EE-511 Lab Task, Fall 2014

Position Control of a DC Motor Using a Digital Observer-Based Controller

Submitted By:

Ahmer Saleem
Muhammad Awais
Muhammad Nasim Kashif
Raheel Javed
Zain Azam

Department of Electrical Engineering
Pakistan Institute of Engineering & Applied Sciences

Course Instructor: Dr. Ghulam Mustafa

Contents

Abstract

This report includes the identification of linear mathematical model of the DC servo system in state space form along with the design and implementation of a digital observer based controller for the position of the system. Digital plant is obtained by zero order hold discretization and the controller is designed by pole placement approach. The plant used here has two states that are position and velocity of shaft. An observer is also designed to estimate the velocity since only position signal is available. The position control system is open loop unstable in nature so a desired position is obtained by applying a voltage signal to motor connected in feedback configuration. In this case the motor is controlled by changing armature voltage. The observer and controller is implemented on computer which also generates the reference signal. It is interfaced to the plant via data acquisition card. The simulation of plant with observer and state feedback controller is done using MATLAB and the results are presented. The responses obtained from the implementation on real plant are also included.

Chapter 1

Introduction

To perform the task of DC motor control, a DC servo system is used.

1.1 Introduction to DC Servo System

A servomechanism or servo is an automatic device that uses error-sensing feedback to correct the performance of a mechanism. The term correctly applies only to systems where the feedback or error correction signals help control mechanical position or other parameters. Modern servomechanisms use solid state power amplifiers, usually built from MOSFET or thyristor devices. Small servos may use power transistors. The origin of the word is believed to come from the French Le Servomoteur or the slavemotor, first used by J. J. L. Farcot in 1868 to describe hydraulic and steam engines for use in ship steering. In the strictest sense, the term servomechanism is restricted to a feedback loop in which the controlled quantity or output is mechanical position or one of its derivatives (velocity and acceleration). Servomechanisms were first used in military fire-control and marine navigation equipment, speed governing of engines, automatic steering of ships, automatic control of guns and electromechanical analog computers. Today, servomechanisms are employed in almost every industrial field. Among the applications are cutting tools for discrete parts manufacturing, rollers in sheet and web processes, elevators, automobile machine tools and aircraft engines, robots, remote manipulators and teleoperators, telescopes, antennas, space vehicles, satellite tracking antennas ,remote control airplanes, anti -aircraft gun control systems, mechanical knee and arm prostheses, and tape, disk, and film drives.

The Feedback MS150 Modular Servo system is particularly intended for experimental use on studies of closed-loop systems. Each of the units of this equipment is fitted with magnetic feet and can be attached to the base-board in any desired position. The main power supplies for the servo amplifier unit and the motor tacho unit are fed through the cables terminating in octal plugs fitted to both Motor-Tacho and Servo Amplifier unit. The lead from the motor should be plugged into the servo amplifier and that from the amplifier into the power supply. Both power supply unit and servo amplifier unit are fitted with 4mm sockets from which 15 d.c supplies can be drawn to operate all other units of the system. The main components of system used in this task are:

1.1.1 Power Supply Unit (PS150E)

This unit supplies a 24v d.c 2A unregulated supply to the motor through an 8-way connector to the Servo Amplifier, as it is this unit that controls the motor. On the front panel there are two sets of 4mm sockets to provide 15v stabilized d.c supplies to operate the smaller amplifiers and provide reference voltage.

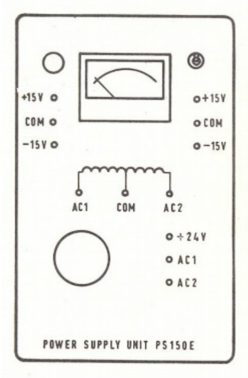

Figure 1.1: PS150E

1.1.2 Motor-Tacho Unit (MT150F)

This unit is made up of 3 parts.

1. A d.c series-wound split-field motor which has an extended shaft, and onto which can be fixed the magnetic brake or inertia disc.
2. A d.c tacho-generator with output on the top of the unit.
3. For control experiments, there is a low-speed shaft driven by a 30:1 reduction gearbox.

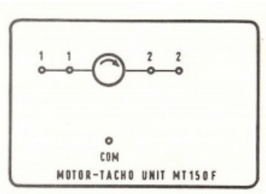

Figure 1.2: MT150F

1.1.3 Servo Amplifier Unit (SA150D)

This unit contains the transistors which drive the motor in either direction. On the front panel connection are provided for patching the armature for different modes of control (e.g. Field controlled or armature controlled).

1.1.4 Input and Output Potentiometer Units (IP150H and OP150K)

These are rotary potentiometers, used in experiments on position control. The input potentiometer is used to set up a reference voltage and the output potentiometer is connected to the low speed shaft of the motor by using the push-on couplings.

1.2 Interfacing with Computer

The signal to control the position of motor is generated by computer where the controller and observer is implemented. DC servo system is connected to computer using 1208FS USB data

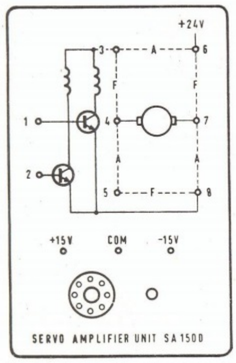

Figure 1.3: SA150D

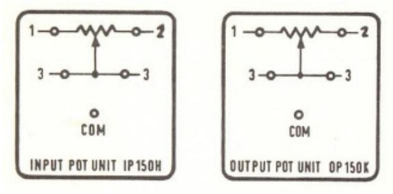

Figure 1.4: IP150H and OP150K

acquisition card by Measurement Computing. The card can be accessed using MATLAB. Some important features are:

- 4 differential (12-bits) or 8 single-ended (11-bits) analog inputs
- ±1V to ±20V range for differential inputs and ±10V for single-ended inputs
- 2 analog outputs (12-bits) with range of 0 to 4.096 V
- Maximum sampling rate of 50KS/s
- 16 Digital I/O lines

Figure 1.5: USB 1208-FS

1.3 Connection Description

The connections of the various parts are shown with the help of block diagram 1.6. The computer

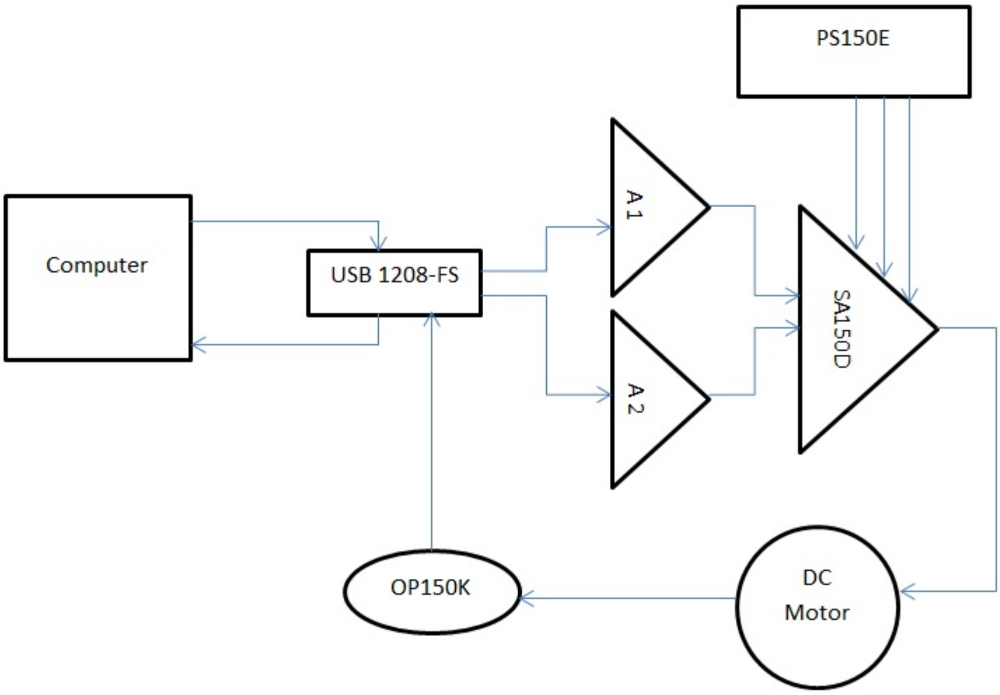

Figure 1.6: Block Diagram of System

provides control signal via DAQ. Since the output signal of DAQ is limited between 0 to 4V and SA150D receives input till 15V, amplifiers A1 and A2 are provided. The two amplifiers serve the purpose of bi-directional rotation of motor. They are supplied by an auxiliary power source. The power supply PS150D supplies ±15V to SA150D and OP150D (not shown in diagram). It is also 24V to motor (not shown in diagram). The connections are made to control the speed of motor through armature winding. The position signal reaches computer through OP150K (Connected to the shaft of motor.) via DAQ.

Chapter 2

Identification of The Plant Model

Mathematical modelling of a dc servo system is essentially that of a dc motor because in open loop the system is simply a dc motor which is armature or field controlled with gears and potentiometer. Consider figure 2.1. A pair of potentiometers acts as an error-measuring device.They convert the input and output positions into proportional electric signals.The command input signal determines the angular position r of the wiper arm of the input potentiometer. The angular position r is the reference input to the system, and the electric potential of the arm is proportional to the angular position of the arm.The output shaft position determines the angular position c of the wiper arm of the output potentiometer. The difference between the input angular position r and the output angular position c is the error signal e, or

$$e = r - c \tag{2.1}$$

The potential difference $e_r - e_c = e_\epsilon$ is the error voltage, where e_r is proportional to r and e_c is proportional to c; that is, $e_r = K_o r$ and $e_c = K_o c$ where K_o is a proportionality constant.The error voltage that appears at the potentiometer terminals is amplified by the amplifier whose gain constant is K,.Theoutput voltage of this amplifier is applied to the armature circuit of the dc motor. A fixed voltage is applied to the field winding.If an error exists,the motor develops a torque to rotate the output load in such a way as to reduce the error to zero. For constant field current, the torque developed by the motor is

$$T = K_2 i_a \tag{2.2}$$

where K_2 is the motor torque constant and i_a is the armature current. When the armature is rotating, a voltage proportional to the product of the flux and angular velocity is induced in the armature. For a constant flux, the induced voltage e_b is directly proportional to the angular velocity $\frac{d\theta}{dt}$,or As shown earlier a DC servo system in open loop is simply a dc motor that may be field or armature controlled. In this case the speed of motor is changed by controlling the voltage to the armature winding of the motor.

$$e_b = K_3 \frac{d\theta}{dt} \tag{2.3}$$

where e_b is the back emf, K_3 is the back emf constant of the motor, and θ is the angular displacement of the motor shaft

2.1 Using Theory

The speed of an armature-controlled dc servomotor is controlled by the armature voltage e_a. (The armature voltage $e_a = K_1 e_\epsilon$, is the output of the amplifier.) Referring to figure 2.1 the differential equation for armature circuit is,

$$L_a \frac{di_a}{dt} + R_a i_a + e_b = e_a$$

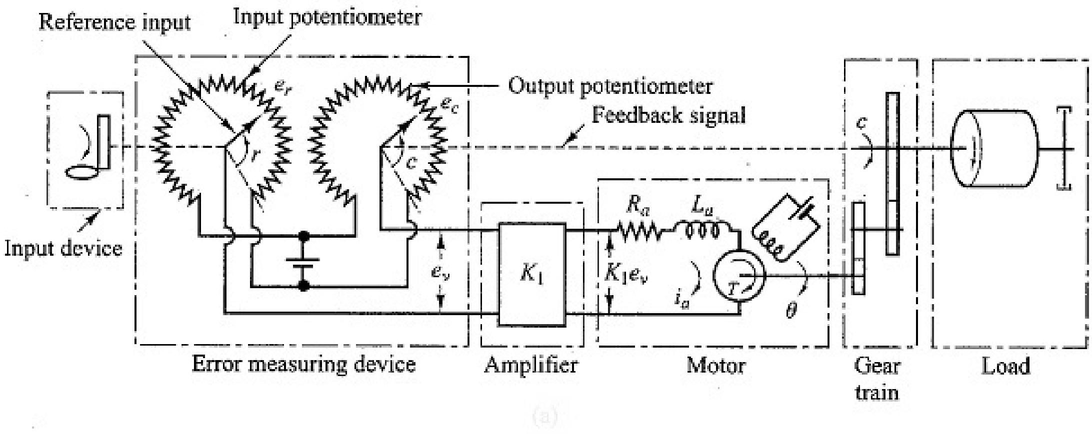

Figure 2.1: Schematic Diagram of DC Servo System

or,

$$L_a\frac{di_a}{dt}+R_ai_a+K_3\frac{d\theta}{dt}=K_1e_v \tag{2.4}$$

The equation for torque equilibrium is,

$$J_o\frac{d^2\theta}{dt^2}+b_o\frac{d\theta}{dt} \quad = \quad T=K_2i_a \tag{2.5}$$

Where J_o is the inertia of the motor, load and gear train refered to the motor shaft and b_o is the viscous coefficient of combination of the motor, load, and gear train referred to the motor shaft. It is assumed that the gear ratio of the gear train is such that the output shaft rotates n times for each revolution of motor shaft. Thus,

$$c(t)=n\theta(t)$$

The relationship among $e_v(t), r(t), andc(t)$ is,

$$e_v(t)=K_o[r(t)-c(t)]=K_oe(t)$$

Equation 2.4 and 2.5 becomes,

$$L_a\frac{di_a}{dt}+R_ai_a+\frac{K_3}{n}\frac{dc}{dt} \quad = \quad K_oK_1[r-c] \tag{2.6}$$

$$\frac{J_o}{n}\frac{d^2c}{dt^2}+\frac{b_o}{n}\frac{dc}{dt} \quad = \quad K_2i_a \tag{2.7}$$

Using state variables,

$$x_1=\theta,\ x_2=\dot{\theta},\ x_3=i_a$$

After manipulation of above equations following state space representation is obtained,

$$\begin{aligned}\dot{x} &= Ax+Bu\\ y &= Cx\end{aligned}$$

Where,

$$A=\begin{bmatrix}0 & 0 & 1\\ \frac{K_2n}{J_o} & 0 & \frac{-b_o}{J_o}\\ \frac{-R_a}{L_a} & \frac{-K_oK_1}{L_a} & \frac{-K_3}{L_an}\end{bmatrix}, B=\begin{bmatrix}K_oK_1\\ 0\\ 0\end{bmatrix}, C=\begin{bmatrix}1 & 0 & 0\end{bmatrix}$$

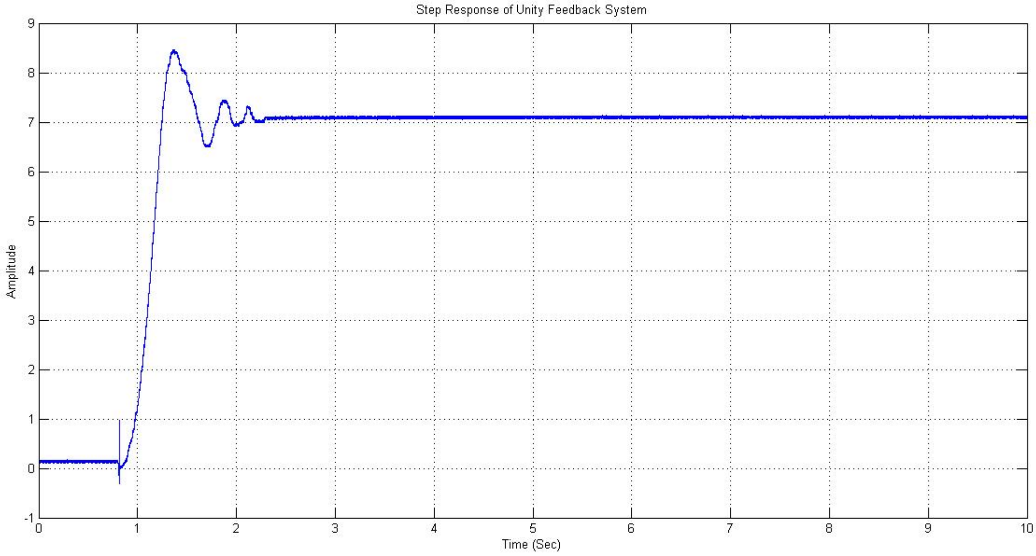

Figure 2.2: Step Response of Unity Feedback System

2.2 Using Response from Actual Plant

The system is connected in unity feedback configuration with a step voltage of 7.45 V applied. The response is obtained in MATLAB using data acquisition card. The sampling frequency is kept high (10000 samples/sec) so that the accuracy of result is not lost. Y-axis shows voltage corresponding to the position of the shaft and X-axis shows time in seconds. The response is shown in fig 2.2.

From the plot,

$T_s = 1.769$ (5% Criterion)
$Peak = 8.459$

The maximum overshoot (MO) is thus, $\frac{8.459-7.45}{7.45} = 0.1914$

Using the relations,

$$\begin{aligned} MO &= exp(\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}) \\ T_s &= \frac{3}{\omega_n T_s} \end{aligned}$$

We have, $\zeta = 0.4657$ and $\omega_n = 3.6413$

Now approximating the system by standard second order system.

$$\begin{aligned} \frac{G(s)}{1+G(s)} &= \frac{\omega_n^2}{s^2+2\zeta\omega_n+\omega_n^2} \\ \frac{G(s)}{1+G(s)} &= \frac{13.2593}{s^2+3.3917s+13.2593} \end{aligned}$$

Solving for $G(s)$ will give the transfer function of open loop plant.

$$G(s) = \frac{13.2593}{s^2 + 3.3917s}$$

In state space form,

$$A = \begin{bmatrix} 1 & 0 \\ 0 & -3.3913 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 13.2593 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Chapter 3

Discretization of the Plant

Since a digital controller is to be designed so it is necessary to discretize the plant. For this purpose the model obtained from actual plant is used since it not only includes all the effects of system but also shows the effects due to variation of constant parameters over time.

3.1 Selection of Sampling Time

The sampling time (h) is calculated using rule of thumb for second order systems. i.e

$N_r = \frac{T_r}{h} = 4$ to 10 times

For standard second order systems,

$T_r = \omega_n^{-1} exp(\frac{\beta}{tan\beta})$

where,$cos\beta = \zeta$

Using this relation and for $N_r = \frac{T_r}{h} = 8$, h comes out to be $60ms$.

3.2 ZOH Discretization

In this scheme it is assumed that the input to the plant between two sampling instants is constant and is equal to the previous sample. The ZOH discretization of the plant is done using Matlab. The results are,

$$\begin{aligned} x(k+1) &= \phi x(k) + \Gamma u \\ y(k) &= Cx(k) \end{aligned}$$

Where,

$$\phi = \begin{bmatrix} 1 & 0.05429 \\ 0 & 8.159 \end{bmatrix}, B = \begin{bmatrix} 0.0.2233 \\ 0.7198 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

Chapter 4

Design of Digital Observer Based Controller

To obtain desired response state feedback controller is used with minimum order observer.

4.1 Specifications of Controller

The controller is designed to limit percentage overshoot to 20% and settling time to 1 sec with zero steady state error. Using these requirements the dominant poles can be found in s-domain using eqns 2.8 and 2.8.

$$s_{1,2}\text{= -3 + 5.8559j, -3 - 5.8559j}$$

Using relation,

$$z = exp(sT)$$

Where T = Sampling period

The poles can be calculated in z-domain. So,

$$z_{1,2} = \text{0.7842 + 0.2875j, 0.7842 - 0.2875j}$$

4.2 Controller and Observer Design

The controller is designed by pole placement approach using Matlab. The obtained gain vector is K=[2.9913 0.2509]. Minimum order observer is designed to estimate velocity state of the system. The gain obtained is K_m=15.0286. The transfer function of observer controller is calculated as follows,

$$\begin{aligned}
\hat{\eta}(k+1) &= (G_bb - K_eG_ab)\hat{\eta}(k) + [(G_bb - K_mG_ab)K_m + G_ba - K_mGaa]y(k) + (H_b - K_mH_a)u(k) \\
u(k) &= -K\hat{x}(k) \\
\hat{x}_2(k) &= K_my(k) + \hat{\eta}(k)
\end{aligned}$$

Where,
G_{aa}, G_{bb}, G_{ab} and G_{ba} are the partitions of $G = \theta$ matrix.
H_a and H_b are the partitions of $H = \Gamma$ vector.

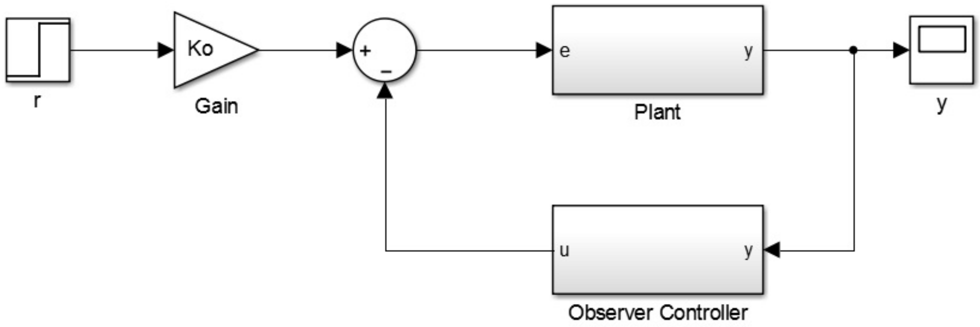

Figure 4.1: Structure of System

Solving the above equation gives,

$$-\frac{U(z)}{Y(z)} = \frac{6.762z - 3.7707}{z + 0.964} \tag{4.1}$$

It is necessary to provide an adjustable gain K_o in the input path so that the gain of entire control system can be determined such that the steady-state output to a unit step input is unity. This is because the pole placement with observer modifies the gain of the entire system. Using reference input as unity and using,

$$Lim_{z->1}[(1 - z^{-1})K_oY(z)] = 1 \tag{4.2}$$

The K_o comes out to be 2.73.

Chapter 5

Simulation Results

First of all the observer-controller design is simulated using discrete controller and plant. Step response is then obtained using MATLAB. The code for simulation is included in appendix.

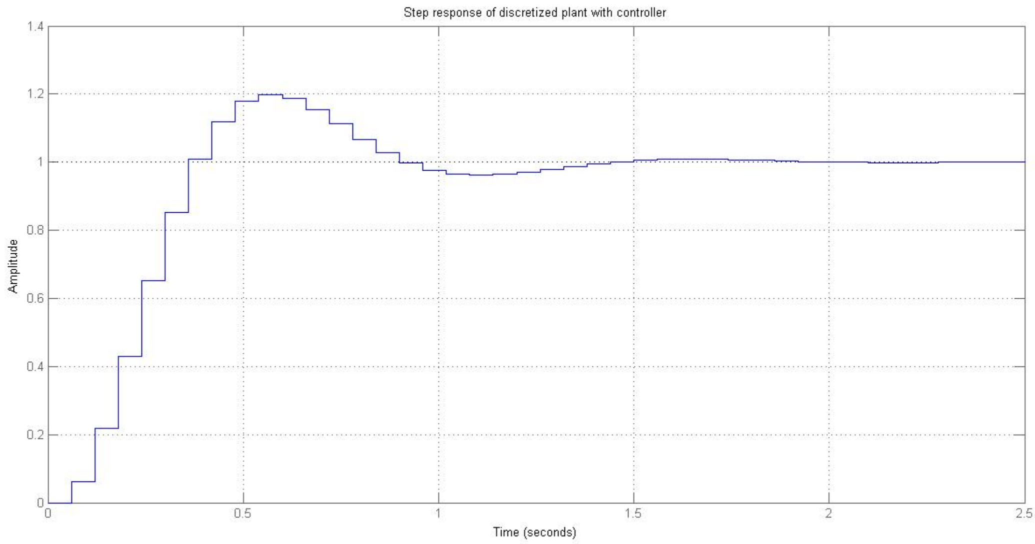

Figure 5.1: Controller with Discrete Plant

For more realistic results the process is repeated for discrete controller and continuous plant (i.e sampling time of plant smaller than the controller).

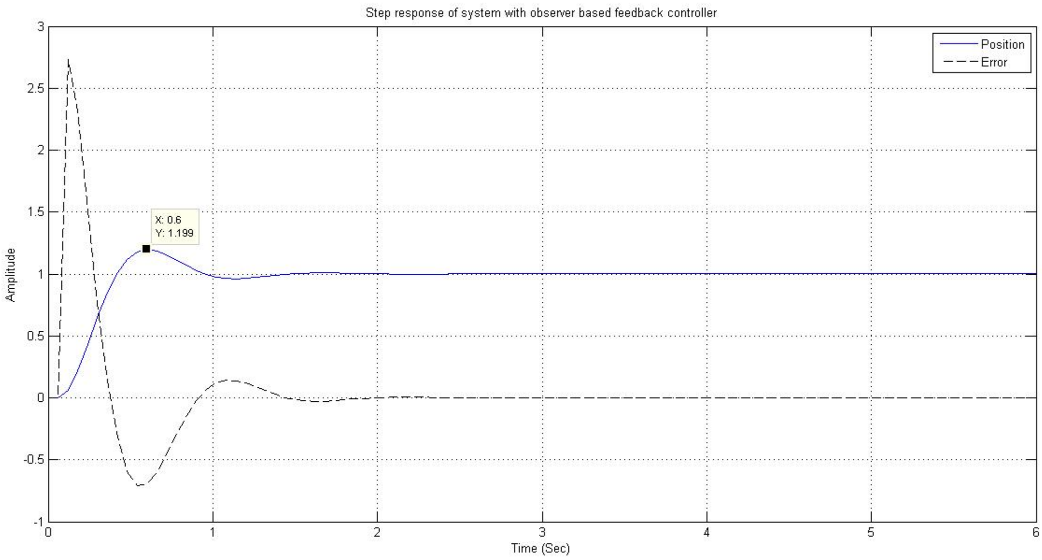

Figure 5.2: Controller with Continuous Plant

Chapter 6

Implementation of Observer based Controller

The results obtained by implementing the controller on real system are shown for a reference voltage of 7V which is generated by computer. Sampling time is kept very small (50000 samples/sec). The noise in the plots is due to position sensor (potentiometer) that can be easily filtered out. The performance of system is satisfactory but there is a steady state error that may be removed by adjusting input gain such that the output reaches reference voltage. A curve between position of the shaft and output voltage can be drawn by calibrating output potentiometer.

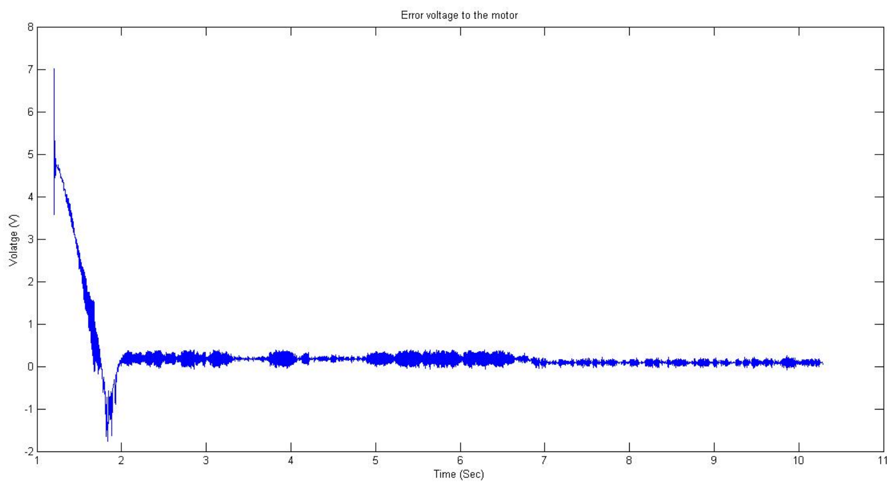

Figure 6.1: Error signal (r-u) that is fed to motor

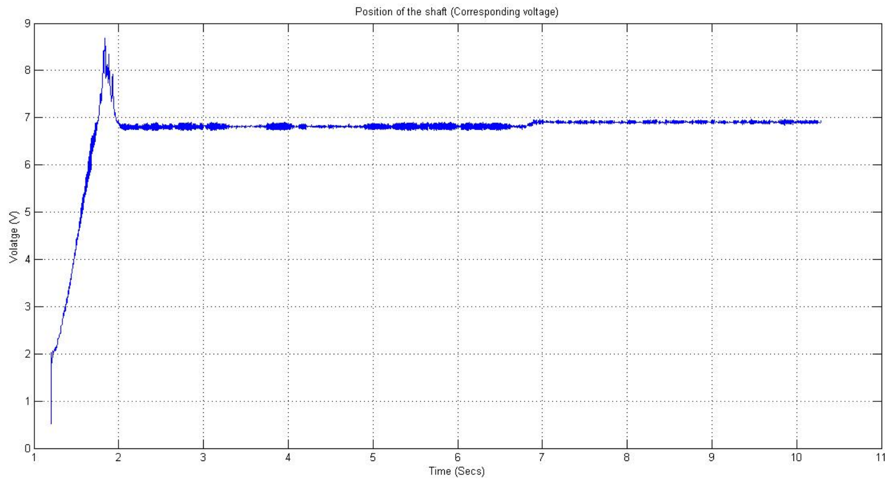

Figure 6.2: Position of the shaft in terms of voltage

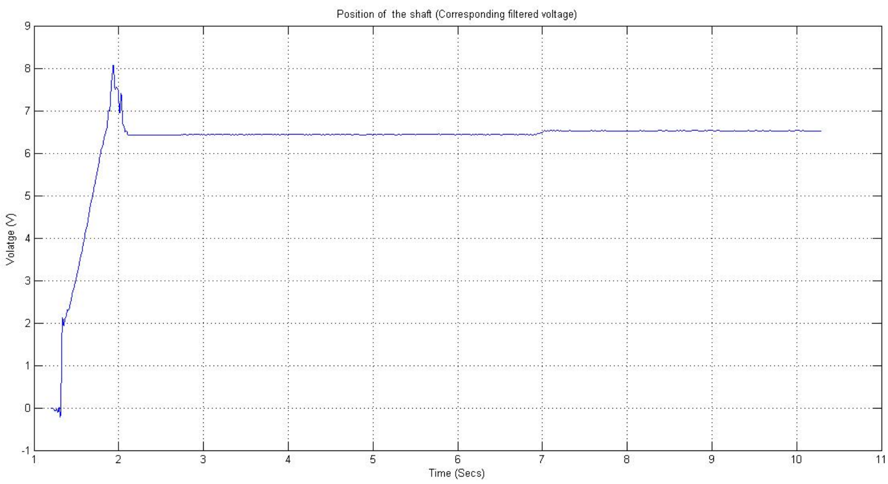

Figure 6.3: Filtered Version of Position Signal

Appendix A

Matlab Code for Simulation

The following code of matlab can be used to design observer based controller and plot the step response using discrete plant and controller.

```
clear
clc

%Open Loop System
A=[0 1; 0 -3.3917];
B=[0; 13.2593];
C=[1 0];
D=0;
G_c=ss(A, B, C, D);
[num, den]=ss2tf(A, B, C, D);
G=tf(num, den);

%Discretization
T=0.06;
[G_d H]=c2d(G_c, T, 'ZOH');

%Controller Specifications
MO=0.2;
Ts=1;

%Dominant Closed Loop Poles of Unity Feedback System (s-domain)
zeta=(abs(log(MO)))/(sqrt(pi^2+(log(MO)^2)));
wn=3/(zeta*Ts);
xtic_eqn=[1 2*zeta*wn wn^2];
d_poles=roots(xtic_eqn)';
a = real(d_poles(1));
b = abs(imag(d_poles(1)));

%Desired Closed Loop Poles (z-domain)
pol_1=exp(a*T)*(cos(b*T)+1j*sin(b*T));
pol_2=exp(a*T)*(cos(b*T)-1j*sin(b*T));
c_pol=[pol_1 pol_2];

%Controller Design
K=place(G_d.A, G_d.B, c_pol);
```

```
%Observer Design
G22=[0.8159];
G21=[0.05429];
Km=place(G22', G21', 0)';


[num, den]=ss2tf(G_d.A, G_d.B, G_d.C, G_d.D);
P=tf(num, den, T);                        %Plant transfer function
C=tf([6.762 -3.7707], [1 0.0964], T);   %Observer-Controller transfer function
S=feedback(P, C);       %Transfer function of complete system
step(S*2.73)
grid
title('Step response of discretized plant with controller')
```

The following code can be used to plot the step response of system with discrete observer-controller and continuous plant.

```
%Main.c
clear
clc

N=100;                   %Number of samples
T=0.06;                  %Sampling Time
r=2.73*ones(1, 100);     %Input Signal

for k=1:1
    y(k)=0;      %Initializing first samples of output, error
                 %and controller output
    u(k)=0;
    e(k)=0;
end
xo=[0; 0];

%Model of the plant
A=[0 1; 0 -3.3917];
B=[0; 13.2593];
C=[1 0];
D=0;
G=ss(A, B, C, D);

%Transfer Function of Controller
D=tf([6.762 -3.771], [1 0.0964], T);

%Step response of system with observer based state feedback controller
for k=2:N
e(k)=r(k)-u(k-1);
%ZOH
l=0:0.06/6:0.06;        %System can change output 6 times during a sampling period
e_h=[e(k) e(k) e(k) e(k) e(k) e(k) e(k)];
[y_h, l, x]=lsim(G, e_h, l, xo);
```

```
xo=x(7, :)';             %Previous States
%Sampling of Output
y(k)=y_h(7);
u(k)=controller(y, u, k, D.num{1}, D.den{1});   %Observer-Controller output
end

%Results
hold off
plot([1:N]*T, y)
hold on
plot([1:N]*T, e, 'k--')
grid
title('Step response of system with observer based state feedback controller')
xlabel('Time (Sec)')
ylabel('Amplitude')


%controller.c
function u_k=controller(e, u, k, num, den)
u_k=-den(2)*u(k-1)+num(1)*e(k)+num(2)*e(k-1);
end
```