'Q' and 'R'
```

```
Q = [10 0 0 0 0;
      0 1 0 0 0;
      0 0 10 0 0;
      0 0 0 1 0;
      0 0 0 0 10;]

R =0.05

% Calculate feedback gains by Algebraic Riccati Equation
[KKK S E] = lqr(AA_a,BB_a,Q,R)

% Separate feedback control gains for the augment state and the original states
KK_111=KKK(5)
KK_222=KKK(1:4)


% Design estimator by LQR method
QQ=1000
RR=0.0001*[5 0 ;
     0 0.01]

% Calculate estimator gains by Algebraic Riccati Equation
[LL1 SS EE]=lqr(AA',CC',BB*QQ*BB', RR)
LL=LL1'

% Plot results for full state feedback control
T=[0:10/length(output):10]

figure
plot(T(1:length(output)), output)
title('Simulation of LQR method for full state feedback control')
xlabel('Time in seconds')
ylabel('The position of the ball on the beam in metres')
grid

figure
plot(T(1:length(output)), output1)
title('Simulation of LQR method for full state feedback control')
xlabel('Time in seconds')
ylabel('The tilt angle of the beam in degrees')
grid


% Plot results for full state estimate control (double click the manual
```

```
% switch in 'full_state_estimate_simulation.mdl' and run simulation again)
figure
plot(T(1:length(output)), output)
title('Simulation of LQR method for full state estimate control')
xlabel('Time in seconds')
ylabel('The position of the ball on the beam in metres')
grid

figure
plot(T(1:length(output)), output1)
title('Simulation of LQR method for full state estimate control')
xlabel('Time in seconds')
ylabel('The tilt angle of the beam in degrees')
grid
```

A.3 Reduced order observers by the LQR method

```
%*********************************************************************%
%**************      Project: Ball & beam system      ******************%
%*************      Supervisor: Dr. Ben Cazzolato        *****************%
%************              Wei Wang              **********************%
%**********            The University of Adelaide        ****************%
%*********           School of Mechanical Engineering          **********%
%*********************************************************************%
clear all
close all
clc

%---- System parameters
Rb= 0.01                                    % ball's radius (m)
m_ball=4/3*pi*(Rb)^3*7800                   % mass of the ball (Kg)
m_beam= 0.381                               % mass of the beam (Kg)
L_beam= 0.7                                 % length of the beam (m)
R=    4.7                                   % Dc motor armuature resistance (ohm)
K=    4.91                                  % Motor torque constant (Nm/A)
Ke=0.5/(2*pi/60)                            % Motor back electromotive-force constant
(Volt/(rad/s))
L= 1.6e-3                                   % Motor electrical inductance (H)
J_beam1= 1/12*(m_beam)*L_beam^2             % moment of inertia of the beam (Kg.m^2)
J_beam=0.019                                % moment of inertia of the beam by
```

```
measurement (Kg.m^2)
J_motor= 43e-3                                  % moment of inertia of the motor (Kg.m^2)
J_bm=J_beam+J_motor                             % moment of inertia of the beam & motor
(Kg.m^2)
b=1.6e-1/(2*pi/60)                              % DC motor damping constant    (Nm/(rad/s))
g=9.81                                          % Acceleration of gravity (m/s^2)
a=0.005                                         % length from the center of the ball to the
beam surface (m)
n=100                                           % gear box reduction ratio

% Simplied state space model of the system (have been linearized)
AA=[0 0 1 0;
      0 0 0 1 ;
     0 g/(1+2/5*(Rb/a)^2) 0 0;
     -m_ball*g/J_bm 0 0 -(b+K^2/R)/J_bm];
BB=[0;0;0;K/(R*J_bm)];
CC=[1 0 0 0; 0 1 0    0];
DD=0

CC1=[1 0]

G=ss(AA,BB,CC,DD,'statename',{'position' 'ball velo' 'beam angle' 'beam rational velo'},...
'inputname',{'voltage'},'outputname',{'position','beam angle'},'notes','Created 5/8/2006' )

% Build the reduced order system
Aaa=AA(1:2,1:2)
Abb=AA(3:4,3:4)
Aab=AA(1:2,3:4)
Aba=AA(3:4,1:2)

Ba=BB(1:2), Bb=BB(3:4)


% Add an augment system for command following
AA_a=[[AA;[1 0 0 0]],zeros(5,1)]
BB_a=[BB;zeros(1,1)]
Cm_a=ctrb(AA_a,BB_a)
Rank_c_a=rank(Cm_a)
cond(Cm_a)



% LQR controller and LQR estimator
% Assign weighting matrix 'Q' and 'R'
```

```
Q = [10 0 0 0 0 ;
      0 10 0 0 0 ;
      0 0 10 0 0 ;
      0 0 0 10 0 ;
      0 0 0 0 2 ]

R =0.05

% Calculate feedback gains by Algebraic Riccati Equation
[KKK S E] = lqr(AA_a,BB_a,Q,R)

% separate feedback gains for the augment state, measured states, and
% observed states
KK_111=KKK(5)
KK_222=KKK(1:4)
Ka=[KK_222(1),KK_222(2)]
Kb=[KK_222(3),KK_222(4)]

% calclulate gain matrix for the LQR estimator by Algebraic Riccati
% Equation
QQ=10
RR=[1 0 ;
    0 0.01 ]

[LL1 SS EE]=lqr(Abb',Aab',Aba*QQ*Aba', RR)
LL=LL1'



% plot results for reduced order system
T=[0:20/length(output):20]
figure

plot(T(1:length(output)), output)
title('Simulation of LQR method for reduced order system')
xlabel('Time in seconds')
ylabel('The position of the ball on the beam in metres')
grid

figure
plot(T(1:length(output)), output1)
title('Simulation of LQR method for reduced order system')
xlabel('Time in seconds')
ylabel('The tilt angle of the beam in degrees')
grid
```

Appendix B

Matlab M-files for system identification

B.1 ID the open loop plant

B.1.1 Main program

```
%************************************************************%
%*************        Project: Ball & beam system      ******************%
%************        Supervisor: Dr. Ben Cazzolato         ****************%
%************                Wei Wang                ********************%
%**********              The University of Adelaide         ****************%
%*********            School of Mechanical Engineering          ***********%
%************************************************************%

clear all
close all
clc


%---- System parameters
Rb= 0.01                                        % ball's radius (m)
m_ball=4/3*pi*(Rb)^3*7800                       % mass of the ball (Kg)
m_beam= 0.381                                   % mass of the beam (Kg)
L_beam= 0.7                                     % length of the beam (m)
R=    4.7                                       % Dc motor armuature resistance (ohm)
K=    4.91                                      % Motor torque constant (Nm/A)
Ke=0.5/(2*pi/60)                % Motor back electromotive-force constant (Volt/(rad/s))
```

```
L= 1.6e-3                                   % Motor electrical inductance (H)
J_beam1= 1/12*(m_beam)*L_beam^2             % moment of inertia of the beam (Kg.m^2)
J_beam=0.019                    % moment of inertia of the beam by measurement (Kg.m^2)
J_motor= 43e-3                              % moment of inertia of the motor (Kg.m^2)
J_bm=J_beam+J_motor             % moment of inertia of the beam & motor (Kg.m^2)
b=1.6e-1/(2*pi/60)                          % DC motor damping constant  (Nm/(rad/s))
g=9.81                                      % Acceleration of gravity (m/s^2)
a=0.005                                     % length from the center of the ball to the
beam surface (m)
n=100                                       % gearbox reduction ratio


% Simplified state space model of the system (have been linearized)
AA=[0 1 0 0;
    0 0 g/(1+2/5*(Rb/a)^2) 0;
    0 0 0 1 ;
    -m_ball*g/J_bm 0 0 -(b+K^2/R)/J_bm];
BB=[0;0;0;K/(R*J_bm)];
CC=[1 0 0 0; 0 0 1 0];
DD=[0;0]


G=ss(AA,BB,CC,DD,'statename',{'position' 'ball velo' 'beam angle' 'beam rational velo'},...
'inputname',{'voltage'},'outputname',{'position','beam angle'},'notes','Created 5/8/2006' )


% Build the structure model
A=G.a
A(find(((A~=1))&((A~=0))))=(find(((A~=1))&((A~=0))))*NaN;
B=G.b
B(find(((B~=1))&((B~=0))))=(find(((B~=1))&((B~=0))))*NaN;

% Define the model which will be used in estimation
ES=idss(G.a,G.b,G.c,G.d,'Ts','0')

% Define structure matrics
ES.as=A;
ES.bs=B
ES.cs=ES.c
ES.ds=ES.d

ES.DisturbanceModel = 'Estimate';
```

```
% Simulate the 'ID_open_loop_model.mdl' to create input and output Data
U=D_input;
Y1=D_output;
Y=Y1(1:length(U),:);
T=[0:0.005:10];


length(U),length(Y),length(T)

% Look at the data
figure
subplot(2,1,1)
title('Input and Output Data')
plot(T,U)
ylabel('Input Data (Volts)')
xlabel('Time (s)')
subplot(2,1,2)
[AX,H1,H2] = plotyy(T,Y(:,1),T,Y(:,2),'plot')
set(get(AX(1),'Ylabel'),'String','Position of the ball (m)')
set(get(AX(2),'Ylabel'),'String','Tilt angle of the beam (rad)')
title('Outputs of the system')
xlabel('Time (s)')
set(H1,'LineStyle','--')
set(H2,'LineStyle',':')


% Check the quality of the data (calculate coherence between input and output)
[dummy, number_of_inputs] = size(G.b)
[number_of_outputs,dummy] = size(G.c)
NFFT = 1024;
COHERENCE = zeros(NFFT/2+1,number_of_outputs);

fs=1/0.005
F=200/1024*[0:1024/2]
figure
for eachoutput = 1:number_of_outputs
   subplot(number_of_outputs,1,eachoutput)
   for eachinput = 1:number_of_inputs
        % Determine the coherence
        COHERENCE(:,eachoutput)                =                COHERENCE(:,eachoutput)+
mscohere(U(:,eachinput),Y(:,eachoutput),hanning(NFFT),0.75*NFFT,NFFT,fs);
   end
   plot(F,COHERENCE(:,eachoutput))
```

```
   if eachoutput < 2
   title ('Coherence between input and position of the ball')
   xlabel('Frequency (Hz)')
   else
   title('Coherence between input and tilt angle of the beam')
   xlabel('Frequency (Hz)')
   end
end


% Define the dataset
Data = iddata(Y,U,1/fs)
set(Data,'InputName',{'Voltage'},'OutputName',{'Position','Tilt angle'});
figure; plot(Data);


% Improve the date quality by filter or smooth methods
gs = spa(Data);              % Smoothed spectral estimate
figure; bode(gs);
refresh;

% Final ID by 'Free parameterisation', 'grey box',and 'black box'approach
disp('Use a "Free parameterisation" approach where we know the structure of the model')
model1 = pem(Data,ES,'Ts',0,'trace','on','Tolerance',1/1000);

disp('Use a "grey" box approach where we know the structure of the model')
ES1 = idgrey('model_ball_beam',[Rb a g m_ball J_bm K R b],'c',[]);
model2 = pem(Data,ES1,'Ts',0,'trace','on','Tolerance',1/1000,'InitialState','zero');


disp('Use a "black" box approach')
model3 = pem(Data,4,'trace','on');


% ____________________________________________________________________________
% Now compare the results from the ID
% ____________________________________________________________________________

clc
disp('Nominal Plant then Estimated Plants')
G.a, model1.a, model2.a,model3.a
G.b, model1.b, model2.b,model3.b
```

```
% The following command compares the time domain fitted model against the actual output
% and the output from the nominal plant
figure; compare(Data,G, model1,'-',model2,':',model3,'-.');

% present results from free parameterization approach
r1a=model1.a
r1b=model1.b
r1c=model1.c
r1d=model1.d


%    present results from grey box approach
r2a=model2.a
r2b=model2.b
r2c=model2.c
r2d=model2.d

% present results from black box approach
r3a=model3.a
r3b=model3.b
r3c=model3.c
r3d=model3.d
```

B.1.2 M-file for ‘grey box’ approach

```
function [A,B,C,D,KK,X0] = model_ball_beam(pars,T,auxarg)
% This file is to build a state model of the ball&beam system

Rb = pars(1);           % ball radium (m)
a = pars(2);              % length from the center of the ball to the beam surface (m)
g = pars(3);          % Acceleration of gravity (m/s^2)
m_ball = pars(4);          % mass of the ball (Kg)
J_bm = pars(5);            % moment of inertia of beam & motor (Kg.m^2)
K = pars(6);               % electromotive force of constant, that is torque constant (Nm/A)
R= pars(7);                % Dc motor armuature resistance (ohm)
b= pars(8);                    % DC motor damping constant    (Nm/(rad/s))



A=[0 1 0 0;
      0 0 g/(1+2/5*(Rb/a)^2) 0;
```

```
        0 0 0 1 ;
        -m_ball*g/J_bm 0 0 -(b+K^2/R)/J_bm];
B=[0;0;0;K/(R*J_bm)];
C=[1 0 0 0; 0 0 1 0];
D=[0;0]

% This is the gain matrix for the disturbances
KK = zeros(4,2);

% This is the initial states
X0 = zeros(4,1);
```

B.2 ID full state feedback

B.2.1 Main program

```
%************************************************************%
%*************      Project: Ball & beam system      ******************%
%************      Supervisor: Dr. Ben Cazzolato      *****************%
%************              Wei Wang              *********************%
%**********           The University of Adelaide       ****************%
%*********          School of Mechanical Engineering       ***********%
%************************************************************%
clear all
close all
clc


%---- System parameters
Rb= 0.01                                    % ball's radius (m)
m_ball=4/3*pi*(Rb)^3*7800                   % mass of the ball (Kg)
m_beam= 0.381                               % mass of the beam (Kg)
L_beam= 0.7                                 % length of the beam (m)
R=    4.7                                   % Dc motor armuature resistance (ohm)
K=    4.91                                  % Motor torque constant (Nm/A)
Ke=0.5/(2*pi/60)              % Motor back electromotive-force constant (Volt/(rad/s))
L= 1.6e-3                                   % Motor electrical inductance (H)
J_beam1= 1/12*(m_beam)*L_beam^2             % moment of inertia of the beam (Kg.m^2)
J_beam=0.019                  % moment of inertia of the beam by measurement (Kg.m^2)
```

```
J_motor= 43e-3                          % moment of inertia of the motor (Kg.m^2)
J_bm=J_beam+J_motor                     % moment of inertia of the beam & motor (Kg.m^2)
b=1.6e-1/(2*pi/60)                      % DC motor damping constant    (Nm/(rad/s))
g=9.81                                  % Acceleration of gravity (m/s^2)
a=0.005                                 % length from the center of the ball to the beam surface (m)
n=100                                   % gearbox reduction ratio


% Simplified state space model of the system (have been linearized)
AA=[0 1 0 0;
    0 0 g/(1+2/5*(Rb/a)^2) 0;
    0 0 0 1 ;
    -m_ball*g/J_bm 0 0 -(b+K^2/R)/J_bm];
BB=[0;0;0;K/(R*J_bm)];
CC=[1 0 0 0; 0 0 1 0];
DD=[0;0]



GG=ss(AA,BB,CC,DD,'statename',{'position' 'ball velo' 'beam angle' 'beam rational velo'},...
'inputname',{'voltage'},'outputname',{'position','beam angle'},'notes','Created 5/8/2006' )

% Full state feedback using pole placement method
% Check controllability and the condition number
Cm=ctrb(AA,BB)
Rank_c=rank(Cm)
cond(Cm)

% Check observability and condition number
Om=obsv(AA,CC)
Rank_o=rank(Om)
cond(Om)

% Assign closed loop poles
damp_c= 1
wn=5
[num,den]=ord2(wn,damp_c)
r_c=roots(den)
pole_c=[r_c(1),r_c(2),-5,-8]

% Calculate the feedback gain matrix via Ackermann's formula
KKK=acker(AA,BB, pole_c)
```

```
% Construct full state feedback    in order to
% derive the structure of the system

G=AA-BB*KKK

% Build the structure model
A=G
A(find(((A~=1))&((A~=0))))=(find(((A~=1))&((A~=0))))*NaN;
B=GG.b
B(find(((B~=1))&((B~=0))))=(find(((B~=1))&((B~=0))))*NaN;

% define the model which will be used in estimation
ES=idss(G,GG.b,GG.c,GG.d,'Ts','0')

% Define structure matrics
ES.as=A;
ES.bs=B
ES.cs=ES.c
ES.ds=ES.d

ES.DisturbanceModel = 'Estimate';

% Simulate the 'full_state_estimate.mdl' to create Data
U=D_input;
Y1=D_output;
Y=Y1(1:length(U),:);
T=[0:0.005:10];


length(U),length(Y),length(T)

% Look at the data
figure
subplot(2,1,1)
title('Input and Output Data')
plot(T,U)
ylabel('Input Data (Volts)')
xlabel('Time (s)')
subplot(2,1,2)
[AX,H1,H2] = plotyy(T,Y(:,1),T,Y(:,2),'plot')
set(get(AX(1),'Ylabel'),'String','Position of the ball (m)')
set(get(AX(2),'Ylabel'),'String','Tilt angle of the beam (rad)')
title('Outputs of the system')
xlabel('Time (s)')
```

```
set(H1,'LineStyle','--')
set(H2,'LineStyle',':')


% check the quality of the data (calculate coherence between input and output)
[dummy, number_of_inputs] = size(GG.b)
[number_of_outputs,dummy] = size(GG.c)
NFFT = 1024;
COHERENCE = zeros(NFFT/2+1,number_of_outputs);

fs=1/0.005
F=200/1024*[0:1024/2]
figure
for eachoutput = 1:number_of_outputs
   subplot(number_of_outputs,1,eachoutput)
   for eachinput = 1:number_of_inputs
        % Determine the coherence
        COHERENCE(:,eachoutput)                    =                    COHERENCE(:,eachoutput)+
mscohere(U(:,eachinput),Y(:,eachoutput),hanning(NFFT),0.75*NFFT,NFFT,fs);
   end
   plot(F,COHERENCE(:,eachoutput))
    if eachoutput < 2
   title ('Coherence between input and position of the ball')
   xlabel('Frequency (Hz)')
   else
   title('Coherence between input and tilt angle of the beam')
   xlabel('Frequency (Hz)')
   end
end


% Define the dataset
Data = iddata(Y,U,1/fs)
set(Data,'InputName',{'Voltage'},'OutputName',{'Position','Tilt angle'})
figure; plot(Data)


% Improve the date quality by filter or smooth methods
gs = spa(Data);              % Smoothed spectral estimate
figure; bode(gs)
refresh

%    Final ID
disp('Use a "Free parameterisation" approach where we know the structure of the model')
```

```
model1 = pem(Data,ES,'Ts',0,'trace','on','Tolerance',1/1000);

disp('Use a "grey" box approach where we know the structure of the model')
ES1 = idgrey('model_ball_beam_closed',[Rb a g m_ball J_bm K R b],'c',[]);
model2 = pem(Data,ES1,'Ts',0,'trace','on','Tolerance',1/1000,'InitialState','zero');

disp('Use a "black" box approach')
model3 = pem(Data,4,'trace','on');


% ____________________________________________________________________________
% Now compare the results from the ID
% ____________________________________________________________________________

clc
disp('Nominal Plant then Estimated Plants')
G, model1.a

% The following command compares the time domain fitted model against the measured output
% and the output from the estimated plant
figure;
clc
compare(Data,GG, model1,'-',model2,':',model3,'-.')

% Separate the state space matrix from estimate model
AAA=model2.a+model2.b*KKK
BBB=model2.b
CCC=model2.c
DDD=model2.d

% compare difference between measured plant and estimated plate
clc
AAA,AA
BBB,BB
CCC,CC
DDD,DD


% present results from free parameterization approach
r1a=model1.a+model1.b*KKK
r1b=model1.b
r1c=model1.c
r1d=model1.d
```

```
% present results from grey box approach
r2a=model2.a+model2.b*KKK
r2b=model2.b
r2c=model2.c
r2d=model2.d


% present results from black box approach
r3a=model3.a+model3.b*KKK
r3b=model3.b
r3c=model3.c
r3d=model3.d
```

B.2.2 M-file for ‘grey box’ approach

```
function [A,B,C,D,KK,X0] = model_ball_beam_closed(pars,T,auxarg)
% This file is to build a state model of the ball&beam system

Rb = pars(1);           % ball radium (m)
a = pars(2);              % length from the center of the ball to the beam surface (m)
g = pars(3);          % Acceleration of gravity (m/s^2)
m_ball = pars(4);          % mass of the ball (Kg)
J_bm = pars(5);           % moment of inertia of beam & motor (Kg.m^2)
K = pars(6);              % electromotive force of constant, that is torque constant (Nm/A)
R= pars(7);               % Dc motor armuature resistance (ohm)
b= pars(8);                   % DC motor damping constant    (Nm/(rad/s))



AA=[0 1 0 0;
      0 0 g/(1+2/5*(Rb/a)^2) 0;
      0 0 0 1 ;
      -m_ball*g/J_bm 0 0 -(b+K^2/R)/J_bm];
B=[0;0;0;K/(R*J_bm)];
C=[1 0 0 0; 0 0 1 0];
D=[0;0]

A=AA-B*[ 15.4226      11.4038      11.5729      -5.0075]
% This is the gain matrix for the disturbances
KK = zeros(4,2);
```

```
% This is the initial states
X0 = zeros(4,1);
```

B.3 ID full state observers

```
%*************************************************************%
%*************       Project: Ball & beam system      ******************%
%************       Supervisor: Dr. Ben Cazzolato         ****************%
%************                 Wei Wang                 *********************%
%**********              The University of Adelaide         ***************%
%*********             School of Mechanical Engineering          **********%
%*************************************************************%
clear all
close all
clc


%---- System parameters
Rb= 0.01                                        % ball's radius (m)
m_ball=4/3*pi*(Rb)^3*7800                       % mass of the ball (Kg)
m_beam= 0.381                                   % mass of the beam (Kg)
L_beam= 0.7                                     % length of the beam (m)
R=    4.7                                       % Dc motor armuature resistance (ohm)
K=    4.91                                      % Motor torque constant (Nm/A)
Ke=0.5/(2*pi/60)                                % Motor back electromotive-force constant
(Volt/(rad/s))
L= 1.6e-3                                       % Motor electrical inductance (H)
J_beam1= 1/12*(m_beam)*L_beam^2                 % moment of inertia of the beam (Kg.m^2)
J_beam=0.019                    % moment of inertia of the beam by measurement (Kg.m^2)
J_motor= 43e-3                                  % moment of inertia of the motor (Kg.m^2)
J_bm=J_beam+J_motor             % moment of inertia of the beam & motor (Kg.m^2)
b=1.6e-1/(2*pi/60)                              % DC motor damping constant    (Nm/(rad/s))
g=9.81                                          % Acceleration of gravity (m/s^2)
a=0.005                                         % length from the center of the ball to the
beam surface (m)
```

```
n=100                                   % gearbox reduction ratio

% Simplified state space model of the system (have been linearized)
AA=[0 1 0 0;
    0 0 g/(1+2/5*(Rb/a)^2) 0;
    0 0 0 1 ;
    -m_ball*g/J_bm 0 0 -(b+K^2/R)/J_bm];
BB=[0;0;0;K/(R*J_bm)];
CC=[1 0 0 0; 0 0 1 0];
DD=[0;0]

GG=ss(AA,BB,CC,DD,'statename',{'position' 'ball velo' 'beam angle' 'beam rational velo'},...
'inputname',{'voltage'},'outputname',{'position','beam angle'},'notes','Created 5/8/2006' )

% Full state feedback using pole placement method
% Check controllability and the condition number
Cm=ctrb(AA,BB)
Rank_c=rank(Cm)
cond(Cm)

% Check observability and condition number
Om=obsv(AA,CC)
Rank_o=rank(Om)
cond(Om)

% Assign closed loop poles
damp_c= 1
wn=3
[num,den]=ord2(wn,damp_c)
r_c=roots(den)
pole_c=[r_c(1),r_c(2),-8,-5]

% Calculate the feedback gain matrix via Ackermann's formula
KK=acker(AA,BB, pole_c)

% Calculate the estimate gain matrix via Ackermann's formula
pole_o=20*pole_c
LL=place(AA',CC',pole_o)'

% Construct full state feedback using full state observers in order to
% derive the structure of the system
GA=[AA -BB*KK;LL*CC AA-BB*KK-LL*CC]
GB=[BB;BB]
GC=[CC zeros(2,4)]
```

```
GD=DD

% Build estimated model
A=GA
A(find(((A~=1))&((A~=0))))=(find(((A~=1))&((A~=0))))*NaN;
B=GB
B(find(((B~=1))&((B~=0))))=(find(((B~=1))&((B~=0))))*NaN;
C=GC
C(find(((C~=1))&((C~=0))))=(find(((C~=1))&((C~=0))))*NaN;
D=GD

% Define the model which will be used in estimation
ES=idss(GA,GB,GC,GD,'Ts','0')

% Define structure matrics
ES.as=A;
ES.bs=B
ES.cs=ES.c
ES.ds=ES.d

ES.DisturbanceModel = 'Estimate';

% simulate the 'full_state_estimate.mdl' to create Data
U=D_input;
Y1=D_output;
Y=Y1(1:length(U),:);
T=[0:0.005:10];


length(U),length(Y),length(T)

% Look at input and output data
figure
subplot(2,1,1)
title('Input and Output Data')
plot(T,U)
ylabel('Input Data (Volts)')
xlabel('Time (s)')
subplot(2,1,2)
[AX,H1,H2] = plotyy(T,Y(:,1),T,Y(:,2),'plot')
set(get(AX(1),'Ylabel'),'String','Position of the ball (m)')
set(get(AX(2),'Ylabel'),'String','Tilt angle of the beam (rad)')
title('Outputs of the system')
xlabel('Time (s)')
```

```
set(H1,'LineStyle','--')
set(H2,'LineStyle',':')


% Check the quality of the data (coherence calculation)
[dummy, number_of_inputs] = size(GG.b)
[number_of_outputs,dummy] = size(GG.c)
NFFT = 1024;
COHERENCE = zeros(NFFT/2+1,number_of_outputs);

fs=1/0.005
F=200/1024*[0:1024/2]
figure
for eachoutput = 1:number_of_outputs
   subplot(number_of_outputs,1,eachoutput)
   for eachinput = 1:number_of_inputs
        % Determine the coherence
        COHERENCE(:,eachoutput)                    =                    COHERENCE(:,eachoutput)+
mscohere(U(:,eachinput),Y(:,eachoutput),hanning(NFFT),0.75*NFFT,NFFT,fs);
   end

   plot(F,COHERENCE(:,eachoutput))
   if eachoutput < 2
   title ('Coherence between input and position of the ball')
   xlabel('Frequency (Hz)')
   else
   title('Coherence between input and tilt angle of the beam')
   xlabel('Frequency (Hz)')
   end
end


% define the dataset for ID
Data = iddata(Y,U,1/fs)
set(Data,'InputName',{'Voltage'},'OutputName',{'Position','Tilt angle'})
figure; plot(Data)


% Improve the date quality by filter or smooth methods
gs = spa(Data);              % Smoothed spectral estimate
figure; bode(gs)
refresh

%    Final ID
```

```
disp('Use a "Free parameterisation" approach where we know the structure of the model')
model1 = pem(Data,ES,'Ts',0,'trace','on','Tolerance',1/1000);

disp('Use a "black" box approach')
model3 = pem(Data,8,'trace','on');


% ____________________________________________________________________________
% Now compare the results from the ID
% ____________________________________________________________________________

clc
disp('Nominal Plant then Estimated Plants')
GA, model1.a

% The following command compares the time domain fitted model against the measured output
% and the output from the estimated plant
figure; compare(Data,GG, model1,':', model3,'-.')

% Derive the state space matrix from estimated model
AAA=model1.a(1:4,1:4)
BBB=model1.b(1:4,:)
CCC=model1.c(:,1:4)
DDD=model1.d

% compare difference between measured plant and estimated plant
clc
AAA,AA
BBB,BB
CCC,CC
DDD,DD
% present results from free parameterization approach
r1a=model1.a(1:4,1:4)
r1b=model1.b(1:4,:)
r1c=model1.c(:,1:4)
r1d=model1.d



% present results from black box approach
r3a=model3.a(1:4,1:4)
r3b=model3.b(1:4,:)
r3c=model3.c(:,1:4)
r3d=model3.d% Compare difference between measured plant and estimated plant
```

B.4 ID Reduced observers

```
%*****************************************************************%
%*************       Project: Ball & beam system      ************%
%*************      Supervisor: Dr. Ben Cazzolato       ************%
%*************              Wei Wang                    ************%
%************           The University of Adelaide        ************%
%*********           School of Mechanical Engineering         **********%
%*****************************************************************%
clear all
close all
clc


%---- system parameters
Rb= 0.01                                        % ball's radius (m)
m_ball=4/3*pi*(Rb)^3*7800                       % mass of the ball (Kg)
m_beam= 0.381                                   % mass of the beam (Kg)
L_beam= 0.7                                     % length of the beam (m)
R=    4.7                                       % Dc motor armuature resistance (ohm)
K=    4.91                                      % Motor torque constant (Nm/A)
Ke=0.5/(2*pi/60)                % Motor back electromotive-force constant (Volt/(rad/s))
L= 1.6e-3                                       % Motor electrical inductance (H)
J_beam1= 1/12*(m_beam)*L_beam^2                 % moment of inertia of the beam (Kg.m^2)
J_beam=0.019                    % moment of inertia of the beam by measurement (Kg.m^2)
J_motor= 43e-3                                  % moment of inertia of the motor (Kg.m^2)
J_bm=J_beam+J_motor             % moment of inertia of the beam & motor (Kg.m^2)
b=1.6e-1/(2*pi/60)                              % DC motor damping constant    (Nm/(rad/s))
g=9.81                                          % Acceleration of gravity (m/s^2)
a=0.005                         % length from the center of the ball to the beam surface (m)
n=100                                           % gearbox reduction ratio


% Simplied state space model of the system (have been linearized)
AA=[0 0 1 0;
      0 0 0 1 ;
     0 g/(1+2/5*(Rb/a)^2) 0 0;
     -m_ball*g/J_bm 0 0 -(b+K^2/R)/J_bm];
BB=[0;0;0;K/(R*J_bm)];
CC=[1 0 0 0; 0 1 0    0];
DD=0

CC1=[1 0]
```

```
GG=ss(AA,BB,CC,DD,'statename',{'position' 'ball velo' 'beam angle' 'beam rational velo'},...
'inputname',{'voltage'},'outputname',{'position','beam angle'},'notes','Created 5/8/2006' )

% Build the reduced order system
Aaa=AA(1:2,1:2)
Abb=AA(3:4,3:4)
Aab=AA(1:2,3:4)
Aba=AA(3:4,1:2)

Ba=BB(1:2), Bb=BB(3:4)


% Design based on pole placement
% Assign closed loop poles
damp_c= 1
wn=5
[num,den]=ord2(wn,damp_c)
r_c=roots(den)
pole_c=[r_c(1),r_c(2),-5,-8]

% Calculate the feedback gain matrix via Ackermann's formula
KKK=acker(AA,BB, pole_c)
Ka=[KKK(1),KKK(2)]
Kb=[KKK(3),KKK(4)]

% Calculate the estimate gain matrix via Ackermann's formula
feedback_poles =pole_c(3:4)
gain =30*feedback_poles
L=place(Abb',Aab',gain)'


% Build structure of the estimate model
Ar=Abb-L*Aab-(Bb-L*Ba)*Kb
Br=Ar*L+Aba-L*Aaa-(Bb-L*Ba)*Ka
Cr=-Kb
Dr=-Ka-Kb*L

GA=[Aaa+Ba*Dr Aab Ba*Cr; Aba+Bb*Dr Abb Bb*Cr ; Br zeros(2,2) Ar ]
GB=[Ba;Bb;0;0]
GC=[ eye(2,2) zeros(2,4)]
GD=[0;0]


% Build the structure model
```

```
A=GA
A(find(((A~=1))&((A~=0))))=(find(((A~=1))&((A~=0))))*NaN;
B=GB
B(find(((B~=1))&((B~=0))))=(find(((B~=1))&((B~=0))))*NaN;
C=GC
C(find(((C~=1))&((C~=0))))=(find(((C~=1))&((C~=0))))*NaN;
D=GD

% Define the model which will be used in estimation
ES=idss(GA,GB,GC,GD,'Ts','0')

% Define structure matrics
ES.as=A;
ES.bs=B
ES.cs=ES.c
ES.ds=ES.d

ES.DisturbanceModel = 'Estimate';

% Simulate the 'ID_reduced.mdl' to create Data
U=input;
Y1=output;
Y=Y1(1:length(U),:);
T=[0:0.005:10];


length(U),length(Y),length(T)

% Look at the data
figure
subplot(2,1,1)
title('Input and Output Data')
plot(T,U)
ylabel('Input Data (Volts)')
xlabel('Time (s)')
subplot(2,1,2)
[AX,H1,H2] = plotyy(T,Y(:,1),T,Y(:,2),'plot')
set(get(AX(1),'Ylabel'),'String','Position of the ball (m)')
set(get(AX(2),'Ylabel'),'String','Tilt angle of the beam (rad)')
title('Outputs of the system')
xlabel('Time (s)')
set(H1,'LineStyle','--')
set(H2,'LineStyle',':')
```

```
% Check the quality of the data (calculate coherence between input and output)
[dummy, number_of_inputs] = size(GG.b)
[number_of_outputs,dummy] = size(GG.c)
NFFT = 1024;
COHERENCE = zeros(NFFT/2+1,number_of_outputs);

fs=1/0.005
F=200/1024*[0:1024/2]
figure
for eachoutput = 1:number_of_outputs
   subplot(number_of_outputs,1,eachoutput)
   for eachinput = 1:number_of_inputs
        % Determine the coherence
        COHERENCE(:,eachoutput)                    =                    COHERENCE(:,eachoutput)+
mscohere(U(:,eachinput),Y(:,eachoutput),hanning(NFFT),0.75*NFFT,NFFT,fs);
   end

   plot(F,COHERENCE(:,eachoutput))
   if eachoutput < 2
   title ('Coherence between input and position of the ball')
   xlabel('Frequency (Hz)')
   else
   title('Coherence between input and tilt angle of the beam')
   xlabel('Frequency (Hz)')
   end
end



% define the dataset
Data = iddata(Y,U,1/fs)
set(Data,'InputName',{'Voltage'},'OutputName',{'Position','Tilt angle'})
figure; plot(Data)



% Improve the date quality by filter or smooth methods
gs = spa(Data);              % Smoothed spectral estimate
figure; bode(gs)
refresh

%    Final ID
disp('Use a "Free parameterisation" approach where we know the structure of the model')
model1 = pem(Data,ES,'Ts',0,'trace','on','Tolerance',1/1000);

disp('Use a "black" box approach')
```

```
model3 = pem(Data,6,'trace','on');

% ____________________________________________________________________
% Now compare the results from the ID
% ____________________________________________________________________

clc
disp('Nominal Plant then Estimated Plants')
GA, model1.a

% The following command compares the time domain fitted model against the measured output
% and the output from the estimated plant
figure; compare(Data,GG, model1,':', model3,'-.')

% Take the state space matrix from estimate model
AAaa=model1.a(1:2,1:2)-model1.b(1:2)*Dr
AAba=model1.a(3:4,1:2)-model1.b(3:4)*Dr
AAab=model1.a(1:2,3:4)
AAbb=model1.a(3:4,3:4)

clc
AAA_nominate=[Aaa, Aab; Aba, Abb]
AAA_estimate=[AAaa, AAab; AAba, AAbb]

BBB_nominate=[Ba; Bb]
BBB_estimate=model1.b(1:4)

% present results from free parameterization approach
r1aa=model1.a(1:2,1:2)-model1.b(1:2)*Dr
r1ba=model1.a(3:4,1:2)-model1.b(3:4)*Dr
r1ab=model1.a(1:2,3:4)
r1bb=model1.a(3:4,3:4)

r1Ba=model1.b(1:2)
r1Bb=model1.b(3:4)

r1C=model1.c(:,1:4)
r1D =model1.d



% present results from black box approach
r3aa=model3.a(1:2,1:2)-model3.b(1:2)*Dr
r3ba=model3.a(3:4,1:2)-model3.b(3:4)*Dr
```

```
r3ab=model3.a(1:2,3:4)
r3bb=model3.a(3:4,3:4)

r3Ba=model3.b(1:2)
r3Bb=model3.b(3:4)

r3C=model3.c(:,1:4)
r3D =model3.d
```

Appendix C

DC motor (RH-11D-3001-E100-AO)

Technical data

C.1 DC motor (RH-11D-3001)

Table C.1: Technical data of the RH-11D-3001-E100-AO (Harmonic drive 2006)

Item	Value	Unit
Rated output power	12.3	W
Related voltage	24	V
Rated current	1.3	A
Rated Output Torque	3.9	Nm
Rated Output Speed	30	rpm
Max. Continuous Stall Torque	4.4	Nm
Peak Current	2.1	A
Maximum Output Torque	7.8	Nm

Maximum Output Speed	50	rpm
Torque Constant	4.91	Nm/A
Voltage Constant (B.E.M.F.)	0.50	V/rpm
Inertia at Output Shaft	43	$Kg\,m^2 \times 10^3$
Mechanical Time Constant	8.5	msec
Torque-Speed Gradient	0.52	Nm/rpm
Viscous Damping Constant	0.017	Nm/rpm
Rated Power Rate	0.36	kW/sec
Thermal Time Constant	10	min
Thermal Resistance	3.3	$^\circ C/W$
Gear Ratio	100	1:R
Maximum Radial Load	245	N
Maximum Axial Load	196	N
Motor Rated Output	20	W
Motor Rated Speed	3000	rpm
Armature Resistance	4.7	ohm
Armature Inductance	1.6	mH
Electrical Time Constant	0.34	ms
Starting Current	0.31	A
No-Load Running Current	0.55	A
Actuator Accuracy	2	arc-min
Actuator Repeatability	± 60	arc-sec

C.2 Digital encoder (E100) for the DC motor

Table C.2: Technical data of digital encoder (Harmonic drive 2006)

Type	Value	Unit
Output Circuit	Open collector (see Figure C.1)	
Power Supply	+4.75 - +12.6	VDC
	60 max	mA
Output Voltage	0.5 max	V
Max. Response Frequency	125	kHz
Resolution	1000	P/rev
Output Signal	A,B,Z (see Figure C.1)	
Max. Voltage	36	VDC
Max. Current	20	A
Moment of Inertia	3×10^{-8}	$kg\,m^2$
Lead Wire	Ø 4 x 600L Ø 0.12/7 Strand	mm

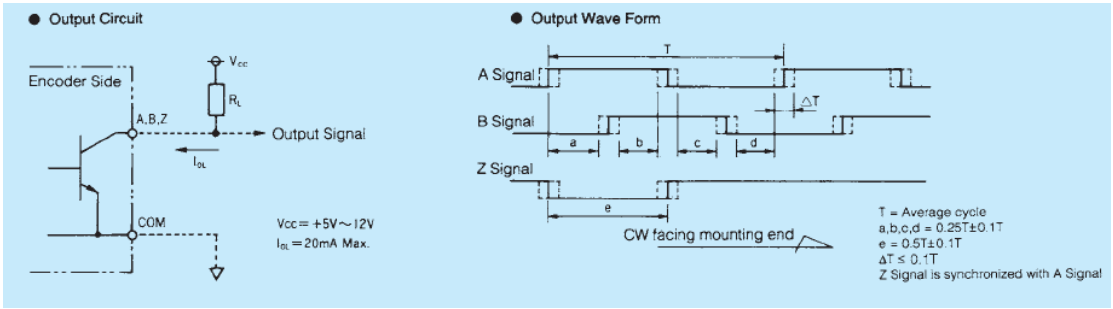

Figure C.1: Output circuit of digital encoder (Open Collector DO) (Harmonic drive 2006)

Color	Line Driver	Open Collector
Brown	A-Signal	A-Signal Output
Blue	$\overline{A}$-Signal	A-Signal Common
Red	B-Signal	B-Signal Output
Green	$\overline{B}$-Signal	B-Signal Common
Yellow	Z-Signal	Z-Signal Output
Orange	$\overline{Z}$-Signal	Z-Signal Common
White	Power Supply	Power Supply
Black	Common	Common
Shield	Floating	Floating

Figure C.2: Encoder lead wire (Harmonic drive 2006)

C.3 Mechanical drawing of the DC motor with Encoder

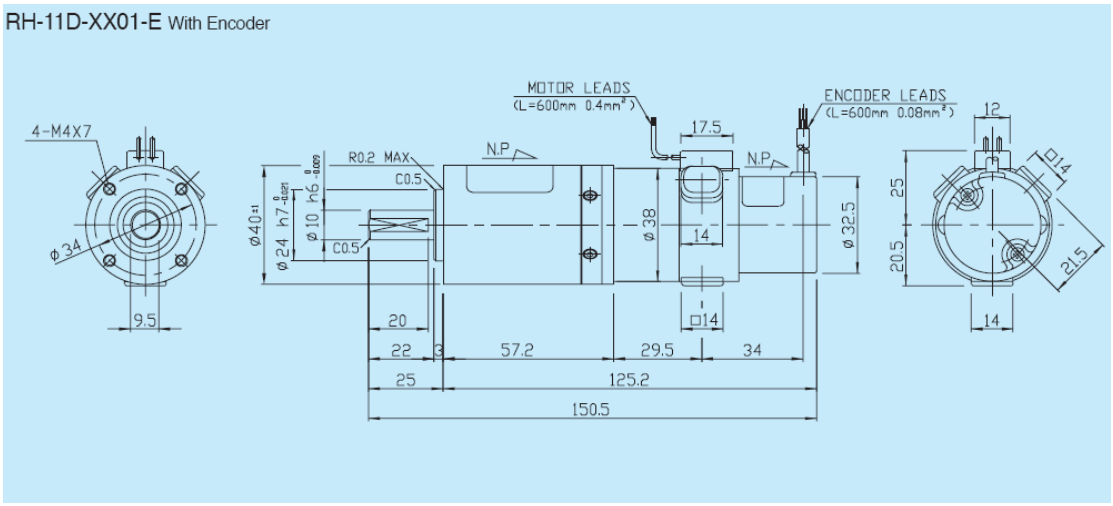

Figure C.3: Mechanical drawing of the DC motor (Harmonic drive 2006)

Appendix D

Mechanical drawings

D.1 Solid model of the entire ball and beam system

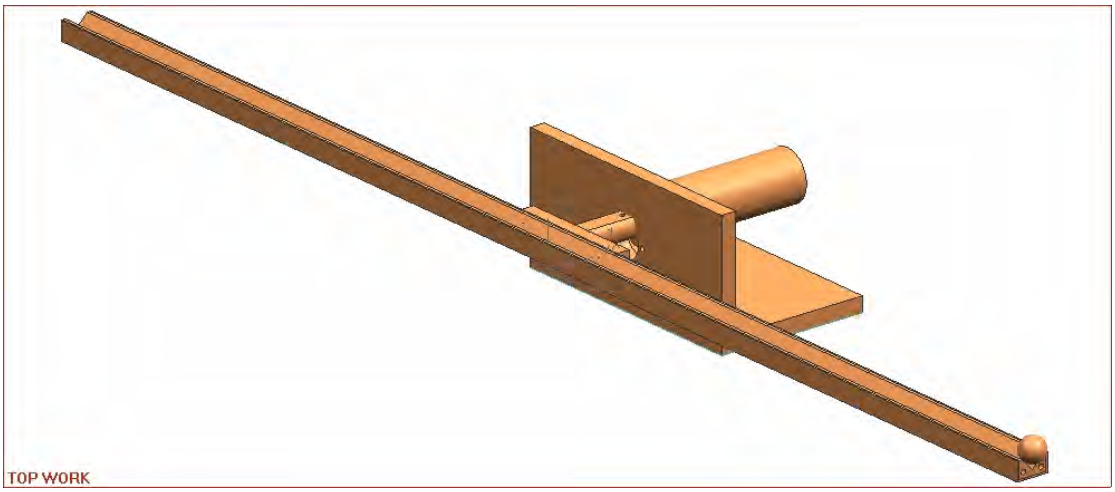

Figure D.1: Overview of the ball and beam system (Solid Edge V19 2006)

D.2 Base

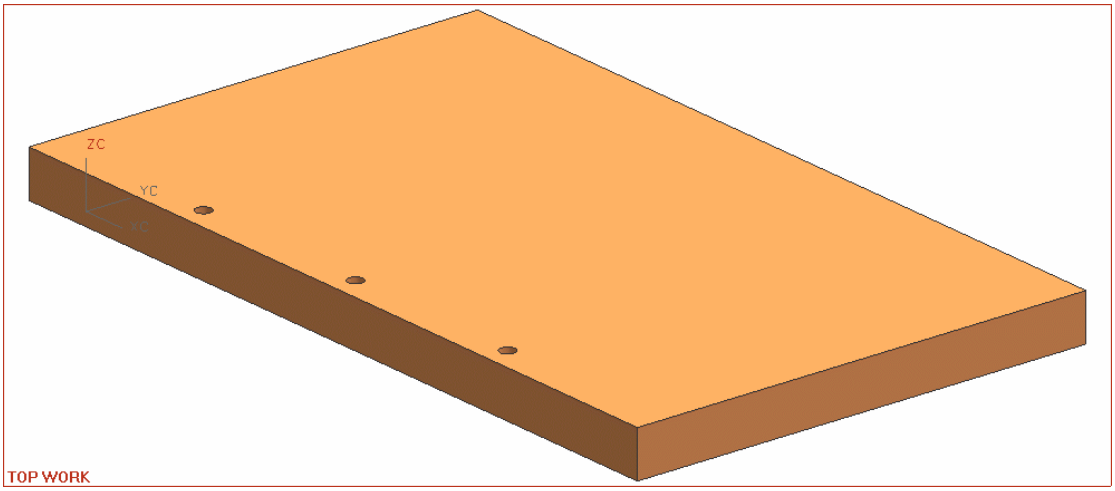

Figure D.2: Overview of the base of the ball and beam system (Solid Edge V19 2006)

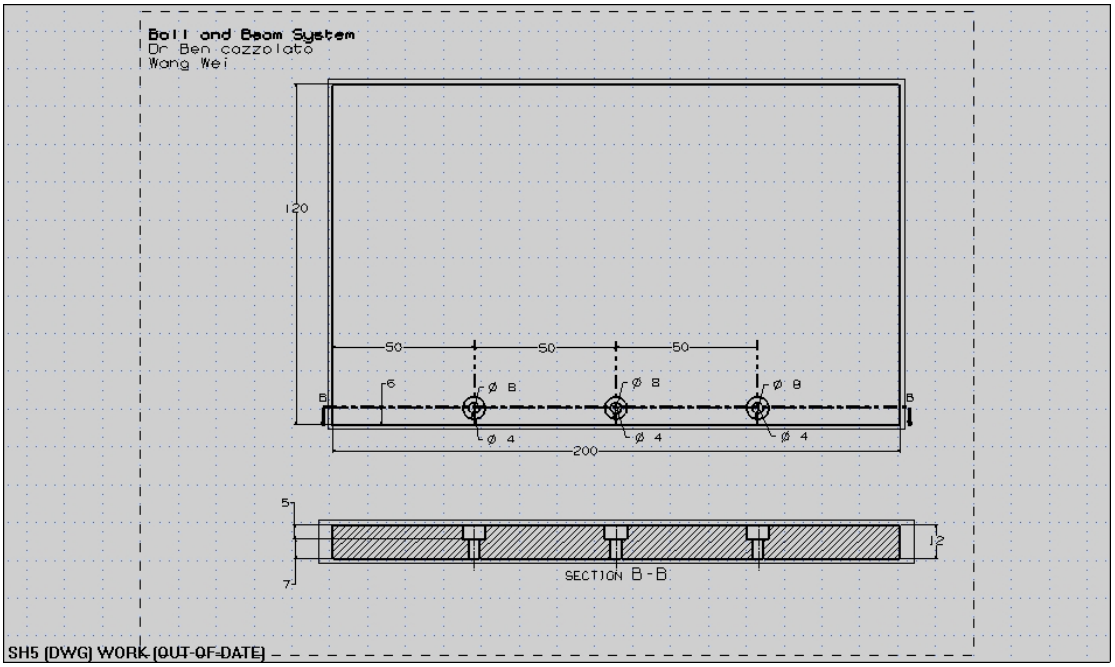

Figure D.3: Base drawing in (mm) (Solid Edge V19 2006)

D.3 Support

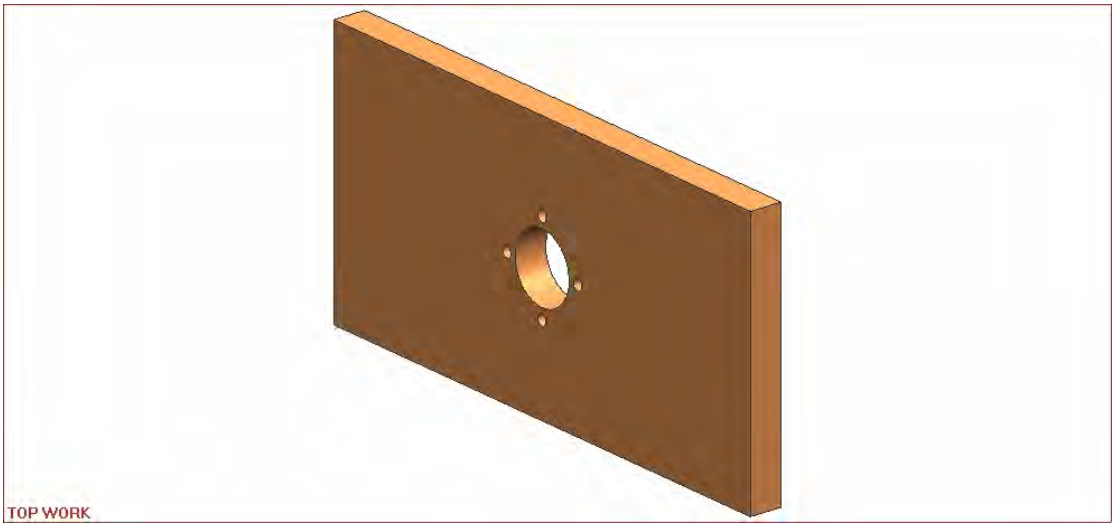

Figure D.4: Overview of the support of the ball and beam system (Solid Edge V19 2006)

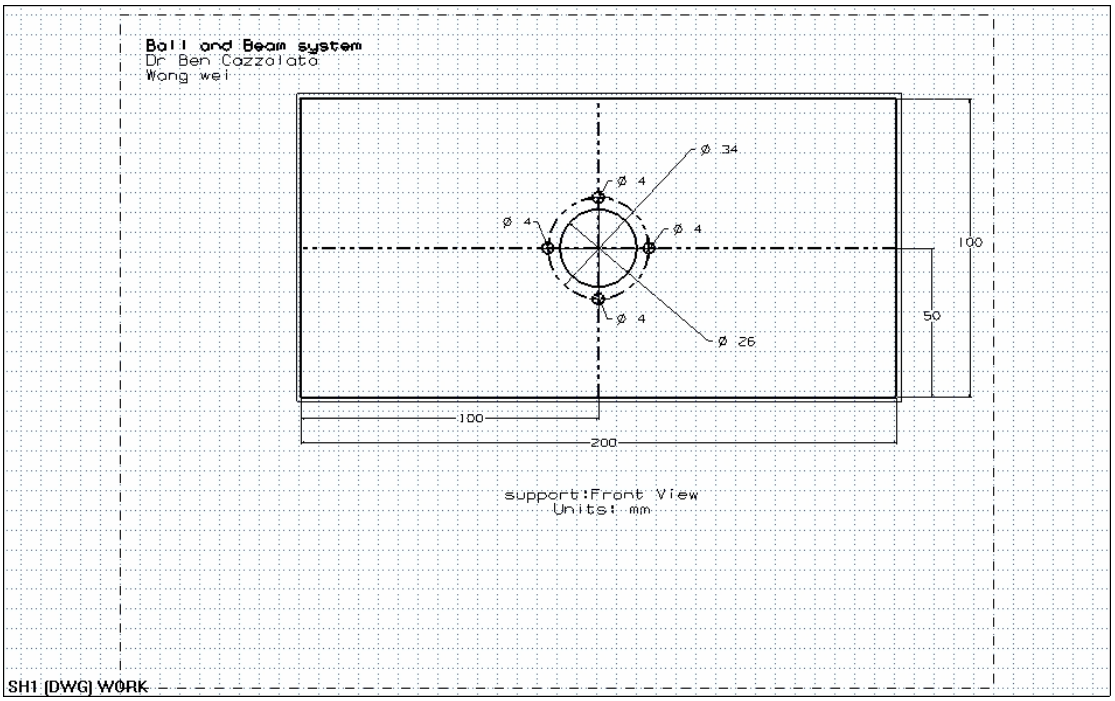

Figure D.5: Support drawing (Front view) in (mm) (Solid Edge V19 2006)

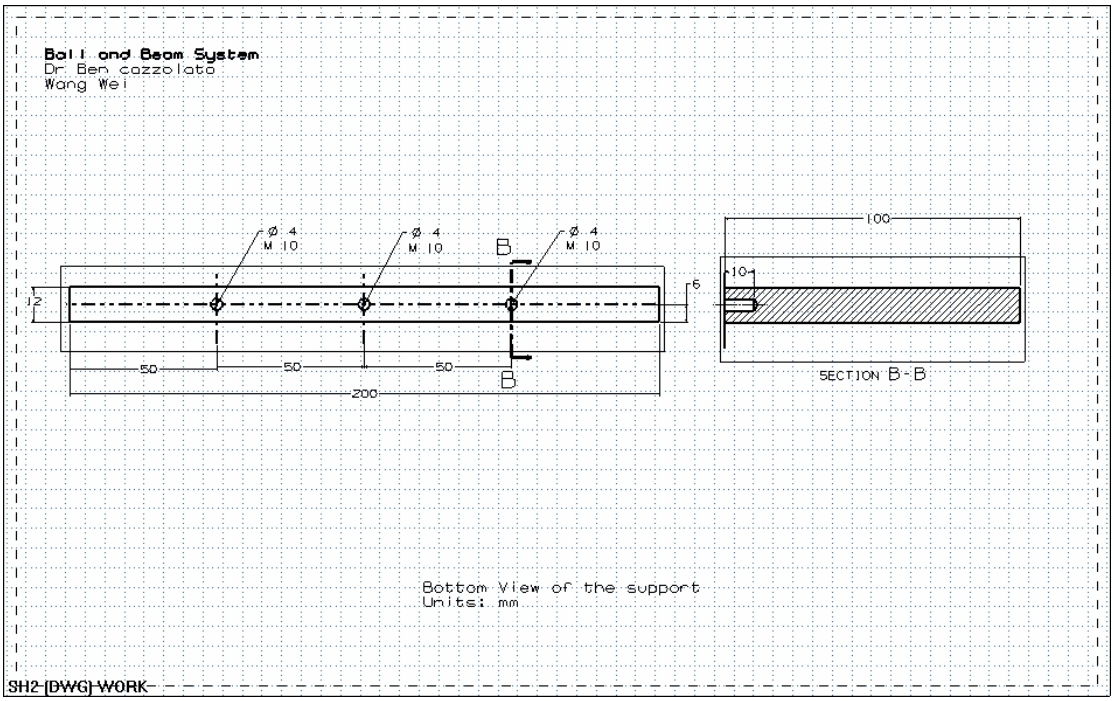

Figure D.6: Support drawing (Bottom view) in (mm) (Solid Edge V19 2006)

D.4 Link

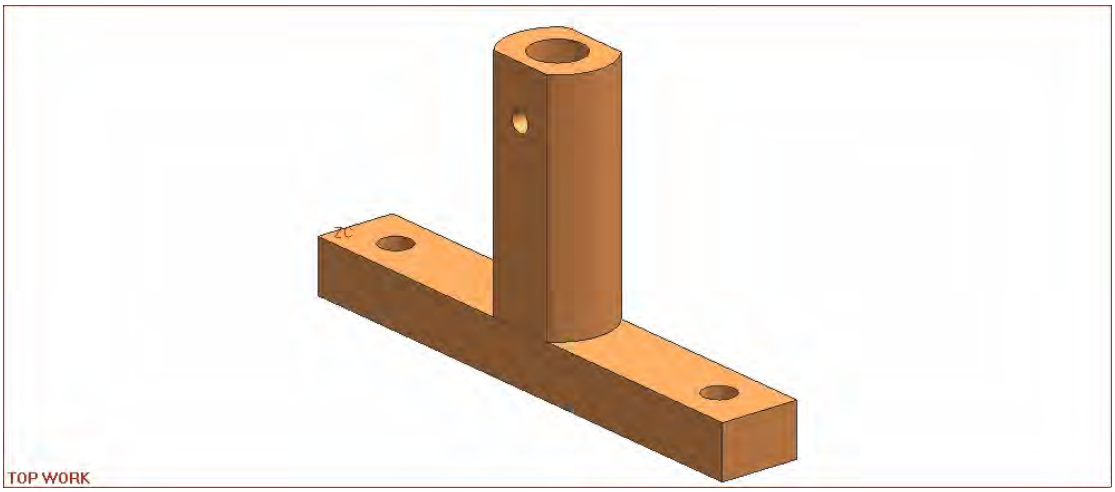

Figure D.7: Overview of the link of the ball and beam system (Solid Edge V19 2006)

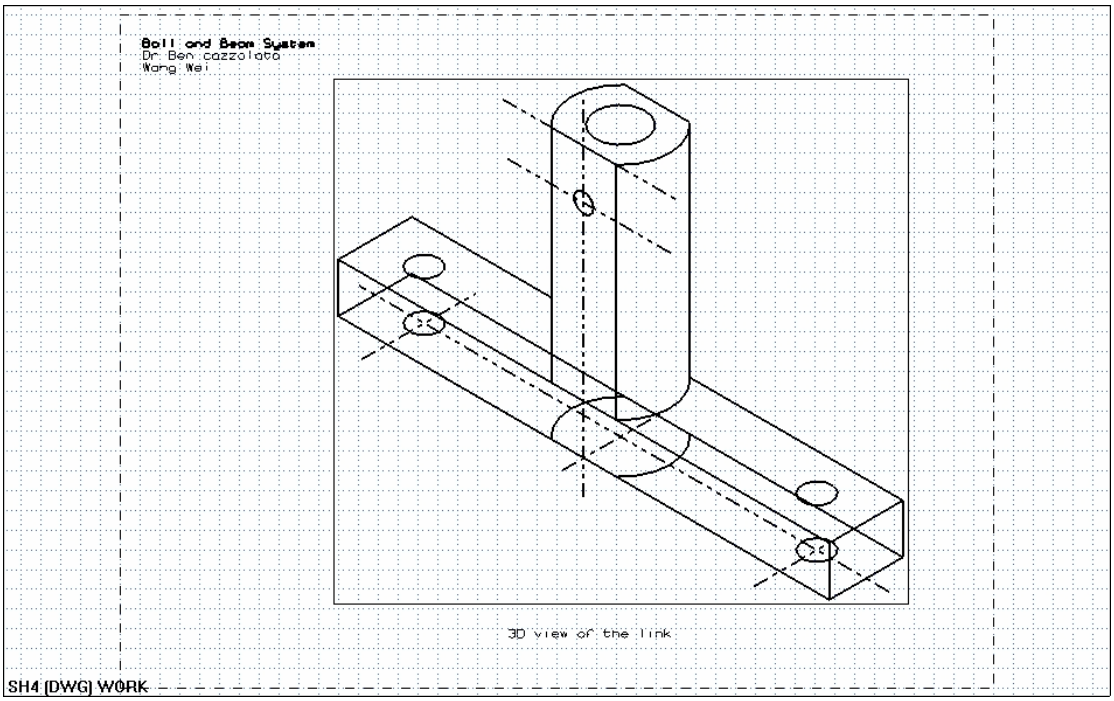

Figure D.8: Transparent Overview of the connect of the ball and beam system

(Solid Edge V19 2006)

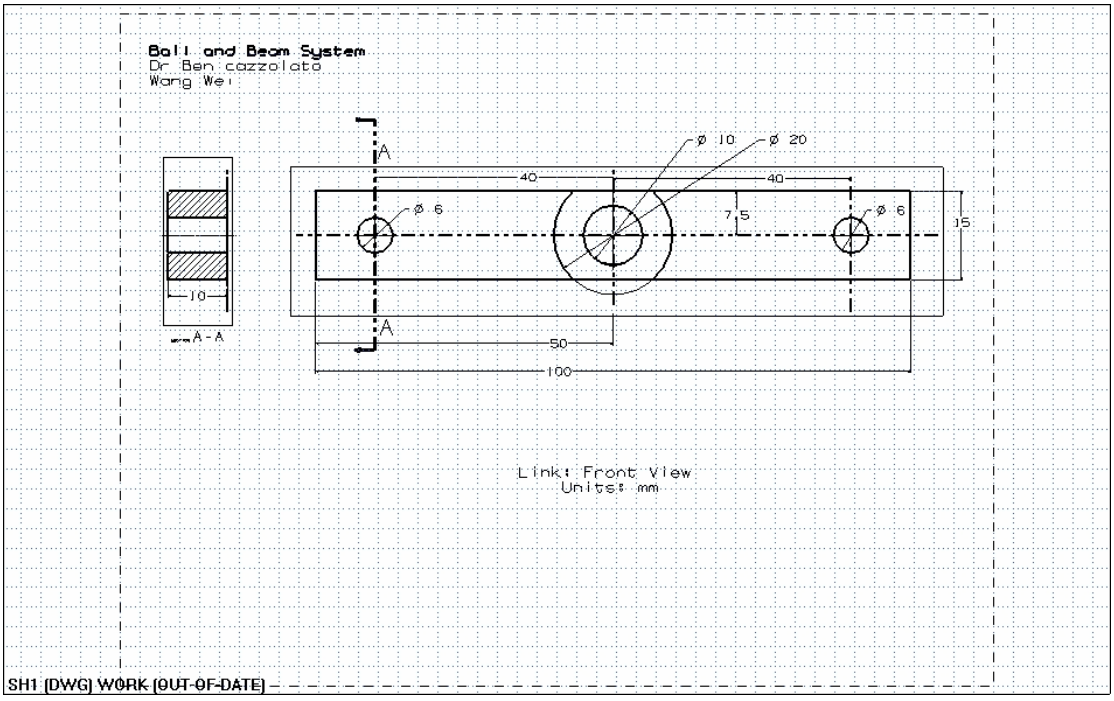

Figure D.9: Connect drawing (Front view) in (mm) (Solid Edge V19 2006)

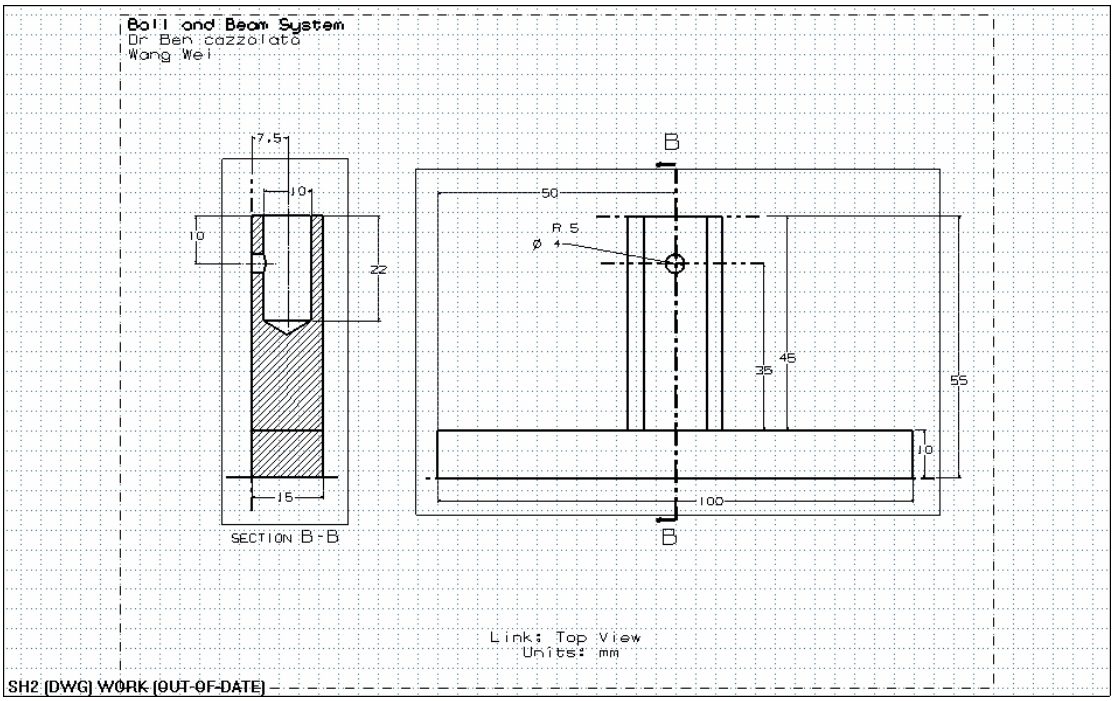

Figure D.10: Connect drawing (Top view) in (mm) (Solid Edge V19 2006)

D.5 Beam

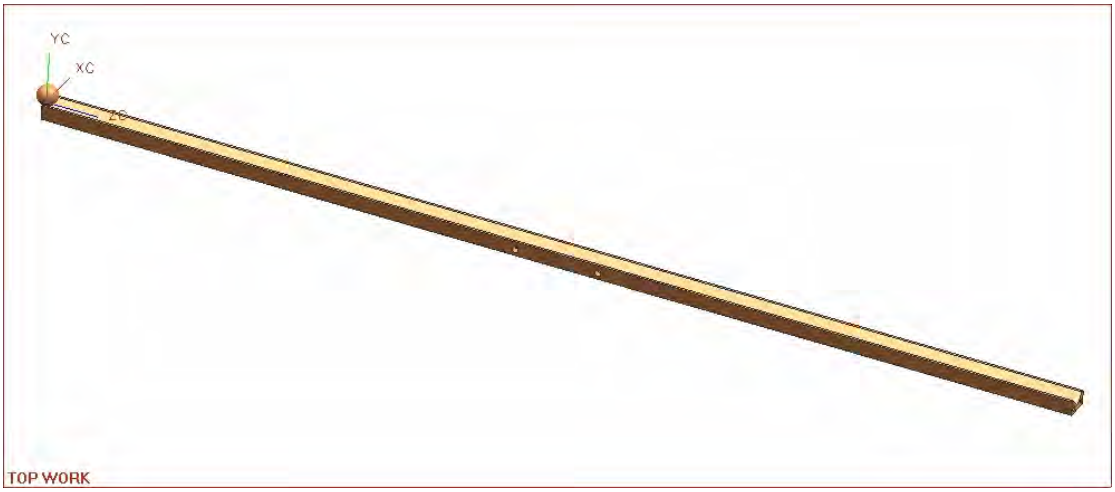

Figure D.11: Overview of the beam and ball of the ball and beam system (Solid Edge V19 2006)

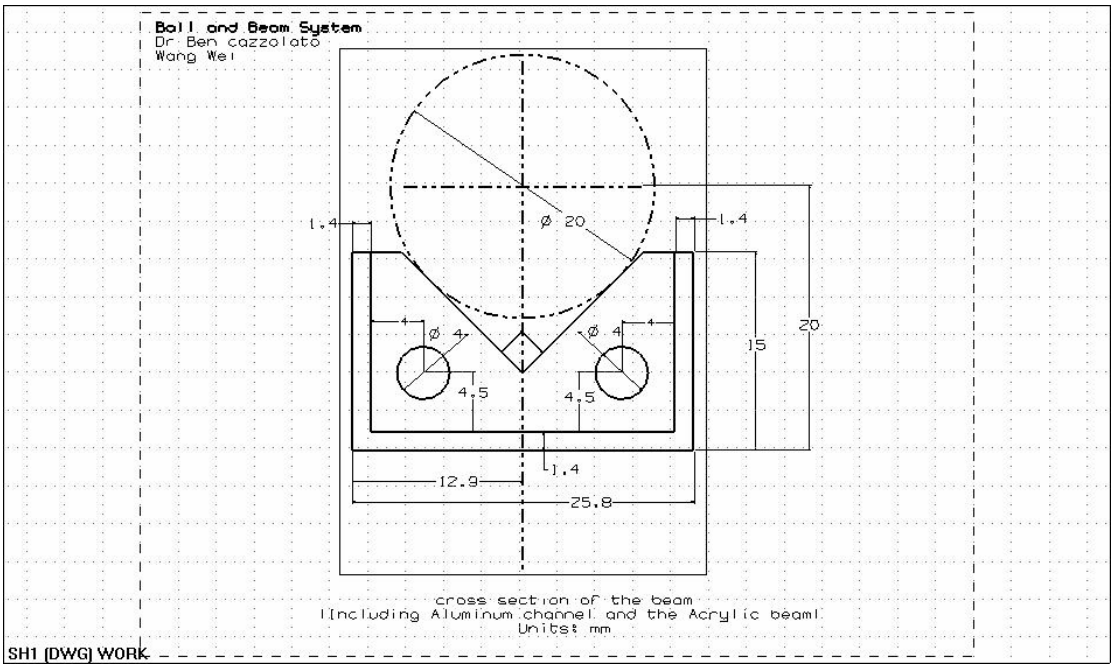

Figure D.12: Beam drawing (side view) in (mm) (Solid Edge V19 2006)

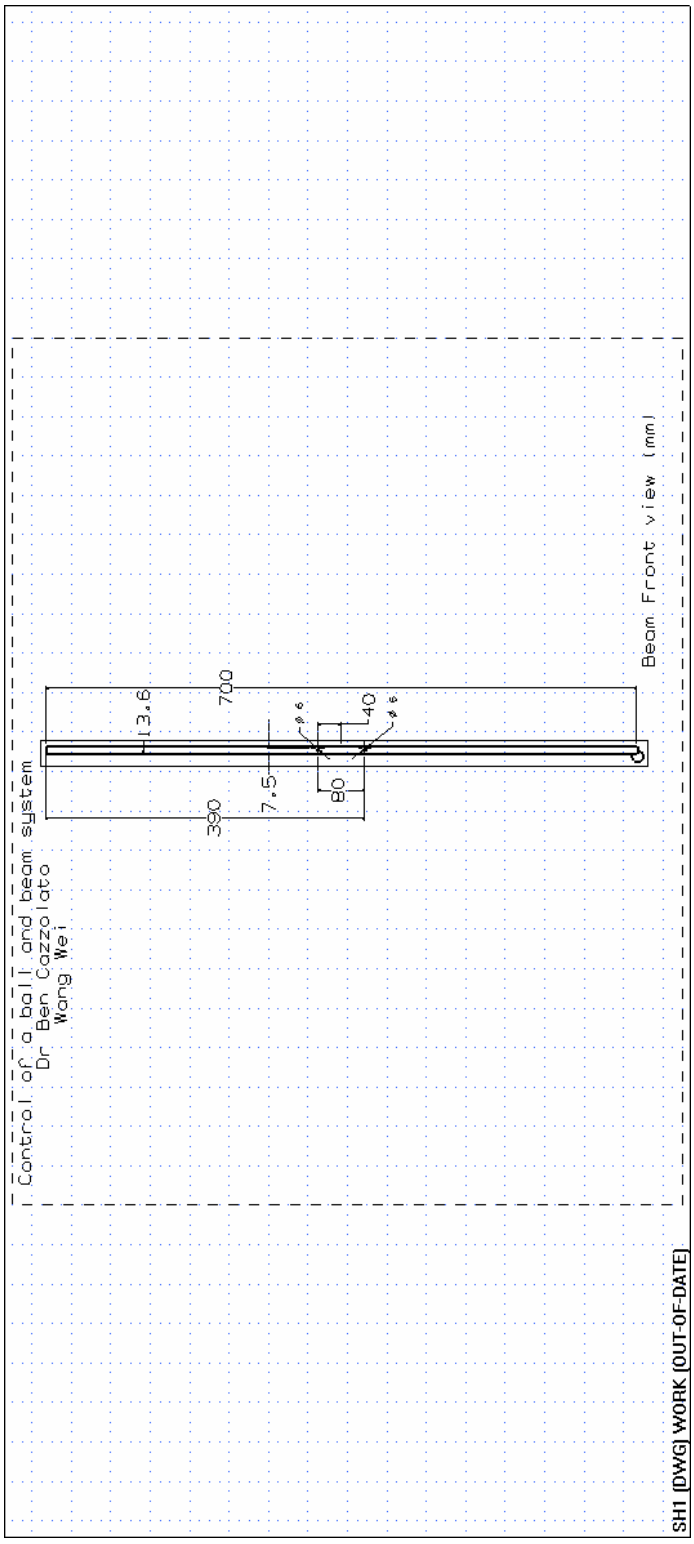

Figure D.13: Beam drawing (side view) in (mm) (Solid Edge V19 2006)

Appendix E

Circuit Diagrams

Circuit Diagrams for the Resistive Wire Position Sensor

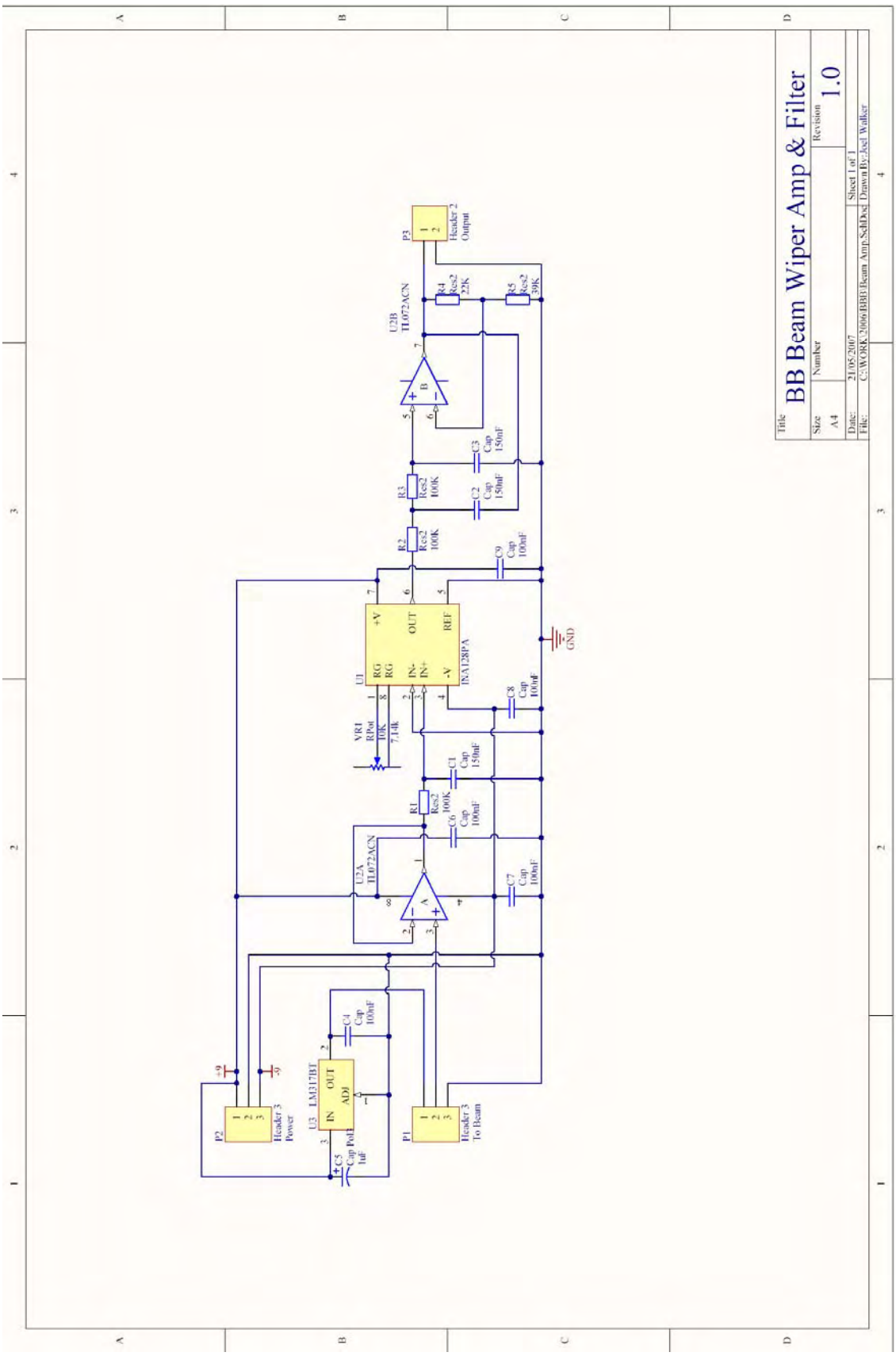

Figure E.1: Amplifier and filter circuit diagram for the resistive wire position sensor (Made by Joel Walker 2007)

Appendix F

The ball and beam system Manual

F.1 Introduction

Control of a ball and beam system consists of the hardware equipment, such as the mechanical system, the power supplier, sensors, the DC motor, and data acquisition and control board, and some software tools, for example, Matlab, and Wincon software (Quanser 2006). This manual introduces the hardware equipment and software tools to be used in the control of a ball and beam system.

F.2 Hardware equipments

The hardware equipment including the DC motor, sensors, power amplifier, and data acquisition board are introduced.

F.2.1 DC motor

The DC motor used for the ball and beam system is RH-11D-3001-E100-AO (Harmonic drive 2006). The technical data of the DC motor can be found in Appendix C. The overview of the motor is shown in Figure 2.6. The DC motor has a gearbox with reduction ratio 100:1. The maximum voltage input is 24V.

Make sure that the voltage input is less than maximum value 24V to protect the motor from burning.

F.2.2 Sensors

There are two sensors used in the ball and beam system, the resistive wire position sensor and the digital encoder. The calibration of these sensors is discussed in Section 6.1. The following calibration factor (Table F.1) for the digital encoder is needed in order to use the sensor in units required.

Table F.1: Calibration factor for the digital encoder

	Conversion (rad)	Conversion (degree)
Encoder	$\frac{2\times\pi}{1000\times100\times4}$	$\frac{2\times180}{1000\times100\times4}$

F.2.3 Power Amplifier

The universal power module (UPM-2405) (Quanser 2006) is a linear power operational amplifier. The MutiQ-PCI data acquisition card cannot suppling enough

power for the DC motor used in the ball and beam system, hence the output control signal from the acquisition card needs to be amplified. The technical date of UMP-2405 is shown in Table F.2. The DC motor used in the ball and beam system has the maximum voltage input 24V with the maximum 12.3W power requirement. Obviously, UPM with the maximum power output 24V with maximum current 5A can meet the DC motors requirements.

Table F.2: UPM 2405 technical data (Quanser 2006)

Model	Maximum output voltage	Maximum continuous Current	Output	Type	Number of outputs
UPM 2405	24 V	5 A	linear	voltage	1

Figure F.1: UPM-2405 (Quanser 2006)

Figure F.2: Top view of UPM-2405 (Quanser 2006)

The UPM-2405 for the ball and beam system is shown in Figure F.1. The top view of the module is shown in Figure F.2.

F.2.4 Data acquisition boards (Quanser Q4 board)

The data acquisition board Q4 (Quanser 2006) consists of two parts: the MultiQ-PCI

data acquisition card and terminal board. The MultiQ-PCI data acquisition card is installed in the PCI slot inside the computer. The terminal board, shown in Figure F.3 interfaces the MultiQ-PCI data acquisition card with the sensors and the actuators, and power module. The Quanser Q4 board is used for all I/O:

- Convert analog signals, such as the position of the ball into digital signal for the computer
- Convert the digital control signal, for example the voltage input of the DC motor, from the computer into the analog signal, and then amplified by the universal power module (UPM) for the DC motor.

The key features of the Q4 board are presented in Table F.3 as follows:

Table F.3: Key features of the Quanser Q4 board (Quanser 2006)

Number	High Resolution - 14-bit inputs
1	Simultaneous sampling of A/D & encoder inputs
2	4 x 14-bit analog inputs
3	4x 12-bit D/A voltage outputs
4	4 quadrature encoder inputs
5	16 programmable digital I/O channels
6	Simultaneous sampling of both analog and encoder sections
7	2x 32-bit dedicated counter/timers
8	4x 24-bit reconfigurable encoder counter/timers
9	2x on-board PWM outputs

10	32-bit, 33 MHz PCI bus interface
11	Supports Quanser real-time control software WinCon (2000/XP)
12	Totem Pole digital I/O for high speed
13	Easy synchronization of multiple Q4 boards

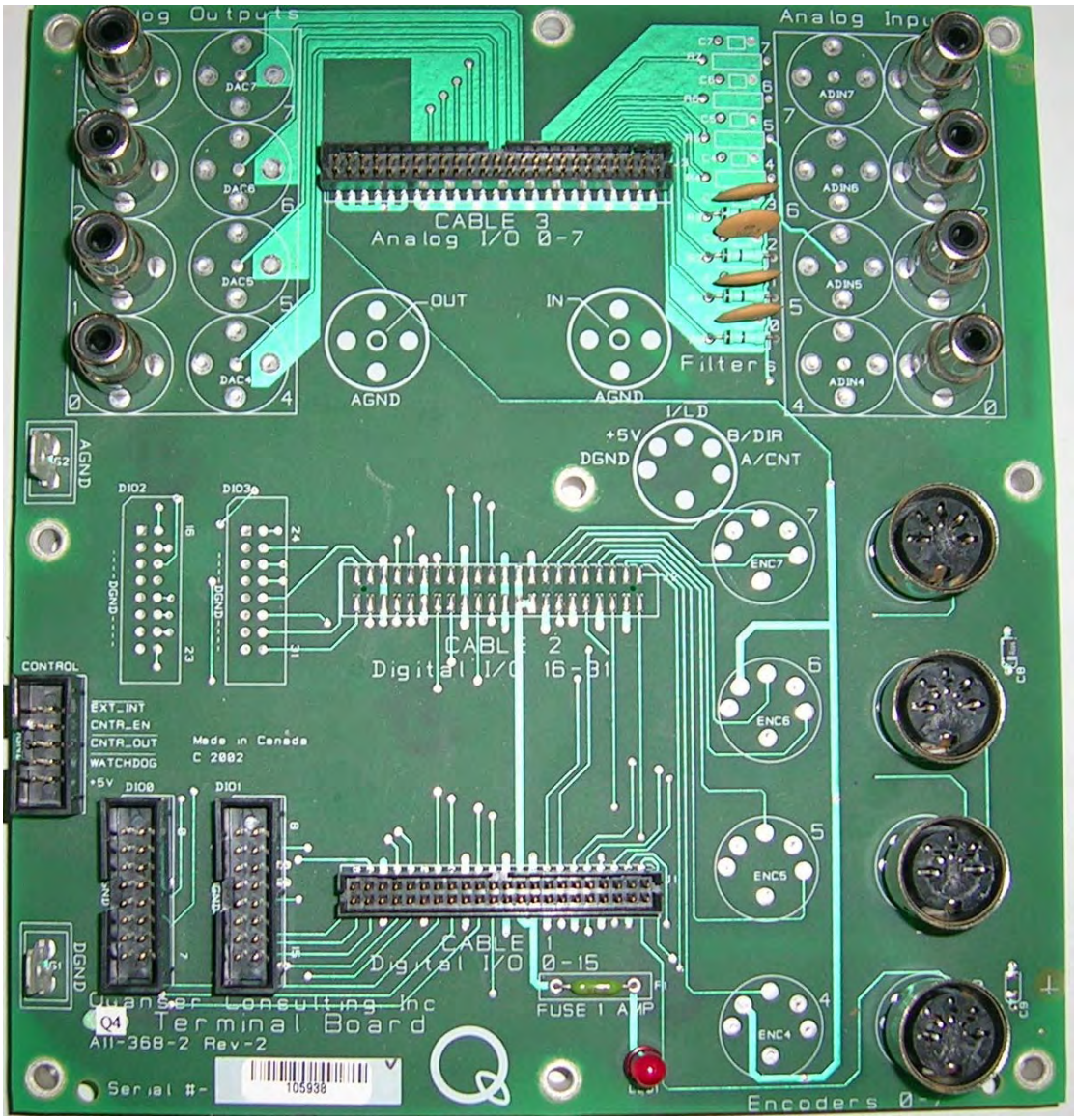

Figure F.3: Quanser Q4 Terminal Board

F.3 Software tools

The software tools involved in the ball and beam system are:

Matlab7.01/ Simulink & System Identification Toolbox

Wincon5.0 (Wincon5.0 2006)

Visual C++6.0 (Visual C++6.0 2006)

F.3.1 Simulink

Simulink provides an environment for modeling physical systems and controllers as block diagrams in Matlab. The simulation parameters can be viewed and edited by selecting Configuration Parameters from the Simulation menu as shown in Figure F.4.

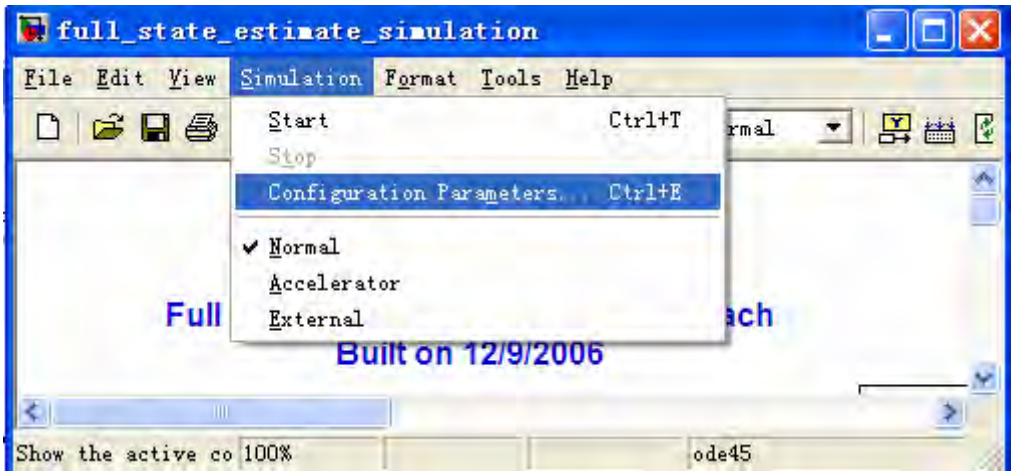

Figure F.4: Select configuration parameters from simulation menu

The simulation parameters for the simulation in Chapter 4 and Chapter 5 are shown in Figure F.5. The solver method is using ode45 (Dormand-Prince) since some blocks for ball and beam system are selected from the 'continuous' toolbox. The start

and stop time are selected as default 0 and 20 seconds. The Sampling time for the simulation is 0.005 second.

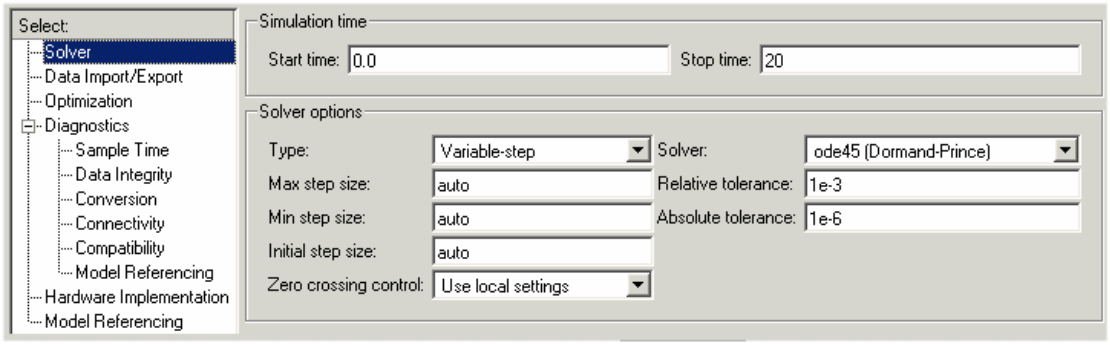

Figure F.5: Simulation parameters for the simulation of the ball and beam system

The simulation parameters for the experimentation in Chapter 4 and Chapter 5 are shown in Figure F.6. The start time is 0, and stop time is infinite for real time application. The type of solver is Fixed step. The method of solver is ode1 (Euler) with a sample time 0.001 second.

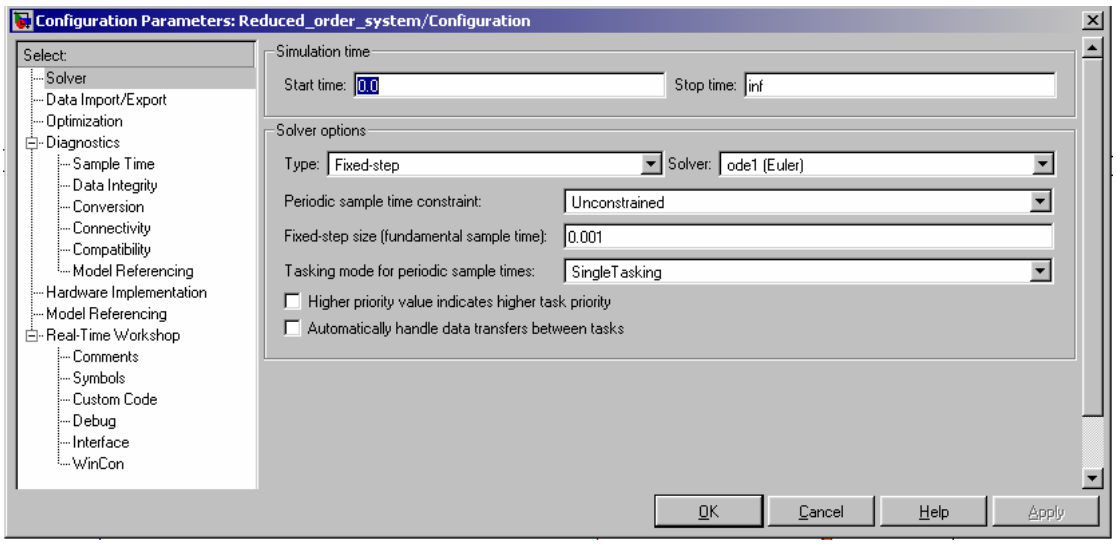

Figure F.6: Simulation parameters for the experiment of the ball and beam system

The input and output channels of Quanser Q4 board (data acquisition board) are modeled in Simulink, and can be selected from the Simulink library, given that Wincon installed properly. Figure F.7 shows the Simulink blocks (dashed box) of the channels of the Quanser Q4 board.

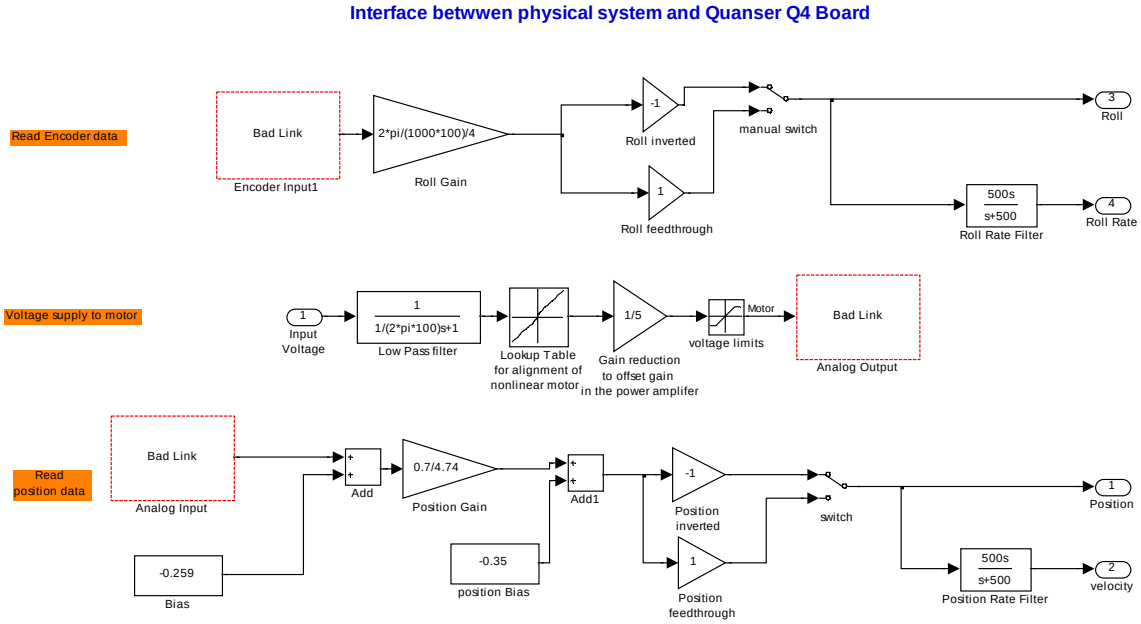

Figure F.7: The interface between the Quanser Q4 board (Quanser 2006) and Simulink of the ball and beam system

F.3.2 Wincon

Wincon presented by Quanser Consulting (Quanser 2006), is *'a real-application that run Simulink generated code using RTX Runtime on a PC under running Windows'* (University of Alberta 2006). Wincon4.1 interfaces Matlab/Simulink with the Quanser Q4 board, and uses the Simulink model of the ball and beam system to generate, compile using Visual C++, and run the program to control Quanser Q4

board. Wincon supports two configurations: the local configuration and the remote configuration. The ball and beam system uses the local configuration, shown in Figure F.8, in which *'Wincon Client, executing the real-time code, runs on the same machine and at the same time as Wincon server (i.e. the user-mode graphical interface)'* (Wincon4.1 2006). In the remote configuration, the client runs in a separate computer from the server.

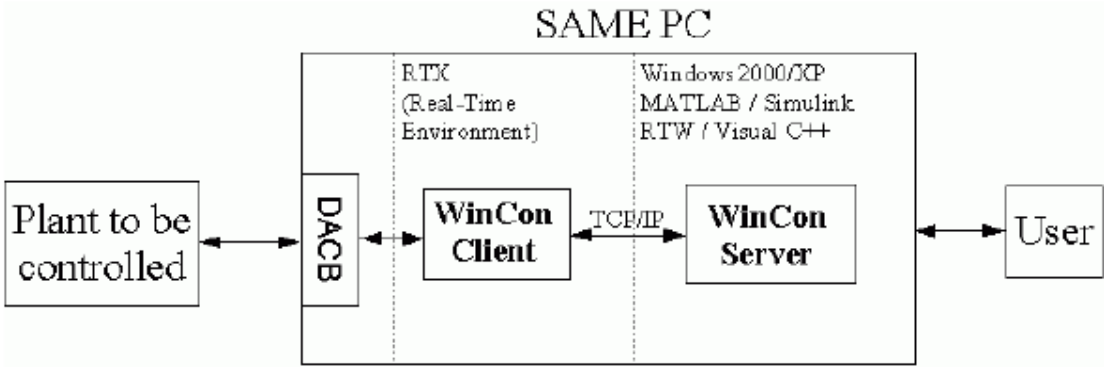

Figure F.8 Local configuration for the ball and beam system (Wincon4.1 2006)

If Wincon is installed properly, a Wincon menu item comes out in the Simulink Window, which allows users to generate and run the real-time code seamlessly. Two windows including Wincon Server, shown in Figure F.9, and Wincon Client shown in Figure F.10 will appear automatically after the constructed Simulink model is successfully built (University of Alberta 2006). Use the button 'Start/Stop' in Figure F.9 to manually start and stop a run-time application. The window of Wincon Client shown in Figure F.10 contains important information of the experiment such as sampling interval 0.001 second, sampling period, and system variables.

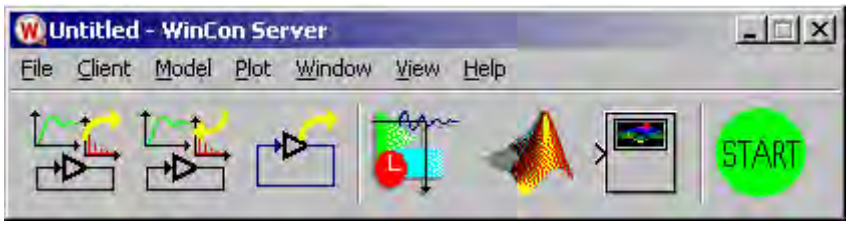

Figure F.9: Pop-up window (Wincon Server) for the real-time application

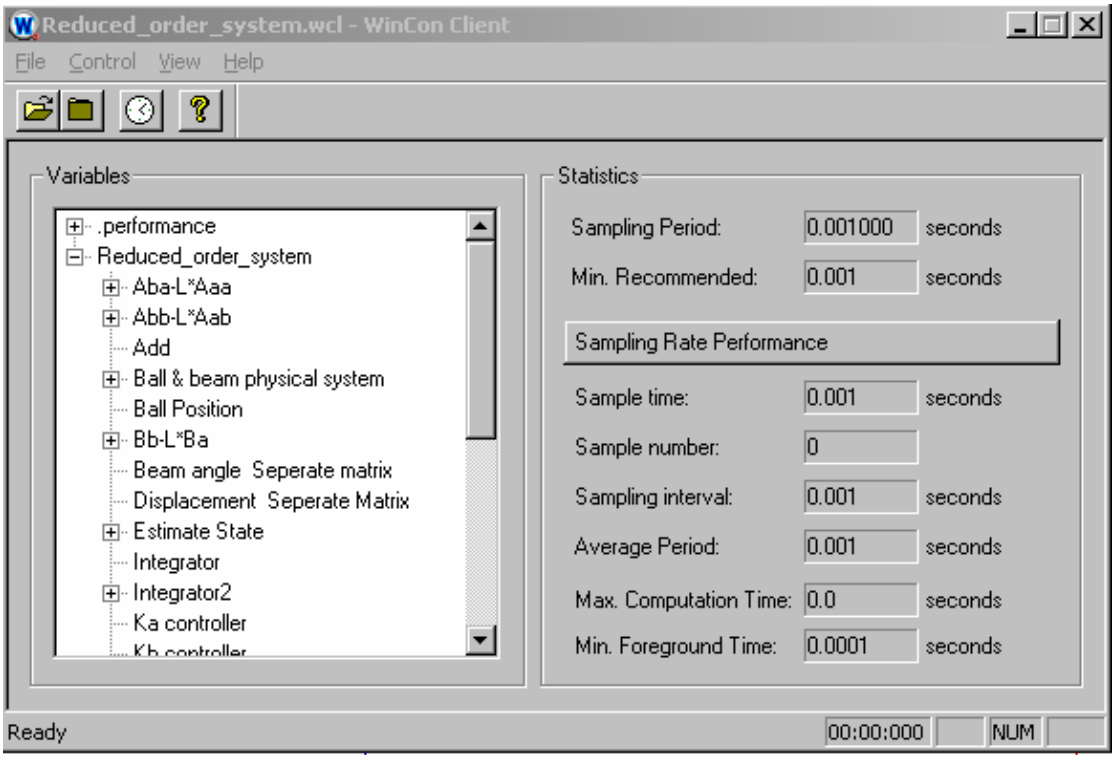

Figure F.10: Automatically appeared window (Wincon Client) for the real-time application

F.4 Operation of instrumentation

Place the system in the place where the beam can freely rotate against its center axis without bumping. Balance the ball on the beam manually. Check the contact condition between the ball and resistive wires. Please take care of the system.

Note: Rotating beam can injure body. Keep a distance with the system when operating

F.4.1 Connections

To operate the ball and beam system successfully, users need to connect the rig to the universal power module (UPM) and Quanser Q4 control board properly, shown in Figure F.11.

1. Connect the digital encoder of the DC motor with Channel 4 of Encoder inputs (Quanser Q4 terminal board).
2. Connect the resistive wire position sensor with Channel 1 of Analog inputs (Q4 terminal board).
3. Connect ‘Channel 1’ of ‘Analog outputs’ (Q4 terminal board) to the ‘From D/A’ of UPM 2405.
4. Connect ‘To Load’ of UPM 2405 to the DC motor.

Figure F.11: Connection of the ball and beam system

F.4.2 Caring for the system

2 Make sure that the ball does not run out of the beam channel.

3 The UPM 2405 used for the ball and beam system with an output channel labeled ‘Gain = 5’. Using this cable, the input voltage should be limited to 4.8 volts, since 4.8 multiplied 5 is 24 volts which is maximum voltage allowed into the DC motor.

4 Make sure the voltage input into the motor never exceeds 24 volts.

5 Put a low pass filter into voltage input channel to protect the DC motor from large current surges caused by the sudden step voltage changes.

6 Clean the resistive wires and ball with tissues or soft cloth

7 If the system is unstable, or hear unusual noise, quickly shut the system and find the reasons.

8 Turn off the universal power module once you finish the lab, and place the system on a stable place.

Appendix G

System identification results

G.1 Nominal plant

Nominal plant					
A_nominal					B_nominal
0.00	1.00	0.00	0.00		0
0.00	0.00	3.77	0.00		0
0.00	0.00	0.00	1.00		0
-5.17	0.00	0.00	-107.38		16.85
C_nominal					D_nominal
1	0	0	0		0
0	0	1	0		0

Nominal reduced order plant					
Aaa		**Aab**			**Ba**
0.00	0.00	1.00	0.00		0
0.00	0.00	0.00	1.00		0
Aba		**Abb**			**Bb**
0.00	3.77	0.00	0.00		0
-5.1696	0	0	-107.38		16.85
C					**D**
1	0	0	0		0
0	1	0	0		0

G.2 Simulation results

G.2.1 ID from Open-loop plant

Estimated plant from open loop system					
Free structured parameterizations approach					
A					B
0.00	1.00	0.00	0.00		0.00
0.00	0.00	3.70	0.00		0.00
0.00	0.00	0.00	1.00		0.00
-7.29	0.00	0.00	-155.48		24.26
C					D
1	0	0	0		0
0	0	1	0		0

Estimated plant from open loop system					
Grey-box approach					
A					**B**
0.00	1.00	0.00	0.00		0
0.00	0.00	3.77	0.00		0
0.00	0.00	0.00	1.00		0
-7.42	0.00	0.00	-156.07		24.36
C					**D**
1	0	0	0		0
0	0	1	0		0

Estimated plant from open loop system					
Black-box approach					
A					**B**
1.00	0.00	0.00	-0.02		0.00
0.00	1.00	0.00	0.01		0.00
0.00	0.00	1.00	0.04		0.00
0.00	-0.02	-0.03	0.58		0.00
C					**D**
-53.28	3.39	-26.53	-0.01		0.00
2.61	8.85	1.37	0.02		0.00

G.2.2 ID from closed-loop plant

Estimated plant from closed loop system					
Free structured parameterizations approach					
A					**B**
0.00	1.00	0.00	0.00		0
0.00	0.00	3.76	0.00		0
0.00	0.00	0.00	1.00		0
-4.84	1.37	-4.30	-103.17		16.225
C					**D**
1	0	0	0		0
0	0	1	0		0

Estimated plant from closed loop system					
Grey-box approach					
A					**B**
0.00	1.00	0.00	0.00		0
0.00	0.00	3.81	0.00		0
0.00	0.00	0.00	1.00		0
-28.39	-17.05	-17.30	-105.39		17.72
C					**D**
1	0	0	0		0
0	0	1	0		0

Estimated plant from closed loop system					
Black-box approach					
A					**B**
0.97	-0.02	-0.04	-0.02		0.00
0.04	1.02	0.04	-0.02		0.00
-0.32	-0.24	0.60	0.15		-0.03
0.08	0.04	0.06	0.98		0.00
C					**D**
-0.09	0.16	0.01	-0.05		0.00
0.74	0.13	-0.02	0.10		0.00

G.2.3 ID from full state observer plant

Estimated plant from full-state-observer system					
Free structured parameterizaition approach					
A					B
0.00	1.00	0.00	0.00		0
0.00	0.00	3.77	0.00		0
0.00	0.00	0.00	1.00		0
-5.20	0.00	0.00	-111.89		24.13
C					D
1	0	0	0		0
0	0	1	0		0

Estimated plant from full-state-observer system					
Black-box approach					
A					B
1.00	0.00	0.00	-0.01		12.509
-0.01	0.98	0.03	0.00		1.046
0.00	-0.04	1.00	0.02		0.008
0.02	0.08	-0.02	0.93		-0.009
C					D
1.1761	-0.47947	0.30618	-0.17661		0
0.20327	2.3987	-2.146	0.6598		0

G.2.4 ID from reduced order observer plant

Estimated plant from reduced order observer system					
Free structured parameterizaition approach					
Aaa		Aab			Ba
0.00	0.00	1.00	0.00		0
0.00	0.00	0.00	1.00		0
Aba		Abb			Bb
0.00	3.77	0.00	0.00		0
18857	-6623.4	0	-118.71		27.78
C					D
1	0	0	0		0
0	1	0	0		0

Estimated plant from reduced order observer system					
Black-box approach					
Aaa		Aab			Ba
4.52	-1.33	-0.01	-0.03		0.002
-0.62	1.23	0.03	0.04		0.000
Aba		Abb			Bb
0.978	-0.401	0.998	0.013		0.001
-5.027	1.870	-0.036	0.889		-0.003
C					D
0.504	0.306	0.132	-0.005		0
0.575	-2.494	-0.864	-0.048		0

G.3 Experiment results

G.3.1 ID from closed loop plant

Estimated plant from closed loop system					
Free structured parameterizaition approach					
A					B
0.00	1.00	0.00	0.00		0
0.00	0.00	5.01	0.00		0
0.00	0.00	0.00	1.00		0
17.16	6.27	9.77	-69.70		10.45
C					D
1	0	0	0		0
0	0	1	0		0

Estimated plant from closed loop system					
Grey-box approach					
A					B
0.00	1.00	0.00	0.00		0
0.00	0.00	4.90	0.00		0
0.00	0.00	0.00	1.00		0
-14.78	0.00	0.00	-50.54		7.3915
C					D
1	0	0	0		0
0	0	1	0		0

Estimated plant from closed loop system					
Black-box approach					
A					B
1.00	0.00	-0.01	0.00		0.00
0.00	1.00	0.00	0.00		0.00
0.02	0.00	0.98	-0.01		0.00
-0.01	0.01	-0.01	1.00		0.00
C					D
1.80	-2.11	0.04	-0.01		0.00
-10.83	0.00	0.04	0.01		0.00

G.3.2 ID from full state observer plant

Estimated plant from full-state-observer system					
Free structured parameterizaition approach					
A					B
0.00	1.00	0.00	0.00		0
0.00	0.00	4.12	0.00		0
0.00	0.00	0.00	1.00		0
-0.65	0.00	0.00	-15.42		14.041
C					D
1	0	0	0		0
0	0	1	0		0

Estimated plant from full-state-observer system					
Black-box approach					
A					B
1.00	0.00	0.01	0.00		0.00
0.00	1.00	0.00	0.00		0.00
-0.01	-0.01	1.00	0.01		0.00
0.00	0.00	-0.02	1.01		0.00
C					D
2.81	-5.40	0.01	0.01		0.00
-12.83	-0.01	-0.05	0.00		0.00

G.3.3 ID from reduced order observer plant

Estimated plant from reduced order observer system					
Free structured parameterizaition approach					
Aaa		**Aab**			**Ba**
0.00	0.00	1.00	0.00		0
0.00	0.00	0.00	1.00		0
Aba		**Abb**			**Bb**
0.00	4.66	0.00	0.00		0
-3036.3	1195.5	0	-27.687		15.148
C					**D**
1	0	0	0		0
0	1	0	0		0

Estimated plant from reduced order observer system					
Black-box approach					
Aaa		**Aab**			**Ba**
1.00	-0.01	0.01	0.00		0.000
0.00	1.00	0.01	0.00		0.000
Aba		**Abb**			**Bb**
-0.428	0.149	0.987	0.045		0.000
-0.124	0.054	-0.016	1.013		0.000
C					**D**
1.285	-1.899	0.024	0.011		0
-8.086	0.000	-0.035	0.000		0