MODELLING AND CONTROLLER DESIGN OF THE GUN AND TURRET SYSTEM FOR AN AIRCRAFT

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

AHMET MERT

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

FEBRUARY 2009

Approval of the thesis:

MODELLING AND CONTROLLER DESIGN OF
THE GUN AND TURRET SYSTEM FOR AN AIRCRAFT

submitted by **AHMET MERT** in partial fulfillment of the requirements for the degree of **Master of Science in Electrical and Electronics Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. İsmet Erkmen _____
Head of Department, **Electrical and Electronics Engineering**

Prof. Dr. Kemal Leblebicioğlu _____
Supervisor, **Electrical and Electronics Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Mübeccel Demirekler _____
Electrical and Electronics Engineering Dept., METU

Prof. Dr. Kemal Leblebicioğlu _____
Electrical and Electronics Engineering Dept., METU

Assist. Prof. Dr. Afşar Saranlı _____
Electrical and Electronics Engineering Dept., METU

Assist. Prof. Dr. Emre Tuna _____
Electrical and Electronics Engineering Dept., METU

M. Sc. Yüksel Serdar _____
Electronics Design Dept., ASELSAN MGEO

Date: _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Ahmet Mert

Signature :

ABSTRACT

MODELLING AND CONTROLLER DESIGN OF
THE GUN AND TURRET SYSTEM FOR AN AIRCRAFT

Mert, Ahmet
M.S., Department of Electrical and Electronics Engineering
Supervisor: Prof. Dr. Kemal Leblebicioğlu

February 2009, 135 Pages

Gun and gun turret systems are the primary units of the weapon systems of an aircraft. They are required to hit targets accurately during operations. That is why a complete, high precision control of weapon systems is required. This function is provided by accurate modeling of the system and the design of a suitable controller.

This study presents the modeling of and controller design for the gun and turret system for an aircraft. For the controller design purpose, first the mathematical model of the system is constructed. Then the controller is designed to position the turret system as the target comes into sight. The reference input to the controller will either be obtained from a FLIR (Forward Looking Infrared) unit or from a HCU (Hand Control Unit). The basic specification for the controller is to hold the

error signal within the 5.5° positioning envelope. This specification is satisfied by designing Linear Quadratic Gaussian and Internal Model Control type controllers.

The performance of the overall system has been examined both by simulation studies and on the real physical system. Results have shown that the designed system is well over being sufficient.

Keywords: Turret System, Modelling and Control, Internal Model Control, Linear Quadratic Gaussian.

ÖZ

HAVA ARACI İÇİN TOP VE TARET SİSTEMİ MODELLENMESİ VE DENETLEYİCİ TASARIMI

Mert, Ahmet
Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Kemal Leblebicioğlu

Şubat 2009, 135 Sayfa

Top ve top taret sistemi hava aracının silah sistemlerinin temel birimleridir. Bunların operasyon esnasında hedefi doğru vurabilmeleri gerekmektedir. Bu sebepten ötürü, silah sisteminin tam ve yüksek hassasiyette denetlenmesi gerekmektedir. Bu fonksiyon sistemin doğru modellenmesi ve uygun denetleyicilerin tasarlanması ile sağlanmaktadır.

Bu çalışma, hava aracı için top ve taret sisteminin modellenmesi ve denetleyici tasarımından oluşmaktadır. Denetleyici tasarımı amacıyla ilk olarak sistemin matematiksel modeli oluşturulmuştur. Daha sonra; hedef görüş açısına girdiğinde taret sisteminin yönlendirilmesi için denetleyici tasarlanmıştır. Denetleyicinin referans girişi ya kızılötesi görüş cihazından (FLIR) ya da el kumanda biriminden

(HCU) elde edilecektir. Denetleyici için temel özellik hata işaretini 5.5 derecelik konum açısı içerisinde tutmaktır. Bu özellik “Linear Quadratic Gaussian” ve “Internal Model Control” tipi denetleyiciler tasarlanarak elde edilmiştir.

Tüm sistemin performansı simulasyon çalışmalarında ve gerçek fiziksel sistem üzerinde sınanmıştır. Sonuçlar tasarlanmış sistem oldukça başarılı olduğunu gösteriyor.

Anahtar Kelimeler: Taret Sistemi, Modelleme ve Kontrol, “Internal Model Control”, “Linear Quadratic Gaussian”.

Bugünlere gelmemi sağlayan anneme ve babama;
Ve bundan sonraki hayatımı paylaşacağım yegane desteğim eşime,

ACKNOWLEDGEMENT

I would like to express my sincere appreciation to my supervisor Prof. Dr. Kemal Leblebicioğlu for his guidance, suggestions, patience, and encouragement throughout the study.

I would like to acknowledge Metin Sancar and Yüksel Serdar sincerely for their support, critics and guidance.

I wish to thank my colleagues Rafet Eroğlu and Sait Sarı during system and hardware design step, Ali Murat Demirtaş and Alpaslan Lorasdağı for their support in software design process, Ufuk Doğan and Deniz Eroğlu at mechanical design.

I would like to thank ASELSAN Inc., Microelectronics, Guidance and Electro-Optics division for the valuable contributions in test facilities, technical documents and equipment.

Special thanks to Emre Rızvanoğlu and Cihan Onmuş, for their support and friendship.

I would also like to express my appreciation to my family for their perpetual support, continued faith in me and patience in my entire life.

Finally, my deepest special thanks go to my fiancee Gül for her motivation, endless love and being with me throughout my whole life.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURES

LIST OF TABLES

TABLES

ABBREVIATIONS

A	State matrix
B	Control input matrix
C	Measurement matrix
D	Disturbance
e	Disturbance / zero-mean white noise
G	Deterministic part of system identification models
G_c	Controller of Internal Model Control part
G_f	Filter of Internal Model Control part
G_p	Process of Internal Model Control part
G_{pm}	Process model of Internal Model Control part
H	Stochastic part of system identification models
I	Set point
K	Kalman filter gain matrix
L	Optimal control gain matrix
M_p	Peak value
Q	Weight matrix for the state vector
R	Weight matrix for the input
T_p	Peak time
T_r	Rise time
T_s	Settling time
u	Control input vector
v	Gaussian white process noise
V	Process noise covariance matrix

w	Gaussian white measurement noise
W	Measurement noise covariance matrix
x	Process state vector
y	Measurement output vector
y*	Output without any measurement error
AR	Autoregressive
ARMAX	Autoregressive Moving Average with Exogenous Input
ARX	Autoregressive with Exogenous Input
BJ	Box-Jenkins
CPLD	Complex Programmable Logic Device
DRC	Discrete to Resolver Converter
DSP	Digital Signal Processor
FLIR	Forward Looking Infrared
FPGA	Field Programmable Gate Array
GM	Gain Margin
HCU	Hand Control Unit
IMC	Internal Model Control
IU	Interface Unit
LQG	Linear Quadratic Gaussian
NVSRAM	Non-Volatile Static Random Access Memory
OE	Output-Error
PI	Proportional Integral
PM	Phase Margin
RAM	Random Access Memory
RDC	Resolver to Discrete Converter
SBSRAM	Synchronous Burst Static Random Access Memory
VHDL	Very High Speed Hardware Description Language

CHAPTER 1

INTRODUCTION

1.1 Introduction

In the battlefield, weapon systems of an aircraft are required to hit both stationary and moving targets as accurately as possible. Moreover, the gunner wants these systems to be easily controllable to perform precise pointing. These issues require a well-developed control system by considering the internal and external dynamics of the weapon system. However, designing a weapon control system with high precision is a complex task in which unpredictable uncertainties, parameter variations, nonlinearity, linearization of nonlinear elements and decision making processes take place [4].

Recently, in the design of high precision pointing systems robust, adaptive and nonlinear controller design algorithms are utilized [3], [5]. In these algorithms the main aim of the design is to achieve rapidly and precisely pointing the gun to the target under uncertainties and disturbances such as coulomb friction, backlash and servo limitation, parameter variations of torsional stiffness and similar disturbances [5]. As a consequence of these studies on the algorithms, it seems that the most effective approach to cope with uncertainties is the robust and intelligent control based designs.

Today the turret system control is one of the ongoing research problems. The control of such turret systems basically depends on constructing a plant model. This model determination stage is followed by choosing an applicable controller that is going to meet the system requirements efficiently. Then by designing this controller, the response of the system under this controller has to be examined. By following the same procedure for various controllers, the best controller is going to be set as the controller of the system. Obviously the mathematical model of the system is not necessarily exact because of uncertainties and unmodeled system dynamics or loss of model accuracy while linearizing the nonlinear parts. Hence designing and implementing a controller to cope with modeling errors for a turret system of an aircraft is a critical issue. Consequently, to meet the system requirements and to ensure system stability, a robust controller design is an important necessity.

There are various methods available for robust controller designs. In particular, Linear Quadratic Gaussian (LQG) design procedure has a number of important advantages as mentioned in [1] and [2]. This is one of the most effective robust controllers used for turret control [6]. Internal Model Control (IMC) type controller design is also used for robust control purposes. All these controller types, different modeling and controller design procedures for gun turret systems have been considered in the literature. In [5], the effect of different control methodologies for high precision weapon control system design is explained. Integration of the turret control system with system dynamics is described in [4]. As frequently pointed out by the previous studies, to control a real system, a physical setup that is able to communicate with the turret system has to be implemented.

1.2 Contributions of the Author

In our study, a setup that supports the communication with the turret system is implemented first. For this purpose, an interface unit is designed for both analog and digital data acquisition and system communication. This unit is basically composed of an interface board. The parts that are designed and implemented by

the author of the thesis inside this interface board are: hardware schematic design and board layout with the help of hardware designers, software development for Digital Signal Processor (DSP) with the help of software designers, coding and programming of Field Programmable Gate Array (FPGA) and Complex Programmable Logic Device (CPLD), controller design and implementation parts in the DSP. Also the data collection from the turret system and system testing are successfully achieved by the author of the thesis. The details about this interface board and interface unit are given in Appendix A.

Furthermore, by using this interface unit, a linear model of the system is obtained based on the measured data associated with the physical system. With the help of this plant model, the controllers are designed using 'MATLAB' and tested on the system. Finally, the responses of the controllers are compared and the best controller which meets the system requirements is chosen as the controller of the system.

1.3 Organization of the Thesis

In chapter 2, first of all, the procedure used for system identification is given. Then the system identification techniques in the literature are mentioned. According to our structure, most suitable system identification technique is determined.

In chapter 3, the data collection and system identification of the system is given. The chapter starts with the data collection of the system. Then, the most suitable system identification technique is selected and examined on the system. This procedure yields the mathematical model of the system. Moreover the procedure used for obtaining mathematical model of the physical system and some observations acquired during the mathematical modeling phase are mentioned.

In chapter 4, a brief description about the general control methodology and detailed explanation about the applied control methods are given. Especially IMC

and LQG methods are mentioned. Moreover, the properties of the controllers and the most suitable controllers for our system are explained.

In chapter 5, the performances of designed digital controllers are checked over some experimental results. On these results, real time outputs of the applied controllers are explained. The response of the real system to ramp input and the comparison between the simulation results and the real system response is made. Based on the observed data, different controllers have been compared and some comments on these controllers are given.

In chapter 6, a summary of the whole study and some observations about this work are given. Aims and results are compared. Certain suggestions for future work and obtained experience during this study are also discussed.

CHAPTER 2

SYSTEM IDENTIFICATION

In this chapter, the general system identification methodologies and their implementations are discussed. The purpose of this chapter is to understand the procedures to obtain the best plant model for our system. There are various model types that can be used as the model for a given system. According to the properties of system identification and our system structure, the most suitable system identification model is determined. Based on the chosen model, the mathematical representation and additional graphical details of this model are given. With the help of the data from the physical system, the approaches used and the objectives of choosing that model are summarized.

2.1 System Identification

General information about system identification and modeling are covered in [9], [10], [11], [12] and [13]. The following pages are based on these references. Especially [9] is an interesting reference handbook on system identification.

The design of a suitable controller to satisfy a set of design criteria usually needs an accurate plant model. However if the plant model differs from the real plant considerably, then the designed controller cannot satisfy the given requirements and cannot work properly.

There are different methods that can be used to obtain a mathematical model for a plant. These methods are: physical modeling, experimental modeling.

Physical Modeling is a kind of modeling that uses fundamental equations, collection of the system parameters and also some calculations. This type of modeling requires the knowledge of all the system parameters. This method is complex and time consuming as a system identification method.

Experimental modeling is an approach that determines the plant by using the responses of the plant. The system is modeled by experimenting. This methodology is also called as black-box for some cases if there isn't any information about the system parameters. But if there is little information about the system then the modeling is grey-box modeling. This methodology requires a prototype to do experimenting and to collect the data.

For our case, the system we use contains some apparatus that are too old to reach to their parameters and characteristics. That is why the physical modeling method is not helpful for us; however experimental modeling is the methodology that is used in obtaining the mathematical model.

There are different approaches used to identify the system. This approach can be linear/non-linear system identification and it can also be parametric/non-parametric methods.

a) Parametric Methods: A parameterized plant model is obtained and then the recorded data maps to this parameterized model directly. This is also called as parameter estimation method. This is a user-defined type model which has transfer function or state space matrices given by the user in hand.

b) Non-parametric methods: The plant model is obtained as a curve in frequency domain or time domain. One of the main advantages of these methods is that it is not required to have knowledge about the system. This model has frequency response, impulse response and/or step responses given by the user.

Figure 2-1 shows parametric and non-parametric methods in time domain and frequency domain analysis.

The systems can be linear or non-linear. The linear systems are easy to control and more flexible in controller design. For our case while constructing the model of the system, most parts of the system are assumed as linear according to the response of the system and some non-linear parts are linearized. Some undesired affects of this linearization have also been taken into account in controller design. The details about these issues are mentioned in 3.1 Modeling of our System part.

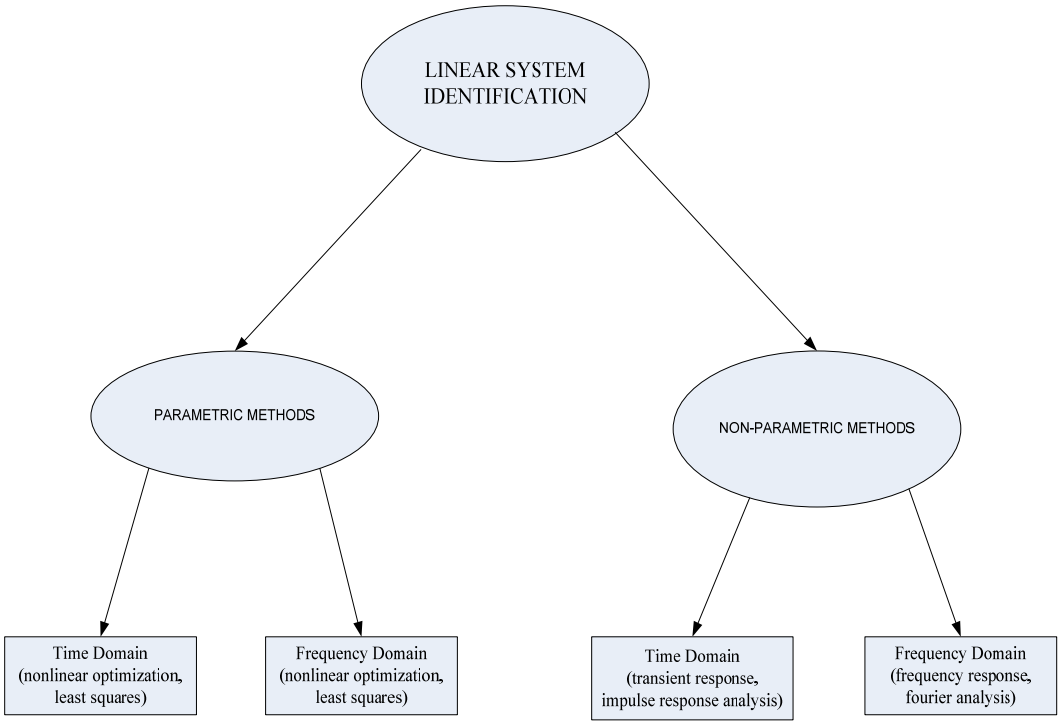


Figure 2-1 Linear System Identification.

System identification is identifying the dynamics of the system to be used in controller design. In order to achieve this job successfully, one has to collect sufficiently rich data about the system. For almost all the systems, Figure 2-2 given below is valid. The output of the system is just the response of the system to the given input and to the disturbance affecting the system.

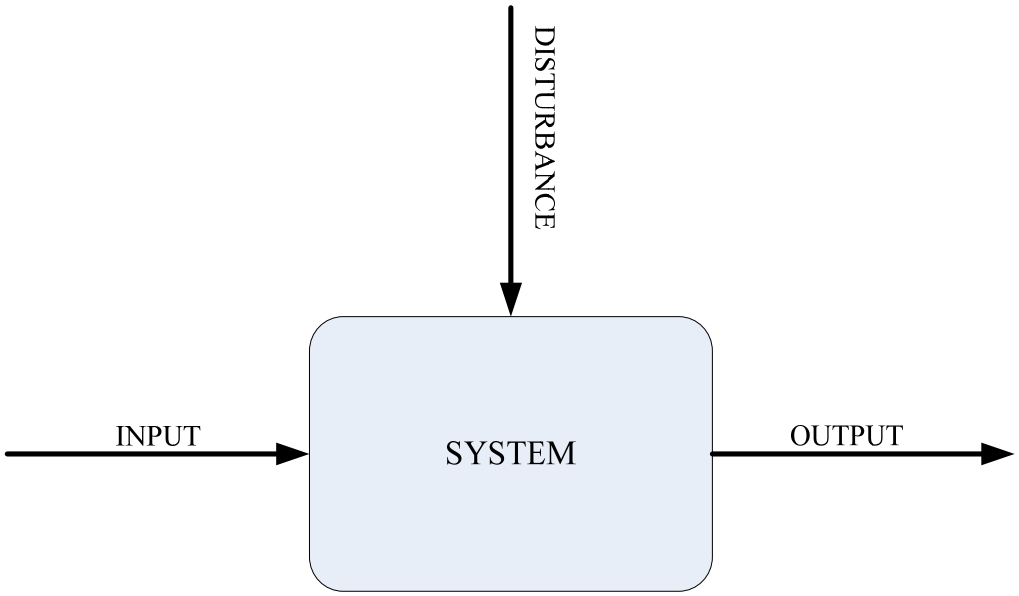


Figure 2-2 General System Structure

This system identification is done by following the procedure outlined below:

- Experiments
- Identification of Model Structure
- Model Parameter Estimation
- Model Validation

This procedure is given in Figure 2-3 below as well and it is the standard procedure explained in references [10], [11].

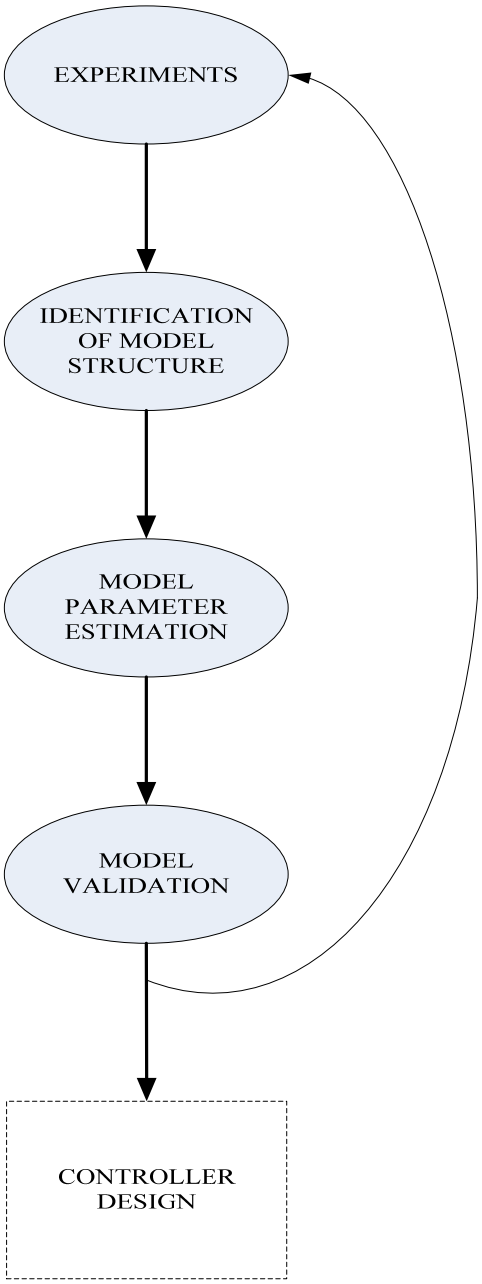


Figure 2-3 System Identification Procedure

2.1.1 Experiment

The design of the experiments and the data collection is explained in this section. Most of the real systems are nonlinear. To identify nonlinearity and to linearize the system, suitable inputs must be chosen. This means that the bandwidth of the

input signals has to be sufficiently large so that accurate model could be obtained and the nonlinearities such as friction can be observed. It is also important to satisfy a good signal-to-noise ratio and meaningful range for input signal amplitude. While choosing this input signal range, the linearity assumptions of the system must not be exceeded. This means that sufficiently large amplitude inputs should be applied to examine the friction but the amplitude range must not exceed the linearity range of the system [10].

After the selection of suitable inputs, the response of the system is observed and the output data is collected. Collected input data and output data sequences are represented as column vectors to be presented to the System Identification Toolbox of MATLAB.

2.1.2 Identification of Model Structure

There are different types of methods for obtaining a model. The experimental modeling method does not need a detailed knowledge about the system; it is used to obtain a model without any knowledge of the system dynamics.

There are various types of system identification models, and by testing different types, a good choice of a model structure could be made. Among these models one may mention Autoregressive (AR), Autoregressive with Exogenous Input (ARX), Autoregressive Moving Average with Exogenous Input (ARMAX), Box-Jenkins (BJ) and Output-Error (OE) models, which are in general called parametric models or models with output feedback. The detailed information about parametric models is given in [12]. The parametric model part of this study is based on this reference.

The systems can be considered as linear can be modeled as

$$y(k) = G(q)u(k) + H(q)e(k) \tag{2.1}$$

The equation given above is shown in Figure 2-4. This is a general linear model view for parametric model structures that includes the stochastic structure and the dynamics of the system. It is obvious that, the suitable model structure should depend on the system dynamics and also noise.

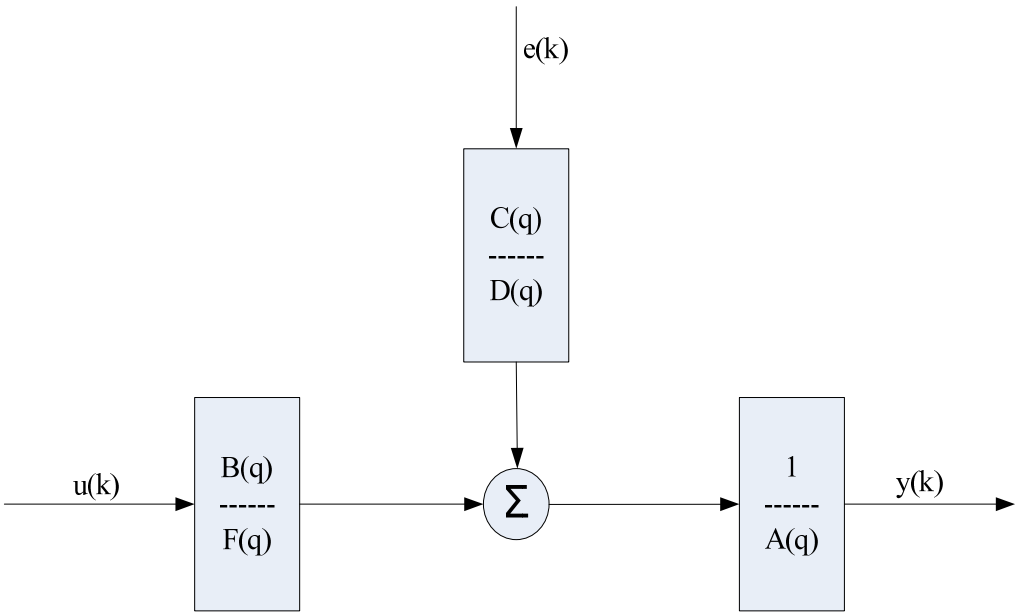


Figure 2-4 General-Linear Model Structure

As it was mentioned before, in general there are different model types used for system identification such as; AR, ARX, ARMAX, BJ and OE which are in general a subsets of parametric models. These models are obtained just by setting one of or some of the A(q), B(q), C(q) or D(q) polynomials equal to 1. These models are detailed in the following sections.

Here is a short summary of models and their characteristics.

2.1.2.1 AR Model

This model structure generates a model in which the outputs dependent on only the previous outputs. The system inputs or disturbances are not taken into consideration and not used by this model type. So this model can be used just for signals not for systems. This model type is generally used for linear prediction

coding or in general for time series analyses. Here is the figure that represents the AR Model;

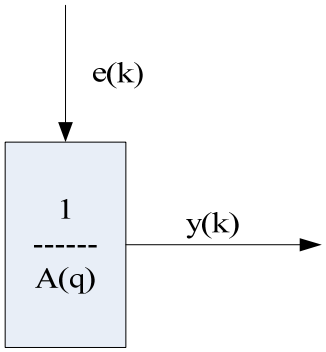


Figure 2-5 AR Model Structure

The model equation is:

$$y(k) = \frac{1}{A(q)} e(k) \tag{2.2}$$

2.1.2.2 ARX Model

This model type is frequently used. The popularity of this model depends on the verification of this model for the global minimum solution of the loss function. If the model order is high then using this model is a good idea. There is a disadvantage of the ARX model; the disturbances on the system is a part of the system dynamics. Here is the figure that represents the ARX Model;

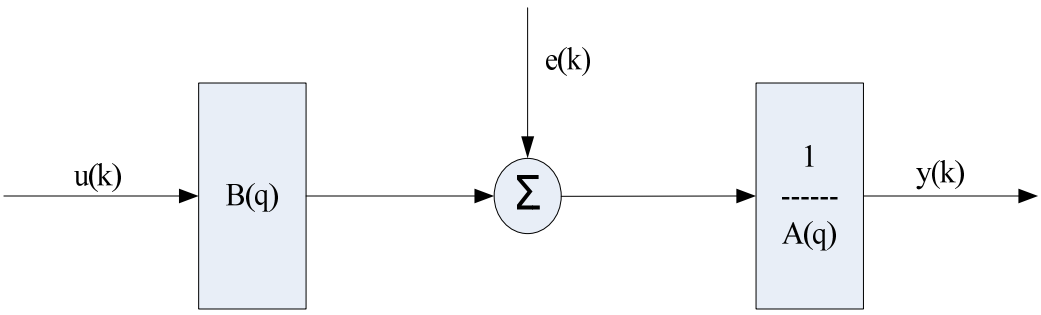


Figure 2-6 ARX Model Structure

The model equation is

$$y(k) = \frac{B(q)}{A(q)}u(k) + \frac{1}{A(q)}e(k) \tag{2.3}$$

2.1.2.3 ARMAX Model

This model is similar to ARX model, but it includes the disturbance dynamics. ARMAX models are helpful for the cases when the disturbances enter the process in the early steps, for example noise in the input. Here is the figure that represents the ARMAX Model;

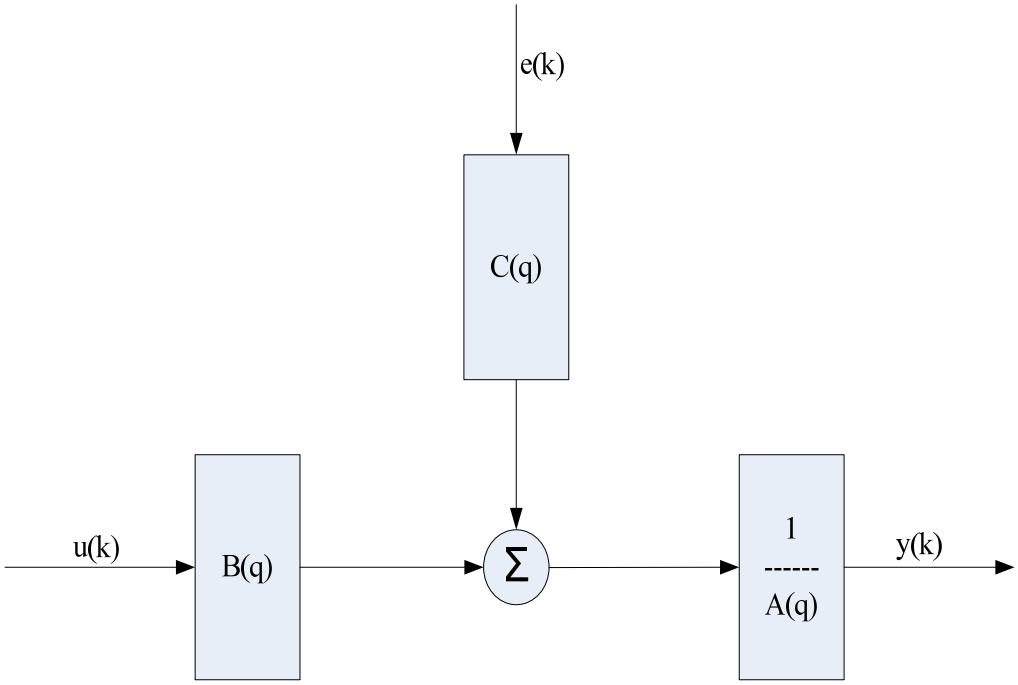


Figure 2-7 ARMAX Model Structure

The model equation is

$$y(k) = \frac{B(q)}{A(q)}u(k) + \frac{C(q)}{A(q)}e(k) \tag{2.4}$$

2.1.2.4 BJ Model

In this model, both the complete system model and the disturbance on the system are modeled separately. These models are helpful for the cases when the disturbances enter the process in the later steps, for example noise on the output at measurement stage. Here is the figure that represents the Box-Jenkins Model;

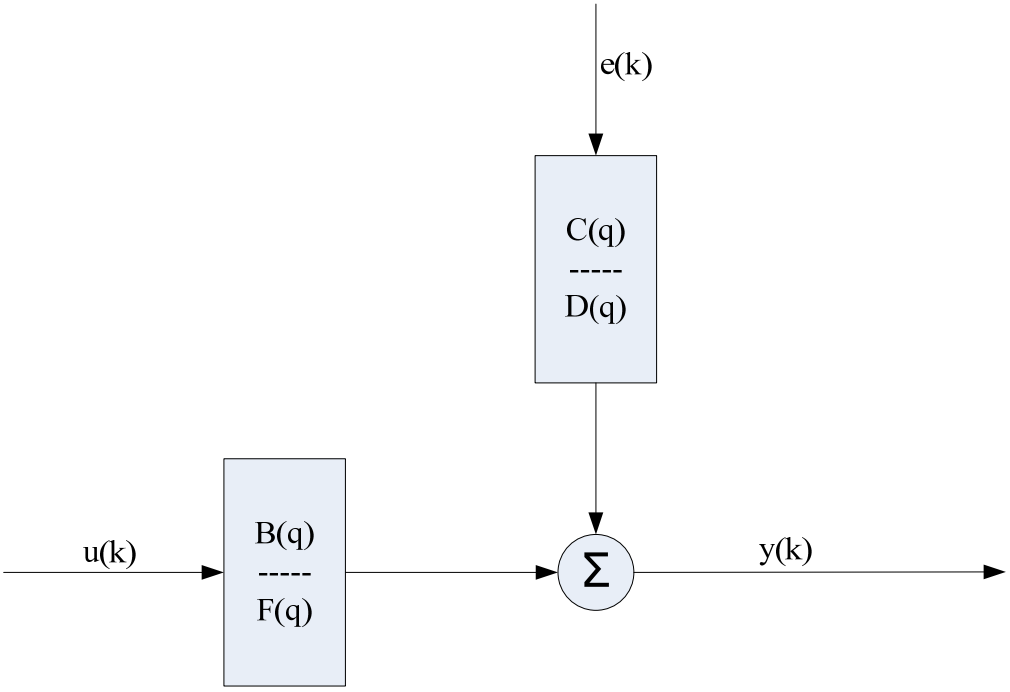


Figure 2-8 Box-Jenkins Model Structure

The model equation is

$$y(k) = \frac{B(q)}{F(q)}u(k) + \frac{C(q)}{D(q)}e(k) \tag{2.5}$$

2.1.2.5 OE Model

In this model, only the complete system dynamics is considered. The disturbance part is not modeled. Here is the figure that represents the Output-Error Model;

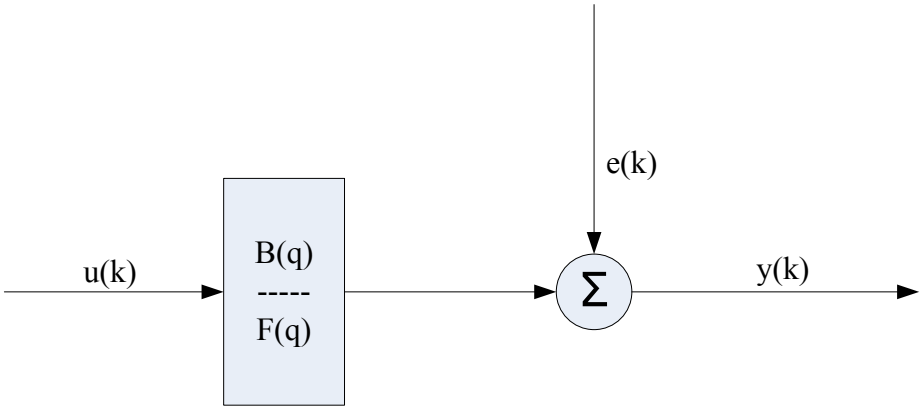


Figure 2-9 OE Model Structure

One of the advantages of using this model is that other model types can get stuck at wrong local minima for higher order polynomial choices for F(q), B(q), etc. The computational effort to determine the system parameters will usually be very high in other models. This model directly separates the error. It is also helpful if robust controllers are planned to be used as controller of the system. Also, as emphasized in
Figure 2-10, if the noise characteristic of the system is white noise, then this model is more suitable rather than AR, ARX, ARMAX and BJ models.

The model equation is:

$$y(k) = \frac{B(q)}{F(q)}u(k) + e(k) \tag{2.6}$$

The parameters of the OE model are

$$nb: \qquad B(q) = b_1 + b_2 q^{-1} + b_3 q^{-2} + \ldots + b_{nb} q^{-nb-1} \tag{2.7}$$

$$nf: \qquad F(q) = 1 + f_1 q^{-1} + f_2 q^{-2} + \ldots + f_{nf} q^{-nf} \tag{2.8}$$

Here nb and nf represent the order of OE model.

For output error models F(q) is used instead of A(q) to emphasize the difference in noise characteristics.

Estimation of these parameters depends on the prediction error method. This method is mentioned in section 2.1.3.1. The details about OE model are given in [9]. For the detailed models above the following classifications are made:

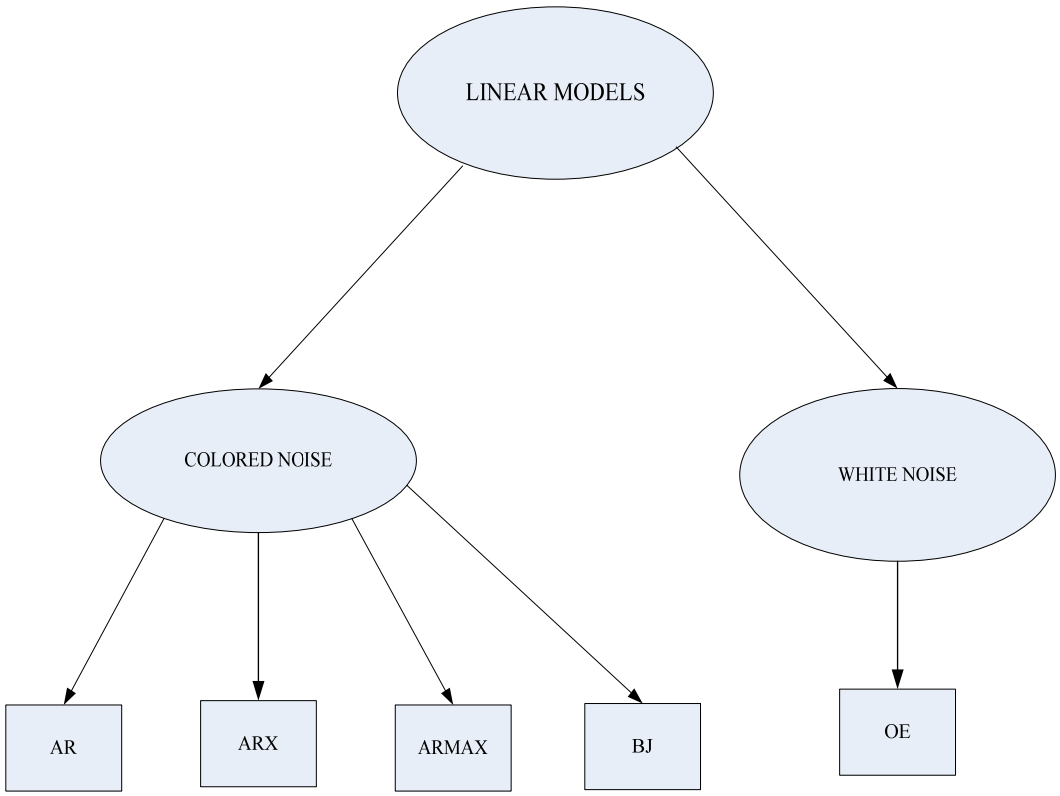


Figure 2-10 Classification According to Noise Properties

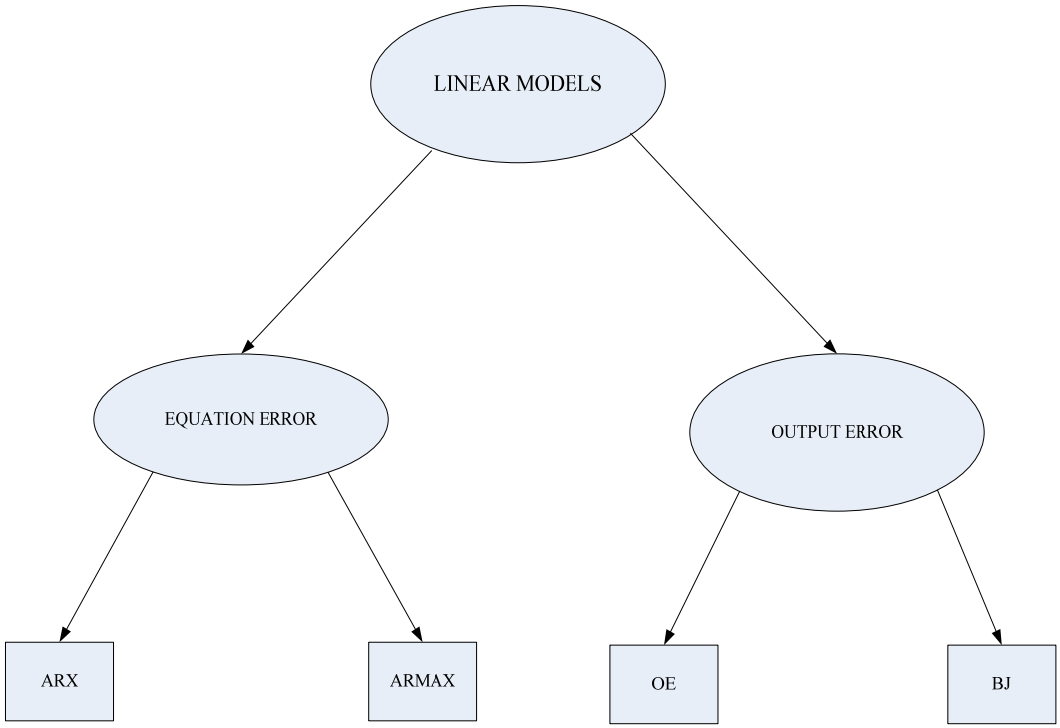


Figure 2-11 Classification According to Equation Error and Output Error

2.1.3 Model Parameter Estimation

As mentioned before according to the characteristics of the system a suitable model structure is chosen, then the next stage in identifying the system is determining the model order to estimate the parameters of the polynomials in the model. There are three basic methods used for parameter estimation which may differ for each parametric model type. For instance; Prediction Error method can be used for AR, ARX, ARMAX, BJ and OE model types. Least square is suitable for AR and ARX and instrumental variable method is suitable for ARMAX model.

Via MATLAB System Identification Toolbox, the transfer function models can be estimated by using the appropriate model parameter estimation method for the predetermined model structure from the input/output data set.

2.1.3.1 Prediction Error Methods, Parameter Estimation

The basic idea behind this method is that, it is the difference between the process output and predicted output performed in the first step. It is typically used by following the steps shown below.

1. Start the procedure by examining nonparametric frequency response function to get the number of resonance peaks. This helps us to get more information on the model order. In general, as a rule of thumb, in the magnitude response, the number of peaks equals to half of the order of $A(q)F(q)$ which are given in Figure 2-4.

2. Get an acceptable estimate of delay in an ARX model by testing or correlation analysis. Next, determine the delay which best fits according to prediction errors.

3. Determine the best fit ARX model and its order with this delay.

4. By following the steps below, an ARX model is obtained that is composed of disturbances and system dynamics. This means the model order is high. Reduce the model order with the help of pole/zero maps and canceling some of them. The resulting model order is the starting point of the OE or ARMAX or BJ models. The initial parameters of these models and the disturbance parameters are obtained.

5. After all, it is expected to obtain a suitable model, but if it fails, then do perform extra search to find whether there are extra signals that can influence the output. After all, start the procedure again [12].

The details about Prediction Error methods are given in [9].

If a successful result could not be obtained then it should be focused on the system to make an observation on non-linear elements and try to modify the previously obtained data set.

According to prediction error view, for the model to fit the data better, the order of the model is increased. This increase means more degrees of freedom so a better fit can be attained with the cost of computation time. In our case, the limitations of the highest order of the model are determined and the modeling part is done with the help of MATLAB programs.

2.1.4 Model Validation

There are different validation procedures. According to the system, different validation types could be chosen such as pole-zero plots, bode plots, comparison of real data with the predicted equation in time domain and some other simulations. The same input used in the experiment part is applied and the output of the estimated model is observed and compared with the results of the real system output data. If the experiment results are similar to the estimated part then everything is correct. Also bode-plots and pole-zero locations of the estimated model are checked and controlled again to verify the conclusion.

2.1.5 Conclusion

There are various model types to model a system but the choice of the best model type depends on the system dynamics and disturbance. These are considered as the main guidance for us to choose the appropriate model.

All these model types are helpful and can model most of the dynamical systems without any information about the parameters of the real system. For our case, from the different types of methodologies, the OE model is chosen as the most suitable one because it provides more flexibility if the stochastic dynamics are more effective in modeling. In our system, there are uncertainties and disturbances so the stochastic properties of the model is very important. Moreover, we have tried different model types for our case and as expected, OE model emerged as the most suitable one because it is simpler and it was successful than the others. Also, this methodology is an input/output principle method. Unlike to ARX and ARMAX models, OE model assumes that noise affects the

system at the output; that is why it is called output error, not equation error. These output error models are more realistic than equation errors so widely used in real life applications. One more point is that, it was assumed that the noise in our system is Gaussian white noise. As a result of the noise assumption, it was expected that OE model would be more successful than the remaining ones. That is why OE is chosen as the main methodology and the estimation of the parameters are obtained by using prediction error method. According to this approach, the linear system model of our real system is obtained by using MATLAB System Identification Toolbox.

In the light of this information, the responses of real system and the linear model of the system are given in section 3.1.

CHAPTER 3

MODELING OF THE SYSTEM

The system that is planned to be modeled has some parameters that are not known. Also there isn't enough information about the system. Therefore the black-box modeling is the best type of modeling for our case. So by using this technique, the data is collected for the azimuth position. All these functions and modeling of the system are detailed throughout this chapter. The way of data collection and collected data set results, the system identification model used and the obtained transfer function of the system, comparison between the model output and the actual output of the system are all given in this chapter.

3.1 Modeling of our System

To model our system, some information about the system and the design of the hardware part that is used to supply the communication between the user and the system has to be analyzed carefully. The details about the system and the designed hardware are given in Appendix A. With the help of given information in Appendix A, the modeling of our system starts by forming the figure shown in Figure 2-2. In this figure, it is indicated that the input should be applied to the system for azimuth positioning and the corresponding noisy output should be recorded. General closed loop system view for data collection and modeling is shown in Figure 3-1.

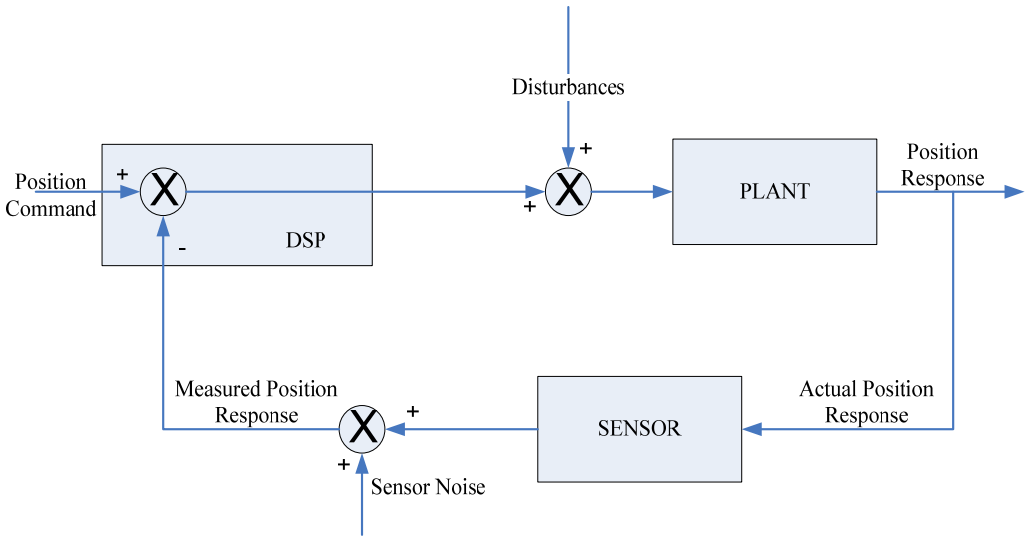


Figure 3-1 System Closed Loop Modeling View

As it shown in Figure 3-1, position command which comes from the user and measured position response, which has noise in it, are collected inside the DSP and sent to the plant by re-arranging these data. Sensors noise, disturbances affecting the system both in the input and output stages and also the system response are collected together when obtaining the plant model. So the plant part is the part that includes everything outside the DSP. Except the controller part which is the part that is going to be formed in the DSP, all the remaining parts are taken as part of the model.

3.1.1 Data Collection

For our case the plant part is taken as the part outside the DSP as shown in Figure 3-2, which shows that the controller according to this plant model is going to be designed inside the DSP. Considering everything outside the DSP is a reasonable approach to obtain the model and it minimizes the uncertainties.

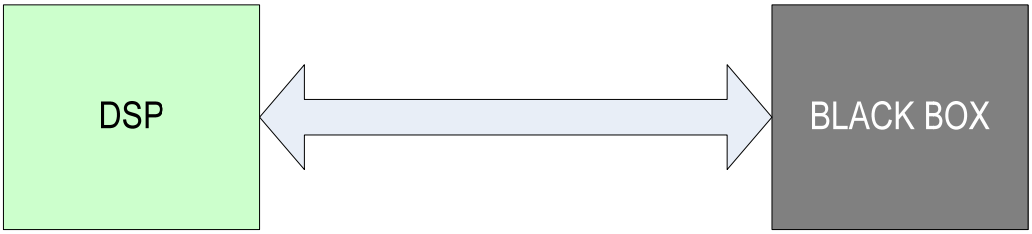


Figure 3-2 Black Box and DSP

According to this approach, a known input is applied, the output is observed and the input/output relationship is examined. This is done for different frequencies. Therefore the system response can be obtained in a frequency range and then the bode-plot is obtained. As shown in Appendix A, the applied input in the DSP part is the digitized part of the actual input, and also the obtained output is also digitized. That is why the applied input data and the obtained response at the output are provided to the MATLAB System Identification Toolbox after passing all these data through a suitable transformation. This transformation is the resolver equation. Since the measured response and the position data send are in the resolver format according to the system specification of the turret system, as mentioned in Appendix A, the resolver equation has been used. This transformation is inserted after the digitized data passes through Discrete to Resolver Converter (DRC) part also before the Resolver to Discrete Converter (RDC) part where the digitized data is obtained. This equation is shown below.

The data of the input to the Turret Control Unit and the data that comes to our Interface Unit is of the form

$$V \times \sin \omega t \times \sin \theta \tag{3.1}$$

In this expression ω is the carrier frequency which is 400Hz and V is the amplitude, $6.8V_{rms}$ and θ is the angle that represents the position of the system. That is why the actual expression is

$$6.8V_{rms} \times \sin 400t \times \sin \theta \tag{3.2}$$

For example, in our case to calculate the digitized data, assume that the position of the system is at 5º which produces

$$6.8V_{rms} \times \sin 400t \times \sin 5 = 0.5926V_{rms} \times \sin 400t \tag{3.3}$$

Here $\sin 400t$ is the carrier part. This is not important and not used by DSP because it does not include any information about the position data. That is why the magnitude that enters the RDC part $0.5926V_{rms} = 0.8381V$ is important. Its' decimal formatted form is 0.8381 = 38E (hex) = 910 (decimal). The decimal value is the value that is seen from the DSP part of the system when the turret is at the 5º of position. For 90º degrees the digitized value is 4000(hex) = 16384 (decimal).

The overall formula that is used before inserting all the obtained data to the MATLAB is

$$\sin(\frac{\mathrm{X} \times 90}{16384}) \times 6.8 \times \sqrt{2} \tag{3.4}$$

Here X represents the angle data that comes from and goes to the DSP part. According to this equation all the data set included inside the DSP is converted into the voltage scale.

By using the above formulation, for a predetermined frequency range some experiments are made. This range is determined according to the turret system specifications. Below are the examples of an input that is applied to the system and the corresponding responses of our system to that input for a specific time period. All the dashed lines show the input and all the solid lines show the output. Also the frequencies of the applied sinusoidal input and the maximum angle of scanning range of the system are written on the title of each graph.

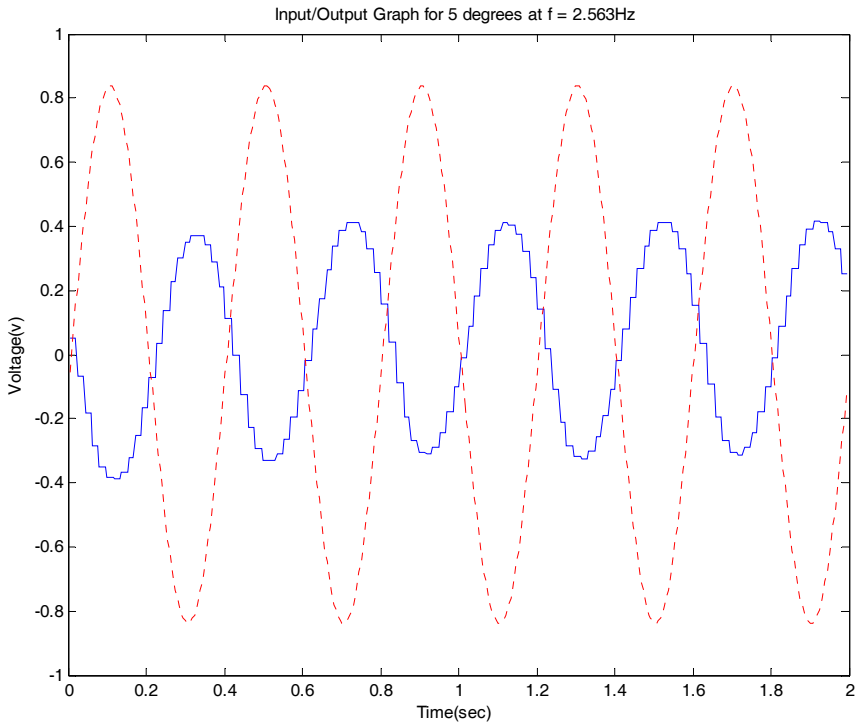


Figure 3-3 Example of Input/Output Graph for 5 Degrees at f = 2.563Hz

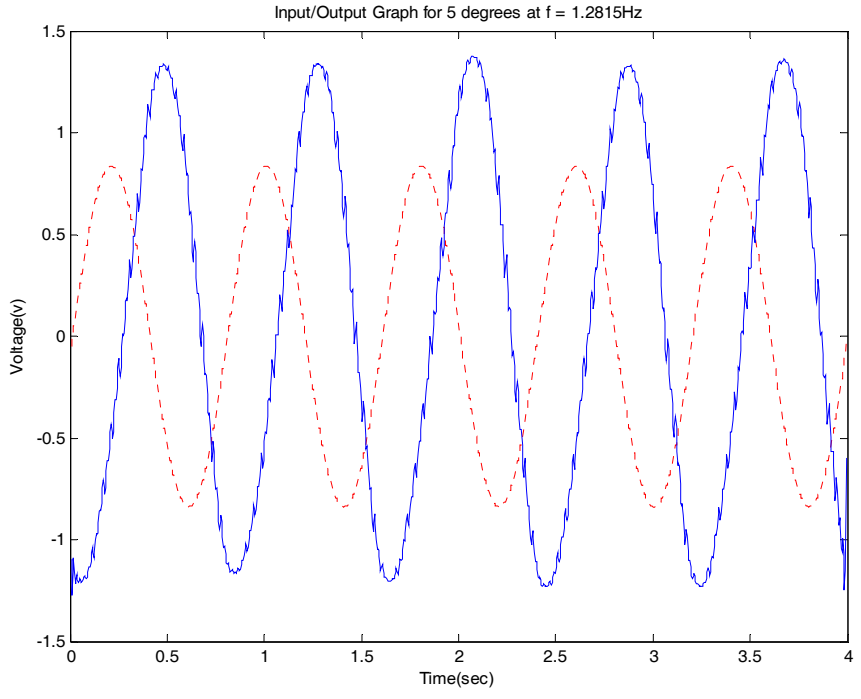


Figure 3-4 Example of Input/Output Graph for 5 Degrees at f = 1.2815Hz

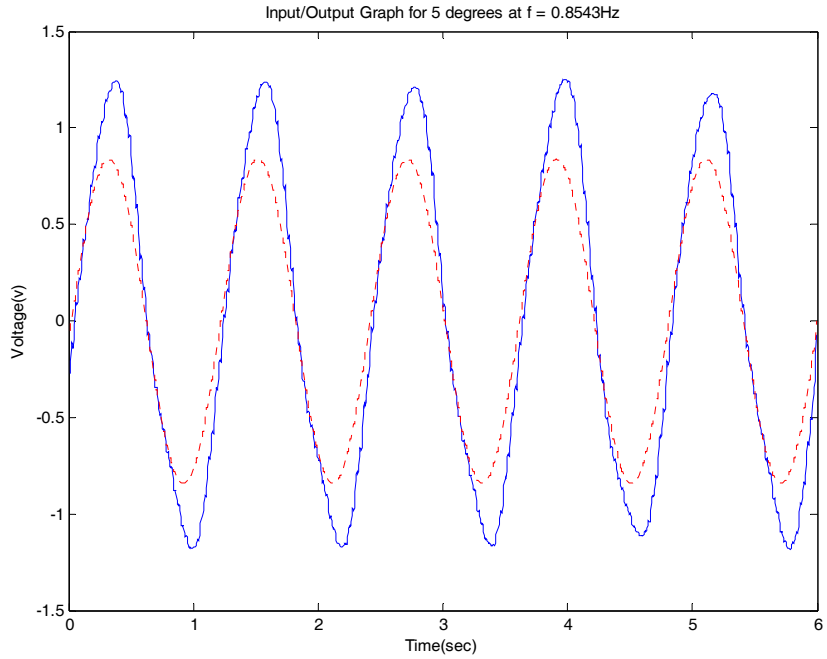


Figure 3-5 Example of Input/Output Graph for 5 Degrees at f = 0.8543Hz

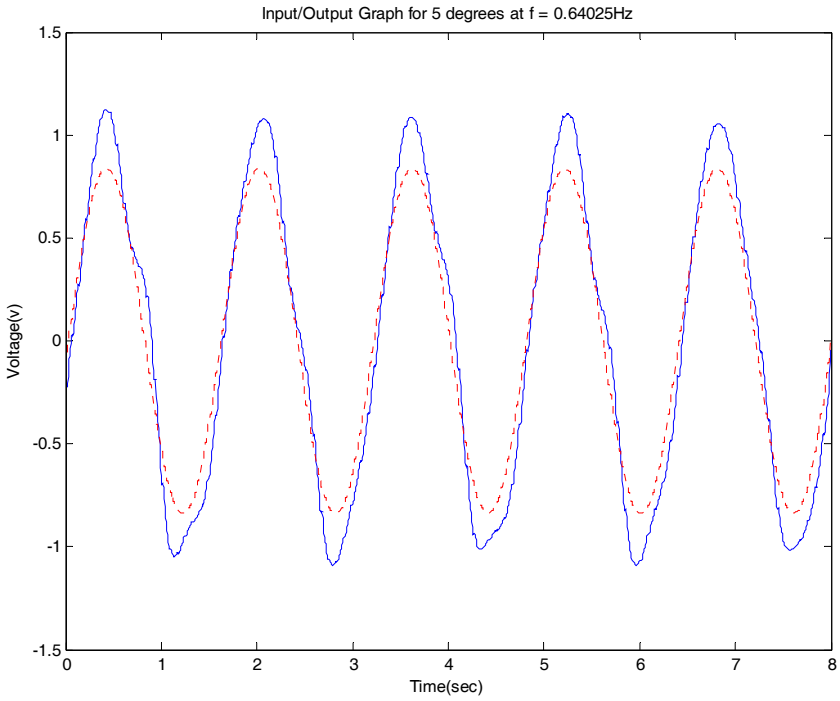


Figure 3-6 Example of Input/Output Graph for 5 Degrees at f = 0.64025Hz

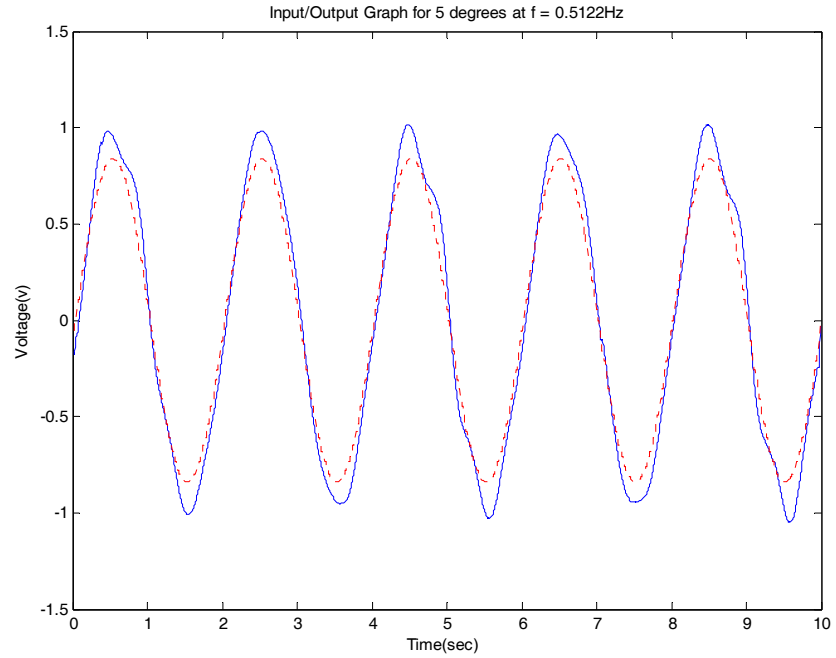


Figure 3-7 Example of Input/Output Graph for 5 Degrees at f = 0.5122Hz

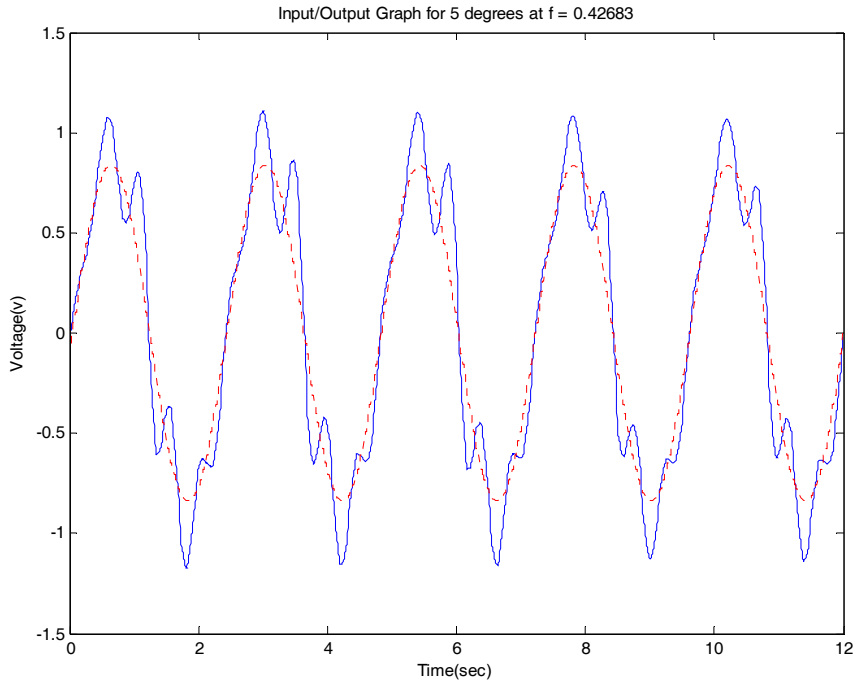


Figure 3-8 Example of Input/Output Graph for 5 Degrees at f = 0.42683Hz

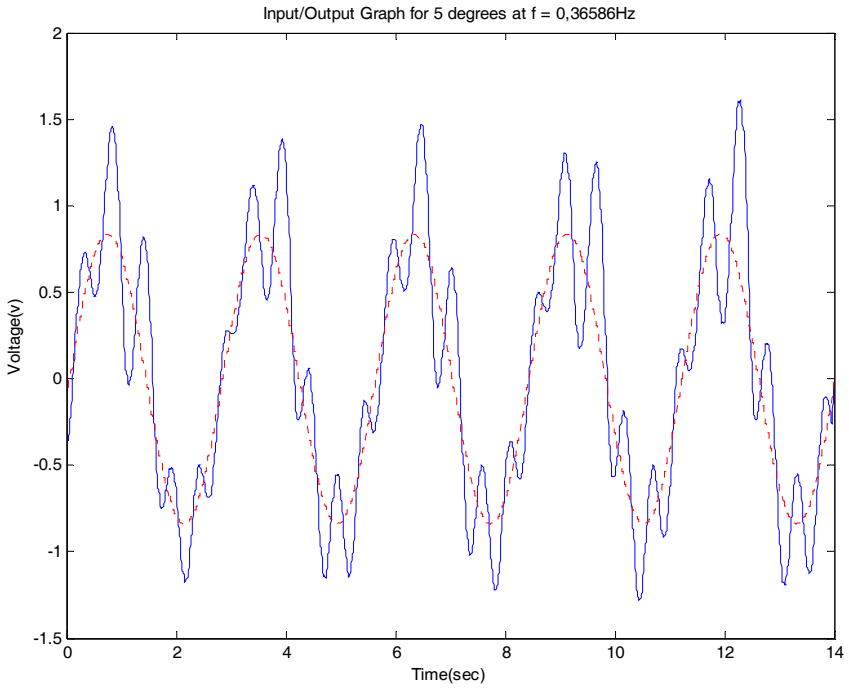


Figure 3-9 Example of Applied Input for 5 Degrees at f = 0.36586Hz

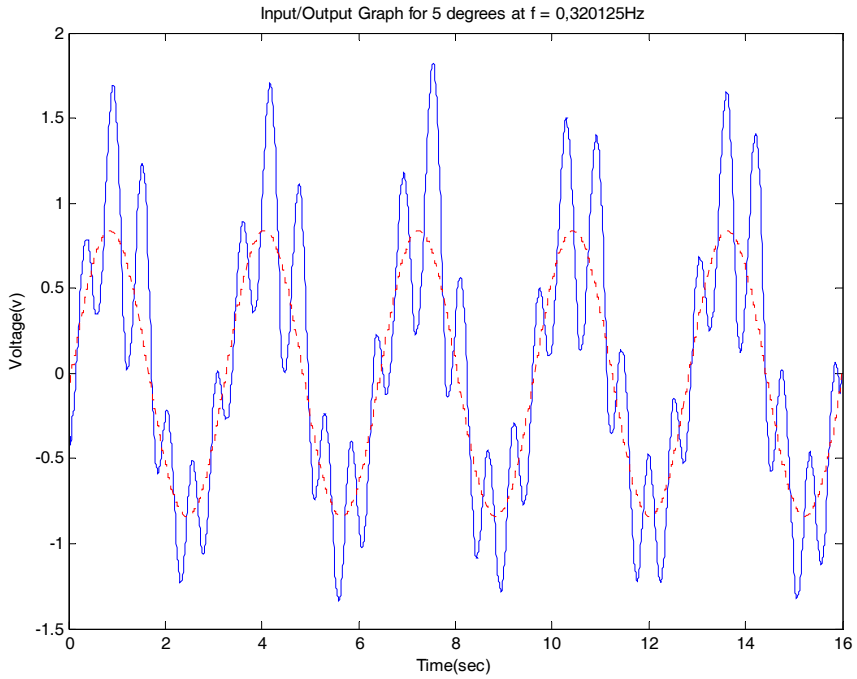


Figure 3-10 Example of Applied Input for 5 Degrees at f = 0.320125Hz

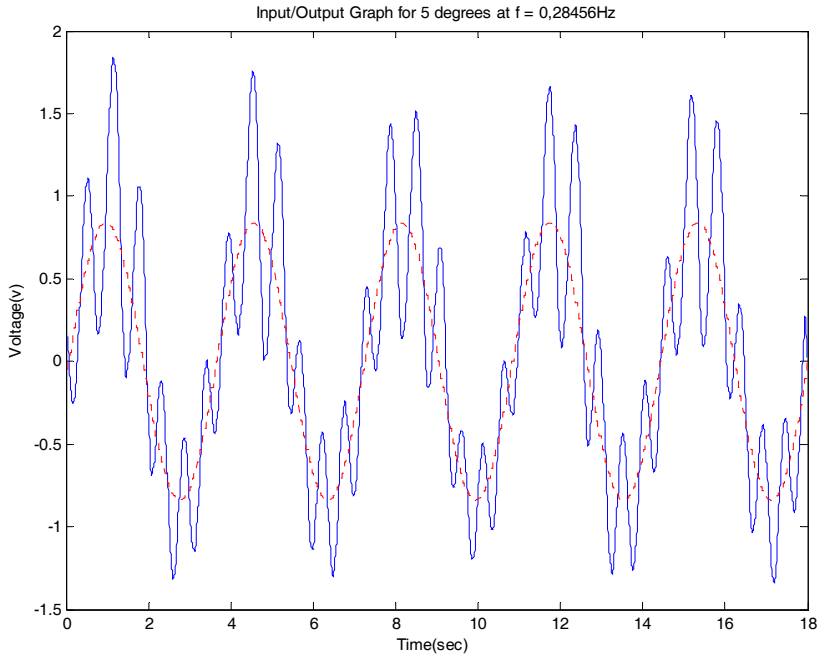


Figure 3-11 Example of Applied Input for 5 Degrees at f = 0.28456Hz

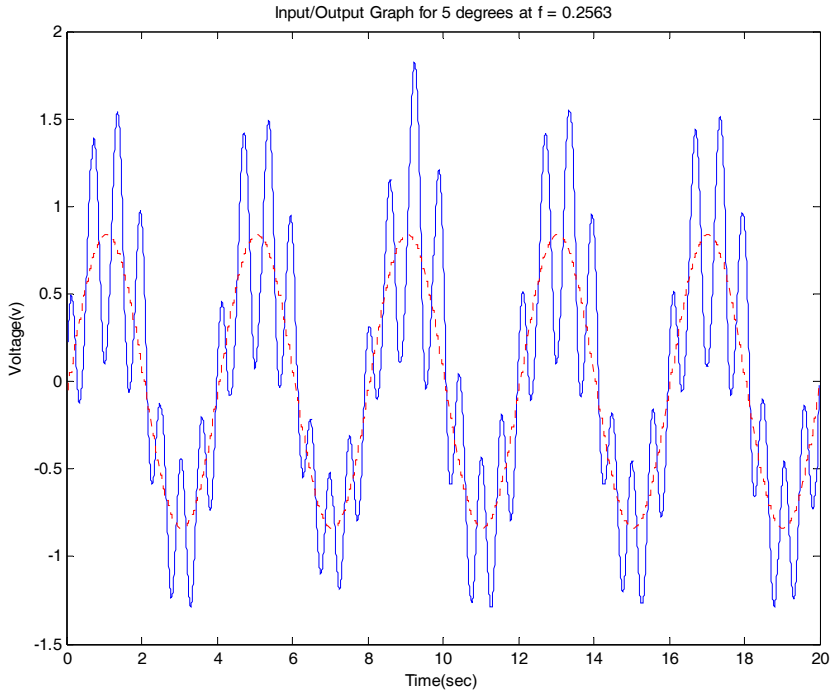


Figure 3-12 Example of Applied Input for 5 Degrees at f = 0.2563Hz

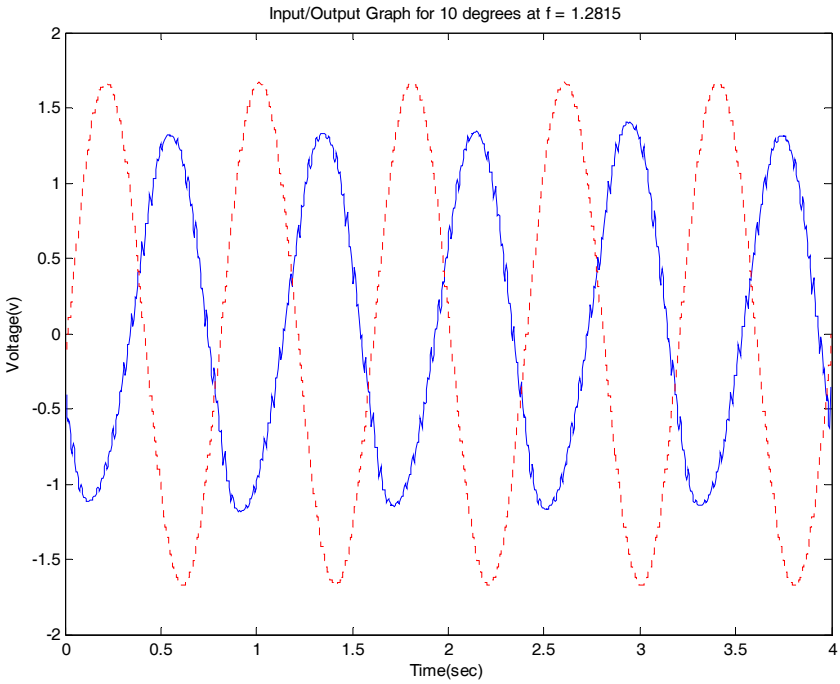


Figure 3-13 Example of Input/Output Graph for 10 Degrees at f = 1.2815Hz

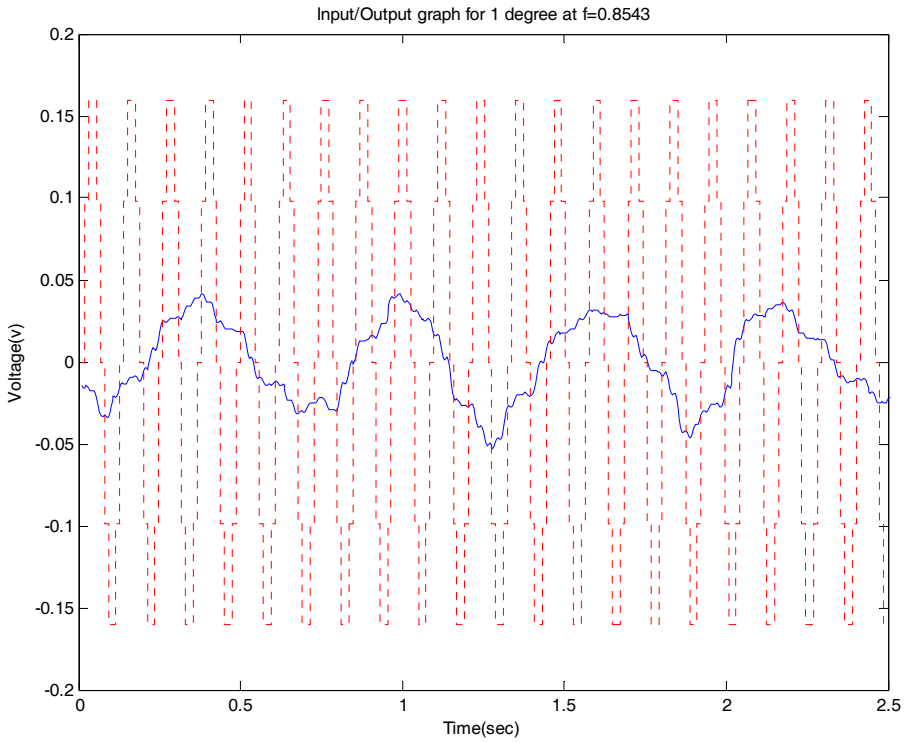


Figure 3-14 Example of Input/Output Graph for 1 Degree at f = 0.8543Hz

All the data set for applied inputs and obtained outputs from the system are taken at 0.2˚, 1˚, 5˚, 10˚ positions. But also 20˚, 30˚, 40˚, 50˚, 60˚, 70˚, 80˚, 90˚ positions data have been obtained as well. However they are not used in the modeling, because they are linear multiples of 5˚, 10˚ data and by adding them to calculations, the mathematical modeling part gets too complicated and it becomes impossible to obtain a model.

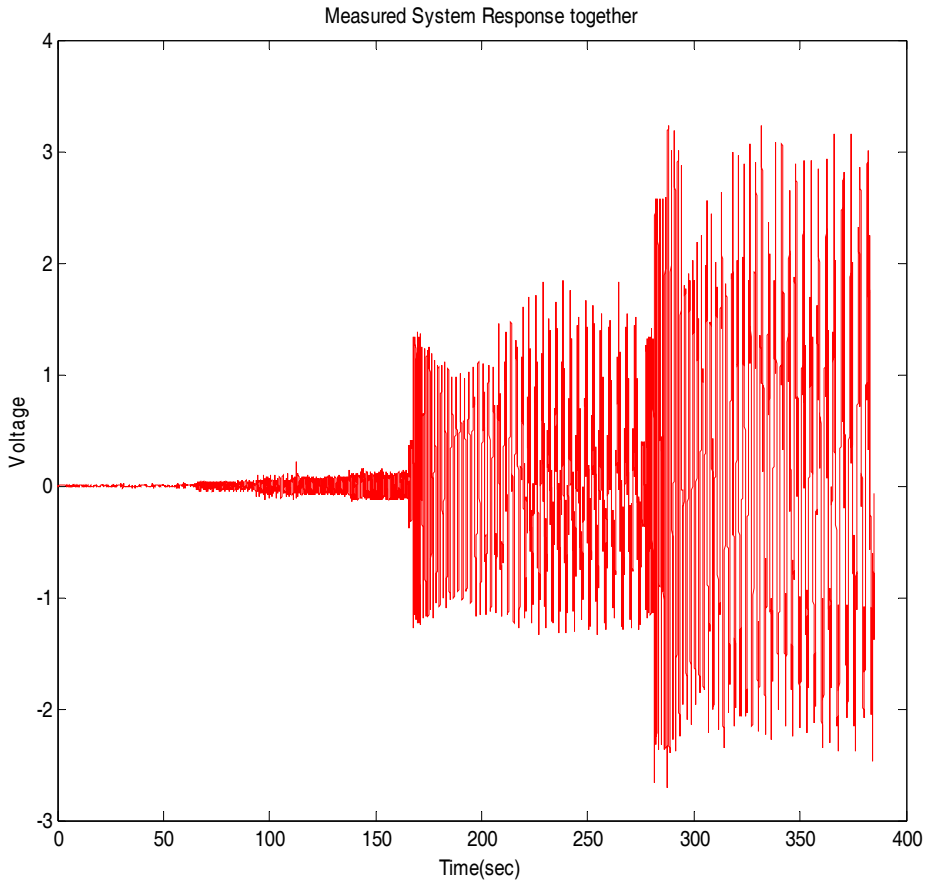
From the graphs, it is seen that there are undesired perturbations on the data which deforms the sinusoid. This can be because of the turrets actual characteristics, modeling errors, noise on the whole system or there is something missing in the setup. With a deeper analysis of the real system, one observation is that this deformation is most probably a result of the shielding problem of the whole system. By making some improvements on the cables and at each connection points, a better performance is obtained, and the response of the system becomes more meaningful. It can be observed that data contains both noise and some other characteristics which can be considered as modeling errors.

Also from these figures, it can be seen that the amplitudes of the output and the input sinusoids are proportional. There is also some phase shift at the output. Actually, the constant of proportionality between the amplitudes of the output and the input, and the phase shift vary with respect to frequencies. The previous figures in this section and Figure 3-15, show the response of our system. It can be observed that, as the number of test data points increase, the frequency increases as well. But, when we want to obtain the model, we get the system I/O graph for each frequency and collect all the data. The Bode plot according to the collected data can be constructed.

When we put both input and the output together, it is obvious from previous figures and from Figure 3-15 that there is a magnitude and phase difference between the input and the output.

Various inputs are applied to the system for different time periods and various outputs are obtained. All the applied inputs and the outputs are converted from digital data to voltage data by using equation (3.4) and also the time periods for each data packet are stored. Then, to obtain a linear model, all the data is re-sampled to 0.0039 seconds. After the re-sampling process, the following Figure 3-15 is obtained which covers all the applied inputs and the obtained outputs.

First graph represents the measured system response and second graph represents the applied input.



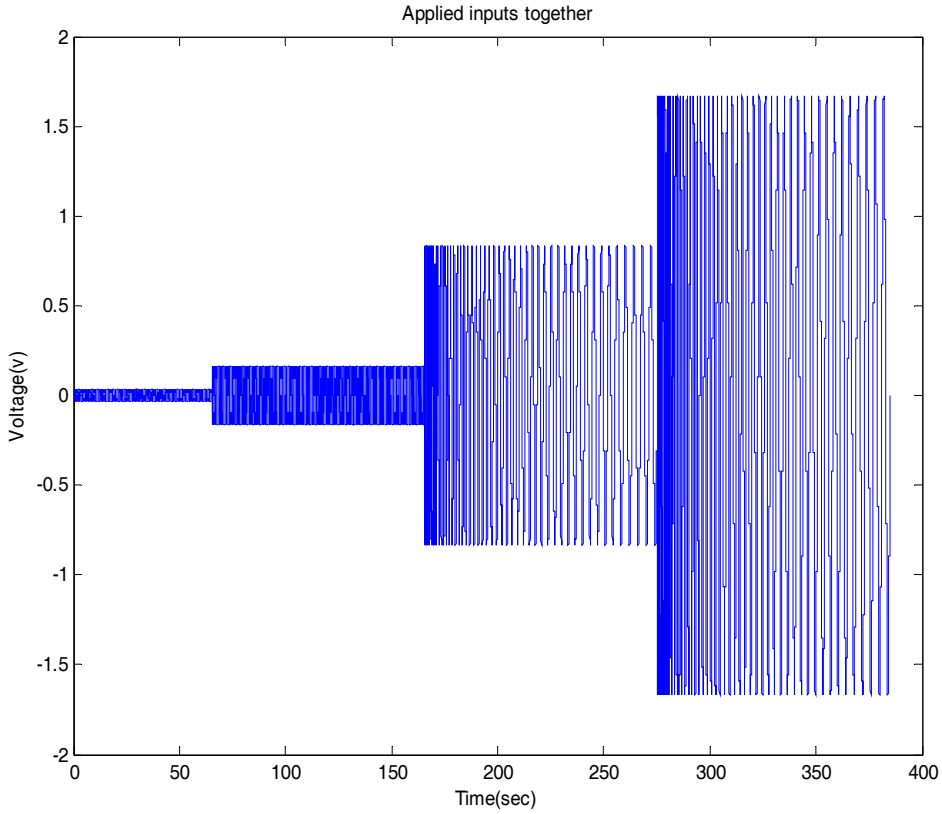


Figure 3-15 Whole Data Set of Output and Input of the System

3.1.2 System Identification

After obtaining the data mentioned in the previous section, system identification process starts by using the system identification tool box of MATLAB to get a linear model of the system.

There are, obviously various different linear models, but as mentioned earlier the most useful one for our case is OE. The details about this model type were given in section 2.1.2.5. Please refer to equation (2.6). The reason that OE model is used is that, in series of experiments, the other models did not perform well as OE has performed. Moreover, this model is easier as well and did not get stuck to a local optimum during the identification process. By using the OE model, the following transfer function is obtained. The equations of OE model are already given in

section 2.1.2.5. For various values of *B*, *F* and *e*, too many equations are obtained and from these equations, the one which has the best bode plot approach, minimum error in matching with the actual output and least equation coefficient is chosen. The name given to the chosen one is OE881. Transfer function of OE881 is as follows.

$$H(z)=\frac{0.4647z^{7}-0.3716z^{6}-0.2874z^{5}-0.5949z^{4}+0.6722z^{3}+0.2713z^{2}+0.2508z-0.4046}{z^{8}-0.8416z^{7}-0.561z^{6}-1.142z^{5}+1.577z^{4}+0.3482z^{3}+0.2944z^{2}-0.9506z^{1}+0.2763} \quad (3.5)$$

Transfer function from error input to output is 0.3746. Sampling time is 0.00399 seconds.

Also the graphs below show the actual measured response of the system and the obtained model response by using the OE model. All the dashed lines show the simulated model's output and all the solid lines show the measured actual system response at different magnitudes of the input.

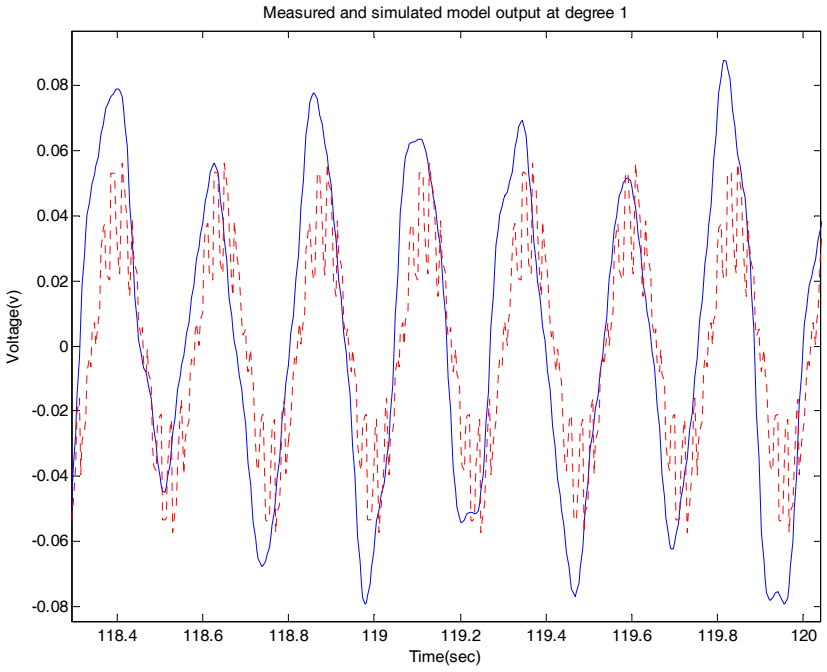


Figure 3-16 Example of Modeled vs. Actual System Response Graph for 1°

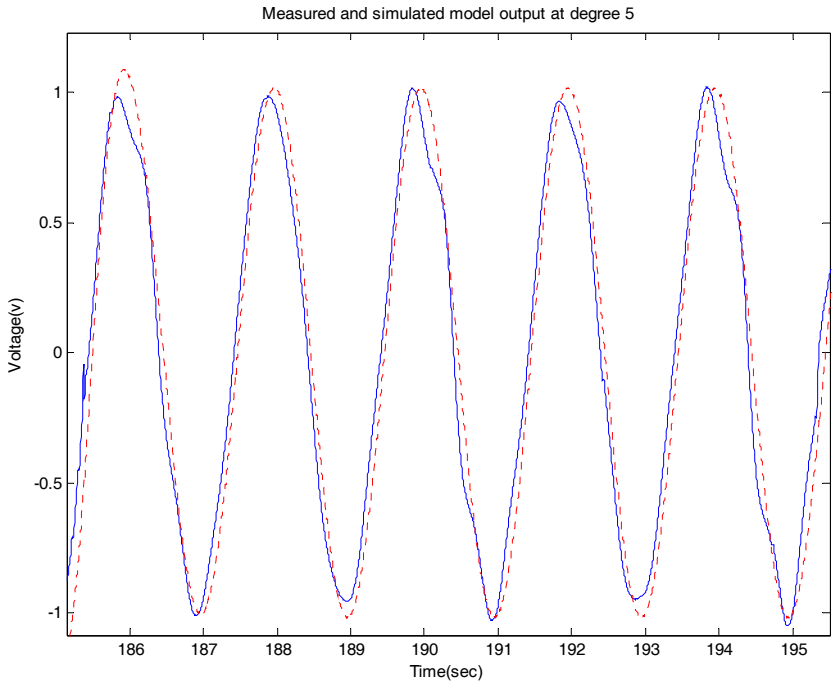


Figure 3-17 Example of Modeled vs. Actual System Response Graph for 5°

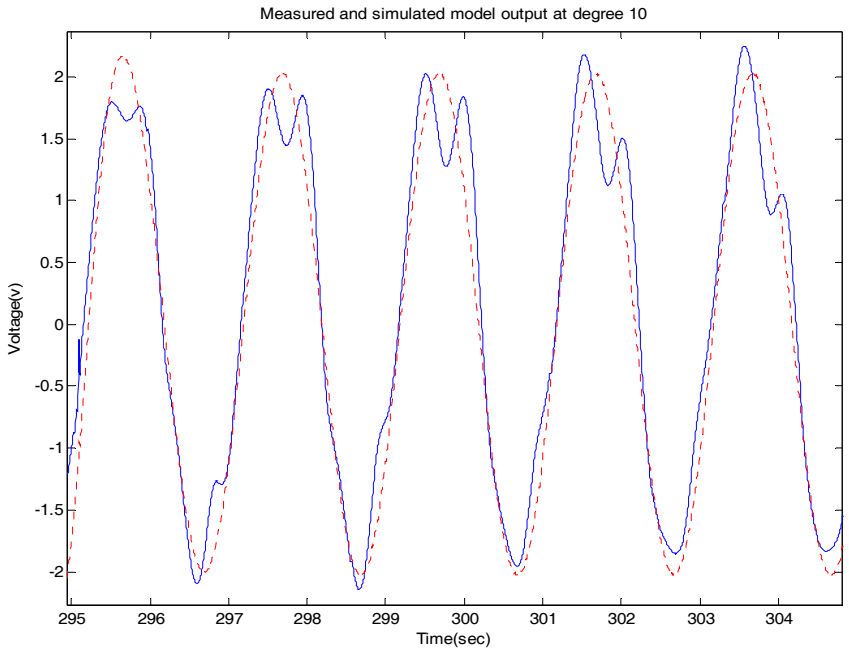


Figure 3-18 Example of Modeled vs. Actual System Response Graph for 10°

The closed loop bode plot of the obtained model OE881 is shown below.

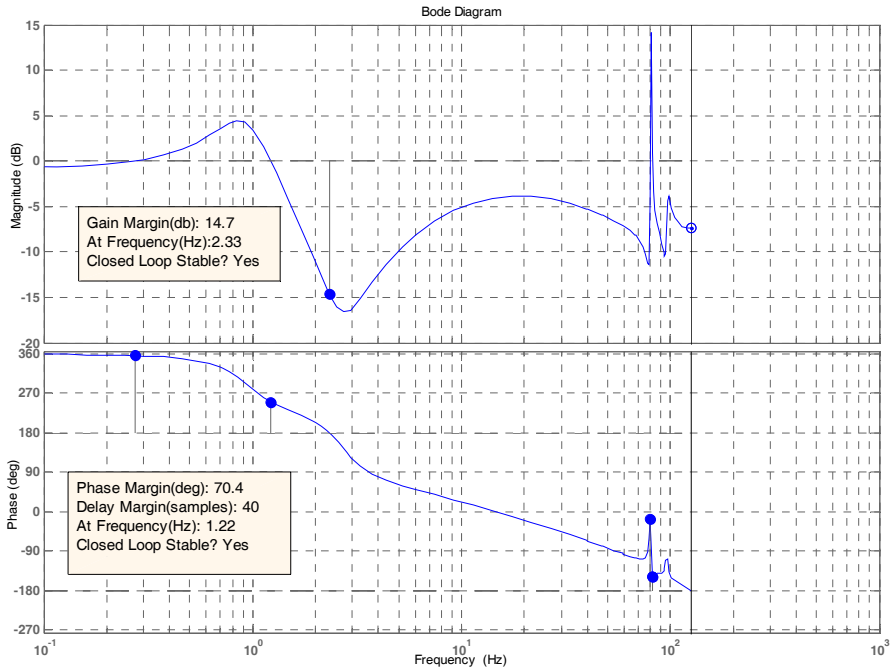


Figure 3-19 Bode Plot of the Obtained Model

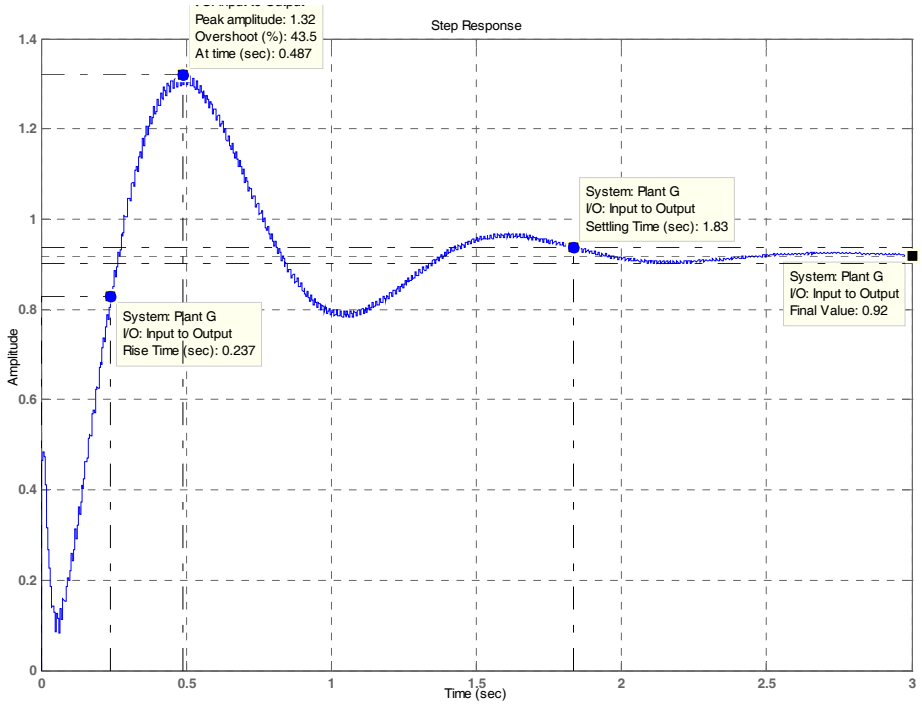


Figure 3-20 Step Response of the Obtained Model

As it is seen from the plots, some reformations are required. This can be done by designing a controller for this plant. This controller part is explained in the following chapter.

3.1.3 Conclusion

As it is seen from the graphs, the applied input scans as mentioned earlier different magnitudes and also different frequencies. Actually the applied input also scans the range that we are not going to use; especially the sinusoidal inputs at high frequencies. Because the maximum system response is 80 degrees per second and the desired range is less than 20 degrees per second. There is an applied sinusoidal input with maximum angle of motion given in degrees. For example, at 10 degrees means, the motion is from 0 degrees to 10 degrees first, then from 10 degrees to -10 degrees and finally again to 0 degrees. This means that 40 degrees of motion. At f = 2.5630 for 10 degrees means 102.52 degrees of motion per second. This is too high for system specifications. In the light of this information, the collected data set is at 0.2 degrees from f = 25.63 to f = 5.122Hz, at 1 degrees from f = 8.543 to f = 3.6586Hz, at 5 degrees from f = 2.5630 to f = 0.2563Hz, and at 10 degrees from f = 2.5630 to f = 0.2563Hz. Moreover, at very small frequencies, because there are too many white noise sources in the system, the output value is high. While determining the best model for our case, these are taken into consideration.

One more important issue while obtaining the model is that not only the data set for 0.2˚, 1˚, 5˚, 10˚ degrees but also 20˚, 30˚, 40˚, 50˚, 60˚, 70˚, 80˚, 90˚ degrees are taken but with a small examination. It is seen that the response of the system to the degrees of 20, 30, 40, 50, 60, 70, 80, 90 is the linear multiples of 10 degrees. For example: the collected data set of 20 degrees data is almost equal to two times of the data set of 10 degrees. This means that system response for 5 degrees or 10 degrees can be used instead of the remaining angle sets. For this reason, the 20, 30, 40, 50, 60, 70, 80, 90 degrees data are not taken into consideration. One other reason for not using these data is that, too much data requires too much processing effort for MATLAB, and by adding these data, it

may be very hard, with today's computers to obtain the mathematical model of the system.

By using the obtained data at 0.2˚, 1˚, 5˚, 10˚ degrees, various different models are obtained by using the OE model and also some other models with the help of MATLAB system identification tool box. From these different models, it is seen that OE model gives the best performance for our system, as expected. Also by using OE model, various transfer functions are obtained, but OE881 is chosen because, degrees lower than 8 have worse matching performance and do not give similar response as the actual response of the system, for degrees higher than 8, response is almost similar to degree 8 and does not make any improvement. So the transfer function of OE881 is chosen as the mathematical model of our system.

CHAPTER 4

CONTROL METHODS

The basic idea behind the control of a turret system is to achieve a response which is not easily affected by disturbances and unmodeled system dynamics. Moreover, it is aimed to satisfy precise and rapid tracking [14]. According to these concepts robust, intelligent and adaptive controllers are generally preferred by the controller designers.

Robust controllers are mostly preferred because they are capable of operating under uncertain and variety of conditions. There are different robust methods like: H_2, LQG which is a specialized version of H_2, generalized singular linear quadratic (GSLQ), H_∞.

Adaptive intelligent control is used to determine and overcome the problems encountered by parameter variations and abnormalities which are discussed detailly in [16].

From all these types of controllers, the suitable ones are determined according to the system requirements. Therefore, for our case the requirements of our system are:

- Robustness against unmeasured system dynamics,

- Flexible to uncertainties occurred at: linearization of nonlinear elements, barrel temperature, unmodeled system parts, helicopter motion, and compliance of the system [6],
- Closed loop stability also in case of disturbances and uncertainties,
- To overcome with the modeling errors,
- Precise and rapid tracking performance.

From these requirements and a careful study on the published studies, three types of controllers are planned to be used as the controllers of our system. These are Proportional Integral (PI), IMC, LQG controllers. Especially IMC and LQG controllers are the main controllers whom we are expecting to response better because of our system specifications. Detailed information about the robust design of the weapon control system is given in reference [15].

First of all, the properties of IMC controller is detailed, and then LQG is explained through this chapter. Also PI controller is used and the details about PI controllers can be found in [32]. Finally the designed controllers and their properties are explained in Chapter 5.

4.1 Internal Model Control

General information about Internal Model Control is covered in [17]. While preparing the following information about IMC, [17] is taken as the main reference.

Control of a system takes place if you want to adjust your system to work in desired range. The desired controller can be reached if the exact model could be achieved. In order to obtain a perfect controller, the system has to be determined precisely. However because of the disturbances, measurement noises and system instabilities, the exact model can not be reached. This is the main idea behind the Internal Model Control design strategy.

IMC is a model based controller design method which is used frequently in control world because it provides robust control of the systems.

The Internal Model principle states that if the system encapsulates some of the process then the control of the system can be obtained [18].

The IMC technique is good for open-loop stable systems but for unstable ones direct use of this is impossible. Therefore, in order to apply IMC technique the process is pre-stabilized by using conventional feedback [18].

First of all, take the general open loop structure:

Figure 4-1 Open loop control

$Y(s) = G_c(s)G_p(s)I(s)$

Then for a perfect controller:

$G_c(s) = G_{pm}(s)^{-1}$ where $G_{pm}(s) = G_p(s)$: process model

This controller does not need any feedback loop and stability can be achieved if the exact process model could be obtained. However, in practice this is not possible; there is always some disturbance and robust characteristics on the process. In order to cope with this problem the following structure is taken into account called IMC.

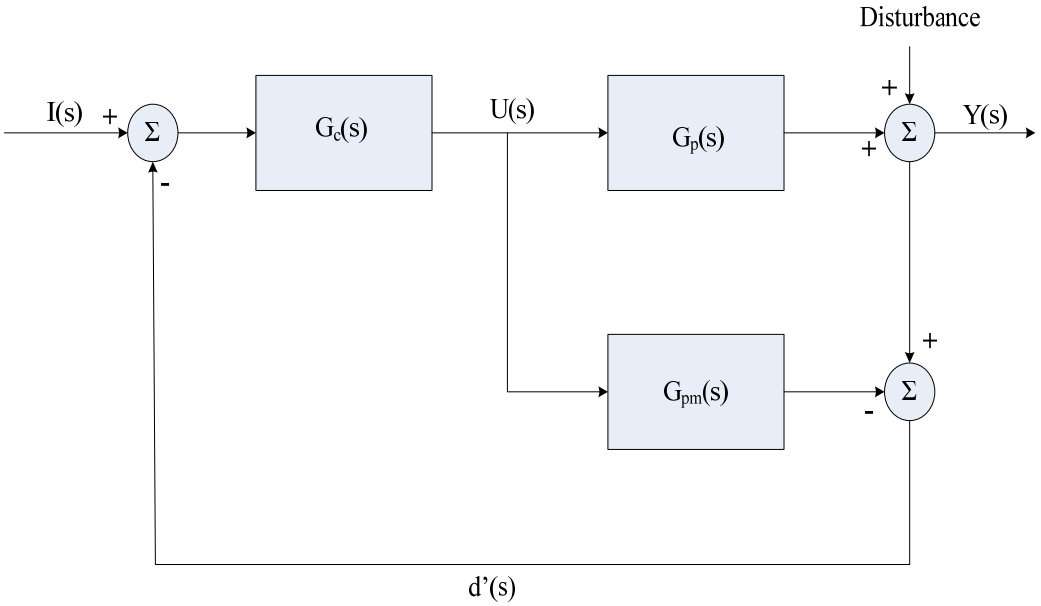


Figure 4-2 IMC

As mentioned in the upper part $G_{pm}(s) = G_p(s)$ is the ideal case. The closed loop is stable if the controller is stable because process is stable in general. From the scheme above, the following equation could be obtained.

$$d'(s) = [\, G_p(s) - G_{pm}(s)]U(s) + D(s) \tag{4.1}$$

If $G_{pm}(s) = G_p(s)$ then $d'(s)$ just represents the disturbance.

For ideal case, $G_c(s) = G_{pm}(s)^{-1}$

If the ideal case can be achieved then the disturbance effect can be observed clearly and the precautions could be taken for almost all disturbance effects.

In general the difference between the process and process model shows the mismatch in the model, uncertainties or disturbances on the system. The main reason for choosing this type of controller is the effective robustness property of the system to the uncertainties or disturbances on the system [17].

It is always very important to develop transfer functions between the disturbance and the set point inputs by re-drawing Figure 4-2 as follows:

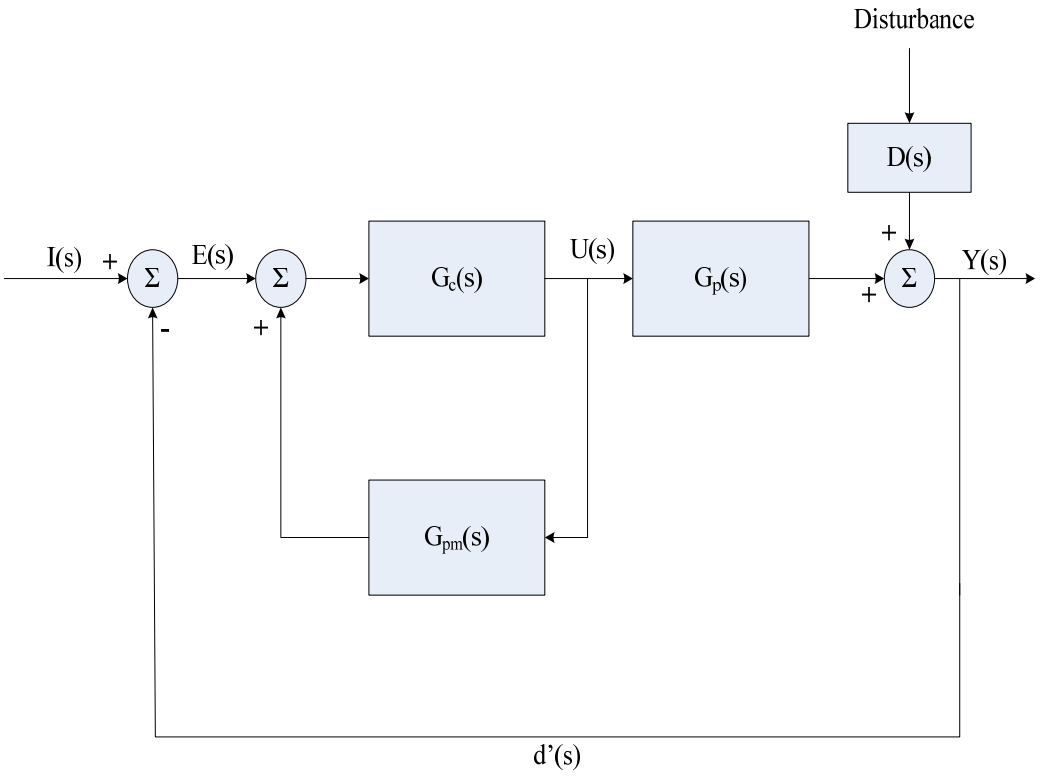


Figure 4-3 Alternate design of IMC

From the figure above, the following equations could be obtained

$$[E(s) + G_{pm}(s)U(s)]\, G_c(s) = U(s) \tag{4.2}$$

$$E(s)\, G_c(s) = U(s)[1- G_{pm}(s)\, G_c(s)] \tag{4.3}$$

$$C(s) = U(s)/E(s) = G_c(s) / [1- G_{pm}(s)\, G_c(s)]; \tag{4.4}$$

$$[I(s) – Y(s)]\, C(s)\, G_p(s)+ D(s) = Y(s) \tag{4.5}$$

$$Y(s) =[I(s)C(s)\, G_p(s) +D(s)] / (1+ C(s)\, G_p(s)) ; \tag{4.6}$$

$$Y(s) = (I(s)\ G_c(s)\ G_p(s) + D(s)\ [1- G_{pm}(s)\ G_c(s)]) / (1- G_{pm}(s)\ G_c(s)+ G_c(s)\ G_p(s)); \quad (4.7)$$

$$Y(s) = (I(s)\ G_c(s)\ G_p(s) + D(s)\ [1- G_{pm}(s)\ G_c(s)]) / (1- G_c(s)[G_{pm}(s) - G_p(s)]); \quad (4.8)$$

So ideal case is $G_c(s) = G_{pm}(s)^{-1}$. And the closed loop system is stable than the system could respond to the input as wanted by eliminating the effects of mismatch and uncertainties.

4.1.1 IMC Design

Firstly the controller is designed according to the assumption mentioned above that $G_{pm}(s) = G_p(s)$ ideally. First of all, the plant uncertainties are not taken into account. However secondly the robustness properties are considered and uncertainties are taken into account. This is done by assigning a compensator $G_f(s)$. So the new controller is in the form

$$G_{IMC}(s) = G_C(s)\ G_f(s)$$

$G_f(s)$ is generally a filter that is used to eliminate the noise on the plant.

After all these are done, a suitable IMC servo controller could be achieved.

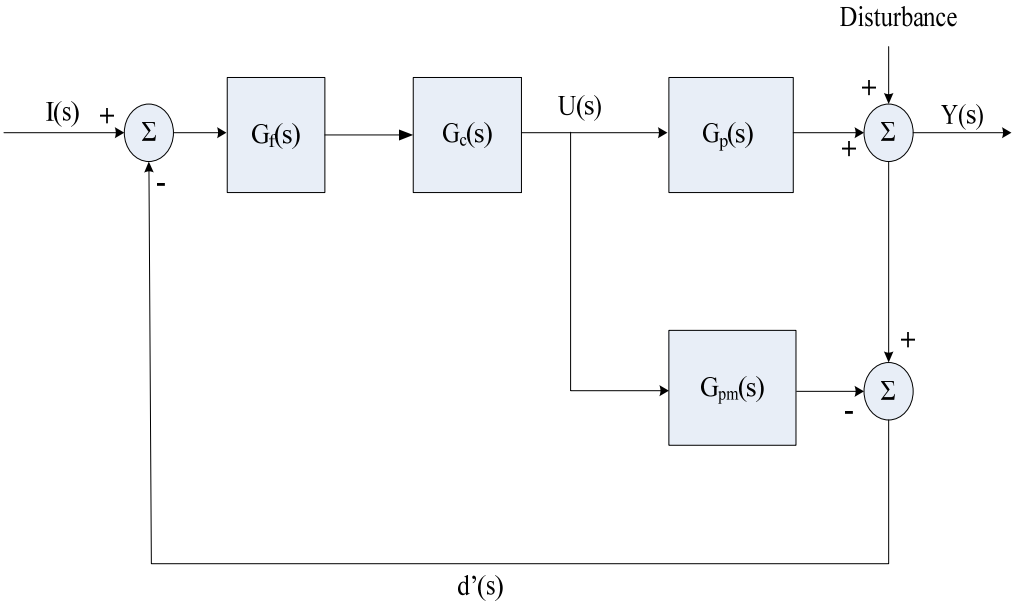


Figure 4-4 IMC with filter

First part: obtaining $G_C(s)$ according to $G_{pm}(s) = G_p(s)$:

1) Distinguish the invertible components ($G_{pmi}(s)$) and non-invertible components ($G_{pmn}(s)$) of the internal model.

$$G_{pm}(s) = G_{pmi}(s)\, G_{pmn}(s)$$

2) take the invertible part and the controller is just;

$$Gc(s) = G_{pmi}(s)^{-1}$$

To use the IMC, the internal model should contain invertible components and should indicate stability. If the internal model does not indicate stability and has only non-invertible components, then it is not suitable to use IMC principle. IMC principle depends on these concepts.

Here non-invertable components are defined as the components that will lead to realisability and instability problems if inverted. E.g. terms that contain time-delays and positive zeros [19].

Second part: obtaining $G_f(s)$:

As it was mentioned, the optimal controller that controls the plant is of the form below,

$$G_{IMC}(s) = G_C(s)\, G_f(s) \tag{4.9}$$

Here $G_f(s)$ is a low-pass filter, used to attenuate mismatches occurred during process modeling. This filter is;

$$G_f(s) = 1 / [1+ \zeta_f s]^n; \tag{4.10}$$

Filter order and the parameters are chosen appropriately so that control of the system is satisfied. In general, the order of the filter is same as the order of the internal model.

4.1.2 Conclusion

One basic property of this type of controller is that there is the robustness in its structure and there is a low-pass filter used to guarantee closed-loop stability [20].

According to the ideal case assumption, the inverse of the process has to be taken but in practice it is not possible so the perfect control is not possible. However, with its robustness property of the IMC, this problem can be overcome.

In practice, obtaining an exact model of the system is almost impossible; there is a mismatch every time. IMC type controllers are also designed to suppress the mismatch and the other uncertainties of the system considering the disturbance.

These types of controllers are widely used in real life, in industry and in process control because of its mismatch and disturbance rejection capabilities and also the robustness properties.

4.2 Linear Quadratic Gaussian

In general, some systems may require robust controllers if there are some measurement errors, nonlinearities, disturbances. These cause uncertainties in the system. The robust controllers are used to come up with these problems and supply some tolerance to these uncertainties.

The system robustness is the measure of stability or the performance of the system to uncertainties and varying operating conditions. That is why robust controllers are designed to ensure stability for all the different plant models that are in the acceptable uncertainty range. Therefore, it can be said that all the models of the system are approximations of the real plant [25].

Using robust controllers is a good issue to achieve system requirements when there are disturbances and uncertainties affecting on the system. In classical control, to overcome this problem the phase margin and gain margin are taken as high as possible. However, in modern control the robust controllers are used. Some examples of most known robust controllers are; H_2, H_∞, LQG...

In the system, the LQG robust controller is used to overcome the uncertainties. LQG is a special case of H_2 and this controller is widely used in aircraft applications. LQG is a good robust controller that is used to design optimal dynamic regulator.

In this method, we define and determine the weighted mean square error criterion as a standard and also we determine the stochastic models for disturbances and uncertainties. The remaining parts are done automatically [26].

Through this LQG section, [33] and [34] are used as the main references.

The general structure of LQG is given in Figure 4-5. As it is seen from the figure, the plant equations are

$$x_{k+1} = Ax_k + Bu_k + v_k \tag{4.11}$$

$$y_k = Cx_k + w_k \tag{4.12}$$

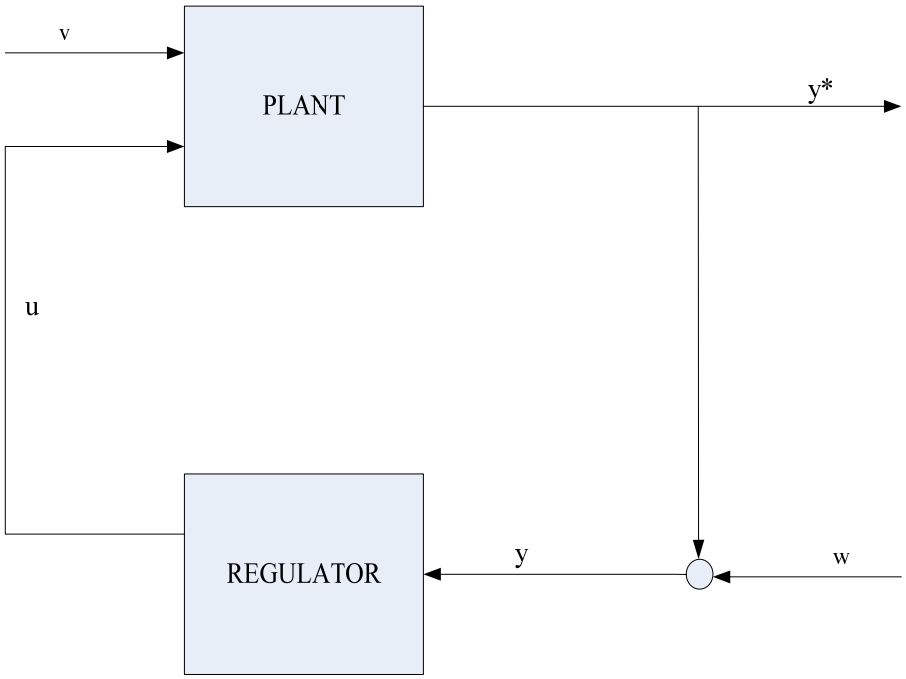


Figure 4-5 General Structure of LQG

In the equation, there are two noises acting on the system which are measurement noise and process noise. These noises are unknown, also the initial state of x is unknown but to make some assumption about these unknown parameters and to eliminate the noise, LQG method is used. This method is composed of two main parts: elimination of noise by Kalman filter part and solving an optimal control

problem. The Figure 4-6 shows how LQG method is formed. As shown on the structure, the output of the Kalman filter is connected to state-feedback.

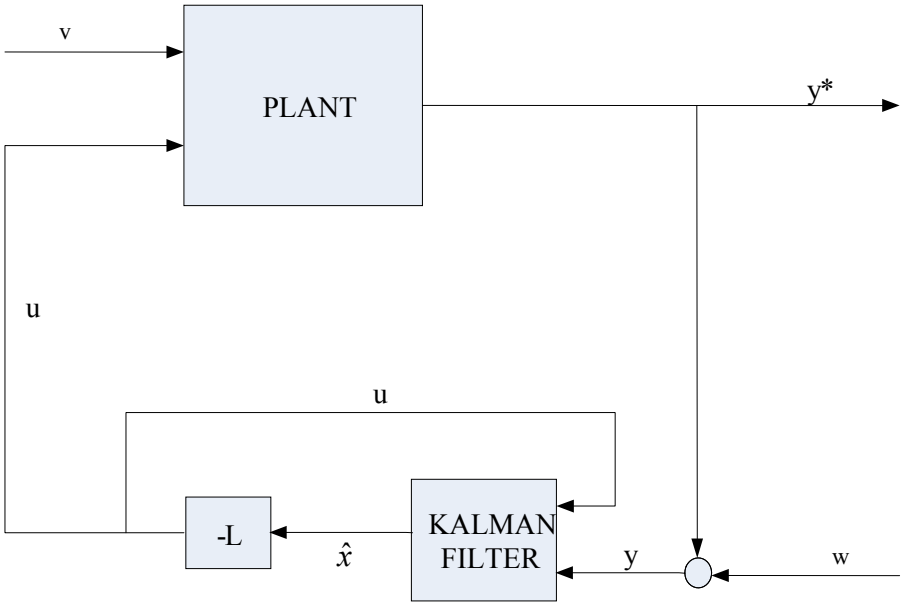


Figure 4-6 Detailed Structure of LQG

Kalman filter is the optimal filter in case of white noise [21]. It is assumed that noise distribution of our system is white Gaussian. Then the elimination of noise can be done by Kalman filter concept.

The LQG design procedure is composed of two main parts. This is done by using Separation Theorem. First of all, an optimal state-feedback regulator is designed; secondly the Kalman filter is designed for the elimination of noises. After all, these two parts are combined together and an optimal compensator is formed [22].

Here now the optimal control problem starts with the optimal state-feedback gain calculation procedure;

4.2.1 Optimal State Feedback

For a classical linear time-invariant system final cost function is given as follows;

$$J = \sum_{k=0}^{\infty} \left[x_k{}^T Q x_k + u_k{}^T R u_k \right] \tag{4.13}$$

This is the cost function that we have to minimize.

Here Q and R are the weight matrices for the state vector and the input, respectively. Where $Q \geq 0$ and $R > 0$.

First part for our case is the optimal-state feedback case.

$$u_k = -L\hat{x}_k \tag{4.14}$$

Here $\hat{x}_k$ represents the estimated value of state vector x_k.

For optimal control gain matrix L, the equation is

$$L = (B^T S B + R)^{-1} B^T S A \tag{4.15}$$

This S matrix can be obtained from the solution of discrete-time algebraic Riccati equation

$$S = A^T \left[S - SB(B^T S B + R)^{-1} B^T S \right] A + Q \tag{4.16}$$

Now the optimal control gain matrix is obtained.

4.2.2 Kalman Filter

The second part is the Kalman filter part:

For Kalman filter the desired state vector equation is

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + K_k(y_k - C\hat{x}_k) \tag{4.17}$$

Where

$$K_k = AP_kC^T\left[CP_kC^T + W_k\right]^{-1} \tag{4.18}$$

For Kalman filter P_k matrix can be obtained from the solution of discrete-time algebraic Riccati equation

$$P_{k+1} = A^T\left[P_k + P_kC^T(CP_kC^T + W_k)^{-1}CP_k\right]A^T + V_k \tag{4.19}$$

Here in the Kalman filter process, the important issue is determining the noise intensity matrices V_k and W_k. The selection of suitable V_k and W_k is determined empirically.

In order to obtain a better performance and optimal control, the maximum information about the system, measurement and system noise is required.

4.2.3 LQG Design

Now LQG is formed by using the Kalman filter and the optimal state-feedback together as shown in Figure 4-6.

In the Figure 4-6, L is known as stated in (4.15) and for Kalman filter K is known that is driven in (4.18). These K and L parameters include V_k, W_k, Q, and R which

are the tuning parameters of the LQG design. Finally the LQG design equation is as follows;

From the equations and the figure above;

$$K_k = AP_kC^T\left[CP_kC^T + W_k\right]^{-1} \text{ observer gain} \tag{4.20}$$

$$L = (B^T SB + R)^{-1} B^T SA \quad \text{state feedback} \tag{4.21}$$

And

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k + K_k(y_k - C\hat{x}_k) \tag{4.22}$$

$$u_k = -L\hat{x}_k \tag{4.23}$$

From the equations above, finally estimated value of state vector $\hat{x}_k$ is:

$$\hat{x}_{k+1} = A\hat{x}_k + BL\hat{x}_k + K_k(y_k - C\hat{x}_k) \tag{4.24}$$

4.2.4 Conclusion

The Kalman filter concept is not only helpful in filtering the uncertainties and noises but also gives us some information about the unmeasured sensors or other devices that are not taken into consideration while modeling [22].

The existence of K and L matrices and stable closed-loop system concludes controllable and observable system with unique positive definite solution of P and $\hat{P}$ [27].

The advantage of using the separation principle is that K and L matrices may be designed separately to observe the behaviors of closed loop plant and the observer [24].

After all, to analyze the performance and maybe to make some changes on the K and L matrices, the closed-loop system is analyzed. This can be done by using frequency response and robustness analysis of the system. By looking at these concepts the optimized K and L matrices can be obtained. This is an easy way and it guarantees stable closed-loop system. The weighting matrices are determined by looking at the system response. Large weights mean small response. The basic concept in determining the weighting matrices is first take square of allowed deviations and then inverse it and assign these values as the diagonal elements of the matrices. But also K and L can be determined by putting some constraints on the control procedure and also on the state variables [28].

CHAPTER 5

CONTROLLER DESIGN AND IMPLEMENTATION

Throughout this chapter, the position command loop is given, control of our system and the designed controllers are discussed. Then, the response of these controllers in the real system and the implementation of the controllers are presented. The purpose of this chapter is to design a controller which satisfies the requirements. Different controller types are designed to make an observation and to command on them. Also the effect of perturbation which is an important issue is analyzed. There may occur some changes on the system because of industrial manufacturing or during system integration. In order to overcome these problems, the effect of perturbation should be considered. This effect is also explained in details in this chapter. Finally according to the responses of these controllers, the best type of controllers, their properties and the commands on the controllers are summarized. At the end of this chapter, the control strategy of the system satisfying all the requirements is explained.

5.1 Control Structures and Requirements

5.1.1 Control Loop

The control loop of the system is implemented in the DSP as given in Appendix A and Figure 3-2. The plant part is taken as the part outside the DSP. The position loop structure of our system is as follows;

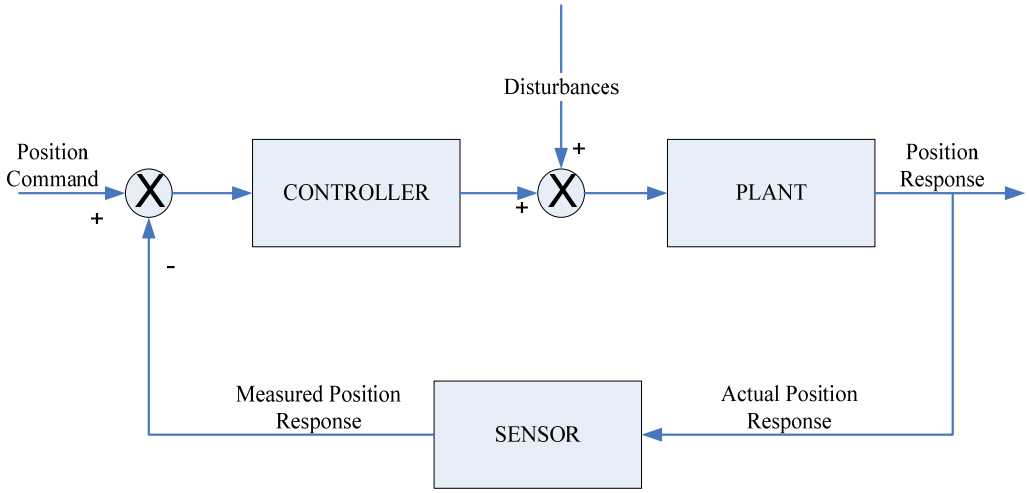


Figure 5-1 Position Loop Block Diagram

From the Figure 5-1, there is a controller, a plant and also a sensor. The sensor is the resolver part of the system which is used to get position data of the system. This position data is sent to DSP as well as the position command is transferred to DSP. In the DSP with the help of controller, the new position command is used and sent to the plant. Then the plant responds to the command. This response is measured by the sensor and the loop starts again. The position loop of the system is implemented by this way. The details are already given in Appendix A.

5.1.2 Design Requirements

The design of a controller basically depends on design requirements and the aim of the controller is to meet the requirements. The basic specification for the controller is to hold the error signal within the 5.5° positioning envelope. Also

desired maximum speed for system is 20˚ per second. This is the safer margin value and real expectation for the speed is less than or equal to 20˚ per second. According to these requirements, some basic parameters are limited to obtain a better performance.

Therefore the designed controller should satisfy the following conditions;

- Hold the error signal within the 5.5° positioning envelope,
- Closed loop stability,
- Disturbance rejection,
- Precise pointing capability,
- Robustness against unmodeled system parameters.

Also additional requirements are given to obtain a better controller, which are;

- Rise Time less than 0.3 seconds,
- Percent overshoot less than 40%,
- Settling time less than 3 seconds,
- Steady-state error should be less than 5%,
- Gain margin is approximately higher than 6 dB,
- Phase margin is higher than 40 dB.

These requirements are the basic requirements while designing a controller. Especially for the phase and gain margin, the limits are chosen according to the limits determined for these type of systems in general. Actually, the designed controllers that have the best performance of all, better parameters and least error signal are chosen as the most suitable controller for the system.

5.2 Integration of Equations to DSP

Assume that the equations of the controllers are ready, now these equations have to be integrated into the DSP. The structure of our system is given in Appendix A. It is seen that the designed controller are embedded into the DSP because the

position command to the turret system is sent by DSP. The procedure used for integrating the equations into the DSP is as follows which is given in the reference [29] in details.

5.2.1 Difference Equation Generation for Azimuth Position Loop Compensation

To get a general idea, start with second order filter derivation and then generalize it for our case.

Assume that the s-domain transfer function is

$$\frac{Y(s)}{X(s)} = \frac{as^2 + bs + c}{ds^2 + es + f} \tag{5.1}$$

Transform this to z-domain

$$s = (\frac{2}{T})(\frac{z-1}{z+1}) \tag{5.2}$$

This Bilinear Transform maps left half s-plane into the z-plane unit circle.

By placing the (5.2) into the (5.1) obtain that,

$$\frac{Y(z)}{X(z)} = \frac{a\left[\left(\frac{2}{T}\right)\left(\frac{z-1}{z+1}\right)\right]^2 + b\left[\left(\frac{2}{T}\right)\left(\frac{z-1}{z+1}\right)\right] + c}{d\left[\left(\frac{2}{T}\right)\left(\frac{z-1}{z+1}\right)\right]^2 + e\left[\left(\frac{2}{T}\right)\left(\frac{z-1}{z+1}\right)\right] + f} \tag{5.3}$$

$$\frac{Y(z)}{X(z)} = \frac{(4az^2 - 8az + 4a) + (2bTz^2 - 2bT) + (cT^2z^2 + 2cT^2z + cT^2)}{(4dz^2 - 8dz + 4d) + (2eTz^2 - 2eT) + (fT^2z^2 + 2fT^2z + fT^2)} \tag{5.4}$$

This is

$$\frac{Y(z)}{X(z)} = \frac{(4a + 2bT + cT^2)z^2 + (2cT^2 - 8a)z + (4a - 2bT + cT^2)}{(4d + 2eT + fT^2)z^2 + (2fT^2 - 8d)z + (4d - 2eT + fT^2)} \quad (5.5)$$

Take that

$$G1 = (4a + 2bT + cT^2) \quad (5.6)$$

$$G2 = (2cT^2 - 8a) \quad (5.7)$$

$$G3 = (4a - 2bT + cT^2) \quad (5.8)$$

$$G4 = (4d + 2eT + fT^2) \quad (5.9)$$

$$G5 = (2fT^2 - 8d) \quad (5.10)$$

$$G6 = (4d - 2eT + fT^2) \quad (5.11)$$

So the new equation is

$$\frac{Y(z)}{X(z)} = \frac{G1z^2 + G2z + G3}{G4z^2 + G5z + G6} \quad (5.12)$$

Multiply both parts by z^{-2} then we get

$$\frac{Y(z)}{X(z)} = \frac{G1 + G2z^{-1} + G3z^{-2}}{G4 + G5z^{-1} + G6z^{-2}} \quad (5.13)$$

Time domain equation

$$Y(n) = \left[\left(\frac{G1}{G4}\right)X(n) + \left(\frac{G2}{G4}\right)X(n-1) + \left(\frac{G3}{G4}\right)X(n-2) - \left(\frac{G5}{G4}\right)Y(n-1) - \left(\frac{G6}{G4}\right)Y(n-2)\right] \quad (5.14)$$

For our case the orders of the controller equations may be higher than 2 so this discretization method can be generalized for other cases like

$$\frac{Y(z)}{X(z)} = \frac{G1z^{r} + G2z^{r-1} .... + (G(r+1))z^{0}}{(G(r+2))z^{m} + (G(r+3))z^{m-1} .... + (G(r+2+m))z^{0}} \tag{5.15}$$

Discretize it

$$Y(n) = \left[\begin{array}{l} \left(\frac{G1}{G(r+2)} \right) X(n) + \left(\frac{G2}{G(r+2)} \right) X(n-1) ... \\ + \left(\frac{G(r+1)}{G(r+2)} \right) X(n-r) - \left(\frac{G(r+3)}{G(r+2)} \right) Y(n-1) - ... \left(\frac{G(r+2+m)}{G(r+2)} \right) Y(n-m) \end{array} \right] \tag{5.16}$$

5.3 Designed Controllers

For the controller design purpose, two types of controllers are examined in the first stage. These two controllers are PI and IMC controllers. Secondly, some perturbations are applied to see the affect of perturbation. Then LQG type of controllers are designed and experimented on the real system. Working principles of these controllers are examined. Ramp response of some of the controllers and the difference between experiments and simulations are also studied.

5.3.1 PI & IMC Controllers Design

For the controller design purpose, both PI and IMC controllers are designed. The results that are collected from the original system are shown below.

The equations and graphs shown below belong to these controllers that show closed loop responses of the real system;

5.3.1.1 Designed Controllers and Characteristics

PI Type 1

Transfer function of CNT 4 (PI Controller)

$$H_{CNT4}(z) = \frac{0.3143z - 0.3004}{z - 1} \tag{5.17}$$

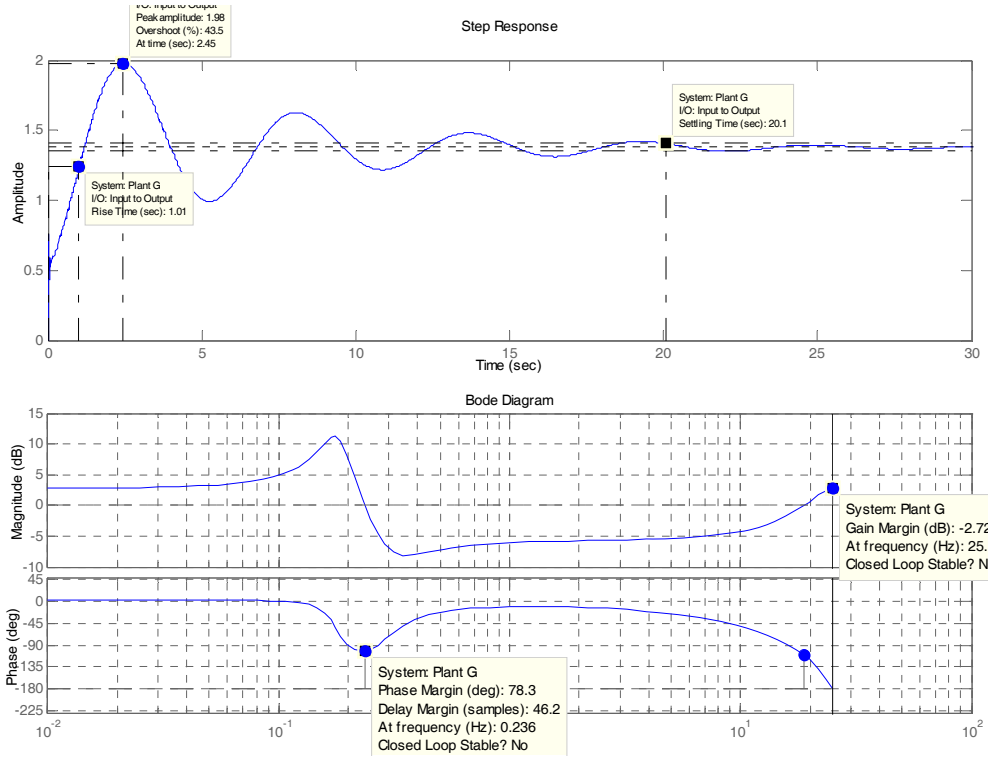


Figure 5-2 Step Response, Impulse Response and Bode Plot of CNT4

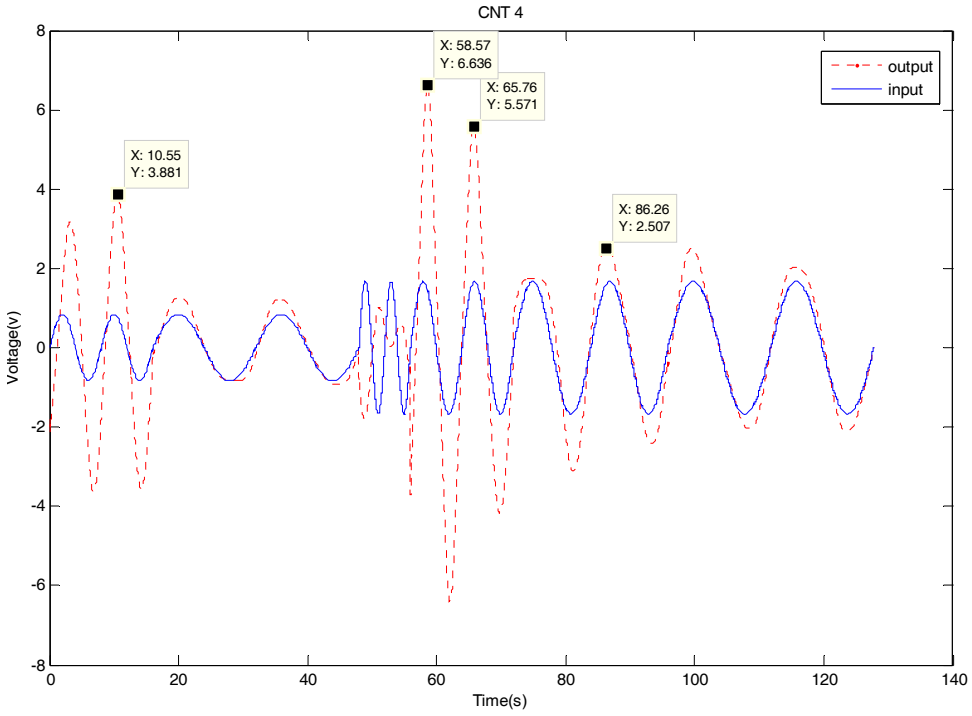


Figure 5-3 Input/Output Graph of CNT4

IMC Type 1

Transfer function of CNT 1 (IMC with 1 notch filter)

$$H_{CNT1}(z)=\frac{-0.03941z^6+0.2741z^5-0.7688z^4+1.123z^3-0.9057z^2+0.3839z^1-0.06697}{z^6-5.85z^5+14.27z^4-18.58z^3+13.62z^2-5.326z^1+0.8688} \quad (5.18)$$

This transfer function is obtained by adding one notch filter on to the following transfer function

$$H_{CNT3}(z)=\frac{-0.04091z^4+0.2033z^3-0.3539z^2+0.2615z^1-0.07009}{z^4-3.936z^3+5.817z^2-3.825z^1+0.9443} \quad (5.19)$$

The properties of added notch filter are;
Natural frequency = 3.3256 Hz,
Damping (zero) = 0.05,
Damping (pole) = 0.5,
Notch Depth (db) = -20,
Notch width (log) = 0.28547.

As a result of this notch filter addition process, the order of the transfer function is increased from 4 to 6.

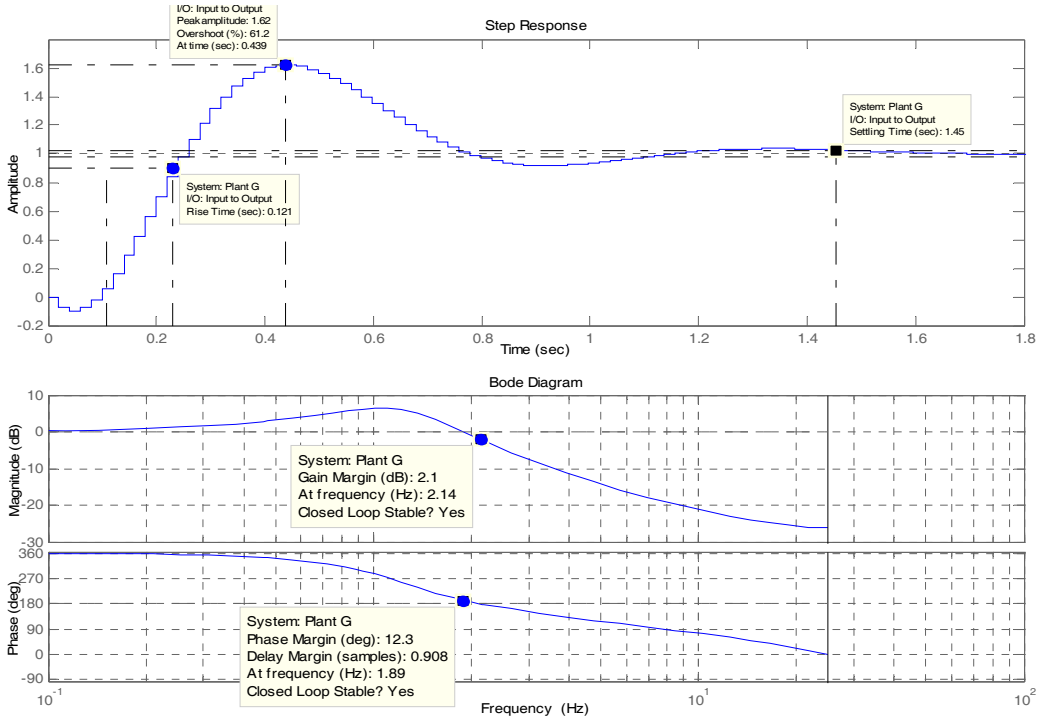


Figure 5-4 Step Response, Impulse Response and Bode Plot of CNT1

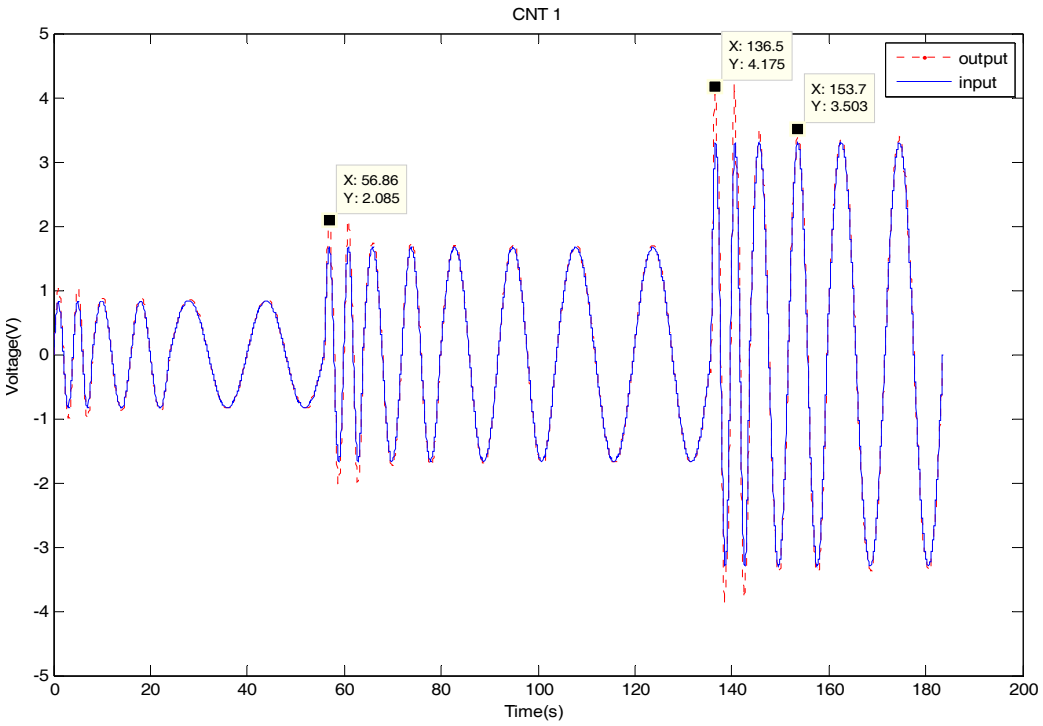


Figure 5-5 Input/Output Graph of CNT1

IMC Type 2

Transfer function of CNT 2 (IMC with 2 notch filter)

$$H_{CNT2}(z)=\frac{-0.03641z^{8}+0.2358z^{7}-0.904z^{6}+1.951z^{5}-2.602z^{4}+2.199z^{3}-1.153z^{2}+0.3428z^{1}-0.04435}{z^{8}-7.725z^{7}+26.13z^{6}-50.54z^{5}+61.14z^{4}-47.37z^{3}+22.96z^{2}-6.362z^{1}+0.772} \quad (5.20)$$

This transfer function is obtained by adding two notch filters on to the following transfer function

$$H_{CNT}(z)=\frac{-0.02889z^{4}+0.1436z^{3}-0.2499z^{2}+0.1847z^{1}-0.0495}{z^{4}-3.936z^{3}+5.817z^{2}-3.825z^{1}+0.9443} \quad (5.21)$$

The properties of added first notch filter are;
Natural frequency = 3.324 Hz
Damping (zero) = 0.05
Damping (pole) = 0.5
Notch Depth (db) = -20
Notch width (log) = 0.28547

The properties of added second notch filter are;
Natural frequency = 4.7131 Hz
Damping (zero) = 0.05
Damping (pole) = 0.5
Notch Depth (db) = -20
Notch width (log) = 0.2854

As a result of this notch filters addition process, the order of the transfer function is increased from 4 to 8.

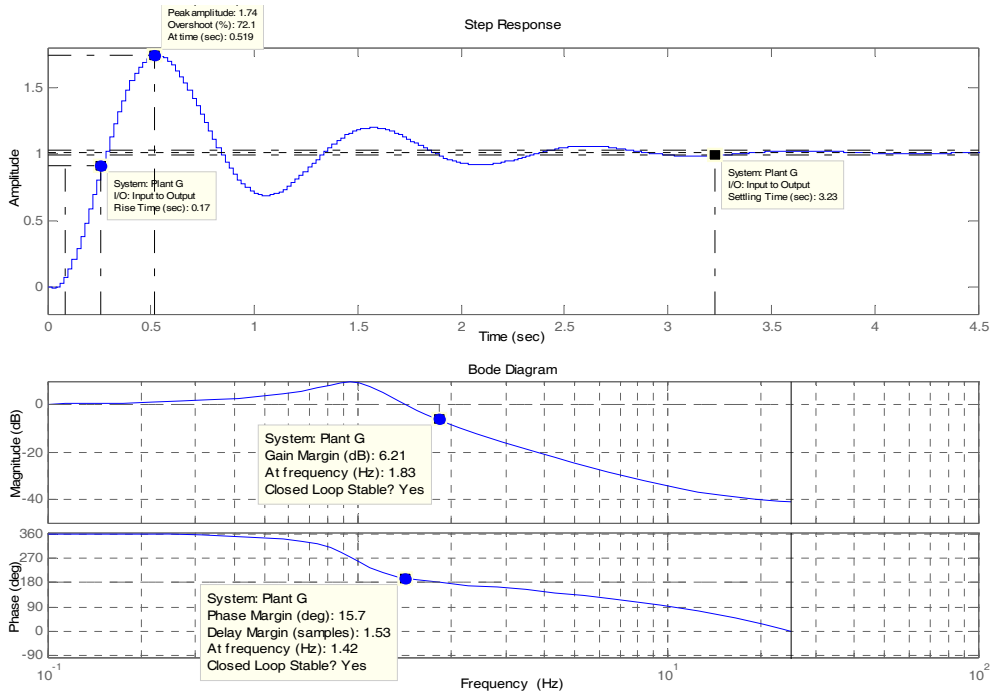


Figure 5-6 Step Response, Impulse Response and Bode Plot of CNT2

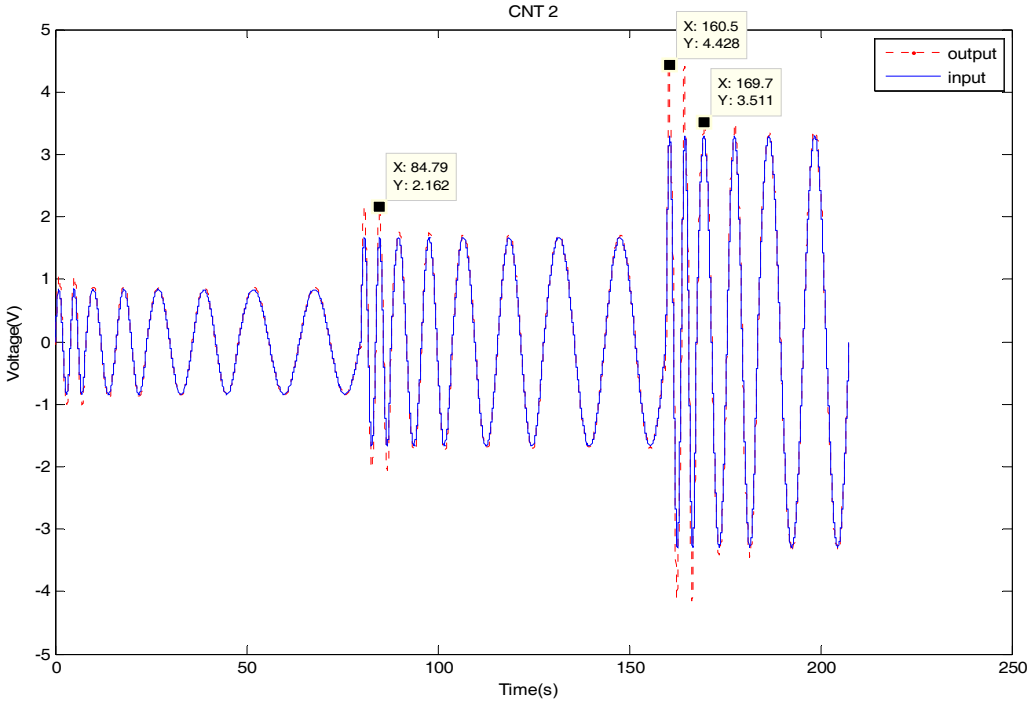


Figure 5-7 Input/Output Graph of CNT2

IMC Type 3

Transfer function of CNT 3 (IMC without any notch filter)

$$H_{CNT3}(z) = \frac{-0.04091z^4 + 0.2033z^3 - 0.3539z^2 + 0.2615z^1 - 0.07009}{z^4 - 3.936z^3 + 5.817z^2 - 3.825z^1 + 0.9443} \qquad (5.22)$$

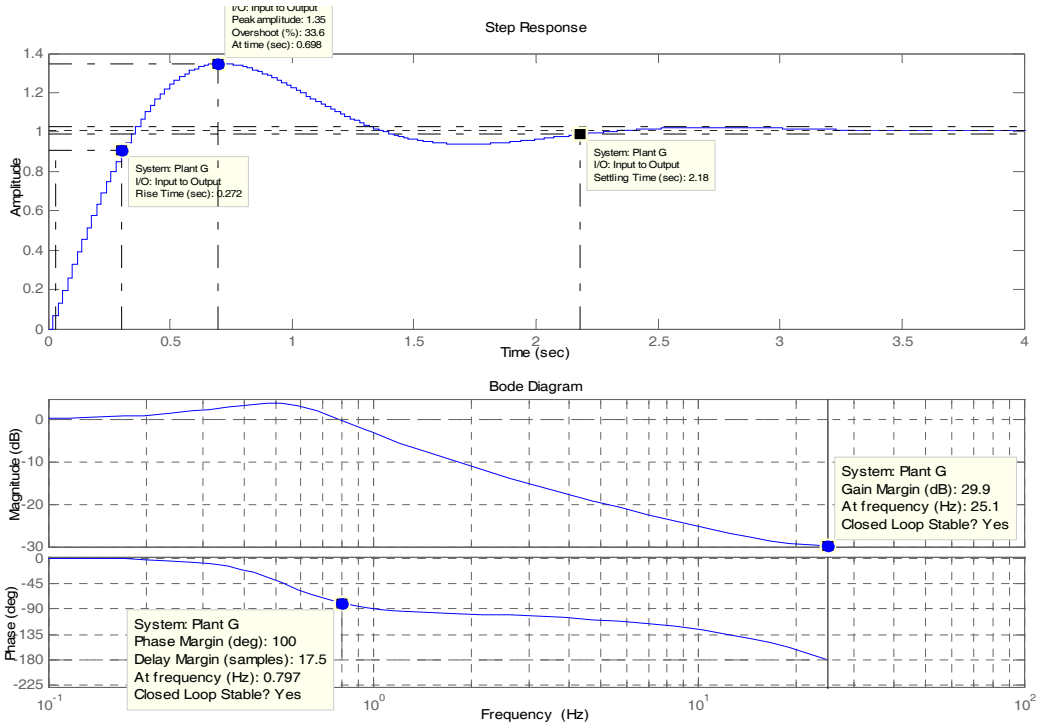


Figure 5-8 Step Response, Impulse Response and Bode Plot of CNT3

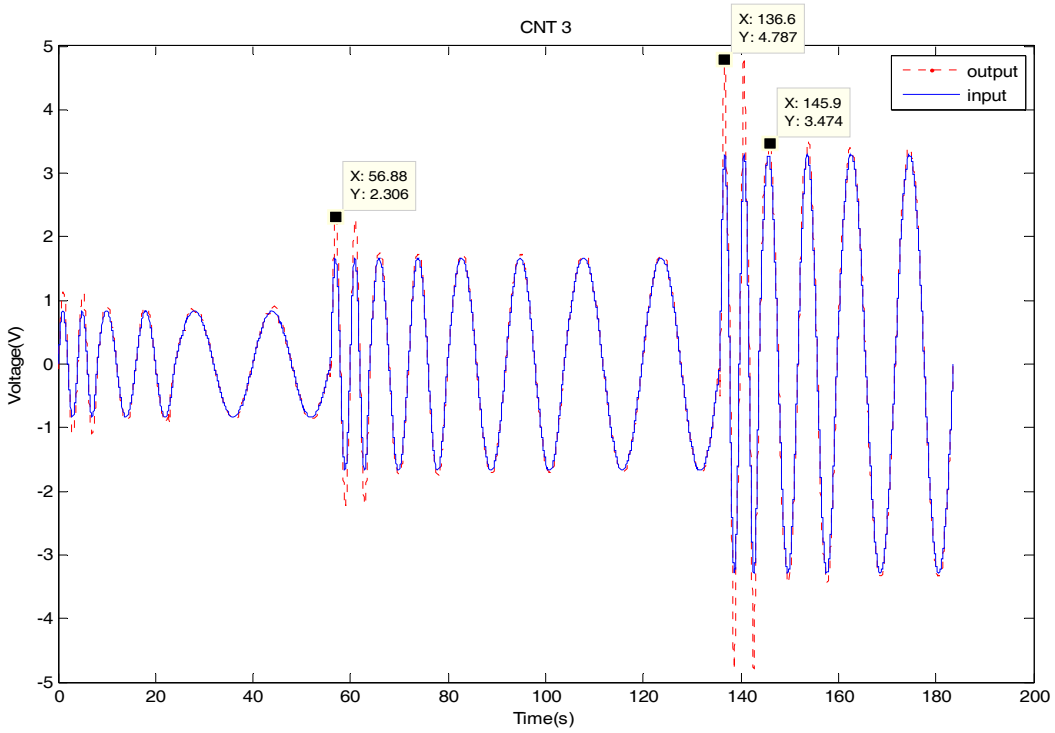


Figure 5-9 Input/Output Graph of CNT3

IMC Type 4

Transfer function of CNT 7 (IMC without any notch filter)

$$H_{CNT7}(z) = \frac{-0.04217z^4 + 0.1574z^3 - 0.2129z^2 + 0.1223z^1 - 0.02461}{0.9552z^4 - 3.849z^3 + 5.841z^2 - 3.945z^1 + 1} \qquad (5.23)$$

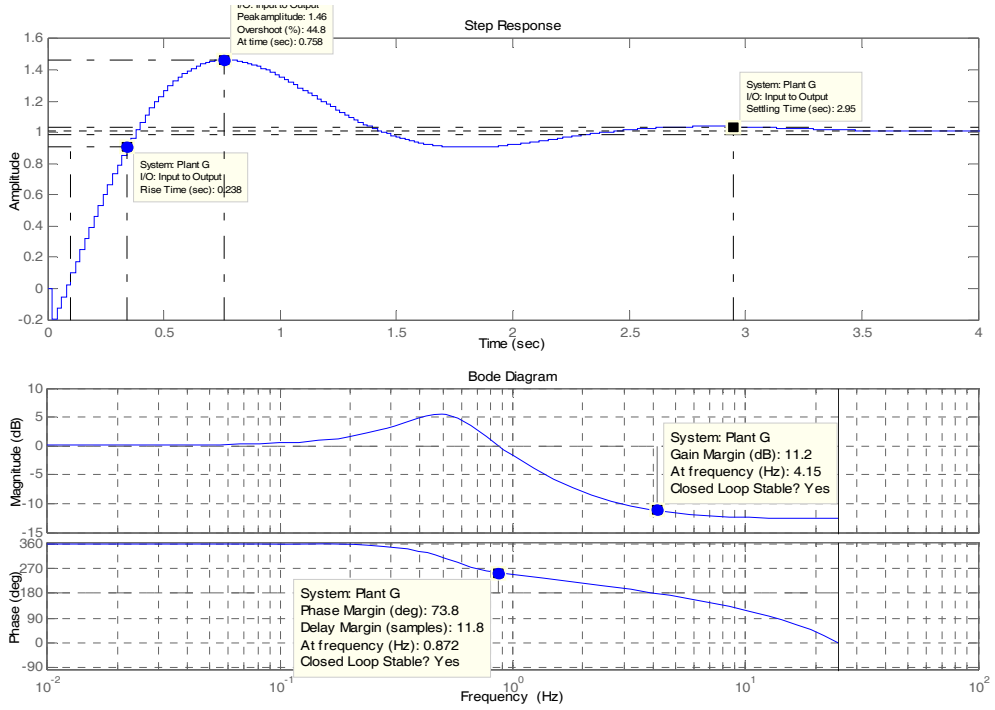


Figure 5-10 Step Response, Impulse Response and Bode Plot of CNT7

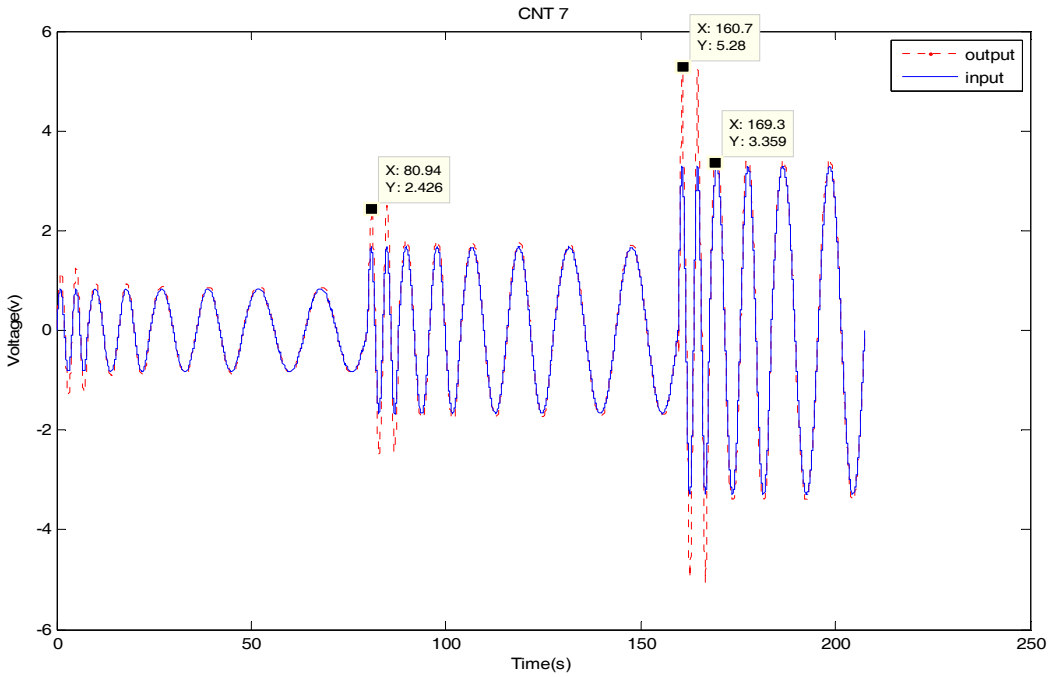


Figure 5-11 Input/Output Graph of CNT7

5.3.1.2 Results and Conclusion

Table 1 PI & IMC Controllers Characteristics

Controller	T_r(s)	M_p / T_p(s)	T_s(s) (1%)	T_s(s) (5%)	GM (dB) / at f(Hz)	PM (dB) / at f(Hz)	Closed Loop Stable
CNT 4	1.01	1.98 / 2.45	25.5	14.4	-2.72 / 25.1	78.3 / 0.236	Yes
CNT 1	0.121	1.62 / 0.439	1.54	1.03	2.1 / 2.14	12.3 / 1.89	Yes
CNT 2	0.17	1.74 / 0.519	3.76	2.25	6.21 / 1.83	15.7 / 1.42	Yes
CNT 3	0.272	1.35 / 0.698	2.98	1.96	29.9 / 25.1	100 / 0.797	Yes
CNT 7	0.238	1.46 / 0.758	3.23	2.14	11.2 / 4.15	73.8 / 0.872	Yes

The controllers and their properties are mentioned before but to give a brief description; CNT 4 is a PI type controller, CNT 1, CNT 2 are IMC type controllers with notch filters, CNT 3, CNT 7 are IMC type controllers without any notch filters. The IMC type controllers with notch filters are examined on the system to observe their performance, to see the effect of compensator on the controllers and to obtain a better bode plot and better response. The characteristics of these notch filters are determined by looking at the bode plots of the IMC controllers. From these bode plots just by changing the applied natural frequency of each notch filter, the response of the system is observed.

Both from the table and also from the input/output graphs, PI controller is worse than all IMC controllers and it is not closed loop stable. So PI controllers cannot be used as the controllers of the system because of noise characteristics of the system. However, by using some compensators these controllers may be used to satisfy the system requirements. For this purpose, a simulation is done and the following table is obtained by using the plant model and PI controller given in (5.17).

Table 2 PI Controller with Compensator Characteristics

Controller	T_r(s)	M_p / T_p(s)	T_s(s) (1%)	T_s(s) (5%)	GM (dB) / at f(Hz)	PM (dB) / at f(Hz)	Closed Loop Stable
CNT 4	0.39	1.09 / 1.43	3.82	2.02	8.08 / 1.5	25.1 / 1.25	Yes
CNT 4-1	0.387	1.06 / 1.4	3.25	1.94	15 / 1.87	44.8 / 1.24	Yes

Controller	T_r(s)	M_p / T_p(s)	T_s(s) (1%)	T_s(s) (5%)	GM (dB) / at f(Hz)	PM (dB) / at f(Hz)	Closed Loop Stable
CNT4-2	0.305	1.12 / 1.46	4.74	2.47	3.69 / 1.35	12.3 / 1.25	Yes
CNT4-3	0.387	1.05 / 1.4	2.88	1.91	16.4 / 1.96	49.8 / 1.24	Yes

This table is obtained by adding Lead and/or Lag filters on the PI controller CNT4. The properties and naming conventions are as follows;

The transfer function for CNT4-1 is

$$H_{CNT4-1}(z) = \frac{0.8705z^3 - 1.92z^2 + 1.314z^1 - 0.262}{z^3 - 1.9z^2 + 0.98z^1 - 0.08} \tag{5.24}$$

CNT4-1 is a PI controller with Lag and Lead filters whose properties are;
For Lag filter Zero = 0.35 and Pole= 0.80.
For Lead filter Zero = 0.9 and Pole = 0.1.

The transfer function for CNT4-2 is

$$H_{CNT4-2}(z) = \frac{0.09672z^2 - 0.1263z^1 + 0.03235}{z^2 - 1.8z^1 + 0.8} \tag{5.25}$$

CNT4-2 is a PI controller with Lag filter whose properties are;
For Lag filter Zero = 0.35 and pole= 0.80.

The transfer function for CNT4-3 is

$$H_{CNT4-3}(z) = \frac{0.943z^2 - 1.75z^1 + 0.8111}{z^2 - 1.7z^1 + 0.7} \tag{5.26}$$

CNT4-3 is a PI controller with Lead filter whose properties are;
For Lead filter Zero = 0.9 and pole= 0.7.

From Table 2, it is obvious that using Lead and Lag filters is a good approach to meet the system requirements. Especially using Lag and Lead filters together makes the system response better according to the simulation results given in Table 2. That is why; these filters are helpful in control architecture and repair the system response.

In addition to PI controller, four different IMC controllers are examined on the system. From the data taken, the graphs and the table shown above are obtained. When we look at the input/output graphs, none of the IMC controllers have phase differences between input and output. But there are amplitude differences. At higher frequencies CNT 1 and CNT 2 which are IMC controllers with notch filters are better than the other IMC controllers according to amplitude difference between output and input. This is because they have better rise time. However their phase and gain margins are worse than the other IMC controllers (CNT 3, CNT 7). And CNT 1 is better than CNT 2.

When we consider the peak value, settling time, gain margin and phase margin, the controllers without any notch filter (CNT 3, CNT 7) are better than the ones with notch filter (CNT 1, CNT 2). Especially CNT 3 is best of all the IMC controllers without any notch filter.

According to the requirements, CNT1 does not satisfy the requirements for phase margin and gain margin. If the system is unique, then there isn't any problem. For our case, the system is unique. However if the controller is planned to be used by other same type turret system then some modifications are required.

When we look at the settling times of the controllers, according to our requirements that steady-state error should be less than 5% and settling time should be less than 3 seconds, all the controllers excluding CNT4 satisfies the requirements at 5%. To analyze the effect of steady state error and settling time, the settling time for steady state error at 1% data is also taken. As expected, the settling time increases with the decrease in steady state error percent. But as

shown in Table 1, the requirement of settling time should be less than 3 seconds is also satisfied at steady state error 1% for CNT1 and CNT3.

Table 3 PI & IMC Controllers Measurements

Degree(º) / Freq(Hz)	**Measured Values**	**CNT1**	**CNT2**	**CNT3**	**CNT4**	**CNT7**
5/0.25	InVol(V)	0.837	0.837	0.837	0.837	0.837
	OutVol(V)	1.023	1.052	1.125	-	1.163
	Freq(Hz)	0.25063	0.25063	0.25063	0.25063	0.25063
	Error %	22.2	25.7	34.4	-	38.9
5/0.125	InVol(V)	0.837	0.837	0.837	0.837	0.837
	OutVol(V)	0.8636	0.862	0.896	3.17	0.924
	Freq(Hz)	0.12539	0.12539	0.12539	0.12539	0.12539
	Error %	3.0	2.98	7.05	279	10.4
5/0.0625	InVol(V)	0.837	0.837	0.837	0.837	0.837
	OutVol(V)	0.853	0.859	0.865	1.253	0.859
	Freq(Hz)	0.06265	0.06265	0.06265	0.06265	0.06265
	Error %	1.9	2.6	3.3	49.7	2.6
10/0.25	InVol(V)	1.669	1.669	1.669	1.669	1.669
	OutVol(V)	2.084	2.153	2.313	-	2.42
	Freq(Hz)	0.25063	0.25063	0.25063	0.25063	0.25063
	Error %	24.7	29	38.6	-	50
10/0.125	InVol(V)	1.669	1.669	1.669	1.669	1.669
	OutVol(V)	1.733	1.758	1.737	6.613	1.804
	Freq(Hz)	0.12539	0.12539	0.12539	0.12539	0.12539
	Error %	3.8	5.3	4	296	8.1
10/0.0835	InVol(V)	1.669	1.669	1.669	1.669	1.669
	OutVol(V)	1.692	1.705	1.722	1.740	1.740
	Freq(Hz)	0.08354	0.08354	0.08354	0.08354	0.08354
	Error %	1.4	2.2	3.2	4.3	4.3
10/0.0625	InVol(V)	1.669	1.669	1.669	1.669	1.669
	OutVol(V)	1.693	1.695	1.685	2.490	1.718
	Freq(Hz)	0.06265	0.06265	0.06265	0.06265	0.06265
	Error %	1.4	1.6	0.9	49.2	2.9

Degree(°) / Freq(Hz)	Measured Values	CNT1	CNT2	CNT3	CNT4	CNT7
20/0.25	InVol(V)	3.287	3.287	3.287	3.287	3.287
	OutVol(V)	4.16	4.420	4.78	-	5.293
	Freq(Hz)	0.25063	0.25063	0.25063	0.25063	0.25063
	Error %	26.6	34.4	45.4	-	61
20/0.125	InVol(V)	3.287	3.287	3.287	3.287	3.287
	OutVol(V)	3.503	3.510	3.467	-	3.336
	Freq(Hz)	0.12539	0.12539	0.12539	0.12539	0.12539
	Error %	6.6	6.8	5.5	-	1.5
20/0.0835	InVol(V)	3.287	3.287	3.287	3.287	3.287
	OutVol(V)	3.343	3.337	3.395	-	3.415
	Freq(Hz)	0.08354	0.08354	0.08354	0.08354	0.08354
	Error %	1.7	1.5	3.3	-	3.9

On the tables including controllers' measurements, there are some parameters taken from the 'input vs. output' graphs of each controller. To explain these parameters some abbreviations are used in the table which are;

InVol = this shows the peak amplitude of the sinusoidal input signal that is applied to the system.

OutVol = this shows the peak amplitude of the output signal that the real system responses.

Freq = this shows the frequency of one period of each sinusoidal signal applied to the system. Actually in general there are four different frequencies; 0.25063Hz = 3.99 sec, 0.12539 Hz = 7.98sec, 0.08354Hz = 11.97sec, 0.06265Hz = 15.96sec. In one period of sinusoidal input signal, 100 different input data is applied and forms a one period of signal and then 100 different output data is taken. So for instance, at 0.25063Hz (3.99 sec) there are 100 input data inside this period so the time passes from applied one input to the other is 3.99msec.

Degree = this shows the scanning range of the applied input in degrees. For instance 20º means that, there is a sinusoidal signal which first goes from 0º to 20º then -20º and then again 0º. So a 20º‘s cycle includes 80º‘s of motion at that applied frequency.

Error% = this shows the percent of the (Output Voltage - Input Voltage) / (Input Voltage). So for instance 10% error for 5 degrees of input means that, the system response to that input is maximum 5.5 degrees. Each error shows percent error for that specific input.

Considering the data given in the table, the comparison between all the controllers can be made. The data shown in Table 3 is taken from the real system. That is why the controller that satisfies our requirements and gives best response to the sinusoidal input is the best one of all. In this study, our main aim is to hold the error signal within the 5.5° positioning envelope and the system can at most go to 100º. So the controllers whose percent error is under 6.11% can be acceptable for 90º. During the experimentation at various frequencies various inputs are applied and the data is taken to see the response. The data taken at 0.25063 Hz actually will not be used in the real system because this frequency is too fast and forces our system to work at higher degrees. For instance, the sinusoidal input at the degrees like 20º, which means that 80º of motion for a complete sinusoidal cycle. Therefore a cycle takes 3.99sec, and we are not expecting from our system to give response at this short time because it may give damage to the system at long term and also this speed is not needed and not used for real systems. However, it is tried to satisfy the requirements at that frequency for higher degrees to be safer whether used or not. To give a small description for the controllers for instance at 10º the percent error under 55% can be acceptable at this degree.

From the tables, the controllers CNT1, CNT2, CNT3 and CNT7 satisfy the requirements also for 0.25063Hz. For its real, wanted working edge, the error is under 4% which is too smaller than 55% that is why these controllers may be used as the controllers of the system. This comparison can be made for all different input voltage ranges. In general, as it was mentioned before both by looking at

Table 1 and Table 3, if the device won't be working at too high frequencies then CNT 3 would be used, however if it will be working at too high frequencies then using CNT 1 instead of CNT 3 is a good idea.

5.3.2 IMC Controller Design with Some Perturbations

For the controller design purpose, two of the previously designed IMC controllers CNT1 and CNT3 were chosen and applied some perturbations on these controllers to see the effect of perturbation on the IMC controllers. The results that are collected from the original system are shown below. Also the effect of perturbation is analyzed on the simulation studies.

The equations and graphs shown below belong to perturbed IMC controllers that are closed loop response of the real system;

5.3.2.1 Designed Controllers and Characteristics

IMC Type 1-1

10% perturbation applied to the first five parameters of the IMC Controller Type 1 at the numerator part. 10% decreased according to original equation.

Transfer function of CNT 1-1

$$H_{CNT1\text{-}1}(z)=\frac{-0.035469z^{6}+0.24669z^{5}-0.69192z^{4}+1.0107z^{3}-0.81513z^{2}+0.3839z^{1}-0.06697}{z^{6}-5.85z^{5}+14.27z^{4}-18.58z^{3}+13.62z^{2}-5.326z^{1}+0.8688} \qquad (5.27)$$

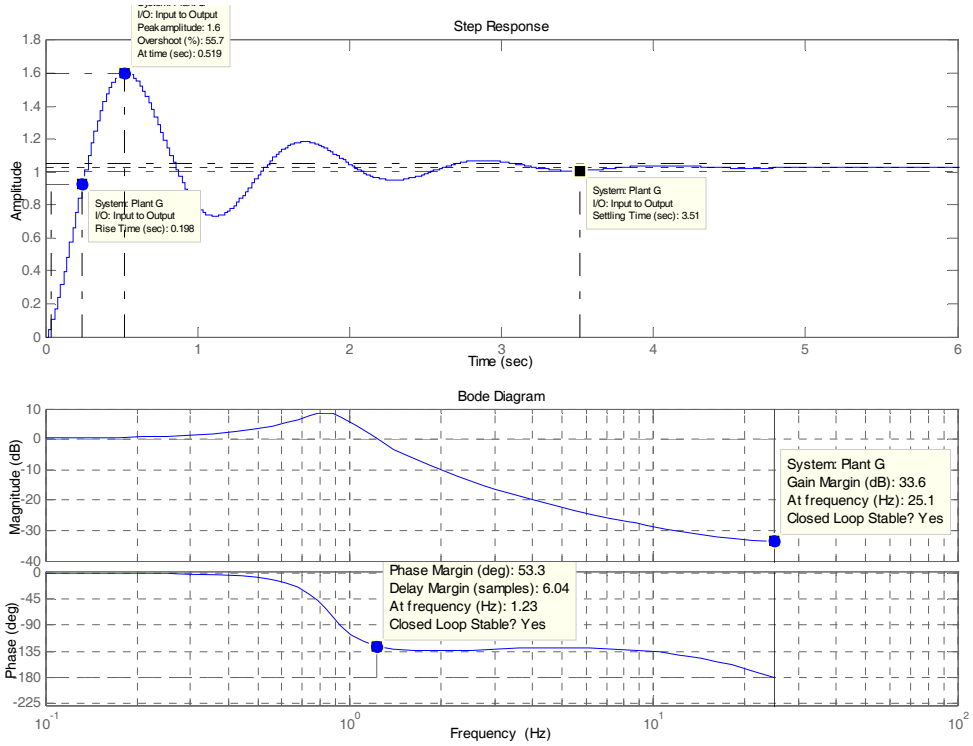


Figure 5-12 Step Response, Impulse Response and Bode Plot of CNT1-1

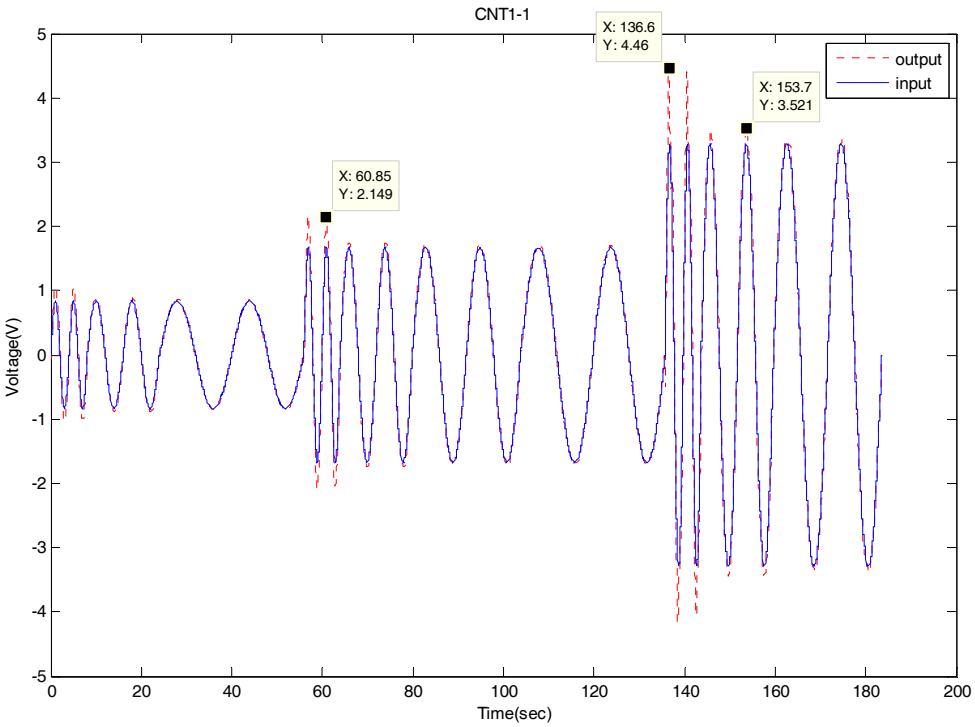


Figure 5-13 Input/Output Graph of CNT1-1

IMC Type 1-2

10% perturbation applied to the first five parameters of the IMC Controller Type 1 at the numerator part. 10% increased according to original equation.

Transfer function of CNT 1-2

$$H_{CNT1\text{-}2}(z)=\frac{-0.043351z^6+0.30151z^5-0.84568z^4+1.2353z^3-0.99627z^2+0.3839z^1-0.06697}{z^6-5.85z^5+14.27z^4-18.58z^3+13.62z^2-5.326z^1+0.8688} \quad (5.28)$$

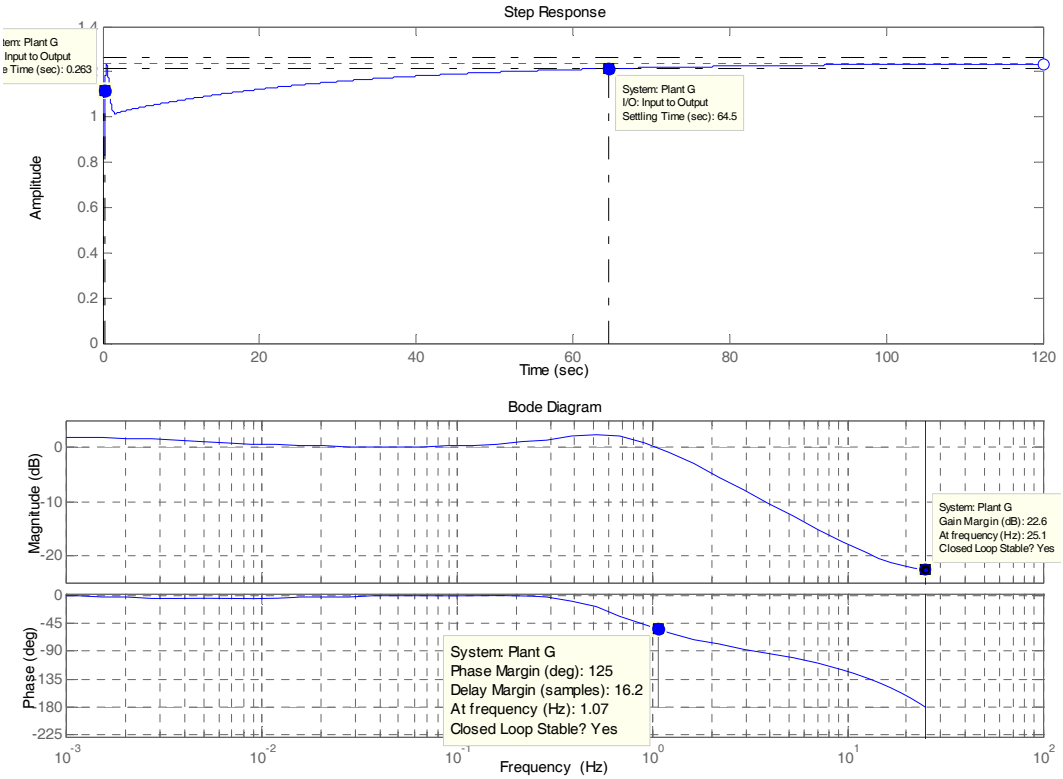


Figure 5-14 Step Response, Impulse Response and Bode Plot of CNT1-2

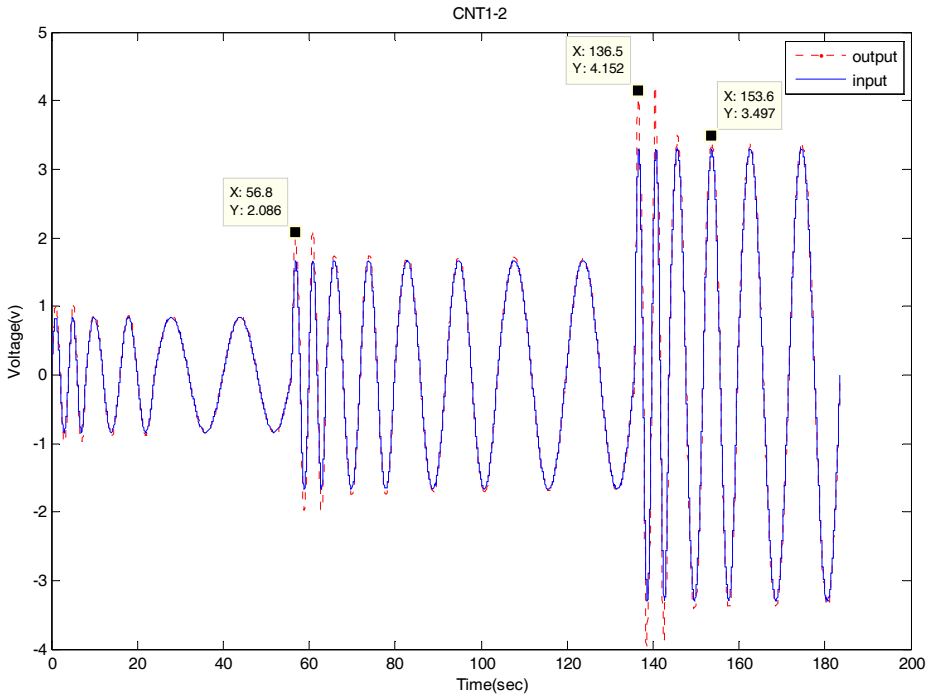


Figure 5-15 Input/Output Graph of CNT1-2

IMC Type 1-3

10% perturbation applied to the first three parameters of the IMC Controller Type 1 at the denominator part. 10% increased according to original equation.

Transfer function of CNT 1-3

$$H_{CNT1\text{–}3}(z)=\frac{-0.03941z^6+0.2741z^5-0.7688z^4+1.123z^3-0.9057z^2+0.3839z^1-0.06697}{1.1z^6-6.435z^5+15.697z^4-18.58z^3+13.62z^2-5.326z^1+0.8688} \qquad (5.29)$$

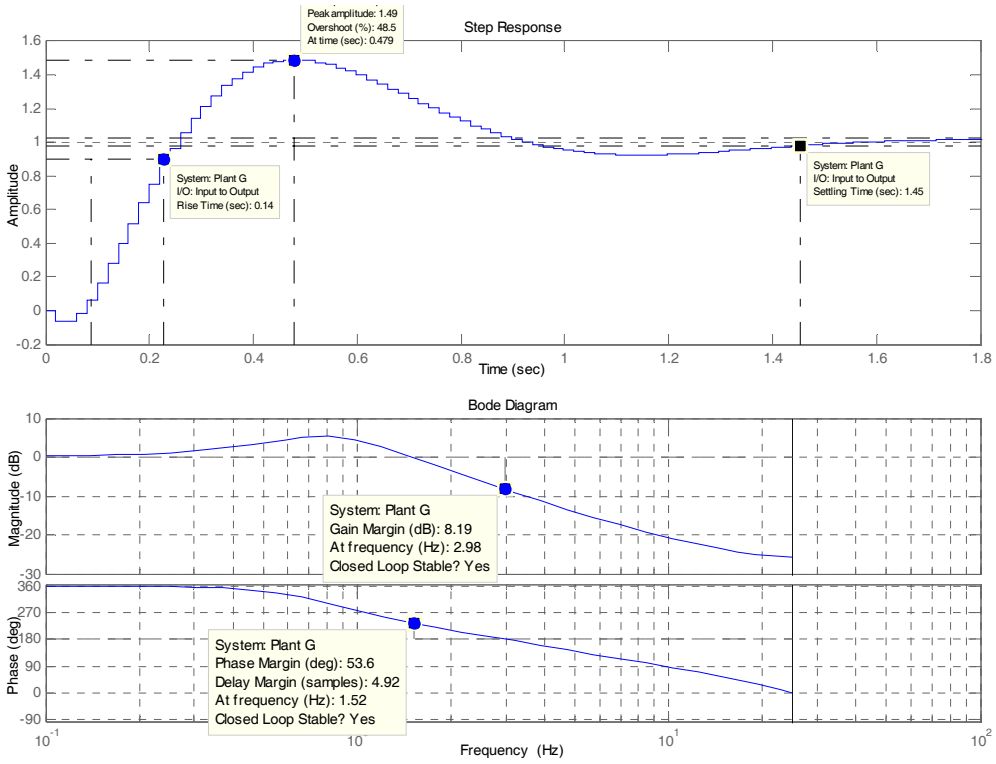


Figure 5-16 Step Response, Impulse Response and Bode Plot of CNT1-3

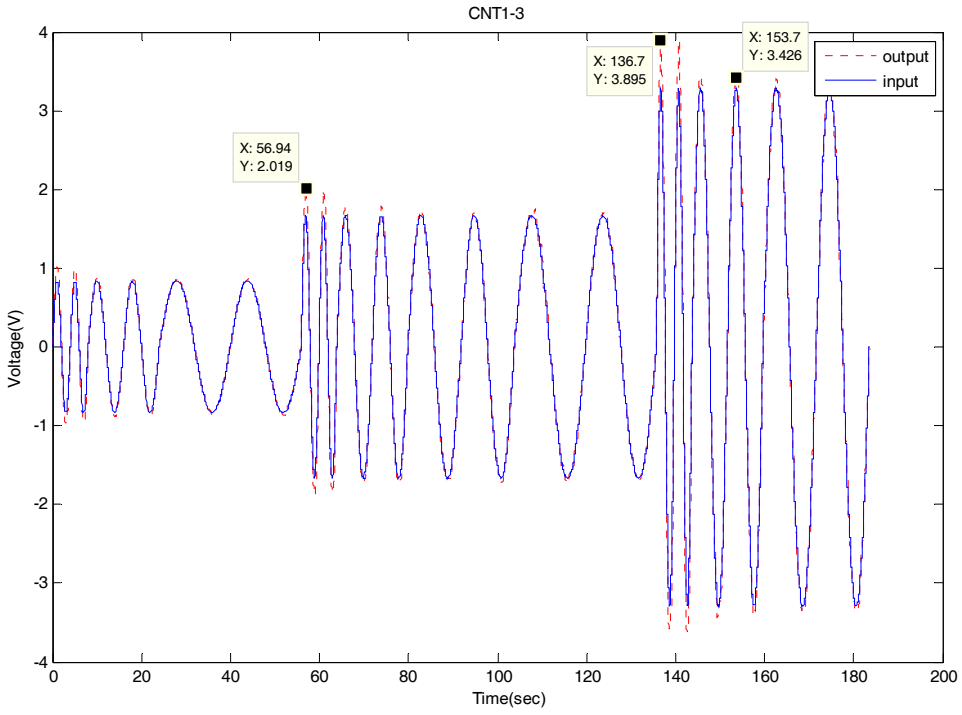


Figure 5-17 Input/Output Graph of CNT1-3

IMC Type 1-4

10% perturbation applied to the last three parameters of the IMC Controller Type 1 at the denominator part. 10% decreased according to original equation.

Transfer function of CNT 1-4

$$H_{CNT1\text{-}4}(z)=\frac{-0.03941z^6+0.2741z^5-0.7688z^4+1.123z^3-0.9057z^2+0.3839z^1-0.06697}{z^6-5.85z^5+14.27z^4-18.58z^3+12.258z^2-4.7934z^1+0.78192} \quad (5.30)$$

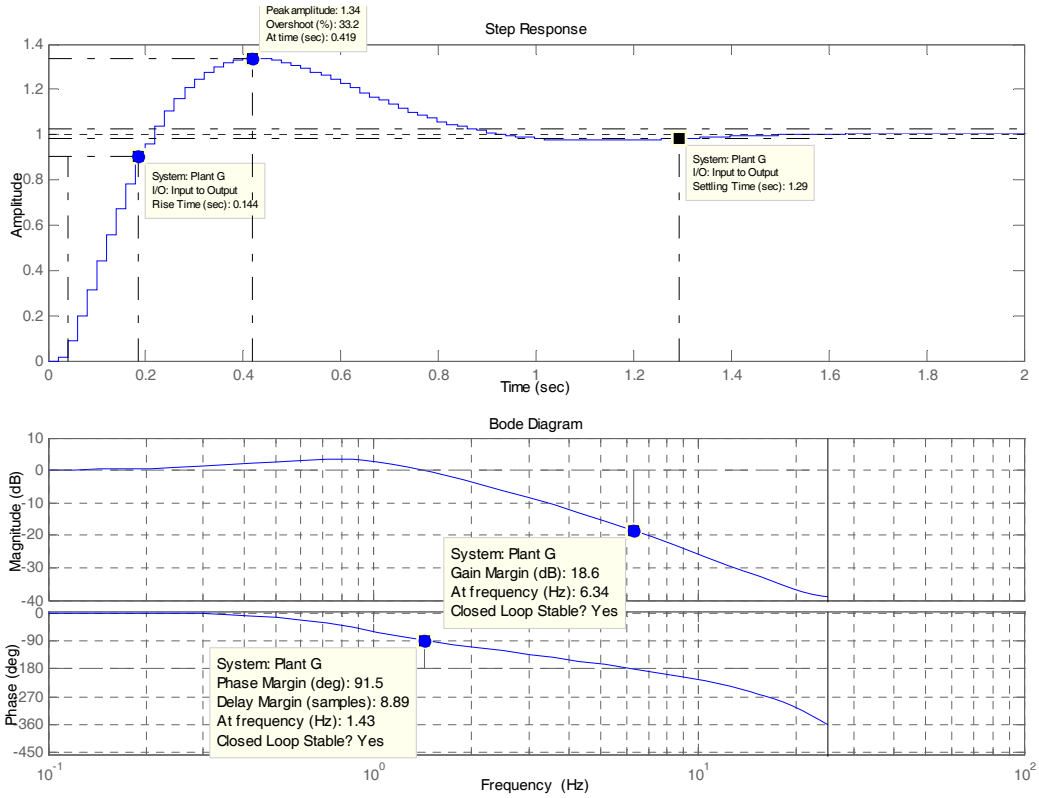


Figure 5-18 Step Response, Impulse Response and Bode Plot of CNT1-4

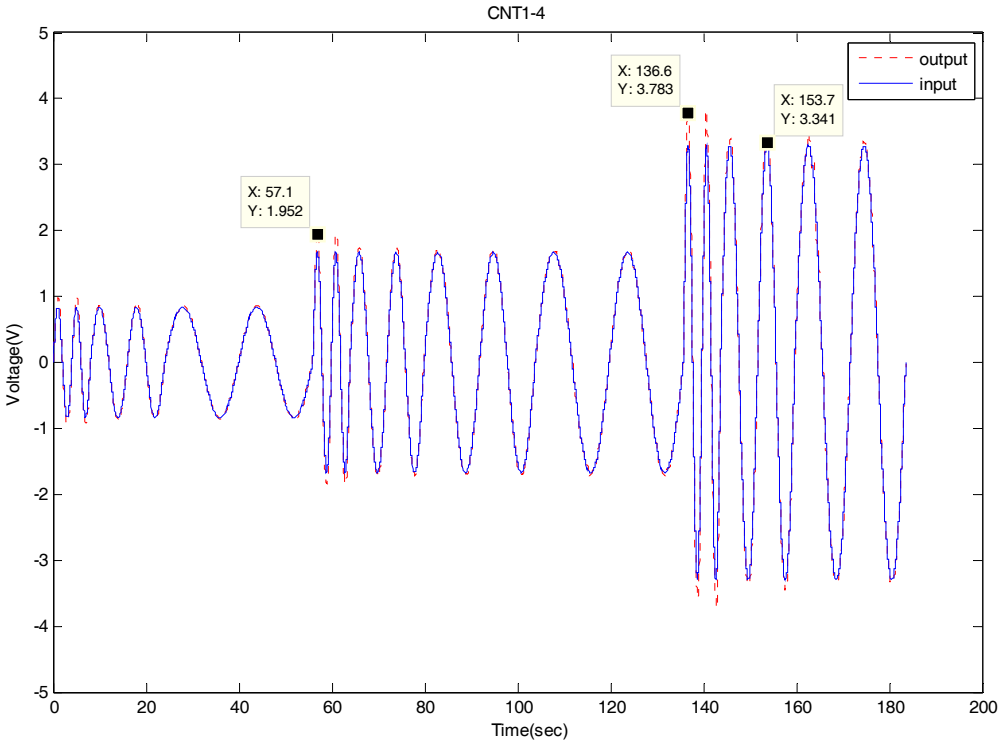


Figure 5-19 Input/Output Graph of CNT1-4

IMC Type 3-1

10% perturbation applied to the first five parameters of the IMC Controller Type 3 at the numerator part. 10% decreased according to original equation.

Transfer function of CNT 3-1

$$H_{CNT3\text{-}1}(z) = \frac{-0.036819z^4 + 0.18297z^3 - 0.31851z^2 + 0.2615z^1 - 0.07009}{z^4 - 3.936z^3 + 5.817z^2 - 3.825z^1 + 0.9443} \qquad (5.31)$$

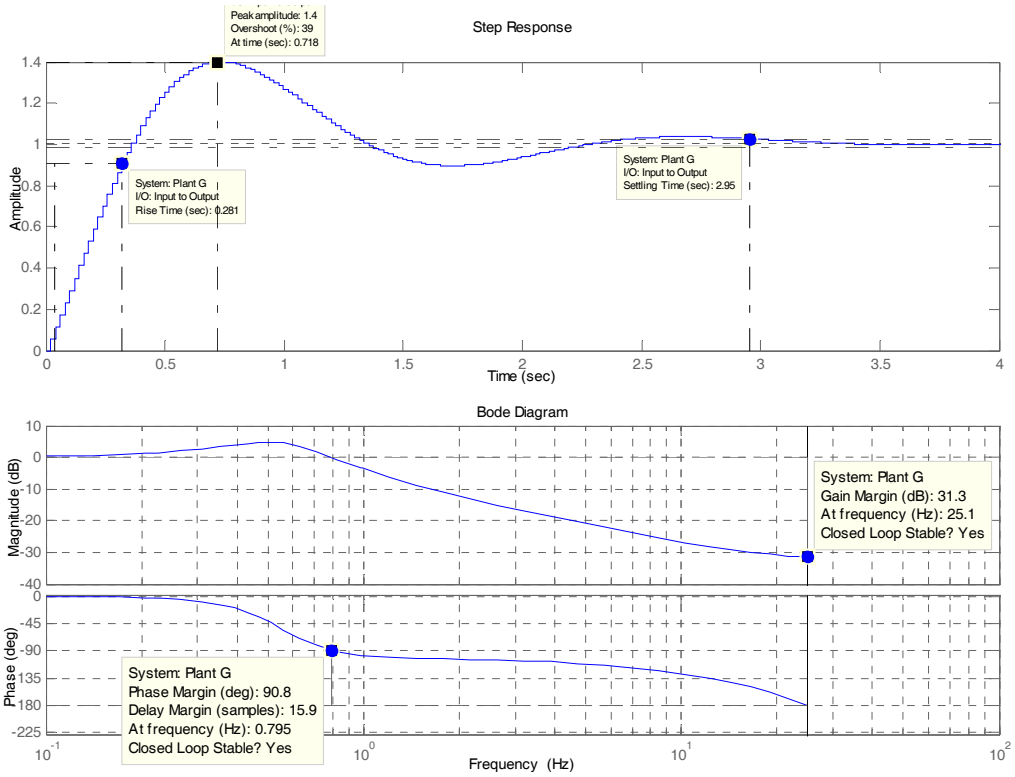


Figure 5-20 Step Response, Impulse Response and Bode Plot of CNT3-1

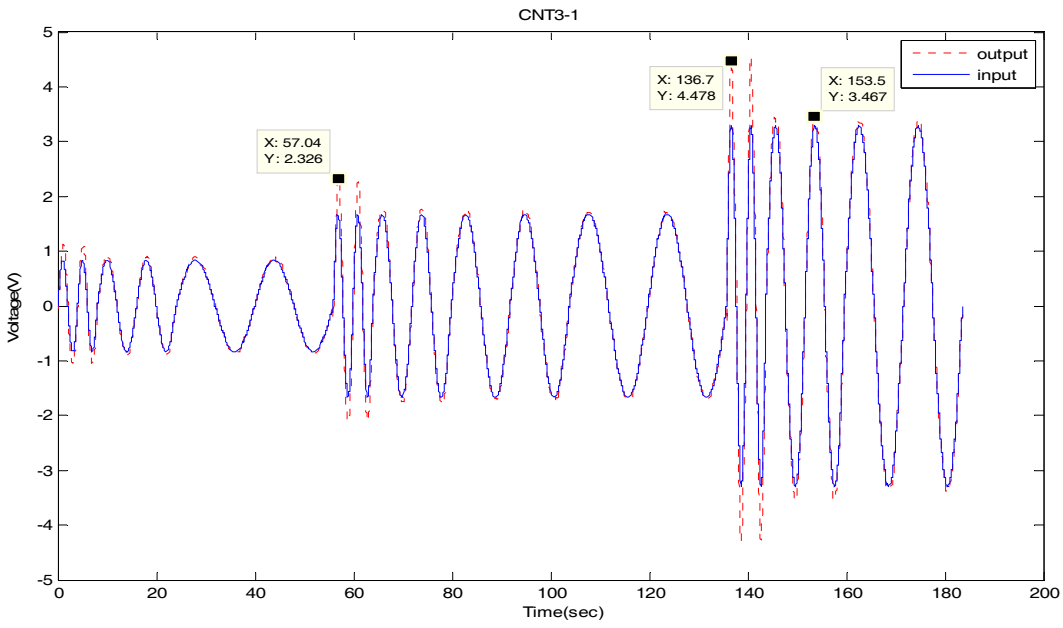


Figure 5-21 Input/Output Graph of CNT3-1

IMC Type 3-2

10% perturbation applied to the first five parameters of the IMC Controller Type 3 at the numerator part. 10% increased according to original equation.

Transfer function of CNT 3-2

$$H_{CNT3-2}(z)=\frac{-0.045001z^4+0.22363z^3-0.38929z^2+0.2615z^1-0.07009}{z^4-3.936z^3+5.817z^2-3.825z^1+0.9443} \tag{5.32}$$

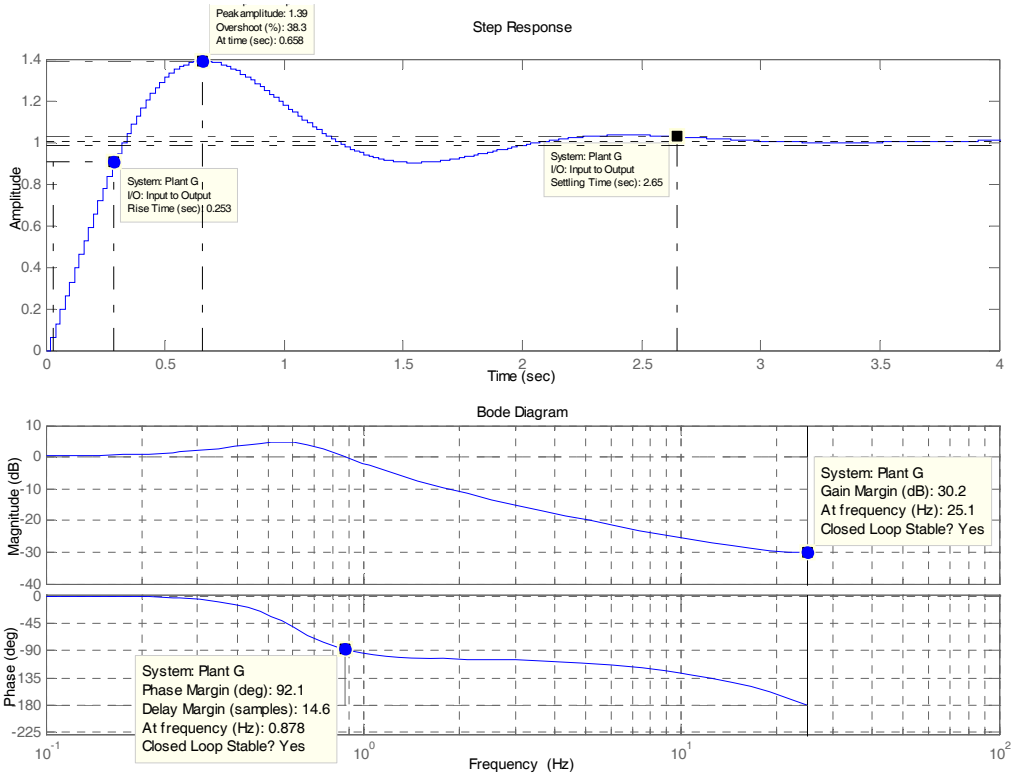


Figure 5-22 Step Response, Impulse Response and Bode Plot of CNT3-2

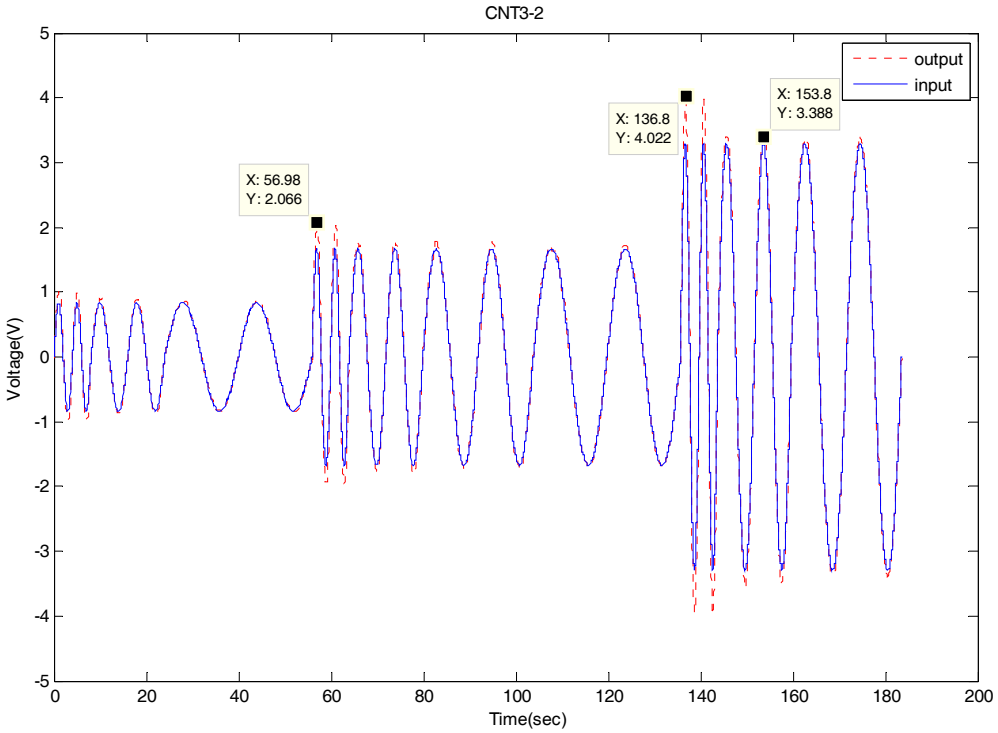


Figure 5-23 Input/Output Graph of CNT3-2

IMC Type 3-3

10% perturbation applied to the first three parameters of the IMC Controller Type 3 at the denominator part. 10% increased according to original equation.

Transfer function of CNT 3-3

$$H_{CNT3-3}(z) = \frac{-0.04091z^4 + 0.2033z^3 - 0.3539z^2 + 0.2615z^1 - 0.07009}{1.1z^4 - 4.3296z^3 + 6.3987z^2 - 3.825z^1 + 0.9443} \qquad (5.33)$$

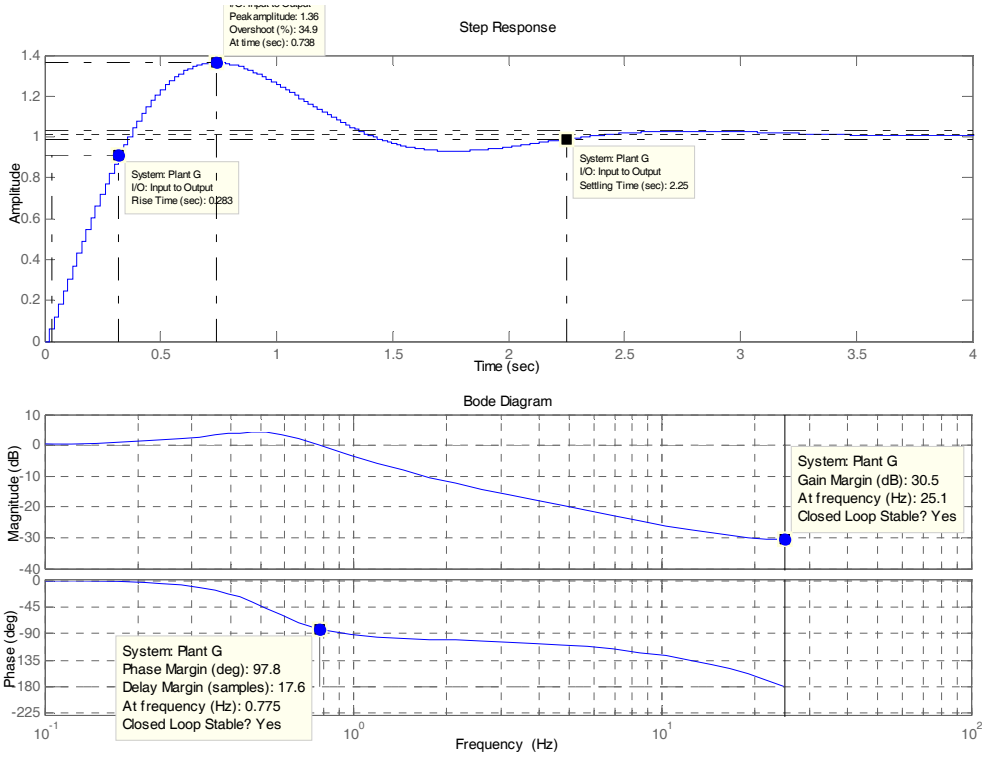


Figure 5-24 Step Response, Impulse Response and Bode Plot of CNT3-3

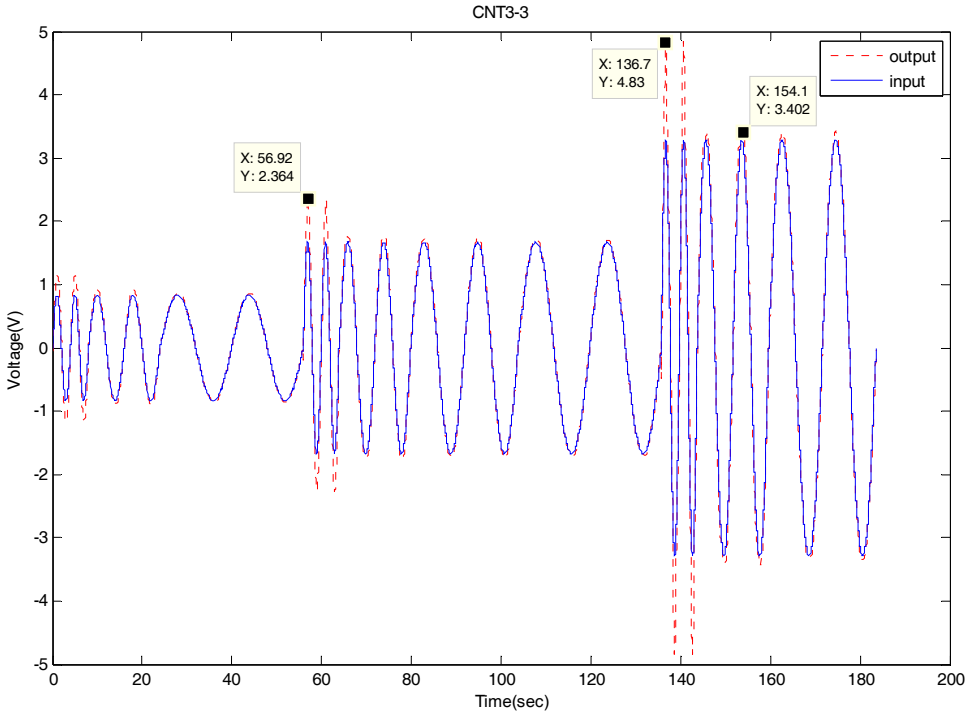


Figure 5-25 Input/Output Graph of CNT3-3

IMC Type 3-4

10% perturbation applied to the last three parameters of the IMC Controller Type 3 at the denominator part. 10% decreased according to original equation.

Transfer function of CNT 3-4

$$H_{CNT3-4}(z) = \frac{-0.04091z^4 + 0.2033z^3 - 0.3539z^2 + 0.2615z^1 - 0.07009}{1.1z^4 - 3.936z^3 + 5.2353z^2 - 3.4425z^1 + 0.84987} \qquad (5.34)$$

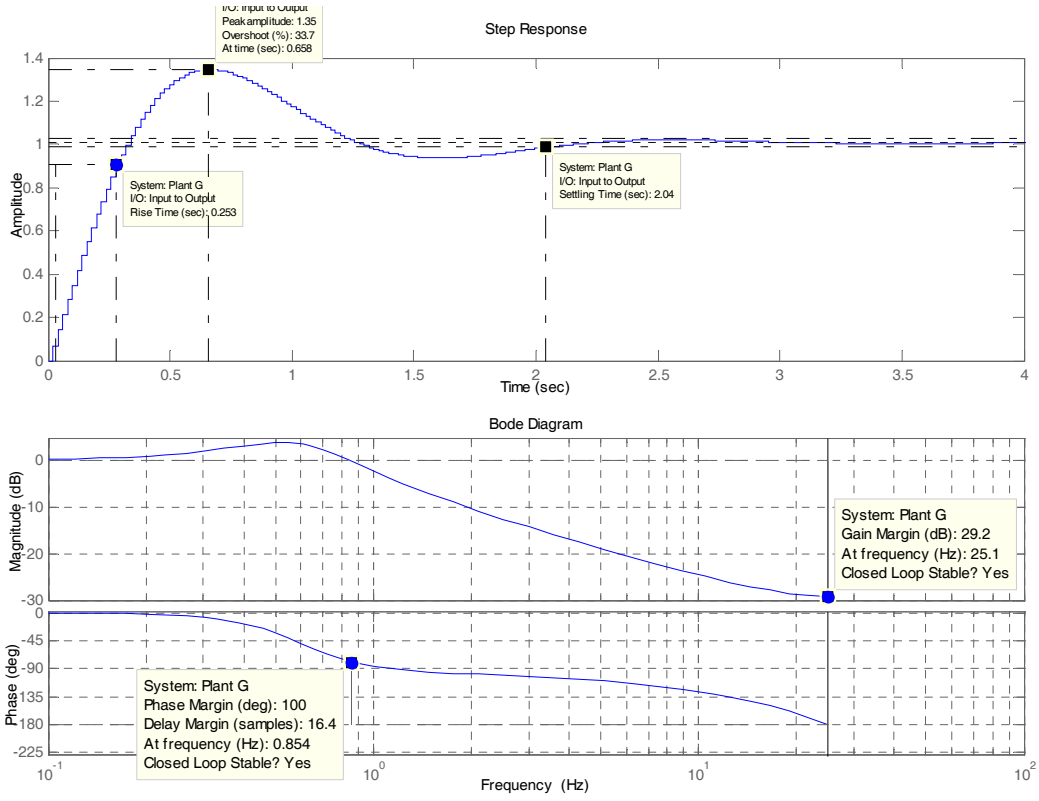


Figure 5-26 Step Response, Impulse Response and Bode Plot of CNT3-4

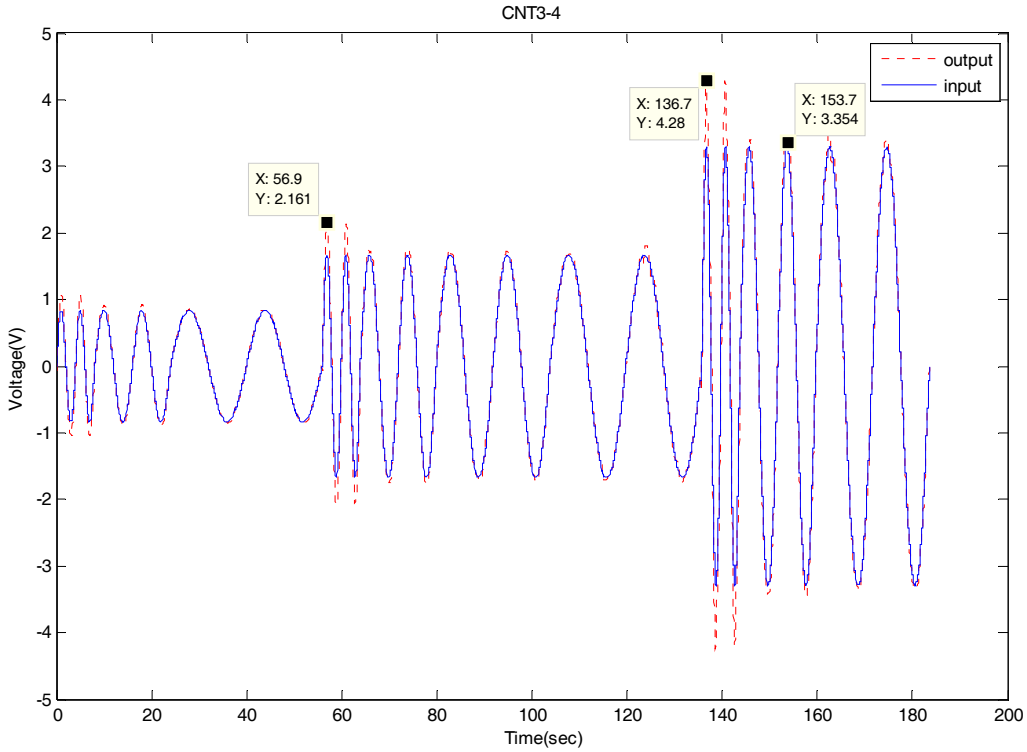


Figure 5-27 Input/Output Graph of CNT3-4

5.3.2.2 Results and Conclusion

Table 4 Perturbed IMC Controllers Characteristics

Controller	**T_r(s)**	**M_p / T_p(s)**	**T_s(s) (1%)**	**T_s(s) (5%)**	**GM (dB) / at f(Hz)**	**PM (dB) / at f(Hz)**	**Closed Loop Stable**
CNT 1	0.121	1.62 / 0.439	1.54	1.03	2.1 / 2.14	12.3 / 1.89	Yes
CNT 1-1	0.198	1.6 / 0.519	4.12	2.45	33.6 / 25.1	53.3 / 1.23	Yes
CNT 1-2	0.263	1,23 / 0.479	84.4	38.1	22.6 / 25.1	125 / 1.07	Yes
CNT 1-3	0.14	1.49 / 0.479	1.8	1.31	8.19 / 2.98	53.6 / 1.52	Yes
CNT 1-4	0.144	1.34 / 0.419	1.42	0.805	18.6 / 6.34	91.5 / 1.43	Yes
CNT 3	0.272	1.35 / 0.698	2.98	1.96	29.9 / 25.1	100 / 0.797	Yes
CNT 3-1	0.281	1.4 / 0.718	3.11	2.07	31.3 / 25.1	90.8 / 0.795	Yes
CNT 3-2	0.253	1.39 / 0.658	2.81	1.87	30.2 / 25.1	92.1 / 0.878	Yes
CNT 3-3	0.283	1.36 / 0.738	3.12	2.06	30.5 / 25.1	97.8 / 0.775	Yes
CNT 3-4	0.253	1.35 / 0658	2.78	1.83	29.2 / 25.1	100 / 0.854	Yes

As it was mentioned in the previous section, two controllers could be chosen as the controllers of the system. If the device won't be working at too high frequencies then CNT 3 would also be used, however if it will be working at too high frequencies then using CNT 1 instead of CNT 3 is a good idea. According to this result to see the affect of perturbations on the system, these two controllers CNT1 and CNT3 are chosen as the main controllers that are going to be perturbed.

Both for CNT1 and CNT3, four different types of perturbations are applied to the controllers. These are:

- 10% perturbation applied to the first five parameters of the specific IMC Controller at the numerator part. 10% decreased according to original equation.
- 10% perturbation applied to the first five parameters of the specific IMC Controller at the numerator part. 10% increased according to original equation.
- 10% perturbation applied to the first three parameters of the specific IMC Controller at the denominator part. 10% increased according to original equation.
- 10% perturbation applied to the last three parameters of the specific IMC Controller at the denominator part. 10% decreased according to original equation.

Both from the table and also from the input/output graphs, perturbations applied to the IMC controllers CNT1 and CNT3 do not affect on the system significantly in a negative way. However in some cases the requirement for settling time, which should be less than 3 seconds, is not satisfied. But in some cases, this requirement is also satisfied for steady state error 1% rather than 5%. Furthermore some perturbations changed some parameters in the positive way.

Moreover, to observe the effect of atmospheric occurrences, motion of the aircraft and wind, Gaussian noise is applied to the plant model. The response of the system in case of Gaussian noise is examined and given in Table 5. In this table, peak values taken from step response of the controllers are shown for each controller for different values of the variance of the Gaussian noise. Also the mean of Gaussian noise is set to 0, and sampling time is set to 0.00399sec.

Table 5 Effect of Gaussian Noise

Variance	CNT2	CNT3	CNT4	LQG9
0.0399	1.36	1.26	1.30	1.33
0.00399	1.2	1.16	1.15	1.04
0.000399	1.15	1.13	1.10	1.02
0	1.13	1.12	1.08	1.01

In Table 5, variance equals to 0 is referred to the case without any noise. For variance equals to 0.0399, the oscillation of the noise amplitude is between 0.8 and -0.6, for variance is 0.00399 this range is between 0.25 and -0.2 but for variance is 0.000399 range is between 0.08 and -0.06.

From Table 5, it is obvious that to some threshold point, the designed controller suppresses the Gaussian noise. For all the controllers shown above, this noise suppression property holds the system in the limits of the requirements. As an outcome of this simulation study, the designed controllers are robust in case of Gaussian noise to some extend.

Table 6 Perturbed IMC Controllers Measurements

Degree(°) / Freq(Hz)	Measured Values	CNT 1-1	CNT 1-2	CNT 1-3	CNT 1-4	CNT 3-1	CNT 3-2	CNT 3-3	CNT 3-4
5/0.25	InVol(V)	0.837	0.837	0.837	0.837	0.837	0.837	0.837	0.837
	OutVol(V)	1.0444	1.0169	1.0123	0.9737	1.1310	1.0068	1.1323	1.0627
	Freq(Hz)	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063
	Error %	24.73	21.49	20.94	16.33	35.12	20.28	35.28	26.96
5/0.125	InVol(V)	0.837	0.837	0.837	0.837	0.837	0.837	0.837	0.837
	OutVol(V)	0.8655	0.8664	0.8691	0.8590	0.8783	0.8912	0.8958	0.9012
	Freq(Hz)	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539
	Error %	3.41	3.51	3.83	2.63	4.93	6.47	7.02	7.67
5/0.0625	InVol(V)	0.837	0.837	0.837	0.837	0.837	0.837	0.837	0.837
	OutVol(V)	0.8664	0.8526	0.8535	0.8517	0.8884	0.8553	0.8563	0.8480
	Freq(Hz)	0.06265	0.06265	0.06265	0.06265	0.06265	0.06265	0.06265	0.06265

Degree(º) / Freq(Hz)	Measured Values	CNT 1-1	CNT 1-2	CNT 1-3	CNT 1-4	CNT 3-1	CNT 3-2	CNT 3-3	CNT 3-4
	Error %	3.51	1.86	1.97	1.76	6.14	2.18	2.54	1.31
10/0.25	InVol(V)	1.669	1.669	1.669	1.669	1.669	1.669	1.669	1.669
	OutVol(V)	2.149	2.0764	1.9926	1.9583	2.2919	2.0376	2.3599	2.1241
	Freq(Hz)	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063
	Error %	28.76	24.41	19.39	17.33	37.32	22.08	41.40	27.28
10/0.125	InVol(V)	1.669	1.669	1.669	1.669	1.669	1.669	1.669	1.669
	OutVol(V)	1.7339	1.7375	1.7402	1.7284	1.7448	1.7411	1.7511	1.7121
	Freq(Hz)	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539
	Error %	3.89	4.10	4.27	3.56	4.54	4.32	4.92	2.58
10/0.0835	InVol(V)	1.669	1.669	1.669	1.669	1.669	1.669	1.669	1.669
	OutVol(V)	1.6931	1.6804	1.7339	1.7076	1.7384	1.7620	1.7030	1.7266
	Freq(Hz)	0.08354	0.08354	0.08354	0.08354	0.08354	0.08354	0.08354	0.08354
	Error %	1.44	0.68	3.89	2.31	4.16	5.57	2.04	3.45
10/0.0625	InVol(V)	1.669	1.669	1.669	1.669	1.669	1.669	1.669	1.669
	OutVol(V)	1.6894	1.7067	1.7040	1.6691	1.7058	1.6831	1.6876	1.6840
	Freq(Hz)	0.06265	0.06265	0.06265	0.06265	0.06265	0.06265	0.06265	0.06265
	Error %	1.22	2.26	2.10	0.005	2.20	0.84	1.11	0.90
20/0.25	InVol(V)	3.287	3.287	3.287	3.287	3.287	3.287	3.287	3.287
	OutVol(V)	4.4067	4.0457	3.8422	3.8092	4.4460	4.0448	4.8304	4.2569
	Freq(Hz)	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063	0.25063
	Error %	34.06	23.08	16.89	15.88	35.26	23.05	46.95	29.51
20/0.125	InVol(V)	3.287	3.287	3.287	3.287	3.287	3.287	3.287	3.287
	OutVol(V)	3.472	3.4678	3.4006	3.3601	3.3644	3.3851	3.3661	3.3834
	Freq(Hz)	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539	0.12539
	Error %	5.63	5.50	3.46	2.22	2.35	2.98	2.41	2.93
20/0.0835	InVol(V)	3.287	3.287	3.287	3.287	3.287	3.287	3.287	3.287
	OutVol(V)	3.2883	3.3350	3.3790	3.4162	3.3471	3.3454	3.3920	3.5845
	Freq(Hz)	0.08354	0.08354	0.08354	0.08354	0.08354	0.08354	0.08354	0.08354
	Error %	0.04	1.46	2.80	3.93	1.83	1.78	3.19	9.05

On the tables including controllers' measurements, there are some parameters taken from the 'input vs. output' graphs of each controller. To explain these parameters some abbreviations are used in the table which are already explained in section 5.3.1.2.

On the system, 8 different perturbed IMC controllers are examined. From the data taken the graphs and the table shown above the following observations are made.

CNT1-1: From Table 4, Rise Time, Peak value, Settling Time parameters are a little worse than CNT1 but Gain Margin, Phase Margin are better than CNT1. Also from Table 6, when we look at the percent error it is observed that CNT1-1 has worse response to the same input. Finally choosing CNT1 as the controller of the system instead of CNT1-1 is a better solution. However, there is nothing changed deeply.

CNT1-2: From Table 4, Rise Time, Peak value, Settling Time parameters are worse than CNT1 but Gain Margin, Phase Margin are better than CNT1. Also from Table 6, when we look at the percent error it is observed that CNT1-2 and CNT1 do not have big differences. Finally choosing CNT1 as the controller of the system instead of CNT1-2 is a better solution. However, there is nothing changed deeply.

CNT1-3: From Table 4, almost all the parameters are better than CNT1. From Table 6, when we look at the percent error it is observed that CNT1-3 and CNT1 are almost similar. Finally choosing CNT1-3 as the controller of the system instead of CNT1 is a better solution. However, there is nothing changed deeply.

CNT1-4: From Table 4, almost all the parameters are better than CNT1. From Table 6, when we look at the percent error it is observed that CNT1-4 has obvious better response to the same input. Finally choosing CNT1-4 as the controller of the system instead of CNT1, CNT1-1, CNT1-2 and CNT1-3 is obviously a better solution.

CNT3-1: From Table 4, Rise Time, Peak value, Settling Time, Phase Margin parameters are a little worse than CNT3 but Gain Margin is better than CNT3. Also from Table 6, when we look at the percent error it is observed that CNT3-1 has a little worse response to the same input. Finally choosing CNT3 as the

controller of the system instead of CNT3-1 is a better solution. However, there is nothing changed deeply.

CNT3-2: From Table 4, almost all the parameters are similar to CNT3. From Table 6, when we look at the percent error it is observed that CNT3-2 has better response to the same input. Finally choosing CNT3-2 as the controller of the system instead of CNT3 is a better solution. However, there is nothing changed deeply.

CNT3-3: From Table 4, almost all the parameters are better than CNT3. From Table 6, when we look at the percent error it is observed that CNT3-3 and CNT3 are almost similar. Finally choosing CNT3-3 or CNT1 as the controller of the system does not matter. However, there is nothing changed deeply.

CNT3-4: From Table 4, almost all the parameters are a little better than CNT3. From Table 6, when we look at the percent error it is observed that CNT3-4 has obvious better response to the same input. Finally choosing CNT3-4 as the controller of the system instead of CNT3, CNT3-1, CNT3-2 and CNT3-3 is obviously a better solution.

Considering the given data in the table, the comparison between all the controllers can be made. The data shown in Table 6 is taken from the real system. That is why the controller that satisfies our requirements and gives best response to the sinusoidal input is the best one of all. Some more observations about our requirements and explanation are already given in 5.3.1.2 Results and Conclusion section.

From the tables all the controllers CNT1, CNT1-1, CNT1-2, CNT1-3, CNT1-4, CNT3, CNT3-1, CNT3-2, CNT3-3 and CNT3-4 satisfy the requirements also for 0.25063Hz excluding 20º. However, for CNT1-4 and CNT3-4, these controllers satisfy the requirements also for 20º, at 0.25 Hz. For its real, wanted working edge, the error is under 4% which is considerably smaller than 55% that is why

these controllers may be used as the controllers of the system. This comparison can be made for all different input voltage ranges.

In general, perturbations applied to the system does not affect on the system in a negative way, even some perturbations make the system response better than the previous version. Especially perturbation type 4 which is; 10% perturbation applied to the last three parameters of the specific IMC Controller at the denominator part, 10% decreased according to original equation, has extremely positive affect on the system. Therefore, applying perturbation type 4 to the system is a good idea.

5.3.3 LQG Controller Design

For the controller design purpose, LQG type controllers are designed. The results that are collected from the original system are shown below.

The equations and graphs shown below belong to these controllers that are closed loop response of the real system;

5.3.3.1 Designed Controllers and Characteristics

LQG Type 1

Transfer function of LQG-1

$$H_{LQG-1}(z) = \frac{-0.2029z^4 + 0.4251z^3 + 0.006269z^2 - 0.4753z^1 + 0.2468}{z^4 - 1.982z^3 + 0.01223z^2 + 1.93z^1 - 0.9598} \qquad (5.35)$$

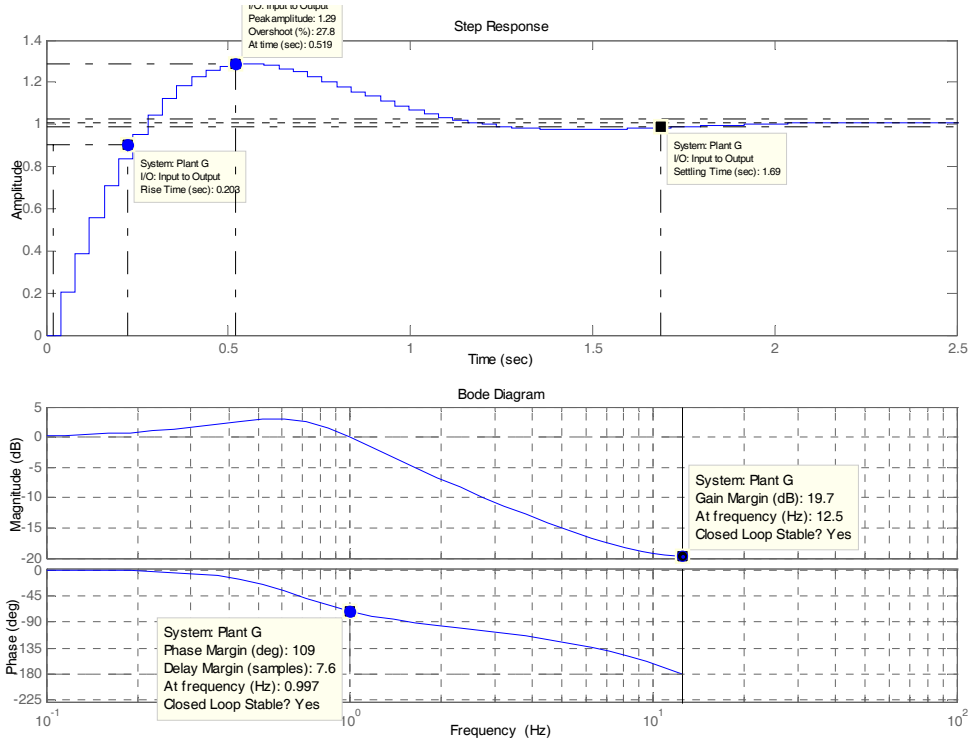


Figure 5-28 Step Response, Impulse Response and Bode Plot of LQG-1

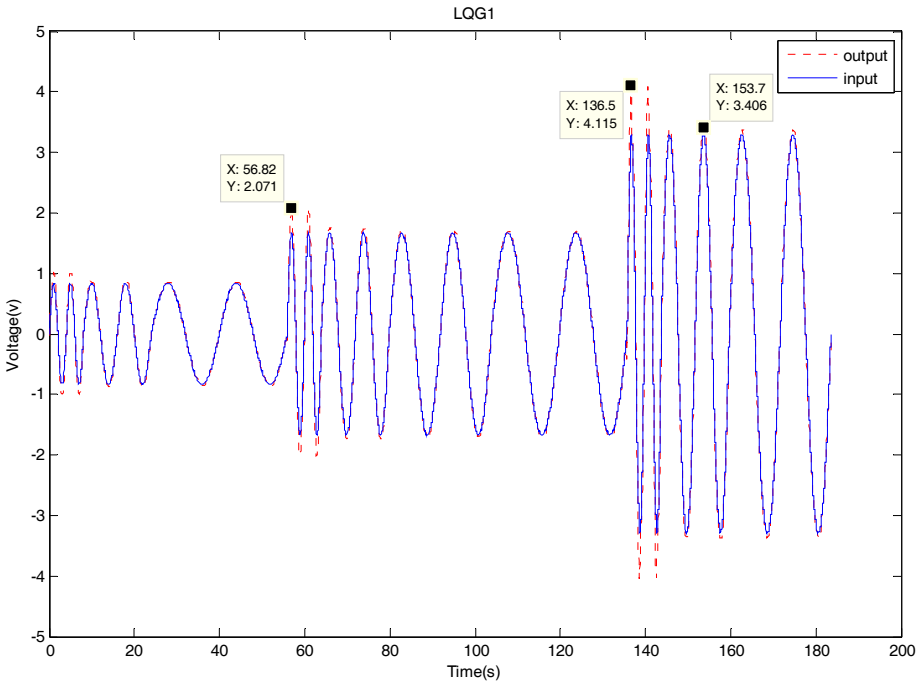


Figure 5-29 Input/Output Graph of LQG-1

LQG Type 2

Transfer function of LQG-6

$$H_{LQG\text{-}6}(z)=\frac{-0.6113z^{6}+0.7494z^{5}+0.3668z^{4}-0.07262z^{3}-0.3586z^{2}-0.5895z^{1}+0.5161}{z^{6}-1.185z^{5}-0.6294z^{4}+0.01838z^{3}+0.6435z^{2}+1.132z^{1}-0.9794} \quad (5.36)$$

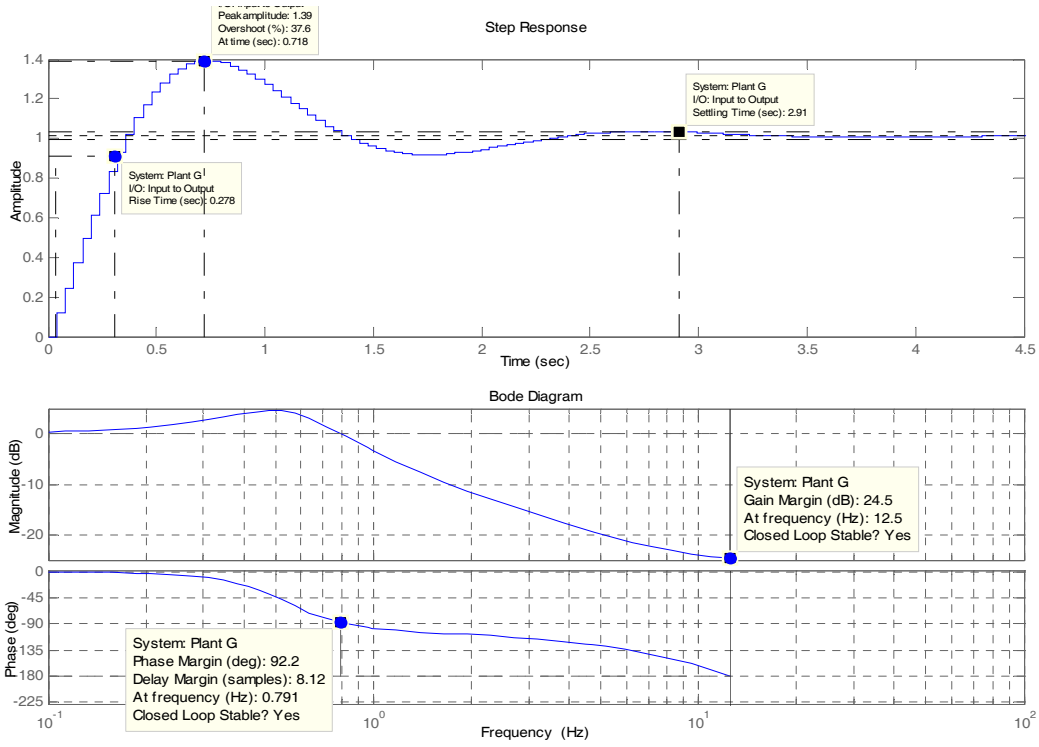


Figure 5-30 Step Response, Impulse Response and Bode Plot of LQG-6

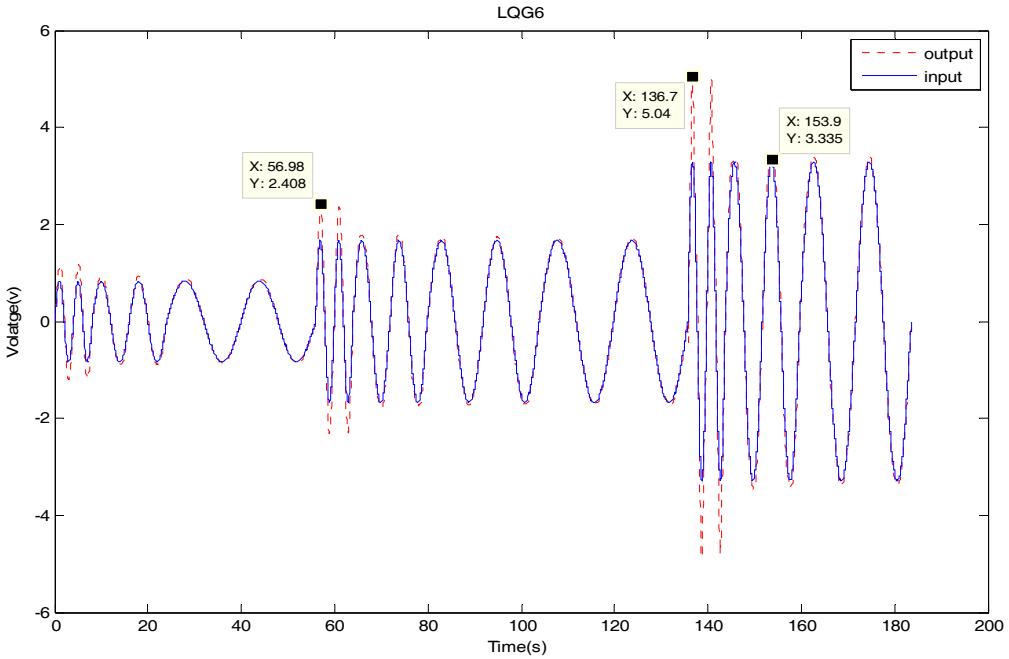


Figure 5-31 Input/Output Graph of LQG-6

LQG Type 3

Transfer function of LQG-9

$$H_{LQG-9}(z)=\frac{-0.1787z^5+0.3985z^4-0.2271z^3+0.1719z^2-0.362z^1+0.1974}{z^5-2.124z^4+1.416z^3-1.393z^2+2.042z^1-0.9419} \qquad (5.37)$$

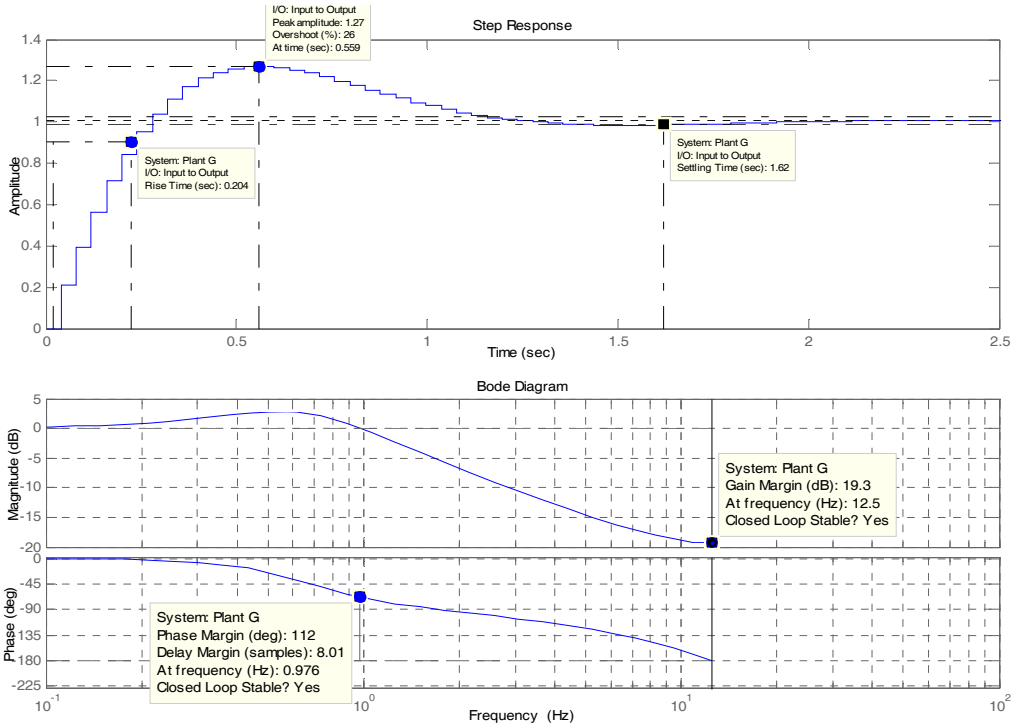


Figure 5-32 Step Response, Impulse Response and Bode Plot of LQG-9

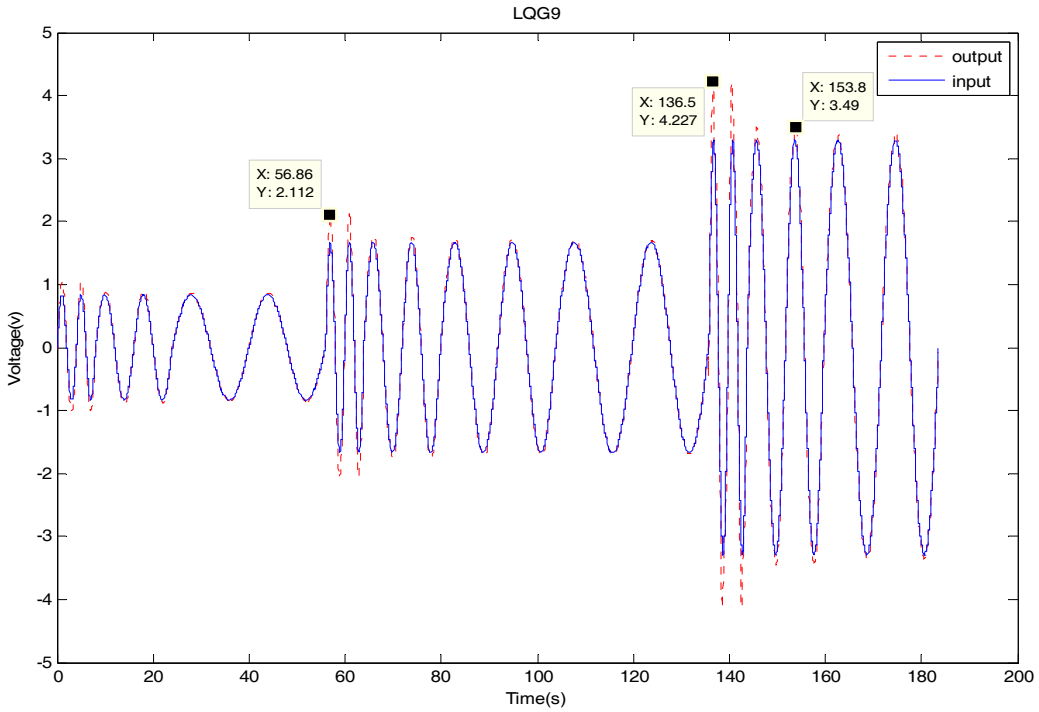


Figure 5-33 Input/Output Graph of LQG-9

5.3.3.2 Results and Conclusion

Table 7 LQG Controllers Characteristics

Controller	T_r(s)	M_p / T_p(s)	T_s(s) (1%)	T_s(s) (5%)	GM (dB) / at f(Hz)	PM (dB) / at f(Hz)	Closed Loop Stable
LQG-1	0.203	1.29 / 0.519	1.84	1.02	19.7 / 12.5	109 / 0.997	Yes
LQG-6	0.278	1.39 / 0.718	3.14	2.08	24.5 / 12.5	92.2 / 0.791	Yes
LQG-9	0.204	1.27 / 0.559	1.86	1.05	19.3 / 12.5	112 / 0.976	Yes

The controllers and their properties are mentioned before but to give a brief description; different LQG type controllers were examined on the system but similar ones are shown above. These different LQG types are designed by changing some parameters on the controller.

Both from the table and also from the input/output graphs, LQG-6 has worse response then the other two LQG-1 and LQG-9. Also rise time, settling time and peak value of LQG-6 is higher. Therefore the expected output response for LQG-6 is worse than the others.

When we look at the settling times of the controllers, as the steady state error percent decreases to 1%, only LQG-6 does not satisfy the requirement of settling time should be less than 3 seconds. Both LQG-1 and LQG-9 satisfy the requirement for settling time. This means that controllers LQG-1 and LQG-9 also works whenever the requirements for steady state and settling time are made more strict.

Table 8 LQG Controllers Measurements

Degree(º) / Freq(Hz)	Measured Values	LQG-1	LQG-6	LQG-9
5/0.25	InVol(V)	0.837	0.837	0.837
	OutVol(V)	1.0077	1.1433	1.0251
	Freq(Hz)	0.25063	0.25063	0.25063
	Error %	20.39	36.59	22.47

Degree(º) / Freq(Hz)	Measured Values	LQG-1	LQG-6	LQG-9
5/0.125	InVol(V)	0.837	0.837	0.837
	OutVol(V)	0.8590	0.9141	0.8517
	Freq(Hz)	0.12539	0.12539	0.12539
	Error %	2.63	9.21	1.76
5/0.0625	InVol(V)	0.837	0.837	0.837
	OutVol(V)	0.8480	0.8489	0.8489
	Freq(Hz)	0.06265	0.06265	0.06265
	Error %	1.31	1.42	1.42
10/0.25	InVol(V)	1.669	1.669	1.669
	OutVol(V)	2.0674	2.3759	2.0989
	Freq(Hz)	0.25063	0.25063	0.25063
	Error %	23.87	42.35	25.76
10/0.125	InVol(V)	1.669	1.669	1.669
	OutVol(V)	1.7375	1.7819	1.7339
	Freq(Hz)	0.12539	0.12539	0.12539
	Error %	4.10	6.76	3.89
10/0.0835	InVol(V)	1.669	1.669	1.669
	OutVol(V)	1.7040	1.7384	1.7040
	Freq(Hz)	0.08354	0.08354	0.08354
	Error %	2.10	4.16	2.10
10/0.0625	InVol(V)	1.669	1.669	1.669
	OutVol(V)	1.6831	1.7030	1.6949
	Freq(Hz)	0.06265	0.06265	0.06265
	Error %	0.85	2.04	1.55
20/0.25	InVol(V)	3.287	3.287	3.287
	OutVol(V)	3.9618	5.0400	3.9046
	Freq(Hz)	0.25063	0.25063	0.25063
	Error %	20.53	53.33	18.79
20/0.125	InVol(V)	3.287	3.287	3.287
	OutVol(V)	3.3929	3.3350	3.4678
	Freq(Hz)	0.12539	0.12539	0.12539
	Error %	3.22	1.46	5.50
20/0.0835	InVol(V)	3.287	3.287	3.287
	OutVol(V)	3.3540	3.3929	3.2692
	Freq(Hz)	0.08354	0.08354	0.08354
	Error %	2.04	3.22	-0.54

On the tables including controllers' measurements, there are some parameters taken from the 'input vs. output' graphs of each controller. To explain these parameters some abbreviations are used in the table which are already explained in 5.3.1.2 Results and Conclusion part.

From the data taken the graphs and the table shown above the following observations are made.

Considering the given data in the table, the comparison between all the controllers can be made. The data shown in Table 8 is taken from the real system. That is why the controller that satisfies our requirements and gives best response to the sinusoidal input is the best one of all. Some more observations about our requirements and explanation are already given in 5.3.1.2.

From the tables all the controllers LQG-1, LQG-6 and LQG-9 satisfy the requirements also for 0.25063Hz excluding 20º. However, for LQG-1 and LQG-9, these controllers satisfy the requirements also for 20º, at 0.25 Hz. For its real, wanted working edge, the error is under 4% which is too small that is why these controllers may be used as the controllers of the system. This comparison can be made for all different input voltage ranges. Also when we look at the rise time, peak value, settling time, gain margin and phase margin, both LQG-1 and LQG-9 have similar almost same values which are all better than LQG-6. So either LQG-1 or LQG-9 can be used as the controller of the system. They both satisfy the requirements.

5.3.4 System Ramp Response to Some Controllers

According to the designed controllers above, to see the performance of the system to ramp input, at different degrees the following experiments are done. These experiments are done for 5 different controllers. These controllers are CNT 1, CNT1-4, CNT3-4, LQG-1, and LQG-9. The details about these controllers are already mentioned.

The equations and graphs shown below belong to these controllers that are closed loop response of the real system;

5.3.4.1 Designed Controllers and Characteristics

For the following figures below, all the dashed lines are showing the input applied to the system and solid lines are showing the response of the system.

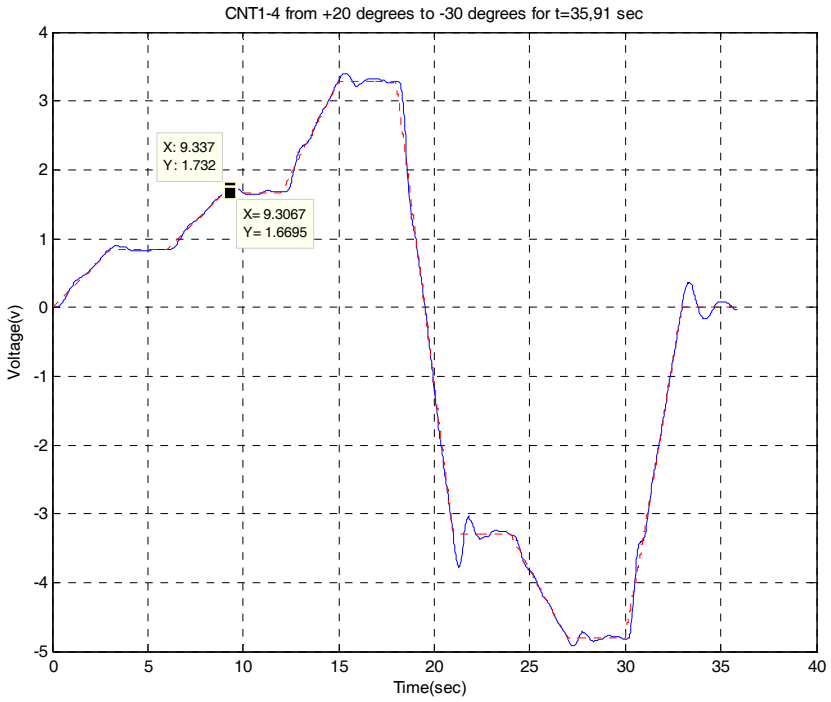


Figure 5-34 Input/Output Graph of CNT1-4 from +20 to -30 degrees t=35,91sec

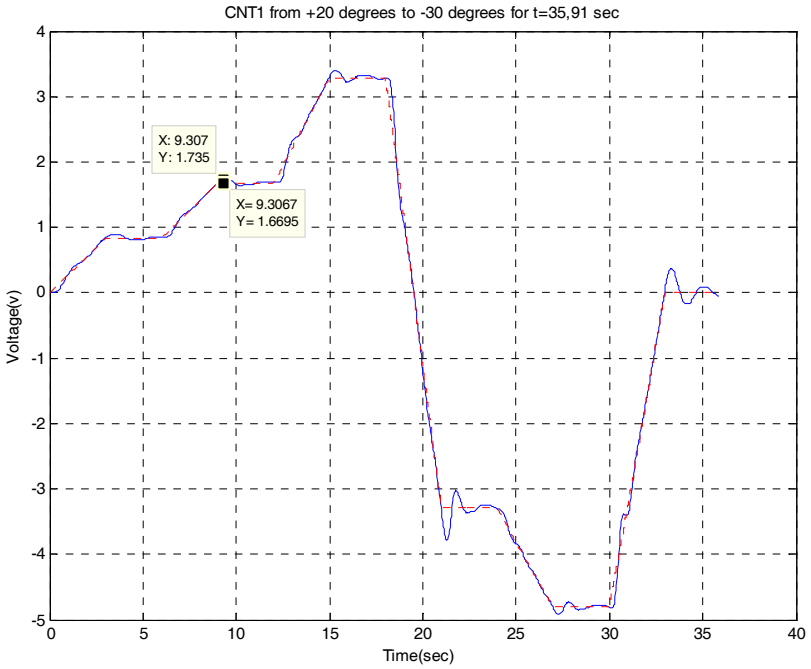


Figure 5-35 Input/Output Graph of CNT 1 from +20 to -30 degrees t=35,91sec

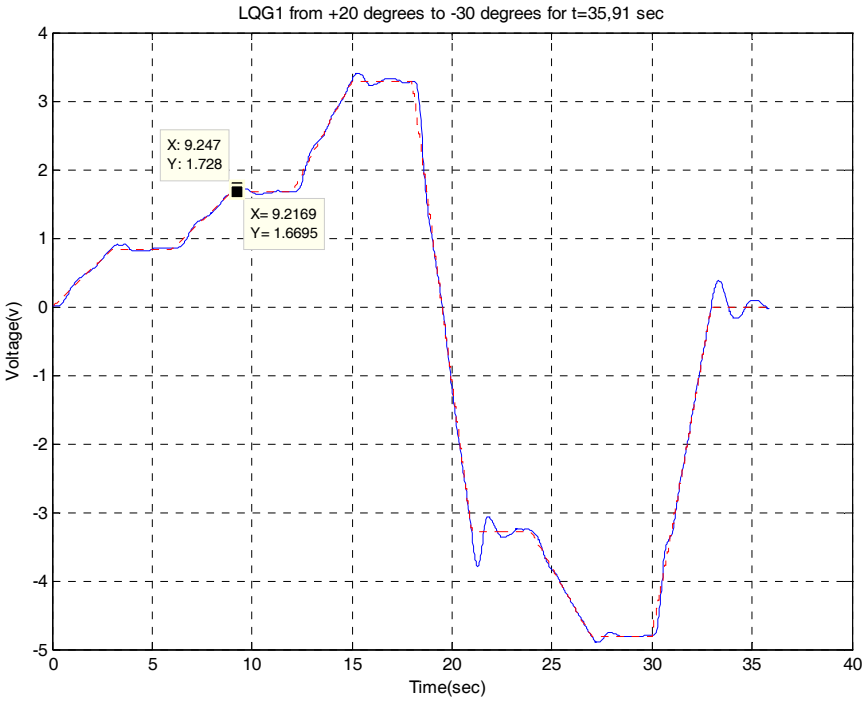


Figure 5-36 Input/Output Graph of LQG-1 from +20 to -30 degrees t=35,91 sec

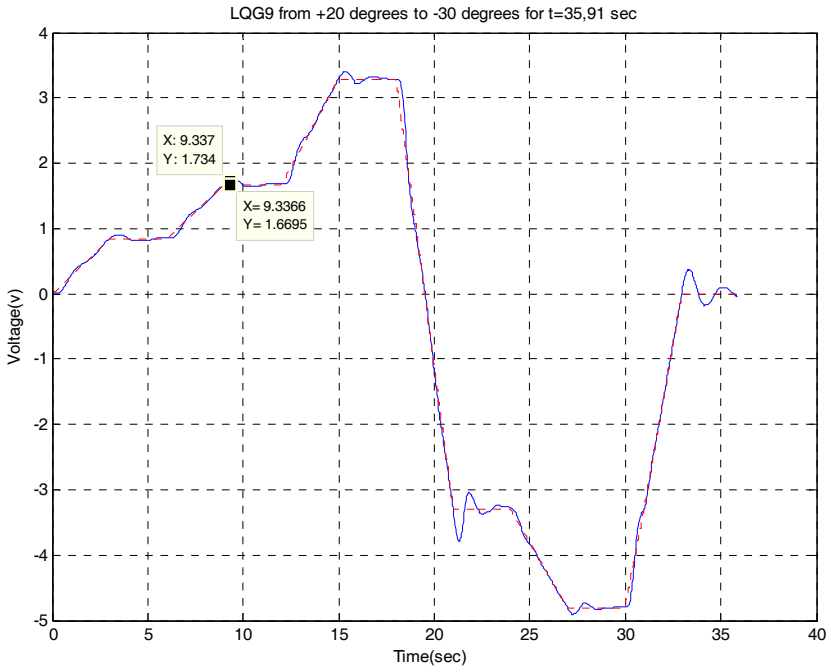


Figure 5-37 Input/Output Graph of LQG-9 from +20 to -30 degrees t=35,91 sec

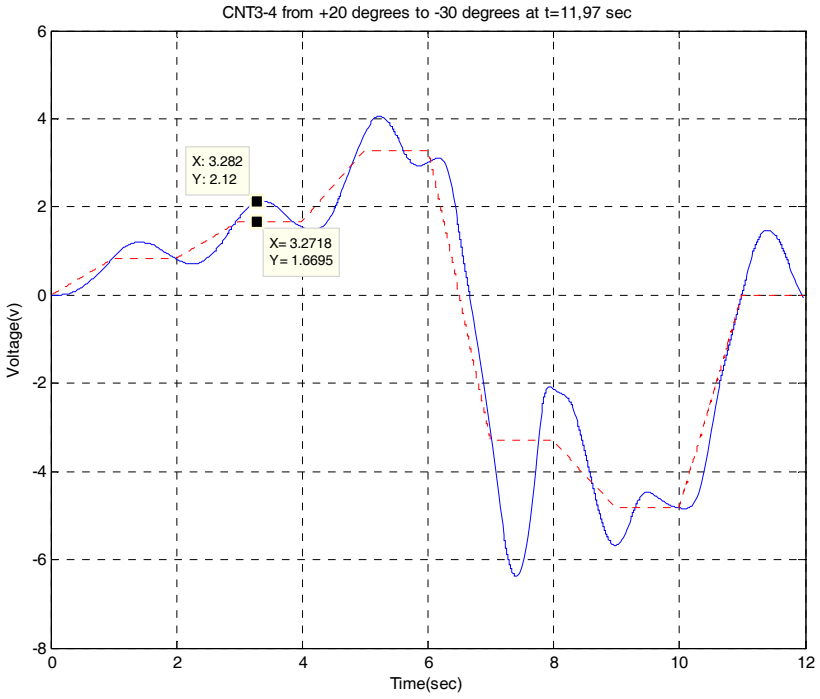


Figure 5-38 Input/Output Graph of CNT3-4 from +20 to -30 degrees t=11,97sec

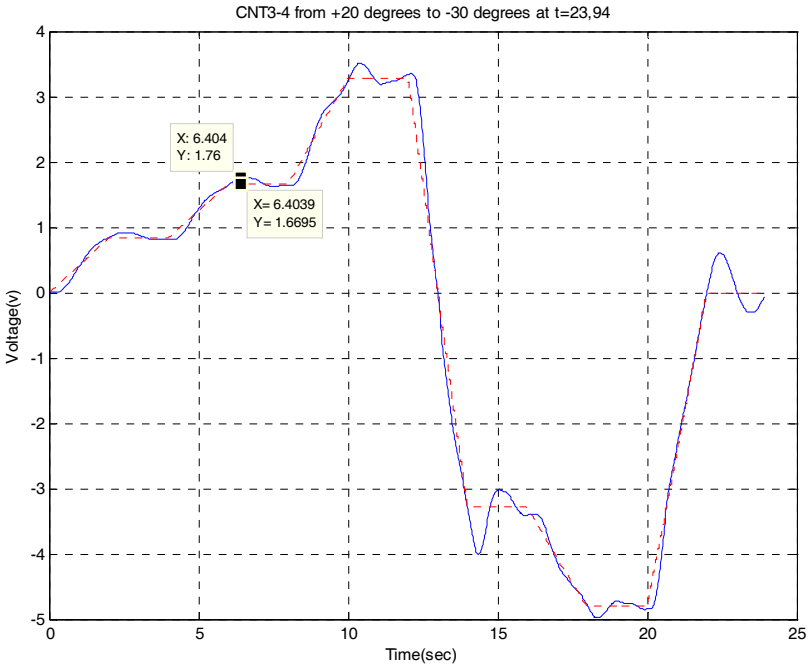


Figure 5-39 Input/Output Graph of CNT3-4 from +20 to -30 degrees t=23,94 sec

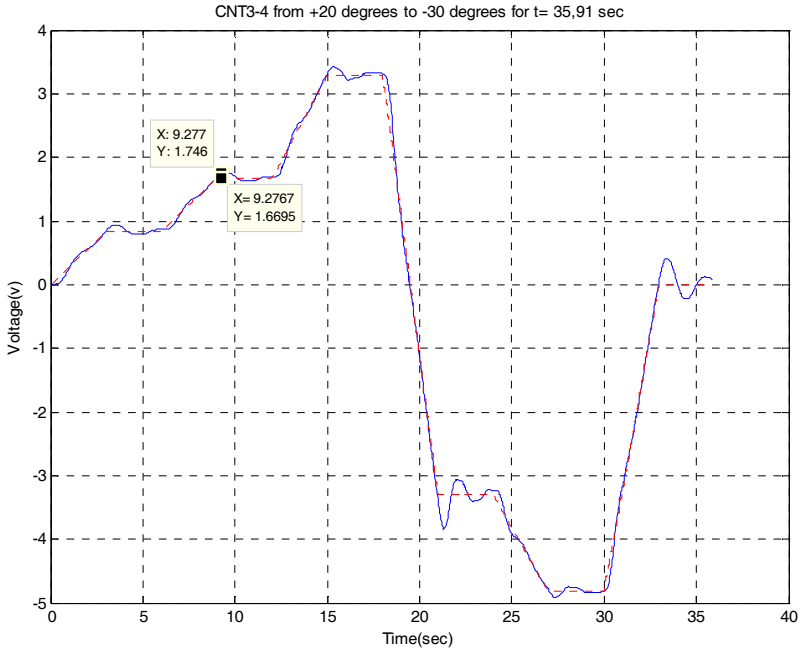


Figure 5-40 Input/Output Graph of CNT3-4 from +20 to -30 degrees t=35,91 sec

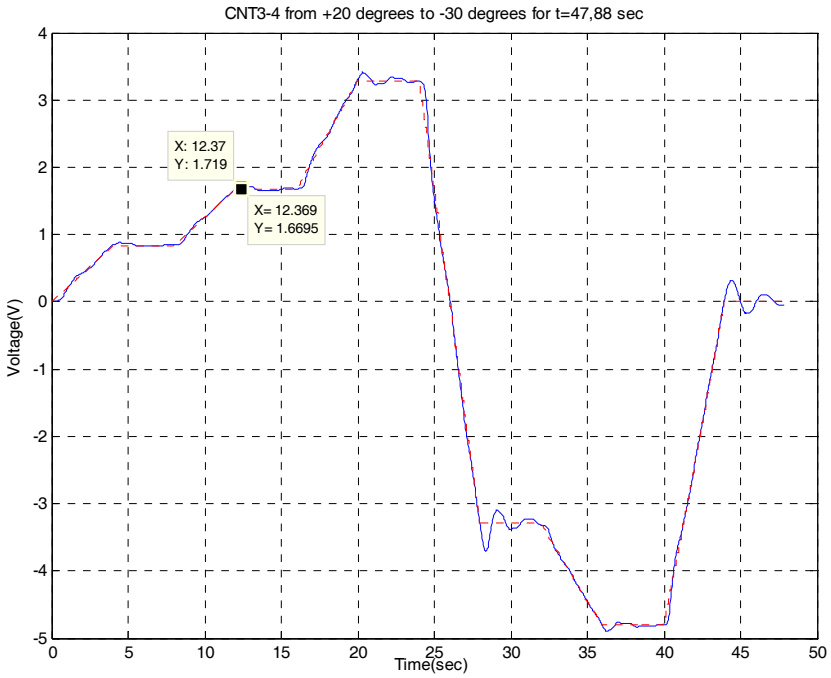


Figure 5-41 Input/Output Graph of CNT3-4 from +20 to -30 degrees t=47,88sec

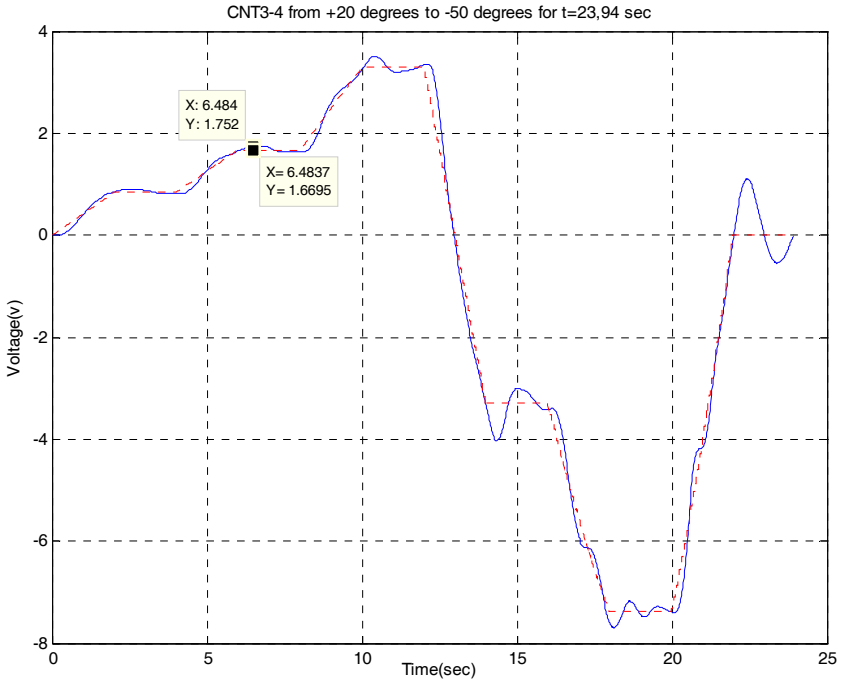


Figure 5-42 Input/Output Graph of CNT3-4 from +20 to -50 degrees t=23,94sec

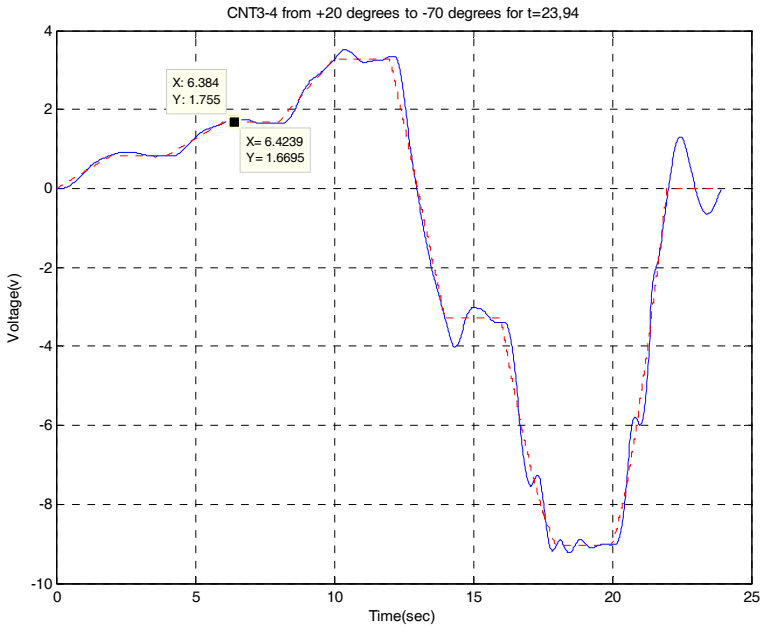


Figure 5-43 Input/Output Graph of CNT3-4 from +20 to -70 degrees t=23,94sec

5.3.4.2 Results and Conclusion

For all of the calculations the input voltage is 1.6695v. This is the voltage level for 10˚.

Table 9 Controllers Response

Controller Type	**Parameters**	**t =11,97 s**	**t =23,94 s**	**t =35,91 s**	**t =47,88 s**
CNT1-4	Output(V)	1,899	1,76	1,732	1,719
	Max Change(%)	13,75	5,42	3,74	2,96
CNT 1	Output(V)	1,897	1,765	1,735	1,719
	Max Change(%)	13,63	5,72	3,92	2,96
CNT3-4	Output(V)	2,12	1,76	1,746	1,719
	Max Change(%)	26,99	5,42	4,58	2,96
LQG-1	Output(V)	1,919	1,767	1,728	1,716
	Max Change(%)	14,94	5,84	3,50	2,78
LQG-9	Output(V)	1,936	1,767	1,734	1,72
	Max Change(%)	15,96	5,84	3,86	3,02

By looking at the figures shown above from Figure 5-34 to Figure 5-43 and the Table 9 the following observations are derived.

First of all, all the figures and tables are obtained by collecting 1200 different data for each set. The values taken at t = 11,97 means that this 1200 data set is completed in 11,97 seconds so a data is taken at every 9,975msec. For a movement from +20˚ to -30˚ means 50˚ of movement there are 100 data set at 0,9975 seconds. This range is high for our system to response correctly and it won't be used at this speed because it may give damage to the system at long term and also this speed is not needed and not used for real systems. So the data taken under the limits of our system isn't important but to give a description about the controllers response at t = 11,97 sec, the controllers CNT1-4, CNT 1, LQG-1 and LQG-9 has approximately 10˚ of error from +20˚ to -30˚ but CNT3-4 has approximately 7˚ of error from +20˚ to -30˚. For the remaining time periods, all the controllers satisfy the requirements from +20˚ to -30˚ range. Also for CNT3-4, the +20˚ to -50˚ range and +20˚ to -70˚ range is scanned at t = 23,94 seconds. At these ranges, the response of the system is in the limits.

In general it can be said that, ramp responses of the controllers are as expected and indicating us that the controllers are working as required. IMC type controllers with perturbation give better response then the LQG type controllers when we look at the figures given above.

5.3.5 The Comparison between Simulation and Real Life

All the designed controllers and their responses on the real system were given above. These controllers are designed to meet the requirements. To see the difference between simulation results and real system response the following observations are made. The examined controllers are CNT 2 and CNT 3.

Here are the simulation graphs and tables that belong to the selected controllers;

5.3.5.1 Designed Controllers and Characteristics

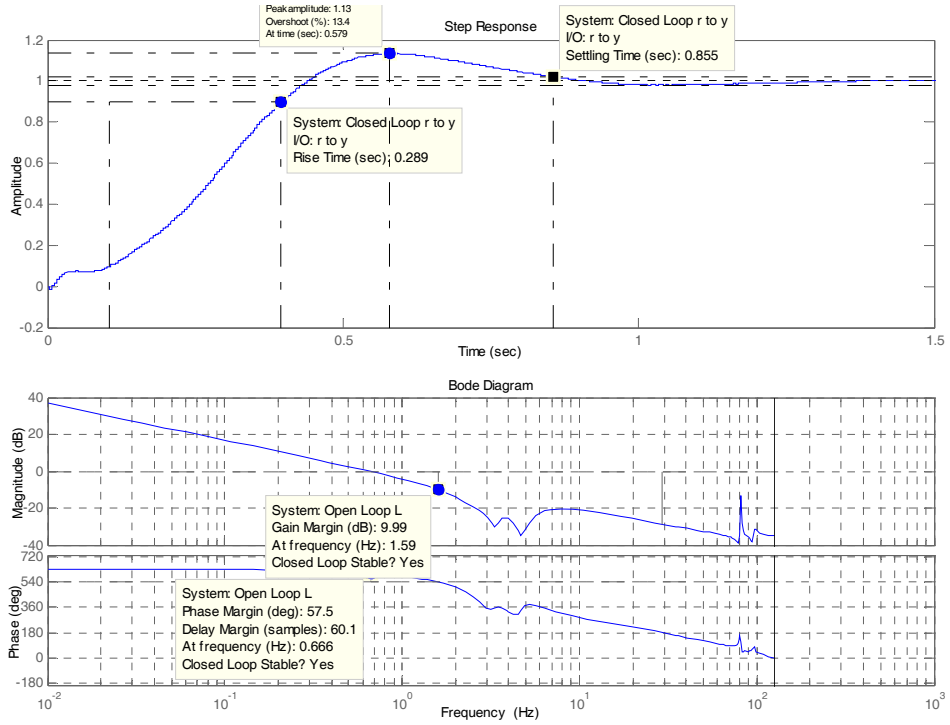


Figure 5-44 Step Response and Closed Loop Bode Plot of designed CNT 2

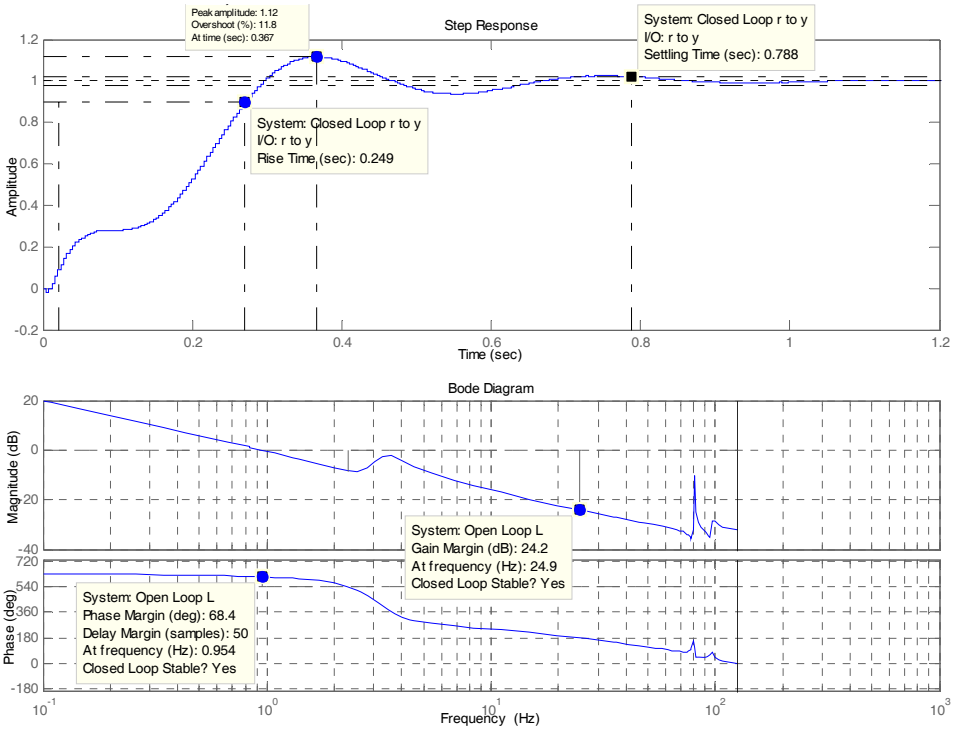


Figure 5-45 Step Response and Bode Plot of designed CNT 3

5.3.5.2 Results and Conclusion

Table 10 Controllers Characteristics

Controller	T_r(s)	M_p / T_p(s)	T_s(s) (1%)	T_s(s) (5%)	GM (dB) / at f(Hz)	PM (dB) / at f(Hz)	Closed Loop Stable
CNT 2 real	0.17	1.74 / 0.519	3.76	2.25	6.21 / 1.83	15.7 / 1.42	Yes
CNT 2 simulation	0.289	1.13 / 0.439	1.22	0.786	9.99 / 1.59	57.5 / 0.666	Yes
CNT 3 real	0.272	1.35 / 0.698	2.98	1.96	29.9 / 25.1	100 / 0.797	Yes
CNT 3 simulation	0.249	1.12 / 0.698	0.828	0.589	24.2 / 24.9	68.4 / 0.954	Yes

According to the given requirements in the 5.1.2 section, suitable controllers are designed. Two of them are shown here to see the difference between the designed controllers response and real time response of the controllers. Both from the Figure 5-44, Figure 5-45 and from the Table 10, the following observations can be made.

There are some differences between the measured response and calculated response. For instance, the percent overshoot increases when the same controller is used in the system. Also the settling time, phase and gain margins get worse. This is most probably because of the modeling error and the noise affecting on the real system. During the linearization of the model, some nonlinear data is not taken into consideration that is why the model is not an one to one model of the system. This is mostly because of the complexity of the system. However, when we are designing a controller, if the parameters are chosen more flexible then, some unknown errors can be tolerated. It is obvious that the simulated results are better than real system response but as it was shown and mentioned before the real time system response holds the error signal within the 5.5° positioning envelope in the wanted speed of system motion.

CHAPTER 6

CONCLUSION

Gun turret systems are the inevitable systems in the battlefield. They have to be precise, because they are wanted to hit the target as desired. Therefore the stabilization and control of these systems are critical issues for such systems. A good performance of the turret system is directly related to successful control of such system. For this purpose, the gun turret system is examined in detail and controllers are designed to hit the target in the required range through this work.

The study is basically composed of real turret system identification, modeling and controlling in the desired range. First of all, the system is described and then to communicate with the system a control interface is designed. The communication is achieved by designing an intelligent unit called interface unit. With help of this communication interface, the system is modeled. By using the model, suitable controllers are designed in the lights of requirements. These designed controllers are tested on the real system and analyzed to observe their performance. By using their performance parameters, most suitable controller for our system is determined. By designing a suitable controller that meets the requirements, the precise control of turret system is achieved.

For complex systems like turret systems using robust controllers is a good approach that reveals promising results. This study represents high performance,

robust, real time control of turret system with the help of hardware and software environment design. Implementation of such setup is given and explained in order to execute real time control of the system. According to our analysis for turret systems robust controllers are more effective than classical control methods. For testing purposes and making some observations on the classical control response, PI controller is examined on the real system and it is observed that these controllers are not effective for such complex systems. PI controller is not closed loop stable, and it is worse than all the other controller types. Therefore, PI controller is not a suitable controller type for the control of this turret system. However, the simulation results of PI controller show that, using lead and lag filters with PI controllers is a good approach to meet the system requirements. These filters are effective in improving system response.

In order to analyze the effect of controllers both IMC and LQG type controllers are examined. It is known that IMC type controllers are preferable if the device is unique and LQG type controllers are used to satisfy stability to uncertainties. Both of the controllers are used for robust control of the systems. From the experiments and measured data, it is obvious that both IMC and LQG type controllers satisfy the requirements. Therefore using these types of controllers for our case and also for similar turret systems is a good approach. Actually, the results of both controllers are similar to each other, so making a comparison between them is difficult. They are both suitable for our case. But in general, LQG type controllers give more stable and reasonable response as a result of its characteristics.

The characteristics and the effect of designed controller on the system are performed both on the real system and mathematical model. The effect of changes in the parameters, on the performance of the system is analyzed. As it was mentioned before, IMC type controllers are used in general for unique system. For our case, the device is unique but it is also examined by applying some perturbations on these controllers to include the production errors, or some other unknown disturbances. The results show that these variations do not make big changes so with these changes the controllers continue to meet the system

requirements. Therefore, it is seen that the designed and implemented controllers are robust to changes in parameters.

Real system response to ramp input is also examined. The ramp response of the system matches with the expected response to ramp input. Therefore the ramp response of the system also indicates that LQG and IMC type controllers are suitable and satisfy the requirements.

As given through the work, both the designed and the measured system responses satisfy the system requirements. However, it is also worth to mention that designed and measured responses slightly differ for parameters such as overshoot, rise time, settling time, phase margin and gain margin in an acceptable range. This small difference is because of the some unknown system parameters, loss during linearization, changes in test setup, coulomb friction, servo limitation and noise on the system. Moreover, there are also some other factors which affects the system response and causes the difference between the simulation results and actual system response. Linearization of non-linear elements, unknown noise, system instabilities, and temperature changes may be the factors that cause change.

As future work, different controller types can be studied, and some improvements could be made in the control algorithms and in the modeling. The effect of uncertainties could be modeled and analyzed in detail. Also, by using an exact model of an aircraft, the effect of wind, motion and some other atmospheric occurrences could be taken into consideration. Moreover, the elevation motion of the system can be examined and integrated with azimuth motion characteristics. Then the working principle in the real battlefield could be tested.

To sum up, a robust controller for the control of turret systems is developed in this thesis. The system identification and modeling of the system are examined. The results show that, it is possible to meet the requirements with precise system identification and modeling in order to obtain a robust control of the turret system of an aircraft as demonstrated in our study.

REFERENCES

[1] M. Athens, "A Tutorial on the LQG/LTR Method," Proceedings of American Control Conference, Seattle, pp 1289-1296, June 1986.

[2] M. Atharts and G. Stein, "The LOG/LTR Procedure for Multivariable Feedback Control Design," IEEE Trans. on Automatic Control, Vol. AC-32, No.2 , pp 105-114, Feb 1987.

[3] American GNC Corp. (AGNC), "High precision, adaptive, robust weapon control system design," Phase I report to ARDEC, Contract NO. DAAA21-91-C-0042, 1991.

[4] S. Banks, N. Coleman, C.F. Lin and T.J. Yu, "Advanced Integrated Control of Nonlinear Flexible Pointing Systems," Proceedings of the Third IEEE Conference on Control Applications, Glasgow, pp.947-952, 1994.

[5] N. Coleman, M. Mattice, S. Banks, C. F. Lin, J. C. Juang, and J. Huang, "High Precision Weapon Control System Design," First IEEE Conference on Control Applications, pp. 726-731, 13-16 Sept 1992.

[6] S.V. Rao, M.S. Mattice and N.P. Coleman, "Design Of Reduced Order LQG/LTR Controllers for Turret-Gun System," Proceedings of American Control Conference, Pittsburgh, June 1989, pp. 312-315.

[7] "Turret Systems", Federation of American Scientists, 2008, Last Access Date: 22 January 2009, <http://www.fas.org/man/dod-101/sys/ac/equip>.

[8] "Turret," TACOM Rock Island Webmaster, September 2008, Last Access Date: 12 February 2009, <http://tri.army.mil/LC/cs/Csa/aawpns.htm>.

[9] O. Nelles, "Nonlinear System Identification," Springer-Verlag, Berlin Heidelberg, Germany, 2001. QA402.N45 2000.

[10] L. Andersson, U Jonsson, K. H. Johansson, J. Bengstsson, "A Manual For System Identification," In Laboratory Exercises in System Identification, KF Sigma I Lund AB. Department of Automatic Control, Lund Institute of Technology, Box 118, S-221 00 Lund, Sweden.

[11] G. Pedersen, IRS7-2006, "System Identification Lecture1," Aalborg Universitet Esbjerg, September 2006, Last Access Date: 05 January 2009, <http://www.cs.aaue.dk/contribution/courses/fall2006/IRS7/SI/lecture1.handout-nup4.pdf>.

[12] "Selecting a Model Structure in the System Identification Process," National Instruments Corporation, 1 July 2008, Last Access Date: 08 January 2009, <http://zone.ni.com/devzone/cda/tut/p/id/4028>.

[13] "Output Error," The Mathworks Inc., 1994-2009, Last Access Date: 06 January 2009, <http://www.mathworks.com>.

[14] S. Banks, N. Coleman, C.F. Lin and T.J. Yu, D.P. Atherton "Real Time, High Performance, Adaptive, Robust Control System Design," Proceedings of the American Control Conference, Seattle, Washington June 1995.

[15] M. Mattice, N. Coleman, S. Banks, J. C.Jnang, and C. F. Lin. “Robust Weapon Control Systems Design”, In Proceedings of Ameriaan Control Conference, 1992.

[16] T. H. S. Li, C. F. Lin, J. C. Jnang, N. Coleman M. Mattice, and S. Banks. “Fuzzy Logic Control of Gun Turret System”, In Proceedings of American Control Conference, 1992.

[17] D.A. Mohutsiwa, “PID Tuning Controller Using Internal Model Control Method,” Bachelor of Engineering Thesis, University of Southern Queensland, Faculty of Engineering and Surveying.

[18] C. E. Garcia and M. Morari, “Internal model control. 1. A unifying review and some new results,” *Ind. Eng. Chem. Process Des. Dev.*, vol. 21, pp. 308–323, 1982.

[19] M.N. Tham “Introduction to Robust Control / Internal Model Control,” Newcastle University, School of Chemical Enginnering and Advanced Materials, January 2002, Last Access Date: 22 January 2009, <http://lorien.ncl.ac.uk/ming/robust/imc.pdf>.

[20] C. Economou, M. Morari, B. Palsson, “Internal Model Control 5. Extensions to Nonlinear Systems,” Ind. Eng. Process Des. Dev., vol. 25, pp. 403–411, Jul. 1986.

[21] X.R. Li, Y.B. Shalom, T. Kirubarajan, “Estimation with Applications to Trackingand Navigation,” John Wiley & Sons, 2001.

[22] K. Isıyel, “Autopilot Design and Guidance Control of Ulisar UUV (Unmanned Underwater Vehicle),” A thesis Submitted to the Graduate School of Natural and Applied Sciences of Middle East Technical University, September 2007.

[23] D. S. Naidu, “Optimal Control Systems”, CRC Press, 2003.

[24] M. Ergun, “Robust Control of A Highly Maneuverable Aircraft,” A thesis Submitted to the Graduate School of Natural and Applied Sciences of Middle East Technical University, April 1998.

[25] C.F. Lin, “Modern Navigation, Guidance and Control Processing,” Volume 2, Prentice Hall Series in Advanced Navigation, Guidance and Control and their applications.

[26] J.C. Doyle, G. Stein, “Multivariable Feedback Design: Concepts for a Classical modern Synthesis,” IEEE Transaction on Automatic Control, Vol.26, No 1, pp.4-16, 1981.

[27] J.M. Maciejowski, “Multivariable Feedback Design,” Addison-Wesley Publishing Company, Inc., 1989.

[28] J.K Astrom, B. Wittenmark, “Computer Controlled System,” Prentice Hall Inc, 1984.

[29] D. Villarosa, “Difference Equation Generation For The Lamps Az/El Rate Loop Compensation”, 08.29.1994.

[30] Y. L. Gu, N. K. Loh, N. Coleman, C. F. Lin, “Control of Weapon Pointing Systems Based on Robotic Formulation,” Invited Paper, Proc. 1992 ACC, Vol.1, pp. 413-418, June 1992.

[31] N.K. Loh, N.Coleman, “Microprocessor Controlled Optimal Helicopter Turret Control System,” Proc. 1988 American Control Conference, Arlington, VA, pp. 1088-1094, June 1982.

[32] R.C. Dorf, R.H. Bishop, “Modern Control Systems,” Ninth Edition, Prentice Hall Inc., 2001.

[33] N. P. Papanikolopoulos, P. K. Khosla, T. Kanade, “Visual Tracking of a Moving Target by a Camera Mounted on a Robot: A Combination of Control and Vision,” IEEE Transactions on Robotics and Automation, Vol. 9, No. 1, February 1993.

[34] “LQG” *Wikipedia, The Free Encyclopedia*, 31 January 2009, 18:00, Wikimedia Foundation Inc., Last Access Date: 17 February 2009, <http://en.wikipedia.org/wiki/Linear-quadratic-Gaussian_control>.

APPENDİX A

SYSTEM DESCRIPTION AND HARDWARE IMPLEMENTATION

In this chapter, turret system and the designed hardware used for providing communication between the turret system and outer world is discussed. The purpose of this chapter is to represent how the expected communication setup is implemented. This setup is basically composed of Interface Unit (IU) and the turret system that is to be controlled. Interface Unit design is formed according to the characteristics of turret system. By considering the characteristics of the system, specifications of the hardware are determined and then required hardware is designed. After that, the communication with the system is constructed. This communication means that the system is now controllable. It indicates that the position measuring and positioning the whole system is achieved. All these activities are detailed through this chapter.

A.1 Turret Subsystem

General Turret Subsystem structure, the detailed information about this system and the turret positioning property of the whole turret system are given in [7], [8]. The following information is originated from these references.

Turret Subsystem is capable of moving both in azimuth and elevation directions. Control of Turret Subsystem is performed by either pilot or gunner by using Hand Control Unit (HCU) or it can automatically follow the Forward Looking Infrared (FLIR) system.

This system is mainly composed of following subassemblies;

A.1.1 Turret

There is a turret system to rotate the system in azimuth and elevation directions which is composed of six (6) main parts: azimuth resolver, azimuth drive, upper support, and lower support, emergency stow control unit, saddle. These parts are shown in Figure A-1 which is taken from reference [8].

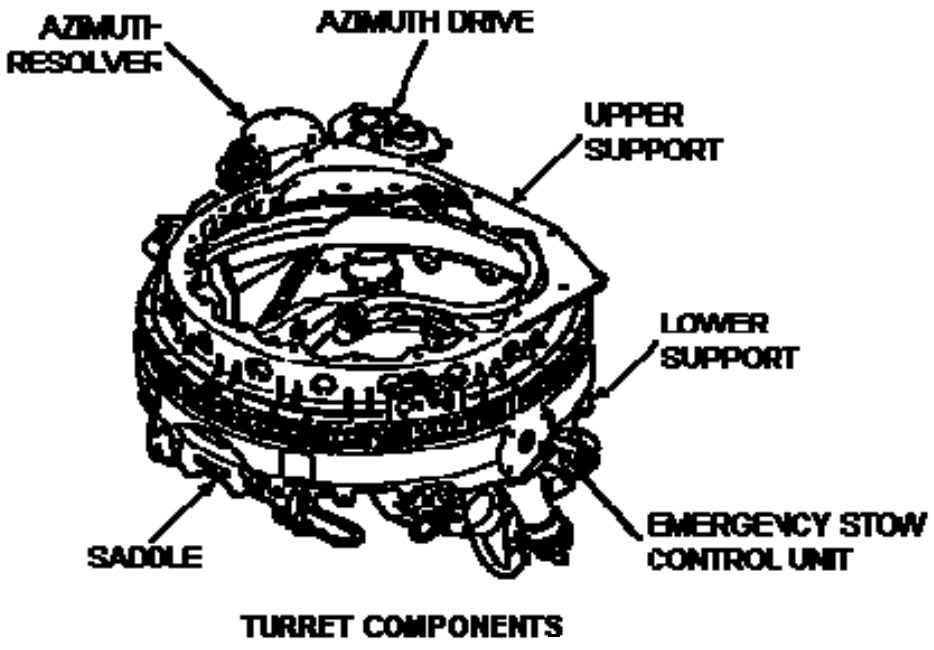


Figure A-1 Turret

Azimuth Resolver: It contains a resolver and a stow switch. Mainly it is used to provide azimuth position of the turret by using resolver. Stow switch is used when there is an emergency condition and it directly positions the system to 0º at azimuth.

Azimuth Drive: It contains gearbox, drive motor and a tachometer. Tachometer gives the azimuth rate signal.

Upper Support: It consists of mounting points to mount the turret to the helicopter.

Lower Support: It contains elevation switches to control elevation motion, elevation drive motor, elevation tachometer and elevation resolver. Tachometer gives the elevation rate signal. Resolver is used to provide elevation position of the turret. Switches are used when there is an emergency condition.

The details about the turret positioning are given in [7], [8].

A.1.2 Barrel

There is a gun with three barrel and turret mounted. The rotor part is the major component. This barrel is mounted into the turret part and form the whole system.

A.1.3 Turret Control Unit

This unit is used to control power that is going to azimuth drive motor and elevation drive motor. These drive motors are used to position the turret.

A.1.4 Power Control Unit

Power Control Unit is used to control voltage of fire, power and solenoid.

All these subparts are used to provide turret positioning with the help of IU. The detailed information about turret positioning can be obtained from [7], [8]. The interface between the system and pilot is shown in Figure A-2.

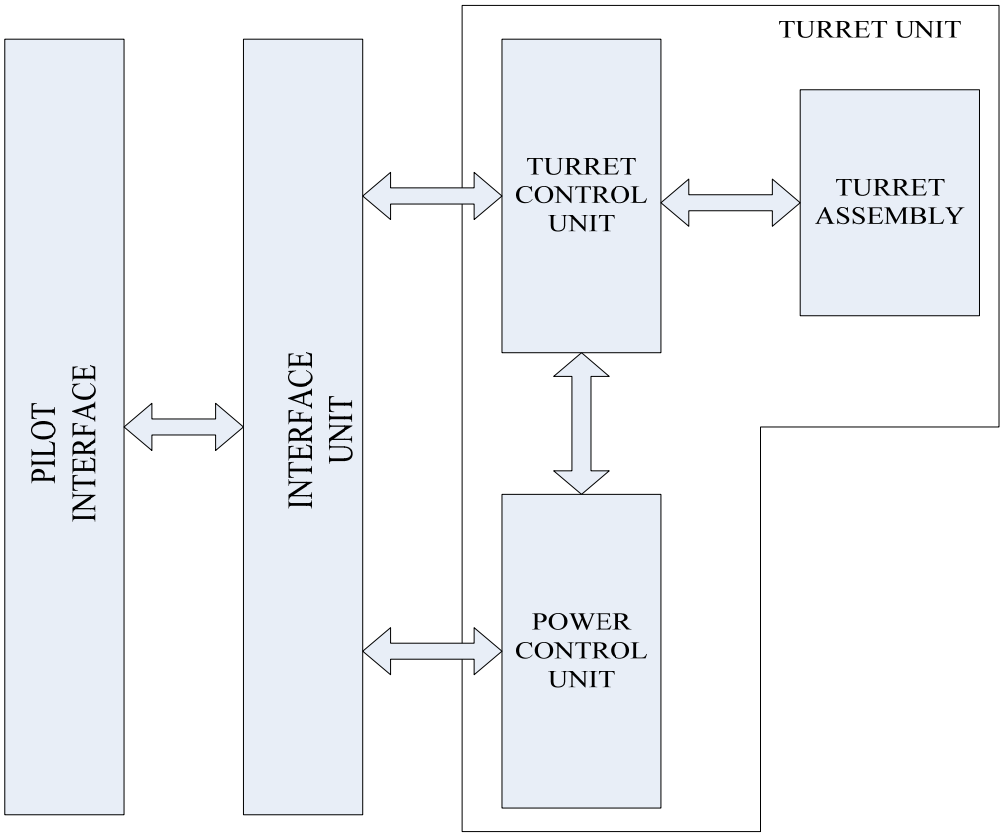


Figure A-2 General System Architecture

As it is seen from Figure A-2, to provide communication between Turret Unit and pilot and to control the Turret Unit, need of a unit arises. This unit is called Interface Unit (IU). The details about this unit are given in the following.

A.2 Interface Unit

Interface Unit is designed to determine required commands like action and fire, to transmit positioning data to Turret Unit in the suitable form and also to provide communication between Turret Unit and pilot. Pilot is the user who uses HCU to send command request. This HCU data is collected by Interface Unit to arrange this command and perform the communication between the user and Turret Unit. By this way the control of the Turret Unit can be achieved.

All these functions are performed by designing IU which is composed of five (5) main parts. These parts are;

- IU EMI filter
- IU Power Board
- IU Interface Board
- IU Main Board
- IU Cabling

All these parts are shown in Figure A-3.

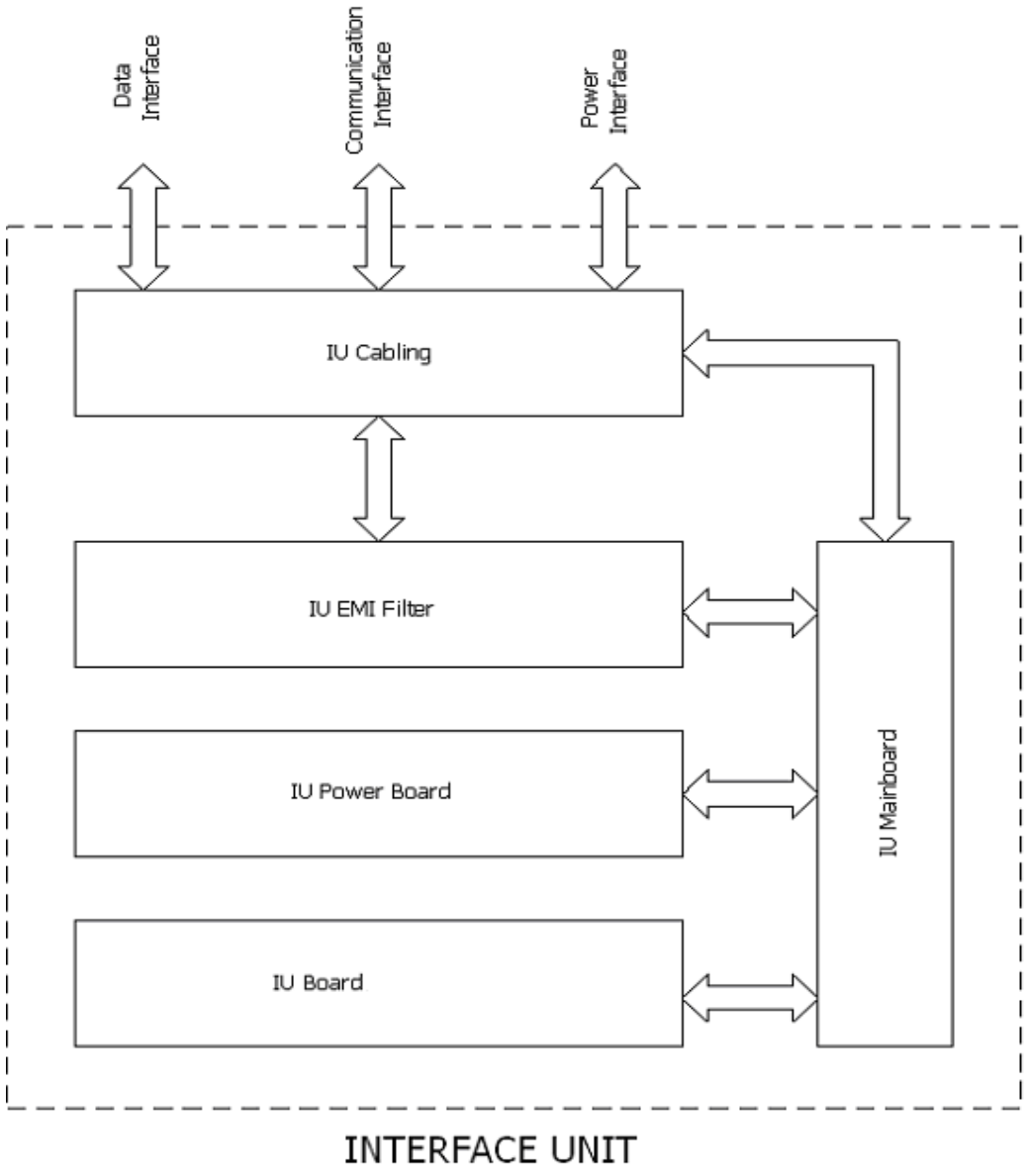


Figure A-3 Interface Unit General System View

These five (5) parts are used to provide three (3) main interfaces; data interface, power interface and transmission interface.

A.2.1 Data Interface

Data interface is composed of IU Board.

IU Board Functional block diagram is shown in Figure A-3. The main functions of IU Board are;

- Converting discrete control signals that come from HCU and the external systems, into digitized format by using Signal Adaptation Circuitry.
- Sending the resolver formatted position data from Turret Unit to Control Circuitry in digitized form by using the Resolver Interface Circuitry. Also sending the digital position error data from Control Circuitry to Turret Unit in resolver formatted form by using the Resolver Interface Circuitry.
- Providing the control and status data between Turret Unit and Control Circuitry by using the I/O Interface Circuitry.
- Providing the communication between the IU and outer world by using the Communication Circuitry on data bus.

The detailed IU Board architecture is given in Figure A-4:

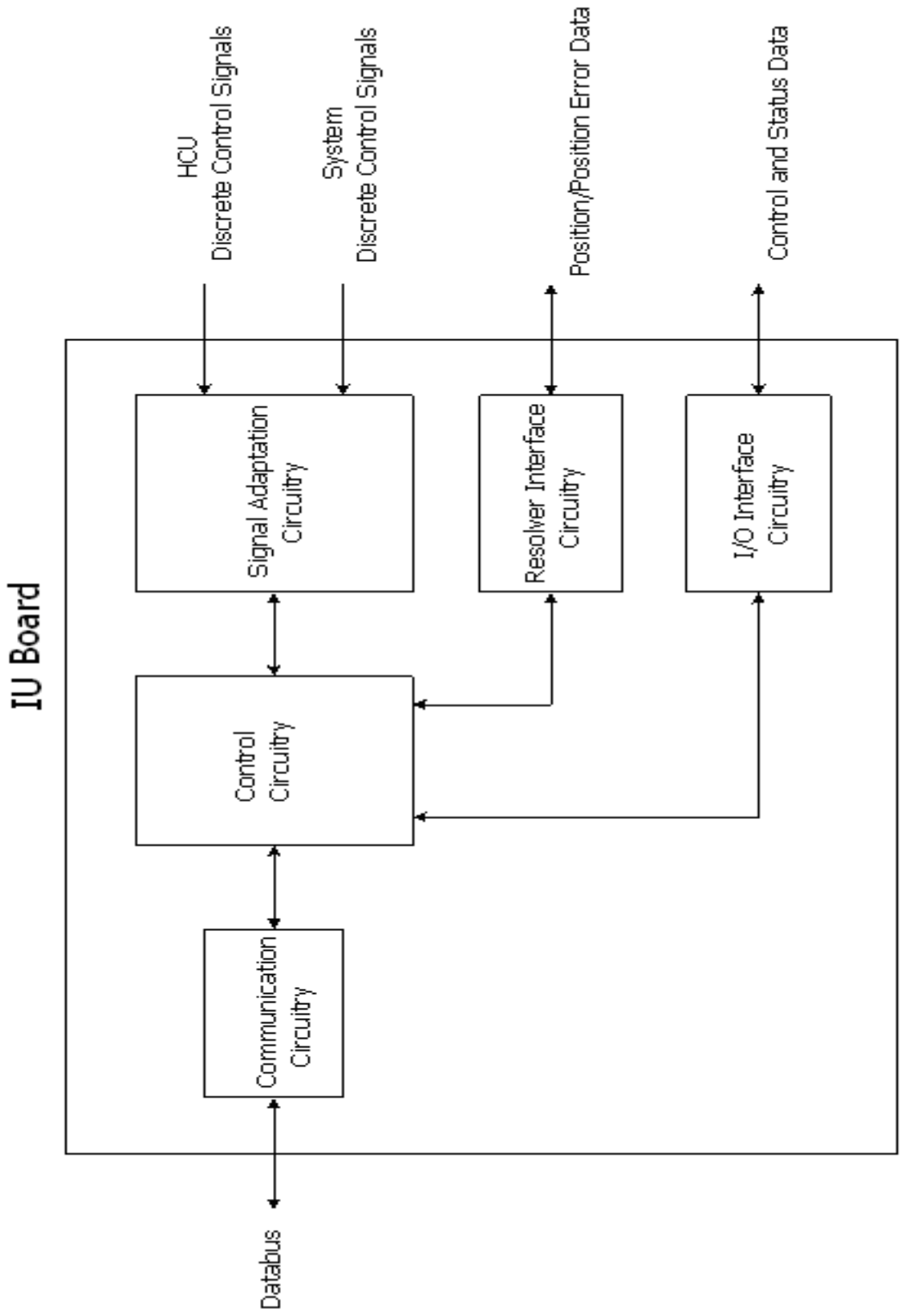


Figure A-4 Interface Unit Board Functional Block

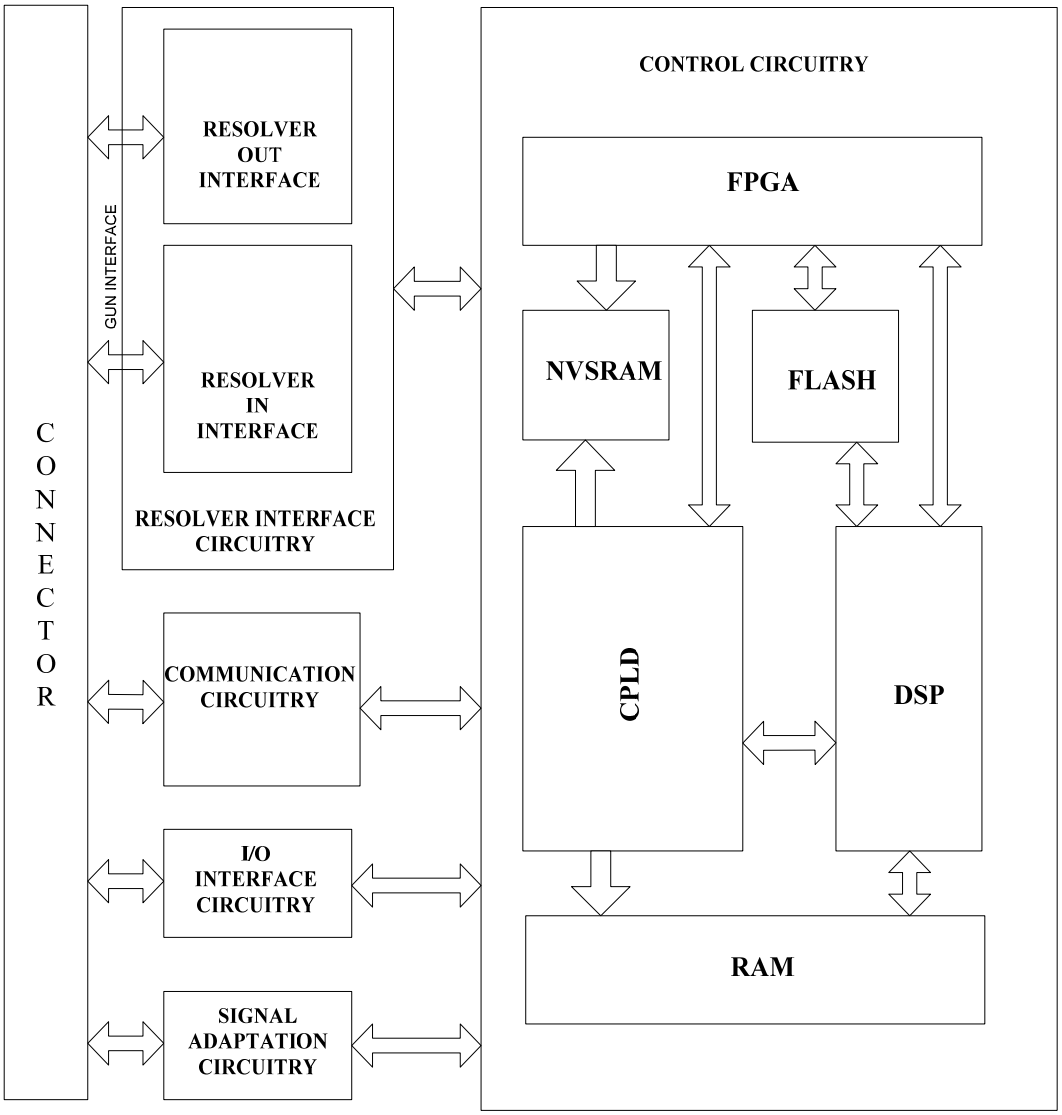


Figure A-5 IU Board Architecture

A.2.1.1 Control Circuitry

All the functions mentioned above are managed and controlled by the control circuitry. Control circuitry is basically composed of DSP, FPGA, program and data memories. Control circuitry architecture is given in Figure A-5:

A.2.1.1.1 DSP

Positioning, control and fire functions of Turret Unit are controlled by the DSP. This is performed by software written in C++ language using Texas Instruments

Code Composer interface. All the modeling and the designed controllers are implemented inside the DSP.

A.2.1.1.2 FPGA

FPGA is used to satisfy the following requirements;

- The communication and control between the resolver output interface and DSP. By this way the positioning error command determined by DSP is going to be sent to Turret Unit. The process is as follows: Firstly data is computed in DSP, then it is sent to FPGA, FPGA writes the data to DRC, whose timings are given Figure A-6 and the data is converted to resolver format. Finally, this resolver formatted data is sent to Turret Unit.
- The communication and control between the DSP and resolver input interface. By this way the position of the Turret Unit is sent to DSP. The process is as follows: Firstly the resolver formatted position data is converted into digital data by RDC whose timings are given in Figure A-7. Then the digitized data is taken by FPGA. Finally position data is sent from FPGA to DSP.
- Signal interface that is used to control Turret Unit.
- The collection of HCU discrete signals and sending them to DSP.
- The communication between the IU Board and outer world.

All these functions are implemented in the FPGA by using Very High Speed Hardware Description Language (VHDL).

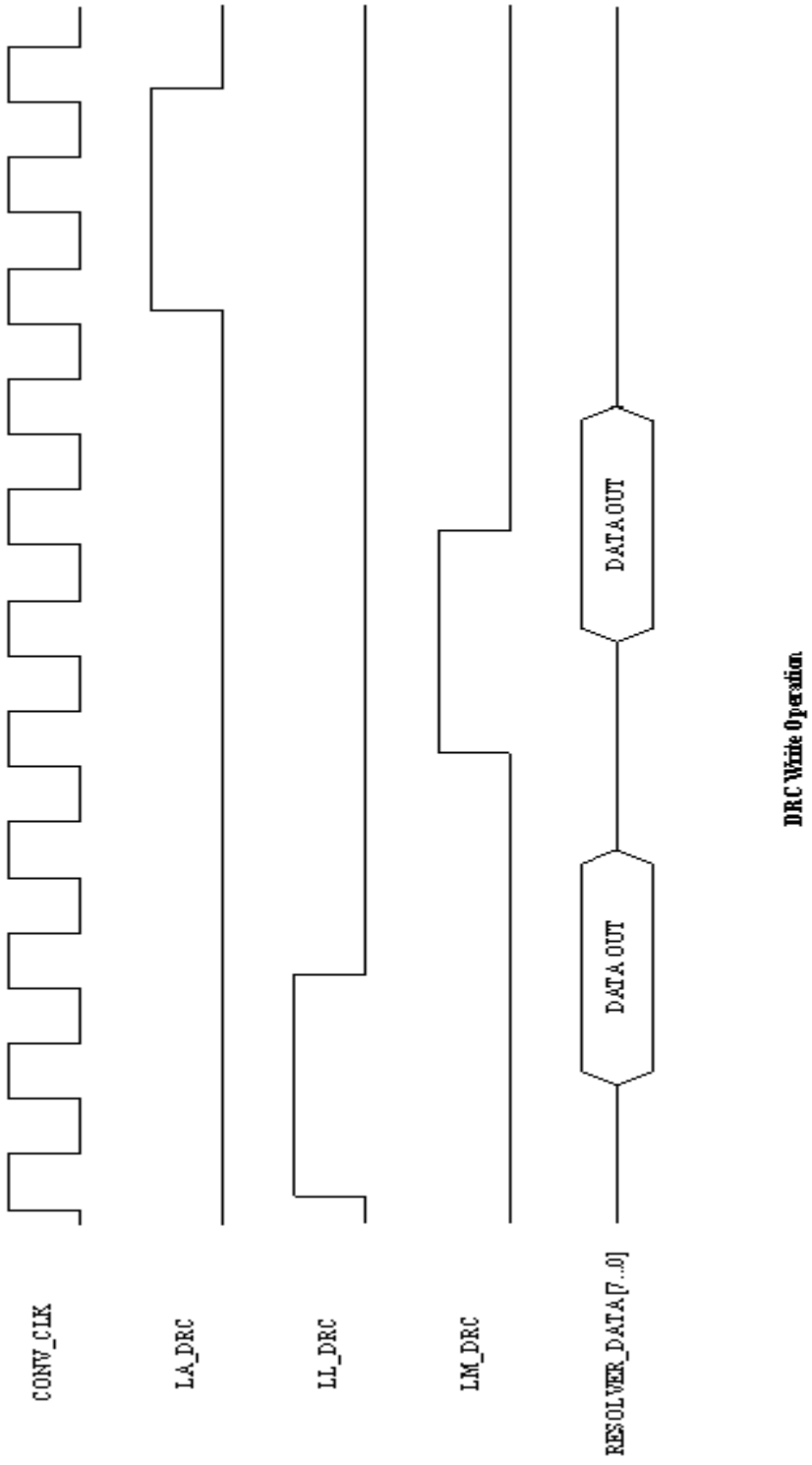


Figure A-6 DRC Write Operation

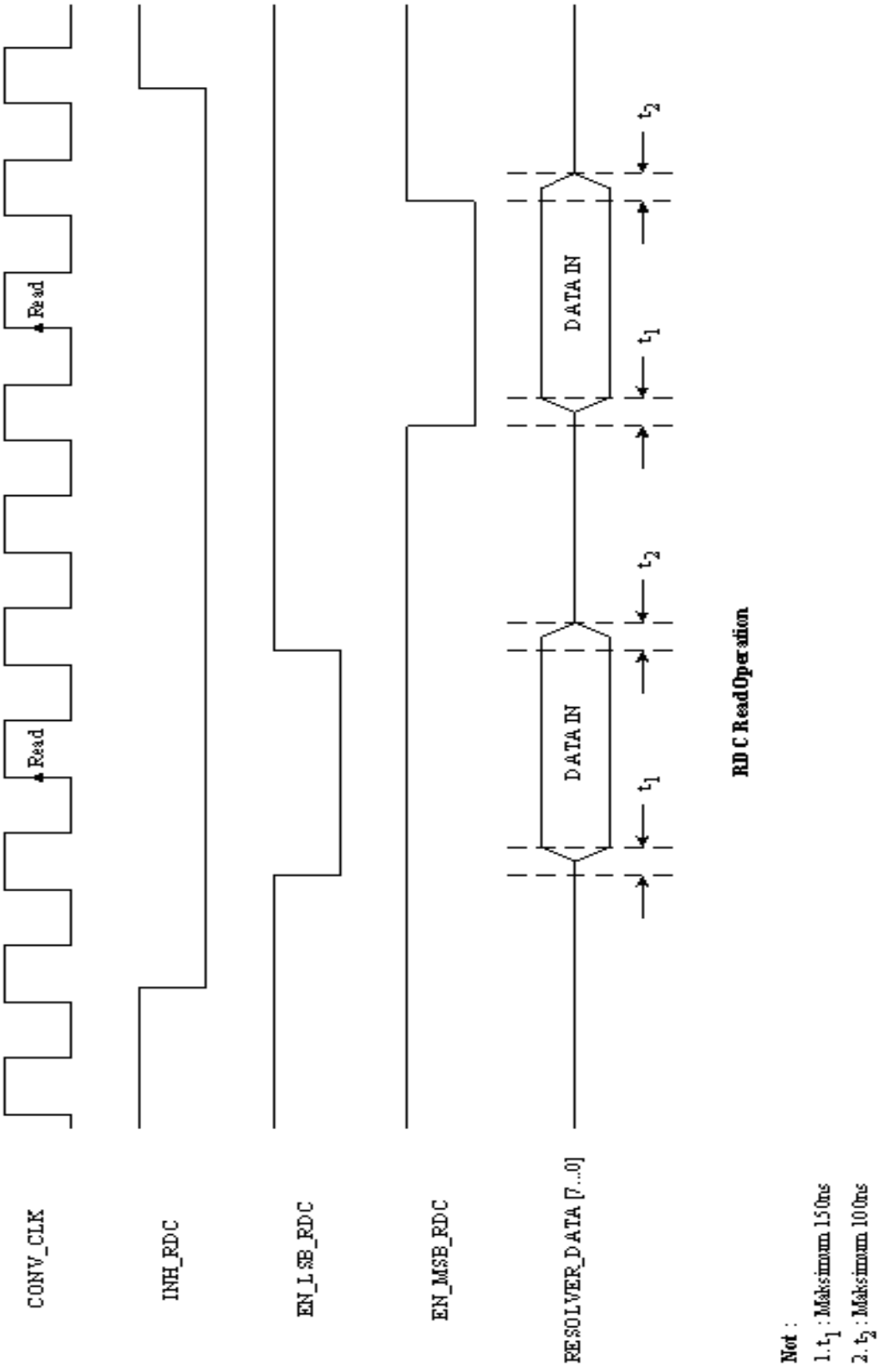


Figure A-7 RDC Read Operation

A.2.1.1.3 CPLD

CPLD is mainly used to produce reset signals for the peripherals on the board. The communication between the DSP and the other peripherals in the Interface Unit Board is provided by FPGA and CPLD devices. These functions are implemented by VHDL coding.

A.2.1.1.4 FLASH

Flash is used to store the DSP software. There is one flash on the system

A.2.1.1.5 RAM

RAM is used to run the DSP software. There is one SBSRAM on the system.

A.2.1.1.6 NVSRAM

NVSRAM is used to store the initial resolver data for Turret Unit. There is one NVSRAM on the system.

A.2.1.2 Resolver Interface Circuitry

A.2.1.2.1 Resolver out Interface

This unit is used to convert the digital data coming from FPGA into resolver formatted data for Turret Unit. This interface is mainly composed of DRC. The converted resolver data has to be 26VAC at 400Hz. The equations of resolver formatted data and the details are mentioned in 3.1.1 Data Collection section. In Figure A-5, the general view of this unit is shown.

A.2.1.2.2 Resolver in Interface

This unit is used to convert the resolver data coming from Turret Unit into digitized data for DSP. This interface is mainly composed of RDC. The incoming resolver data has to be 26VAC at 400Hz. The equations of resolver formatted data and the details are mentioned in 3.1.1 Data Collection section. In Figure A-5, the general view of this unit is shown.

A.2.1.3 Communication Circuitry

This interface is designed for the communication between IU Board and outer world from RS232 serial communication. FPGA writes the serial data coming from the outer world to FIFO for DSP. It also takes data from FIFO written by DSP for outer world. Here FIFO is used to store the data that is going to be transmitted and received for serial interface.

A.2.1.4 Signal Adaptation Circuitry

Signal Adaptation Circuitry is used to isolate and collect the discrete I/O interface signals that are going into or coming out of the IU Board.

A.2.1.5 I/O Interface Circuitry

I/O Interface Circuitry is used to collect control and status data that are going into or coming out of Turret Unit.

A.2.2 Power Unit

This Unit is designed to supply the required power for IU. It is composed of IU Power Board and IU EMI Filter.

A.2.3 Data Transmission Unit

This unit provides internal and external data transmission for IU. It is composed of IU Cabling and IU Main board.

All these functions and units are collected in the IU whose general system view is shown in Figure A-8.

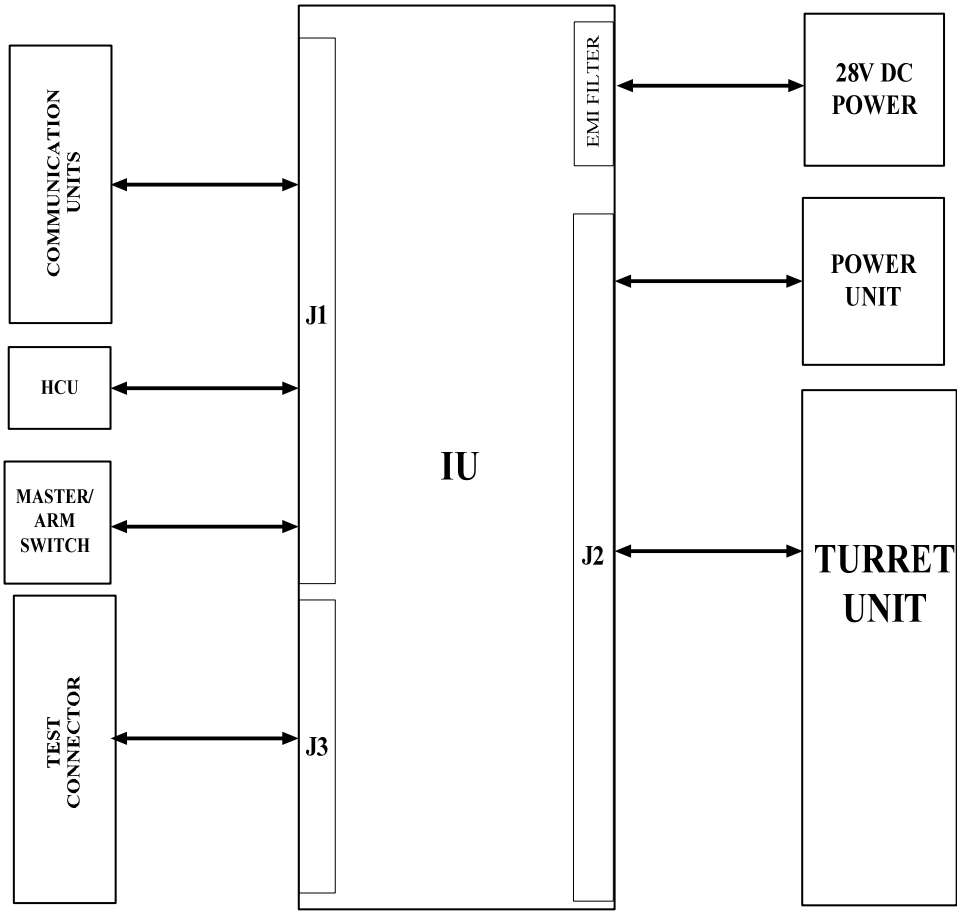


Figure A-8 General View Of IU Box

The previews of IU box are shown in Figure A-9 and Figure A-10.

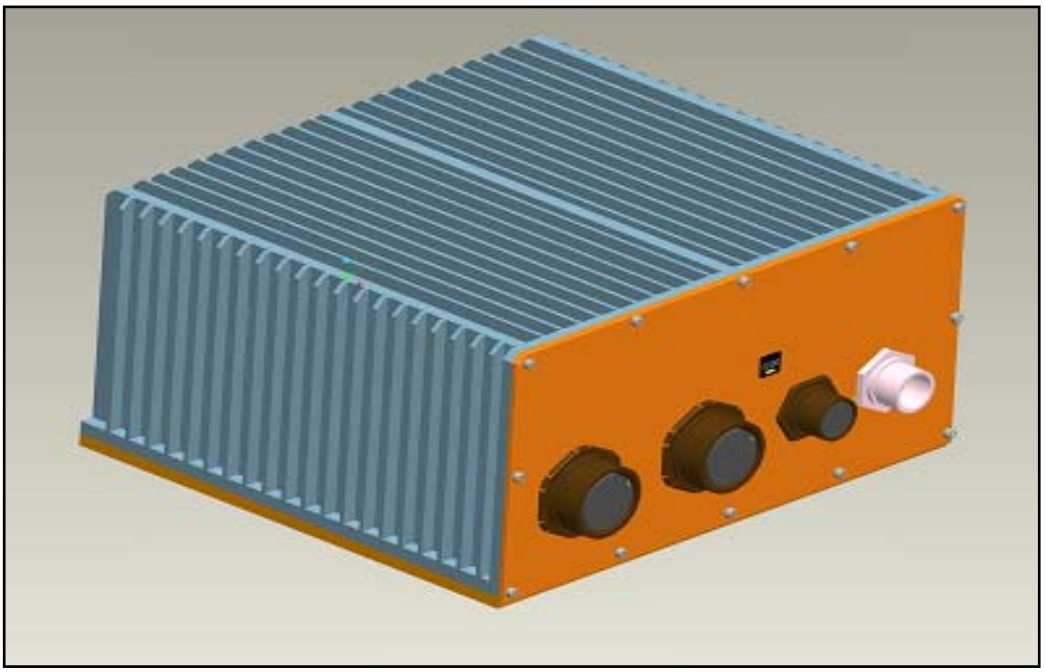


Figure A-9 IU Box View 1

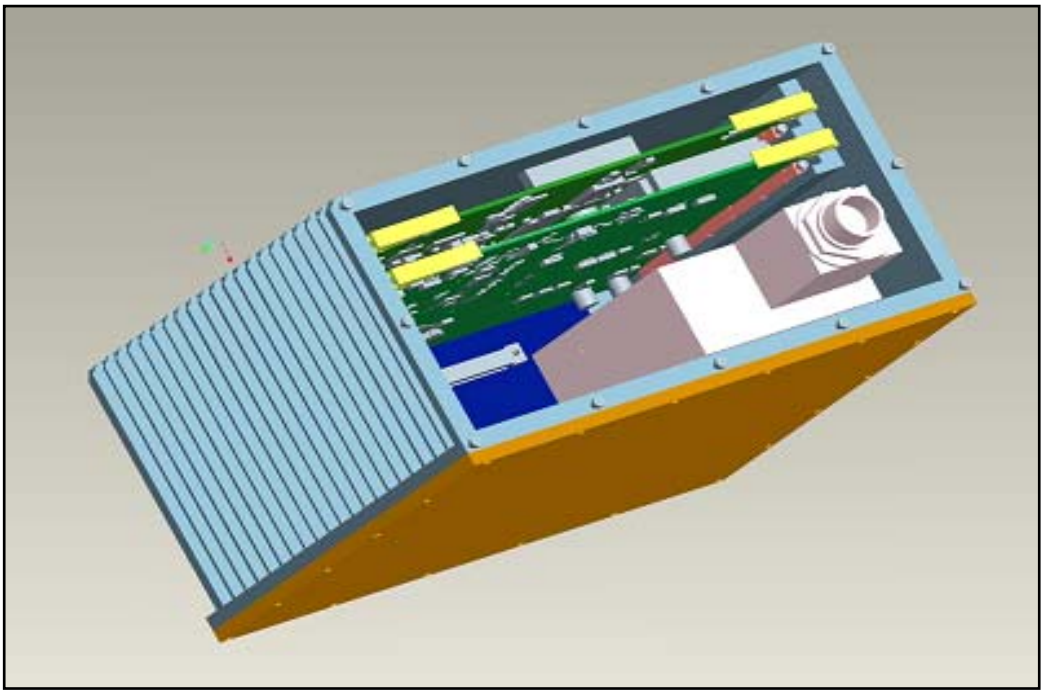


Figure A-10 IU Box View 2

A.3 Communication between Turret and IU

The whole structure of the system that is used for modeling and controlling the turret, is explained in this section. First of all, turret system is briefly explained and then some explanations about the Interface Unit Board are given.

Figure A-11 shows communication interface of our whole system and the provided general digital/analog structures of our case.

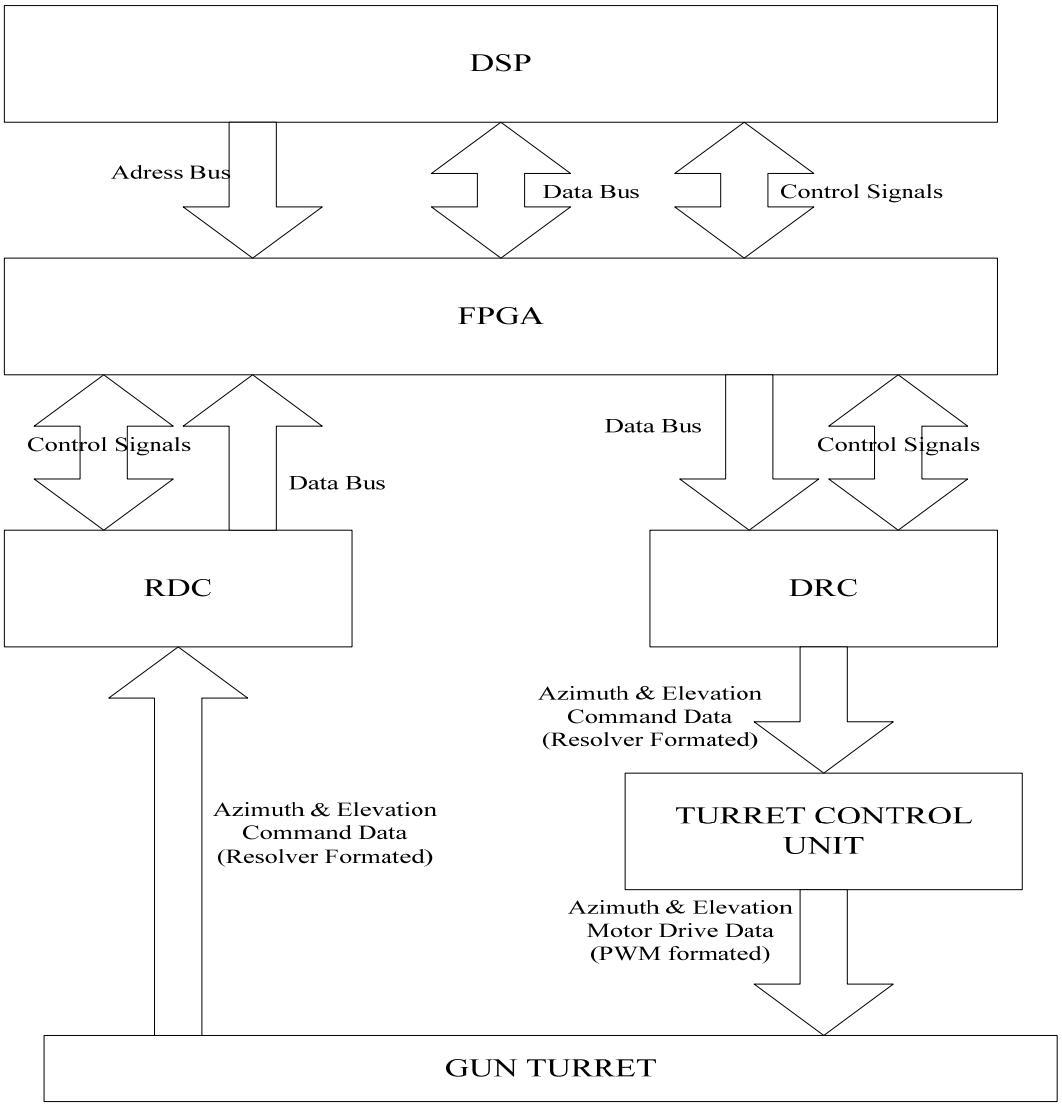


Figure A-11 General System View

As it is seen from the graph, there is a Turret Control Unit that is used to drive the motor. There is a motor that rotates the turret to the desired position. Motor is guided by the Motor Drive Circuitry. Motor Drive Circuitry is the part where the motor drive assembly is built up which is shown in Figure A-12. The Motor Drive Circuitry requires the data in the resolver format.. This data format is prepared by the units shown in Figure A-11. Firstly the desired position command data is formed in DSP part and send to FPGA to supply the communication and send the data to DRC. After the DRC device, the required resolver data for the Turret Control Unit is obtained. Also while forming the desired data, and for the controller part, the position of the system has to be known. Position information comes from the resolver assembly that is on the turret. This position information is in the resolver format so to make it meaningful for the DSP, the resolver formatted data has to be converted to the digital format. This procedure is as follows. Firstly, the resolver data reaches to the RDC. Then the digital data is taken from the RDC by FPGA. After that, the digitized resolver data or in other words measured position data is sent to DSP by FPGA. DSP interprets this data by using its controller and resents the suitable data to position the turret system. This position loop inside DSP is as shown in Figure 3-1. The position command shown in Figure 3-1 is active if the action command comes, if not the system stays in the fixed forward position, this is 0 degree azimuth, 0 degree elevation position. So this action command has to be controlled by DSP. This is performed by taking command for the action including the required positioning data from the pilot or gunner by using HCU or from FLIR. This action command is firstly taken by FPGA and then it is sent to DSP. Finally, DSP forms the action command to control the turret.

All the following structure is obtained and now ready to communicate with turret.

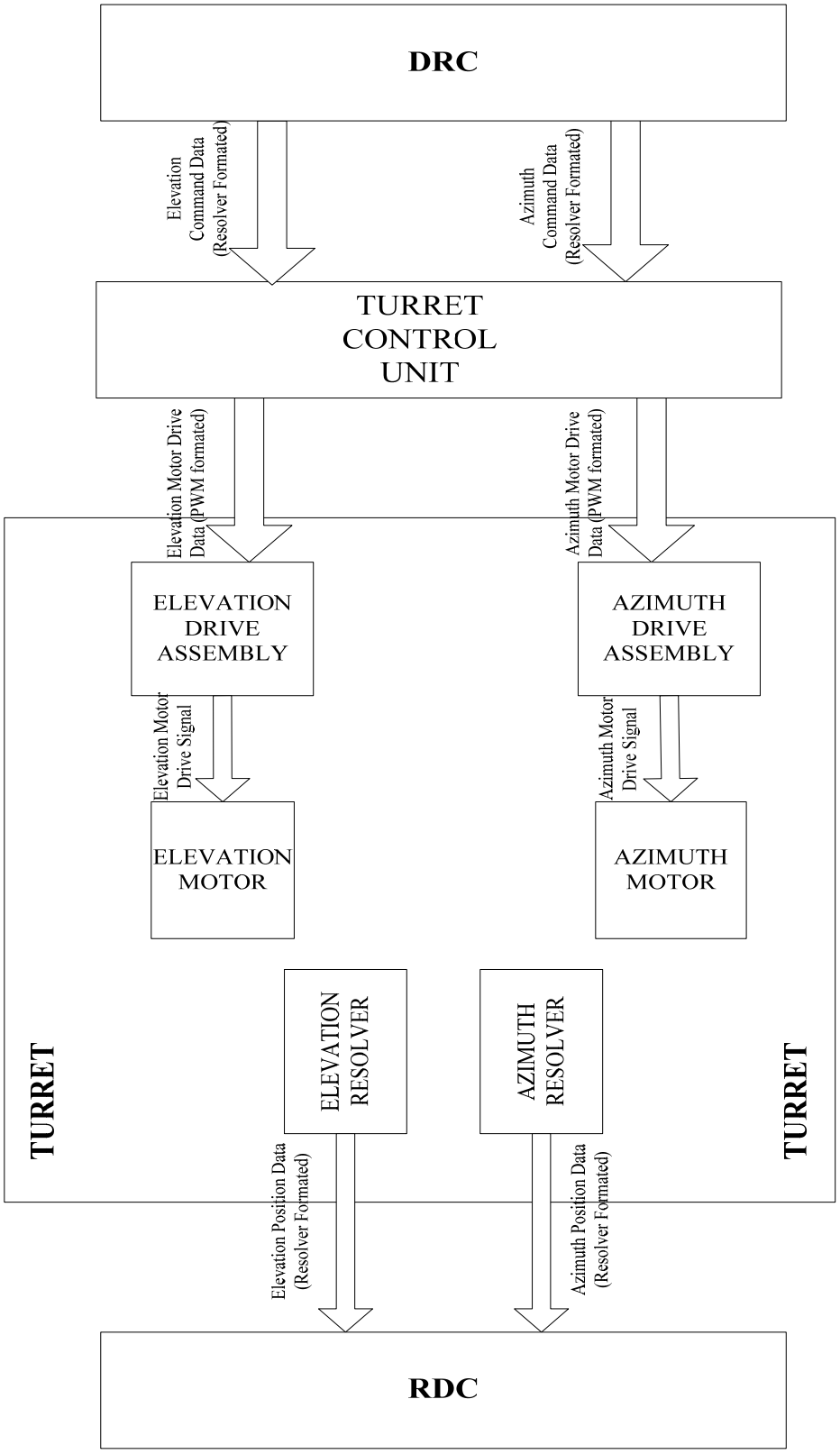


Figure A-12 Turret Detailed View

Hence there is another control outside the system to make the pilot be sure to fire. This is done by Master/Arm switch. If this switch is “OFF” then the turret system is also “OFF”. It means that turret can not fire and turret can not rotate. If this switch is taken to “STANDBY” mode then the control of the turret rotation function is active only, it means that the Pilot/Gunner has only the ability to rotate the turret but can not fire. After this switch is taken to “ARM” position then the turret can be fired and turret system can be rotated following the reference input that is obtained from a FLIR or from HCU.

After the setup is implemented then everything is ready to obtain the model and design a controller. So firstly the system identification and modeling parts has to be done because designing a controller requires identification and parameters of the plant.