1. INTRODUCTION

Time delay has been a common phenomenon to overcome whenever it is closed to a feedback loop for the purpose of controlling any system. Recent increase of control applications in variety gives more importance to systematic methods to cope with time delay (Yamanaka, 2003).

Levine states that transportation and measurement lags, analysis times, computation and communication lags all introduce time delays or dead times into control loops. Dead times are also inherent in distributed parameter systems and frequently are used to compensate for model reduction where high-order systems are represented by low-order models with delays. The presence of dead times in the control loops have two major consequences:

1. It greatly complicates the analysis and the design of feedback controllers for such systems.

2. It makes satisfactory control more difficult to achieve.

There are numerous of systems, which have dead time, in the industrial processes. It is really hard to control the systems which have dead time. There are several introduced ways to control these systems such as Internal Model Control, Smith Prediction, Predictive Control. Because of its simple structure, convenient parameters tuning and its strong robustness, IMC is many experts' favorite in recent years. However, there are not only stable processes with dead time, but also there are unstable processes with dead time. And this kind of unstable process with dead time is very difficult to control (Jin, Sun, Liu, Zhang, 2004).

Proportional-integral-derivative (PID) controllers are still widely used in industrial systems despite the significant developments of recent years in control theory and technology. This is because they perform well for a wide class of processes.Also, they give robust performance for a wide range of operating conditions. Furthermore, they are easy to implement using analogue or digital hardware and familiar to engineers (Kaya, 2004).

However, plants with long time-delays can often not be controlled effectively using a simple PID controller. The main reason for this is that the additional phase lagcontributed by the time-delay tends to destabilise the closed-loop system. The stability problem can be solved by decreasing the controller gain. However, in this case the response obtained is very sluggish (Kaya, 2004).

A control system design is expected to provide a fast and accurate set-point tracking, that is, the output of the system should follow the input signal as close as possible. Also, any external disturbances must be corrected by the control system as efficiently as possible. The first requirement can be achieved by an open loop control system. With an open loop control scheme, the stability of the system is guaranteed provided that both the plant and controller transfer functions are stable. Also, the design of the controller in an open loop control scheme may simply be chosen as $G_C(s) = G^{-1}(s)$, where $G_c(s)$ and $G(s)$ are respectively the controller and plant transfer functions. The drawback of an open loop control system is the sensitivity to modelling errors and inability to deal with external disturbances entering the system. In this case, a closed-loop system can be used to deal with disturbances and modelling errors (Kaya, 2004).

There are several solutions for overcoming with dead time like Smith Predictor, Internal Model Control. When being emphasized to these subjects, the main ideas of these rules are same. Moreover, the basic idea of Smith Predictor depends on the Internal Model Control. In the internal model control, the model of the process is an internal part of the controller. Thus, the application of internal model control is easy than the other control schemes that can be designed for the control of dead time dominant systems. In this homework, the internal model control will be analyzed and then an application of the internal model control with PLC will be introduced.

In the second chapter, the basic principles of internal model control will be introduced. And then the IMC based PID controller will be emphasized. By this way, finding of the PID constants from the internal model control principle will be introduced.

In the third chapter, initially the Flapper Position Control system will be introduced. And then the application of the IMC based PID control will be introduced. Moreover, the fuzzy algorithm application on the system for finding the filter constant in the internal model control will be analyzed. The application of the both fuzzy algorithm application and IMC based PID control application was applied to the system with PLC. Finally, the results of the applications will be discussed.

In the fourth chapter, the results of the applications to the real system will be discussed.

2. INTERNAL MODEL CONTROL (IMC)

2.1 Time-Delay Systems

For all physical systems there will be a duration of time, possibly very short, between the application of stimulus, and the detection of the corresponding response. In some process plants the duration of the delay between stimulus and onset of response may be minutes, or even longer (Gorecki and Korytowski, 1992).

Gorecki and Korytowski state that a pure time delay, an essential element in the modeling and description of these systems, has the property that input and output are identical in form, the only difference being that of translation along the time axis. The stimulus x(t) results in a response y(t) equal to x(t-h) where h is the delay. In many complex systems there will be more than one delay, and these delays may be, for example, in measurement at an output, in control at an input or in a feedback path.

According to Gorecki and Korytowski, the solution of the delay differential equation can be given such as the following:

$$\dot{y}(t) = -y(t-h) + H(t) \qquad (2.1.1)$$

It is assumed that this equation describes a system quiescent at t<0, the only stimulus being the step H (t) applied at t = 0. h is the time delay. The following figure shows a system to which equation (1.1) applies.
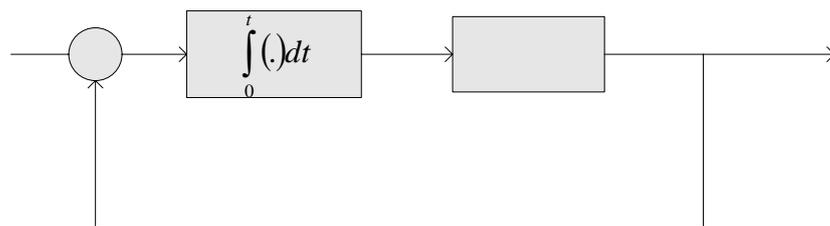
Figure 2.1.1

* If t = h, the derivative term is unaffected by the output so that the following equation is obtained:

$$\dot{y}(t) = H(t) \qquad (2.1.2)$$

If this differential equation is solved, it is obvious that the following solution is obtained:

$$y(t) = tH(t) \qquad (2.1.3)$$

* When h < t < 2h, the following differential equation is obtained:

$$\dot{y}(t) = H(t) - (t - h)H(t - h) \tag{2.1.4}$$

If this differential equation is solved, the following solution is obtained:

$$y(t) = tH(t) - \frac{1}{2}(t - h)^2 H(t - h) \tag{2.1.5}$$

Many differential equations can be found for the other intervals. If all these differential equations are solved correctly, a general rule can be obtained such as the following:

$$y(t) = tH(t) - \frac{1}{2}(t - h)^2 H(t - h) + \frac{1}{6}(t - 2h)^3 H(t - 2h) + ... + \frac{(-1)^r}{(r+1)!}(t - rh)^{r+1} H(t - rh) + ...$$
$$\tag{2.1.6}$$

This method is described as 'the method of steps', extends in a natural way to higher-order systems, the only additional difficulties arising from the initial conditions on the differential coefficients at the start of each step (Gorecki and Korytowski, 1992).

The solution of the differential equation can be obtained by using the Laplace transform more quickly (Gorecki and Korytowski, 1992).

The transfer function of the delay time, h, is written as e^{-sh}. A general delay theorem can be written in the Laplace transform such as the following:

$$L\{x(t - h)H(t - h)\} = X(s)e^{-sh} \text{ and } X(s) = L\{x(t)H(t)\} \tag{2.1.7}$$

If the Laplace transform is applied to the equation (2.1.7), the following solution is obtained:

$$sY(s) + e^{-sh}Y(s) = \frac{1}{s} \tag{2.1.8}$$

Y(s) can be obtained from this equation such as the following equation:

$$Y(s)\left[s + e^{-sh}\right] = \frac{1}{s} \tag{2.1.9}$$

$$Y(s) = \frac{1}{s} \frac{1}{s + e^{-sh}} = \frac{1}{s^2} \frac{1}{1 + \frac{1}{s}e^{-sh}} \tag{2.1.10}$$

The expansion of this equation can be given as:

$$Y(s) = \frac{1}{s^2} - \frac{1}{s^3}e^{-sh} + \frac{1}{s^4}e^{-2sh}... = L\left\{tH(t) - \frac{(t - h)^2}{2!}H(t - h) + \frac{(t - 2h)^3}{3!}H(t - 2h)...\right\}$$

$$\tag{2.1.11}$$

By taking the inverse Laplace transformation of Y(s), the following equation is obtained:

$$y(t) = \sum_{r=0}^{\infty} (-1)^r \frac{(t-rh)^{r+1}}{(r+1)!} H(t-rh) \qquad (2.1.12)$$

Time delay occurs in control systems when there is a delay between the commanded response and the start of the output response. (Nise, 2004)
General structure of time delay systems
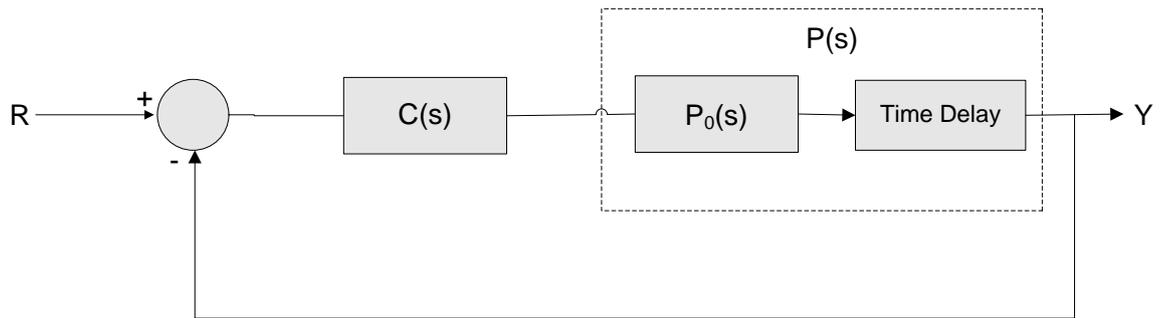
Figure 2.1.2: General Structure of Time Delay systems

Time delay is demonstrated as e^{-sh}.

Controlling the time delay systems is really hard and complex. Pole assignment is not a good solution for the time delay systems. There are different ways to control time delay systems.

2.2 Principles of Internal Model Control (IMC)

Internal Model Control is one of the ways of controlling the dead time systems. Comparing with the other control structures that are used to control the dead time systems, it is obvious that the structure of internal model control is simpler than the other control structures.

The internal model principle is a general method for design of control systems that can be applied to PID control (Äström, Hägglund, 1995).
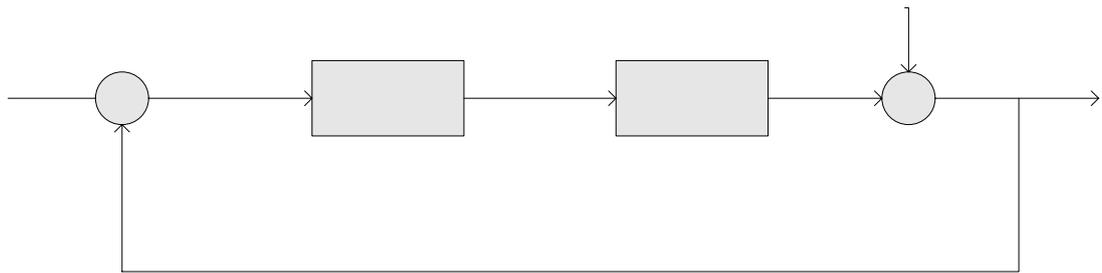
Figure 2.2.1: Classical Feedback Structure
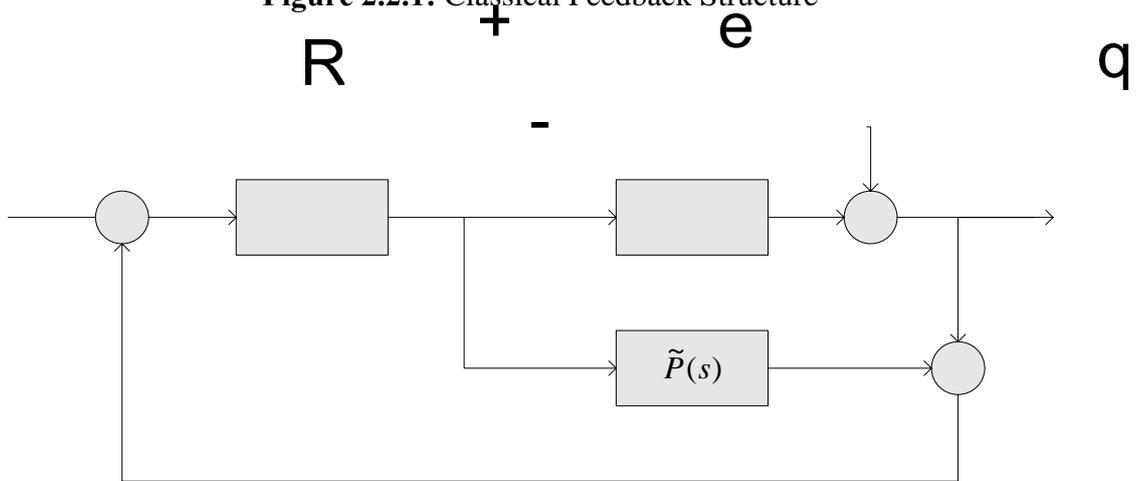
Figure 2.2.2: Internal Model Control Feedback Structure

In the model $\tilde{P}$ indicates the model of the true process. And q symbolizes the IMC controller.

Internal Model Control (IMC) is a control structure which consists of the process model as an internal part of the controller (Rivera, 1999).

In the first graph, it is shown that a classical feedback structure. The IMC Feedback structure can be derived by manipulating the classical feedback structure.
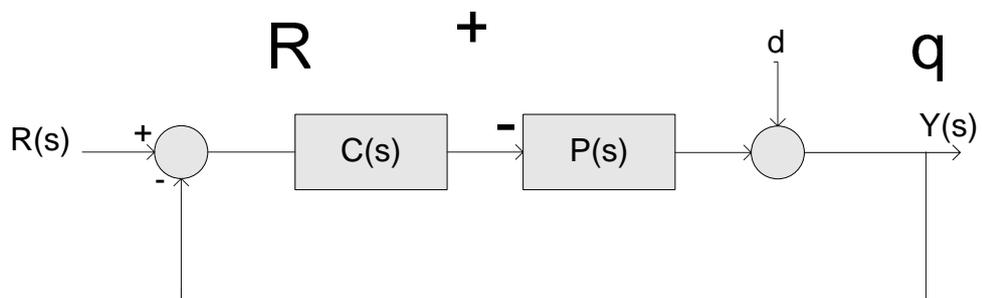
Figure 2.2.3: Classical Feedback Structure

The true model of the process ($\tilde{P}$) can be added between the controller (q(s)) and the system (P(s)) as it is shown in the following figure. If it is added by this way, there is no difference between the system above and the system below.
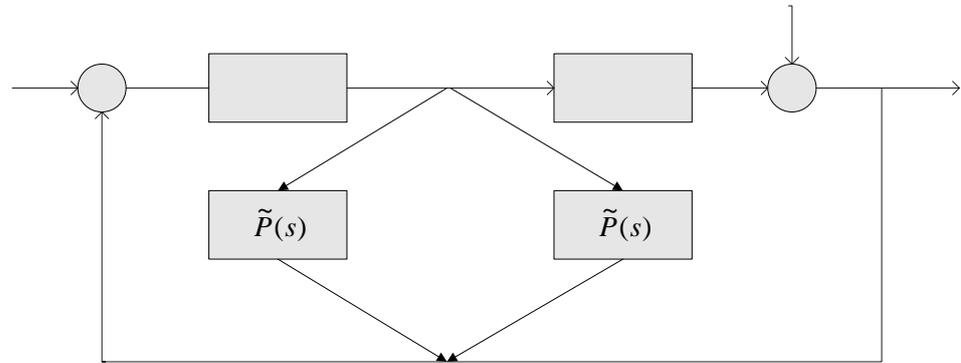
Figure 2.2.4

This graph can be configured again. So, if the model of the process is added as a negative feedback to the classical feedback controller as it is seen at the following graph, there won't be any differences between this graph and the previous one. Moreover, the other system model can be added as a negative feedback to the system model as it is seen at the following graph without changing any structure.
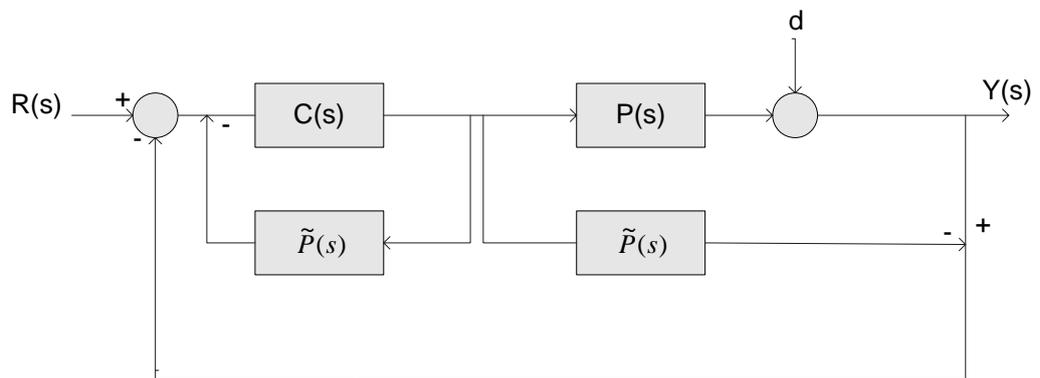
Figure 2.2.5

A new controller, q(s), can be formed instead of C(s) with its negative feedback $\tilde{P}$. So, the following model can be derived:

7
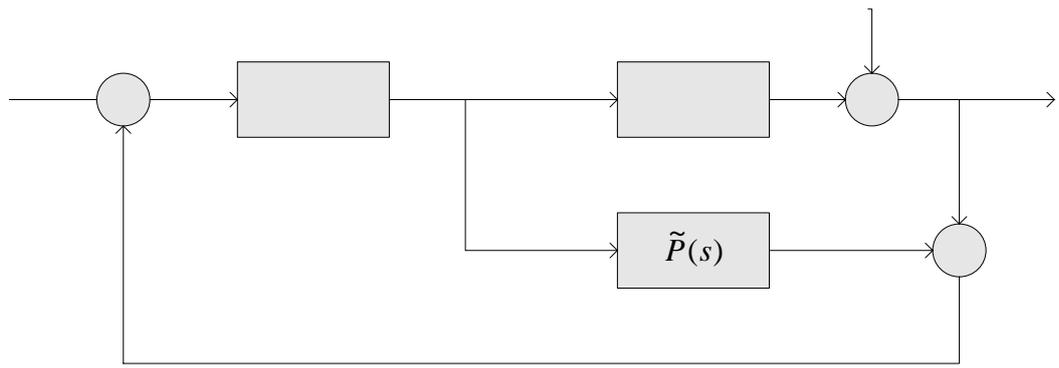
Figure 2.2.6: Internal Model Control Feedback Structure

According to Rivera, the IMC structure has some advantages with respect to the classical feedback structure:

- For the classical feedback structure, we need to solve the characteristic polynomial of the closed loop system to find the roots of the polynomial. But, in the IMC structure we may only search the poles of q.

- It may be searched for the q instead of C(s) without any loss of generality.

2.3 IMC Structure
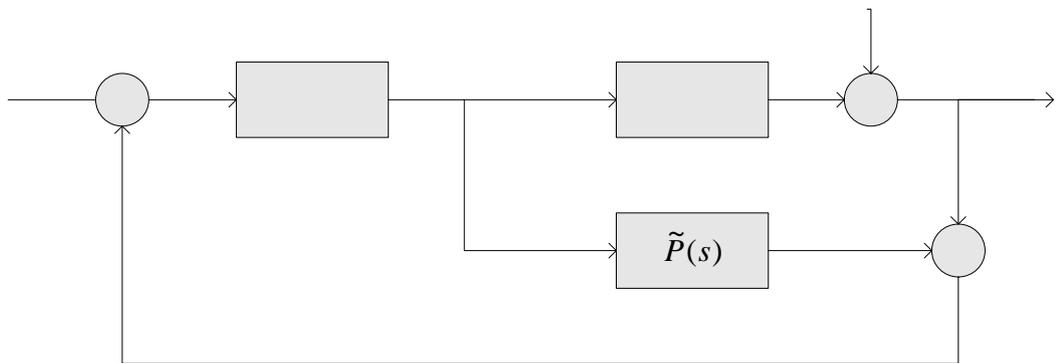
Figure 2.3.1: Internal Model Control Feedback Structure

At the above model, P(s) demonstrates the process model, $\tilde{P}(s)$ demonstrates the model of the true process and the q demonstrates the IMC controller. The equation for the output function can be obtained as following equation:

$$Y(s) = \frac{P(s)q}{1 + q(P(s) - \tilde{P}(s))} R(s) + \frac{1 - \tilde{P}(s)q}{1 + q(P(s) - \tilde{P}(s))} d \qquad (2.3.1)$$

For making similar this equation, the following equations can be written:

$$\eta(s) = \frac{P(s)q}{1 + q(P(s) - \tilde{P}(s))} \qquad \varepsilon(s) = \frac{1 - \tilde{P}(s)q}{1 + q(P(s) - \tilde{P}(s))} \quad (2.3.2)$$

can be written. By the way, it is obvious that there is a relation between the function of $\eta(s)$ and the function of $\varepsilon(s)$ such as:

$$\varepsilon(s) = 1 - \eta(s) \qquad (2.3.4)$$

So, the output function can be rewritten such as

$$Y(s) = \eta(s)R(s) + \varepsilon(s)d(s) \qquad (2.3.5)$$

- If the process model is not equal to the true process model, the closed loop transfer function cannot be simplified anymore. Than the IMC control structure cannot be simpler than the classical control feedback structure. (Levine,)

- If the process model is equal to the true process model $(P(s) = \tilde{P}(s))$, the closed loop transfer function can be simplified. (Levine,) By this way Y(s) simplifies to

$$Y(s) = \frac{P(s)q}{1 + q\underbrace{(P(s) - \tilde{P}(s))}_{0}} R(s) + \frac{1 - \tilde{P}(s)q}{1 + q\underbrace{(P(s) - \tilde{P}(s))}_{0}} d \qquad (2.3.6)$$

If the output function is rewritten again, the equation

$$Y(s) = \tilde{P}(s)qR(s) + (1 - \tilde{P}(s)q)d(s) \qquad (2.3.7)$$

is derived. Consequently, error function, control signal function can be calculated as below:

$$e(s) = (1 - \tilde{P}(s)q)R(s) - (1 - \tilde{P}(s)q)d(s) \qquad (2.3.8)$$

$$u(s) = qR(s) - qd(s) \qquad (2.3.9)$$

In order to result in physically realizable manipulated variable responses, the IMC controller (q) must satisfy the following criteria (Rivera, 1999):

- *Stability:* In order to generate bounded responses to bounded inputs, all poles of the IMC controller (q) must situate in the open Left – Half S Plane.

9

- **Properness:** In order to avoid pure differentiation signals, q(s) must be proper. So,

$$\lim_{|s|\to\infty} q(s) \quad \text{must be finite.}$$

- **Causality:** q(s) must be causal. Q(s) mustn't require any prediction for to be causal.

2.4 Internal Model Control Design Procedure

There are two main steps for designing IMC. The first step will satisfy that the internal model controller's stability and causality, the second step will require the internal model controller's properness.

Step 1: Factor of the model of the true process $(\tilde{P}(s))$ can be written into two parts such as

$$\tilde{P}(s) = \tilde{P}_+(s)\tilde{P}_-(s) \tag{2.4.1}$$

In that equation, $\tilde{P}_+(s)$ contains all Right Half S Plane zeros and poles, which are all *"Nonminimum Phase Elements"* in the model of the process. On the other hand, $\tilde{P}_-(s)$ is *"Minimum Phase"*. Therefore, the IMC Controller which is defined as $\tilde{q}(s) = \tilde{P}_-^{-1}(s)$ is stable and causal. $\tilde{q}(s)$ doesn't contain any time delays and right half s plane zeros.

Step 2: In order to improve the IMC controller, a filter can be added to the controller. Finally,

$$q(s) = \tilde{q}(s)f(s) \tag{2.4.2}$$

is obtained for the IMC controller..

Now, q(s) is stable, causal and proper.

A common filter such as

$$f(s) = \frac{1}{(\lambda s + 1)^n} \text{ (low band pass filter)} \tag{2.4.3}$$

is chosen (Rivera, 1999).

The filter order can be selected large enough to make q proper. λ, which is an flexible parameter, determines the speed of response. If λ increases, the speed response of the control system becomes slower. On the other hand, if λ decreases, the speed response of the control systems becomes faster. More over, λ affects the

robustness of the control system. The higher values of the λ, the higher robustness are got for the system (Rivera, 1999).
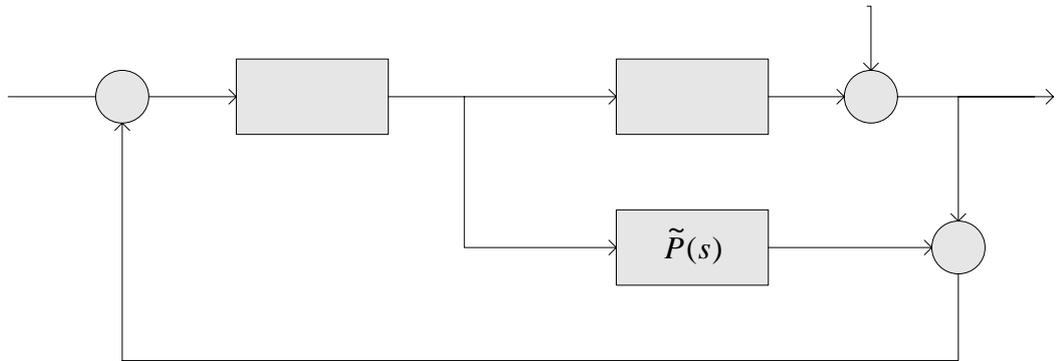
Figure 2.4.1: Internal Model Control Feedback Structure

The equivalent standard feedback control system is obtained by using the transformation of

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} \qquad (2.4.4)$$

This equation also can be proved by block diagram manipulation. At the block diagram, it is shown that the IMC structure. The equation of classical controller can be obtained by manipulating the block diagrams step by step.
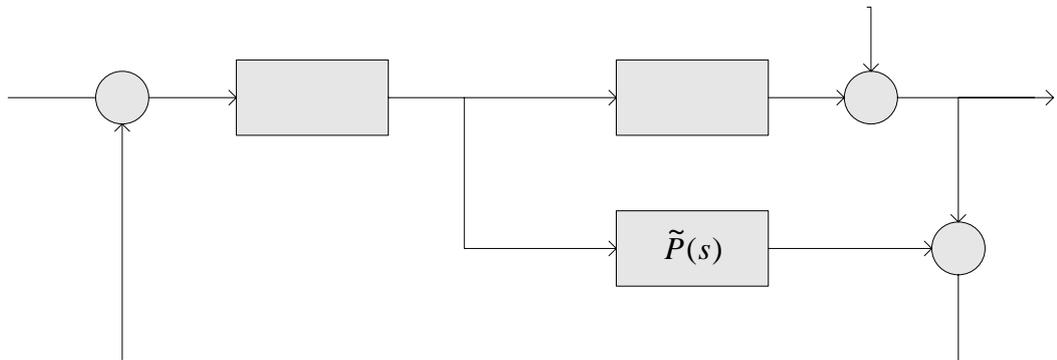
Figure 2.4.2: Internal Model Control Feedback Structure

As it is seen from the block diagram above, the model of the true process $(\tilde{P}(s))$ is connected to unit feedback by negatively. And its input is the control signal. Hence, $\tilde{P}(s)$ can be reconnected as it is shown as the block diagram below.
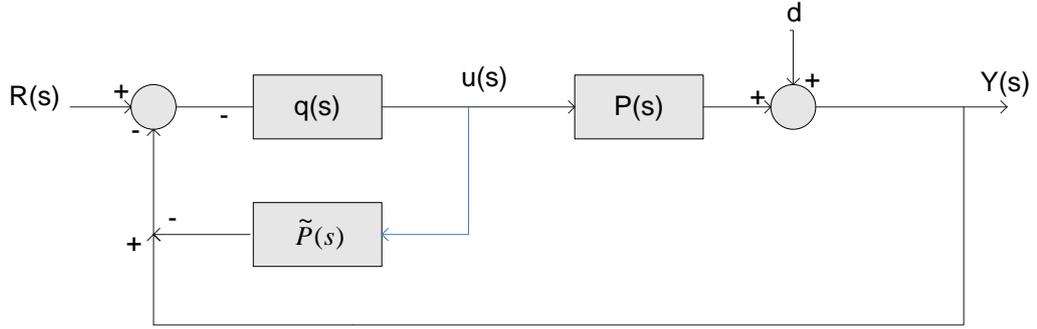
Figure 2.4.3

2.5 The IMC Based PID Controllers

The following steps are used for designing IMC based PID controllers.

1. First of all, the IMC controller transfer function must be defined. As it is stated above, the IMC controller, q(s), must include a filter f(s) for making q(s) semi proper or giving it derivative action (order of the numerator of q(s) is one order greater that the denominator of q(s)) (Bequette, 1994). Thus, q(s) can be written as " $q(s) = \tilde{q}(s)f(s)$ "and it can be written " $\tilde{q}(s) = \tilde{P}_-^{-1}(s)$ " for $\tilde{q}(s)$.

2. The equivalent standard feedback controller transformation must be found by using the transformation of

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} \qquad (2.5.1)$$

3. The equation C(s) can be shown in a PID form. Thereby, the PID constants K_c, T_i, T_D can be found. General equation for IMC based PID can be obtained by

$$C(s) = K_c \left[\frac{T_i T_D s^2 + T_i s + 1}{T_i s} \right] \left[\frac{1}{\lambda s + 1} \right] \qquad (2.5.2)$$

In this equation $\left[\dfrac{1}{\lambda s + 1} \right]$ demonstrates the transfer function of low pass filter.

4. After finishing all these steps, the closed loop simulations must be performed to obtain an efficient controller. The filter constant λ is chosen according to performance (speed of the system result) and robustness.

Example 1:

For this example, the following first order delay free process is considered:

$\tilde{P}(s) = \dfrac{K_P}{T_P s + 1}$ is given as a model of a process.

For finding the PID equivalent to IMC, the steps, expressed below, should be followed.

Step 1: The IMC controller transfer function, q(s), must be found. As it is stated above, q(s) can be found by the formulation of

$$q(s) = \tilde{q}(s) f(s) = \tilde{P}_-^{-1}(s) f(s) \qquad (2.5.2\text{-}1.1)$$

Therefore, q(s) is obtained such as:

$$q(s) = \frac{T_P s + 1}{K_P} \frac{1}{\lambda s + 1} = \frac{1}{K_P} \frac{T_P s + 1}{\lambda s + 1} \qquad (2.5.2\text{-}1.2)$$

Step 2: The equivalent standard feedback controller must be found by the transformation of C(s).

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s) q(s)} \qquad (2.5.2\text{-}1.3)$$

Equivalency of this formula is stated as the transformation of classical feedback controller with respect to IMC. C(s) can be obtained as

$$C(s) = \frac{\dfrac{1}{K_P} \dfrac{T_P s + 1}{\lambda s + 1}}{1 - \dfrac{K_P}{T_P s + 1} \dfrac{1}{K_P} \dfrac{T_P s + 1}{\lambda s + 1}} \qquad (2.5.2\text{-}1.4)$$

If it is made similar,

$$C(s) = \frac{\dfrac{1}{K_P}(T_P s + 1)}{\lambda s + 1 - 1} = \frac{\dfrac{1}{K_P}(T_P s + 1)}{\lambda s} \qquad (2.5.2\text{-}1.5)$$

is obtained.

Step 3: The general transfer function of PI controller can be written as

$$C_1(s) = \frac{K_c(T_i s + 1)}{T_i s} \qquad (2.5.2\text{-}1.6)$$

By rearranging the equation that is found at the *Step 2,* it is obtained that

$$C_2(s) = \frac{T_P}{T_P} \frac{\frac{1}{K_P}(T_P s + 1)}{\lambda s} \qquad (2.5.2\text{-}1.7)$$

and by configuring it, C_2 can be obtained like

$$C_2(s) = \frac{\left(\frac{T_P}{K_P \lambda}\right)(T_P s + 1)}{T_P s} \qquad (2.5.2\text{-}1.8)$$

By equation of $C_1(s)$ and $C_2(s)$, the constants of the PID controller are found as

$$K_C = \frac{T_P}{K_P \lambda} \quad \text{and} \quad T_i = T_P \qquad (2.5.2\text{-}1.9)$$

Example 2:

For this example, the following first order dead time process is considered:

$$\tilde{P}(s) = \frac{K_p e^{-\theta s}}{T_p s + 1}$$

For finding the PID equivalent to IMC, the steps, expressed below, should be followed.

Step 1: There is a dead time in the system. A first-order Pade' approximation can be used for dead time. (Riviera, 1994)

$$\tilde{P}(s) = \frac{K_p e^{-\theta s}}{T_p s + 1} = \frac{K_p\left(\frac{-\theta}{2}s + 1\right)}{\left(T_p s + 1\right)\left(\frac{\theta}{2}s + 1\right)} \qquad (2.5.2\text{-}2.1)$$

Step 2: As it is stated above, $\tilde{P}(s)$ can be separated as $\tilde{P}_+(s)$ and $\tilde{P}_-(s)$.

$$\tilde{P}_-(s) = \frac{K_p}{\left(T_p s + 1\right)\left(\frac{\theta}{2}s + 1\right)} \qquad \tilde{P}_+(s) = \left(\frac{-\theta}{2}s + 1\right) \qquad (2.5.2\text{-}2.2)$$

Step 3: The transfer function of IMC controller must be defined. Before defining the IMC controller, q(s), $\tilde{q}(s)$ must be defined. $\tilde{q}(s)$ can be defined as follows:

$$\tilde{q}(s) = \tilde{P}_-^{-1}(s) = \frac{\left(T_p s + 1\right)\left(\frac{\theta}{2}s + 1\right)}{K_p} \qquad (2.5.2\text{-}2.3)$$

Step 4: The IMC controller, q(s), should be proper. Thus, a filter must be added to make the q(s) proper. Otherwise, a PID controller will not result (Riviera, 1994).

$$q(s) = \tilde{q}(s)f(s) = \tilde{P}_-^{-1}(s)f(s) = \frac{(T_p s + 1)\left(\frac{\theta}{2} s + 1\right)}{K_p} \frac{1}{\lambda s + 1} \quad (2.5.2\text{-}2.4)$$

Now, the PID equivalent of the controller can be found. The classical feedback controller has been found at the above such as:

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} = \frac{\tilde{q}(s)}{1 - \tilde{P}(s)\tilde{q}(s)f(s)} \quad (2.5.2\text{-}2.5)$$

$$C(s) = \frac{\tilde{q}(s)f(s)}{1 - \tilde{P}_-(s)\tilde{P}_+(s)\tilde{P}_-^{-1}(s)f(s)} \quad (2.5.2\text{-}2.6)$$

$$C(s) = \frac{\tilde{q}(s)f(s)}{1 - \tilde{P}_+(s)f(s)} = \left[\frac{1}{K_p}\right] \frac{(T_p s + 1)\left(\frac{\theta}{2} s + 1\right)}{\left(\lambda + \frac{\theta}{2}\right)s} \quad (2.5.2\text{-}2.7)$$

For obtaining a PID equivalent to this controller, the numerator of the controller, C(s), should be expanded such as:

$$C(s) = \frac{\tilde{q}(s)f(s)}{1 - \tilde{P}_+(s)f(s)} = \left[\frac{1}{K_p}\right] \frac{(T_p s + 1)\left(\frac{\theta}{2} s + 1\right)}{\left(\lambda + \frac{\theta}{2}\right)s} = \left[\frac{1}{K_p}\right] \frac{\left(\left(\frac{T_p \theta}{2}\right)s^2 + \left(T_p + \frac{\theta}{2}\right)s + 1\right)}{\left(\lambda + \frac{\theta}{2}\right)s}$$

$$(2.5.2\text{-}2.8)$$

The PID controller transfer function can be demonstrated as:

$$C(s) = K_C \left[\frac{T_D T_I s^2 + T_I s + 1}{T_I s}\right] \quad (2.5.2\text{-}2.9)$$

The denominator and the numerator of the system should be multiplied

by $\dfrac{\left(T_p + \dfrac{\theta}{2}\right)}{\left(T_p + \dfrac{\theta}{2}\right)}$, because of finding a PID equivalent of the system. It is obvious that,

there won't be any changes when the system multiplied by $\dfrac{\left(T_p + \dfrac{\theta}{2}\right)}{\left(T_p + \dfrac{\theta}{2}\right)}$. So, the

following equation is obtained:

$$C(s) = \frac{1}{K_P} \frac{\left(T_p + \dfrac{\theta}{2}\right)}{\left(\lambda + \dfrac{\theta}{2}\right)} \left[\frac{\left(\left(\dfrac{T_p \theta}{2}\right)s^2 + \left(T_p + \dfrac{\theta}{2}\right)s + 1\right)}{\left(T_p + \dfrac{\theta}{2}\right)s} \right] \qquad (2.5.2\text{-}2.10)$$

PID constants can be found easily from this equation such as the following constants:

$$K_C = \frac{1}{K_P} \frac{\left(T_p + \dfrac{\theta}{2}\right)}{\left(\lambda + \dfrac{\theta}{2}\right)} \qquad\qquad T_I = T_p + \frac{\theta}{2} \qquad\qquad T_D = \frac{T_p \theta}{2T_p + \theta} \qquad (2.5.2\text{-}2.11)$$

2.6 PID Tuning Parameters for Open-Loop Stable Processes

	$\tilde{P}(s)$	$\tilde{P}(s)q(s)$	K_C	T_I	T_D	T_F
A	$\dfrac{K_P}{T_p s + 1}$	$\dfrac{1}{\lambda s + 1}$	$\dfrac{T_p}{K_P \lambda}$	T_p	-	-
B	$\dfrac{K_P}{(T_1 s + 1)(T_2 s + 1)}$	$\dfrac{1}{\lambda s + 1}$	$\dfrac{T_1 + T_2}{K_P \lambda}$	$T_1 + T_2$	$\dfrac{T_1 T_2}{T_1 + T_2}$	-
C	$\dfrac{K_P}{T^2 s^2 + 2\zeta T s + 1}$	$\dfrac{1}{\lambda s + 1}$	$\dfrac{2\zeta T}{K_P \lambda}$	$2\zeta T$	$\dfrac{T}{2\zeta}$	-
D	$\dfrac{K_P(-\beta s + 1)}{T^2 s^2 + 2\zeta T s + 1}$	$\dfrac{(-\beta s + 1)}{(\beta s + 1)(\lambda s + 1)}$	$\dfrac{2\zeta T}{K_P(2\beta + \lambda)}$	$2\zeta T$	$\dfrac{T}{2\zeta}$	$\dfrac{\beta \lambda}{2\beta + \lambda}$
E						
F	$\dfrac{K_p}{s}$	$\dfrac{1}{\lambda s + 1}$	$\dfrac{1}{K_P \lambda}$	-	-	-
G	$\dfrac{K_P}{s(T_p s + 1)}$	$\dfrac{1}{\lambda s + 1}$	$\dfrac{1}{K_P \lambda}$	-	T_p	-

Table 2.6.1: PID Tuning Parameters for Open-Loop Stable Processes

In D and E it is assumed that $\beta > 0$, that is, the process has inverse response characteristics (right-half-plane zeros) (Riviera, 1994).

The controller for D is PID + lag, that is

$$C(s) = K_C \left[\frac{T_I T_D s^2 + T_I s + 1}{T_I s} \right] \left[\frac{1}{T_F s + 1} \right] \quad \text{(Riviera, 1994)} \quad (2.6.1)$$

2.7 PID Tuning Parameters for First-Order + Time Delay Processes

Controller	K_C	T_I	T_D	T_F	Notes
PID	$\dfrac{T_p + \dfrac{\theta}{2}}{K_P(\theta + \lambda)}$	$T_p + \dfrac{\theta}{2}$	$\dfrac{T_p \theta}{2T_p + \theta}$	$\dfrac{\theta\lambda}{2(\theta + \lambda)}$	(1)
PID	$\dfrac{T_p + \dfrac{\theta}{2}}{K_P\left(\lambda + \dfrac{\theta}{2}\right)}$	$T_p + \dfrac{\theta}{2}$	$\dfrac{T_p \theta}{2T_p + \theta}$	-	(2)
PI	$\dfrac{T_p}{K_P \lambda}$	T_p	-	-	(3)
Improved PI	$\dfrac{T_p + \dfrac{\theta}{2}}{K_P \lambda}$	$T_p + \dfrac{\theta}{2}$	-	-	(4)

Table 2.7.1: PID Tuning Parameters for First-Order + Time Delay Processes

(1)   With an "all-pass" factorization. Recommended $\lambda > 0.25\theta$

(2)   Without an "all-pass" factorization. Recommended $\lambda > 0.8\theta$

(3)   With zero-order Pade' approximation $(e^{-\theta s} = 0)$. Recommended $\lambda > 1.7\theta$

(4)   With the approximation $\dfrac{K_P e^{-\theta s}}{T_p s + 1} = \dfrac{K_P}{\left(T_p + \dfrac{\theta}{2}\right)s + 1}$. Recommended

$\lambda > 1.7\theta$

3. APPLICATION OF INTERNAL MODEL CONTROL ON FLAPPER POSITION CONTROL SYSTEM

3.1 The Flapper Position Control System

In the Flapper Position Control System, the aim is to control the angle of the flapper with a fan which is run by an asynchronous motor.

The asynchronous motor is run by the asynchronous motor driver of Siemens Sinamics G110. The frequency of the voltage, which is applied to the asynchronous motor, is calibrated proportional by the driver according to the voltage, which is applied to the analogous input of the driver. The analogous input calibrates the frequency of the voltage, applied to the asynchronous motor, from 0 Hz to 50 Hz with a 0 – 10 V voltage signal.
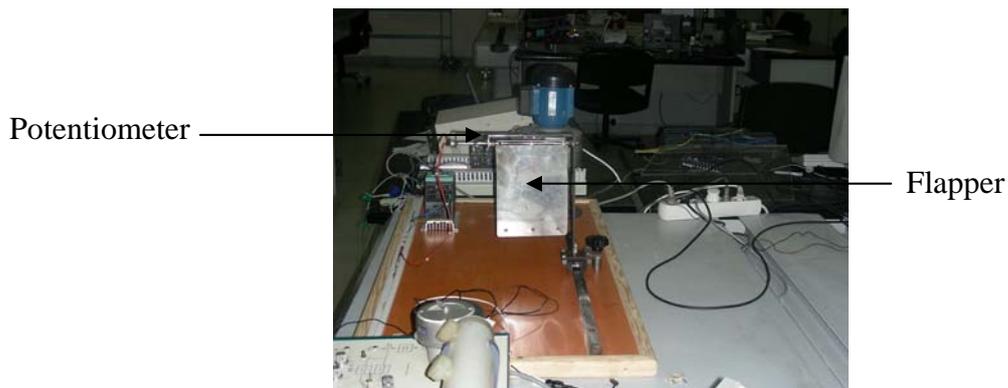
Figure 3.1.1: General Structure of Flapper Position Control System

The angle of the flapper is measured by a potentiometer which is attached to the rotation axis of the flapper. Consequently, the flapper control system is a single

input–single output system (SISO), which has input with a 0 – 10 V voltage, and has an output that is a voltage, measured by potentiometer.

The negative control signal cannot be applied to the system. The interval of the control signal is between 0V and 10V. Furthermore, if the angle between the flapper and vertical axis is 0, the angle actually is not zero. Thus, there is a slipped measurement at the potentiometer.

The flapper position control system is controlled by a Siemens S7 300 PLC. Thereby, an input signal with the interval of 0V – 10V can be applied to the system by the analogue module of this PLC. Moreover, the output information can be obtained as voltage information.

For generating the voltage between the intervals of 0V – 10V from the PLC's analogue module, the integer values between 0 and 27648 must be written to the memory of the PLC's analogue module. If +27648 is written to the memory of the analogue module, it generates +10V output voltage. There is a proportion like that between the integer values and the output voltages.

The minimum value of the data that is read from the system output is 13500, and the maximum value of the data that is read from the system output is 18500. The value of 13500 indicates that the angle between the flapper and the vertical axis is $0°$. Briefly, the output of the system can be controlled between the values 13500 and 18500. The error signal is equals to the subtraction of output signal and the reference signal. So, if the reference signal is given to the system between the values of 13500 and 18500, the system can be controlled.

3.2 The Modeling of the Flapper Position Control System

There are different ways to model the dynamical systems. But first of all, the step responses of the systems are obtained before the modeling. Then, the way of the modeling can be decided.

For the modeling of the flapper control system, the step responses for the different frequencies must be obtained. Because, having the response for only one reference cannot be beneficial for modeling the systems.

The systems in the industrial processes are usually nonlinear dynamical systems. For determining the linearity of the systems, the step responses in the different frequencies can be applied to the systems. Therefore, before the modeling

the flapper control system, the step responses according to the different frequencies are obtained to determine the system's linearity.

The step responses of the flapper control systems for the 10 Hz, 15 Hz, 20 Hz, 25 Hz, 30 Hz, 35 Hz, 40 Hz, 45 Hz and 50 Hz was obtained by applying input signals in these values. After applying inputs in these frequencies, the output signals were collected and the responses of the system drew by Matlab such as below:
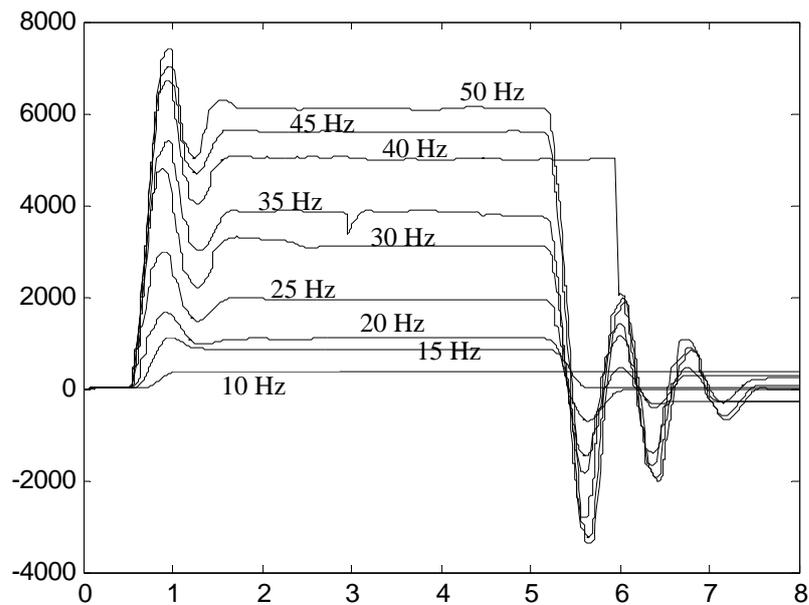
Figure 3.2.1: Responses of the Flapper Position Control system in Different Frequencies

In this graph, the graphics indicate the step responses of the frequencies of 10 Hz, 15 Hz, 20 Hz, 25 Hz, 30 Hz, 35 Hz, 40 Hz, 45 Hz and 50 Hz from up to down respectfully.

It is obvious that, the flapper position control system is a non-linear system, because the system behaves different for all the frequencies.

Non-linearity creates problems when modeling the systems. Moreover, it is really hard to control non-linear systems, because they don't have an exact model for all frequencies. But, when the step responses of the flapper position control are analyzed, the system is not strictly a nonlinear system. The step responses for the frequencies of 35 Hz, 25 Hz and 15 Hz are identical. Thus, the system can be thought as a linear system in these frequencies. Furthermore, the model of the system in these frequencies can be accepted as the general model of the system.

3.2.1 Modeling of the Flapper Control System in 50 Hz

For modeling the flapper control system in 50 Hz, the step response must be obtained initially. 50 Hz equals to 32000 as integer value data. Thus, the 32000 integer value was applied as the input signal to the system. But the driver of the system dont let to apply 32000 to the system. So, 27648 was applied when the 32000 was tried to applied to the system. For higher integer values than 27648, the system model don't change. As a result, the following step response was obtained:
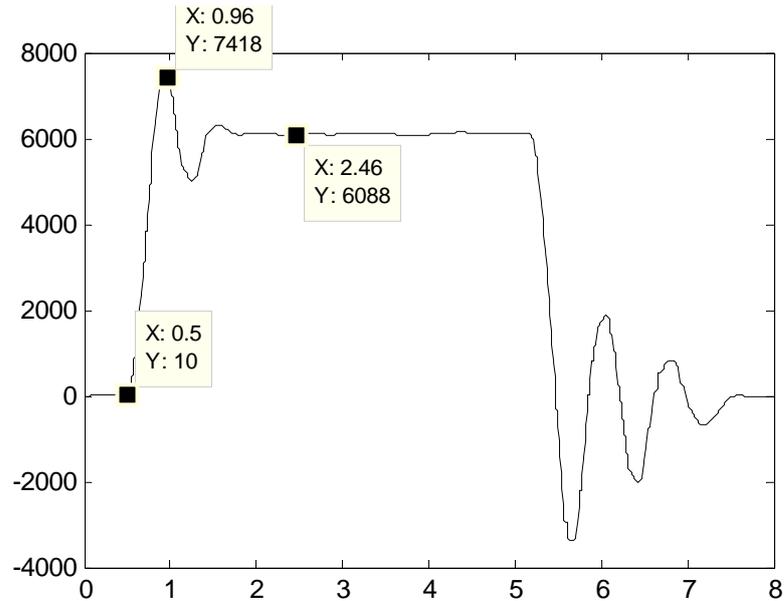
Figure 3.2.1.1: The Response of the System in 50 Hz

The system has a dead time that equals to 0.5 s.

For modeling a system from a step response, the steady state value (y$_{ss}$), the peak value (y$_p$), natural frequency (ω_n), damping ratio are all needed. All these values can be found and calculated from the step response.

A general transfer function for a second order system can be written with a dead time as the following:

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} e^{-hs} \qquad (3.2.1.1)$$

In this equation, K demonstrates the 'Gain', ω_n demonstrates the 'Natural Frequency' and ζ demonstrates the damping ratio, h demonstrated the 'Dead Time' of the system.

According to the step response of the system in 50 Hz frequency, the peak value (y_p) of the step response is 7418. Moreover the steady state value (y_{ss}) is 6088. But all these value are not normalized values. Therefore, it is not true to model the system with these data. So, all these values must be normalized.

For normalizing these values, all these values must be divided by 32000, because these values were obtained by sending 32000 integer value data. Thus, the peak value and the steady state value of the system are obtained such as the following:

$$y_p = 7418/32000 = 0.2318125 \qquad (3.2.1.2)$$

$$y_{ss} = 6088/32000 = 0.19025 \qquad (3.2.1.3)$$

The steady state value also equals to the gain of the system. So, the gain (K) of the system in 50 Hz is 0.19025.

For finding ω_n and ζ the overshoot must be calculated. The overshoot can be calculated such as the following equations:

$$Overshoot\% = \frac{y_p - y_{ss}}{y_{ss}} \times 100 \qquad (3.2.1.4)$$

$$Overshoot\% = \frac{0.2318125 - 0.19025}{0.19025} \times 100 \qquad (3.2.1.5)$$

$$Overshoot\% = 21.8462549\% \qquad (3.2.1.6)$$

The damp ratio (ζ) can be calculated from the overshoot such as the following equations:

$$\zeta = \frac{-\ln(Overshoot)}{\sqrt{\pi^2 + \ln^2(Overshoot)}} \qquad (3.2.1.7)$$

$$\zeta = \frac{-\ln(0.218462549)}{\sqrt{(3.14)^2 + \ln^2(0.218462549)}} \qquad (3.2.1.8)$$

$$\zeta = 0.435796584 \qquad (3.2.1.9)$$

After finding the damp ratio, the natural frequency is needed to complete the transfer function of the flapper position control system. Natural frequency can be found from the settling time. The formula for the settling time can be given for the 2 % band such as the equation of (3.2.1.10).

$$T_s = \frac{4}{\zeta \omega_n} \qquad (3.2.1.10)$$

The settling time of the system can be read from the step response of the system in 50 Hz frequency.

X: 1.82
Y: 6088
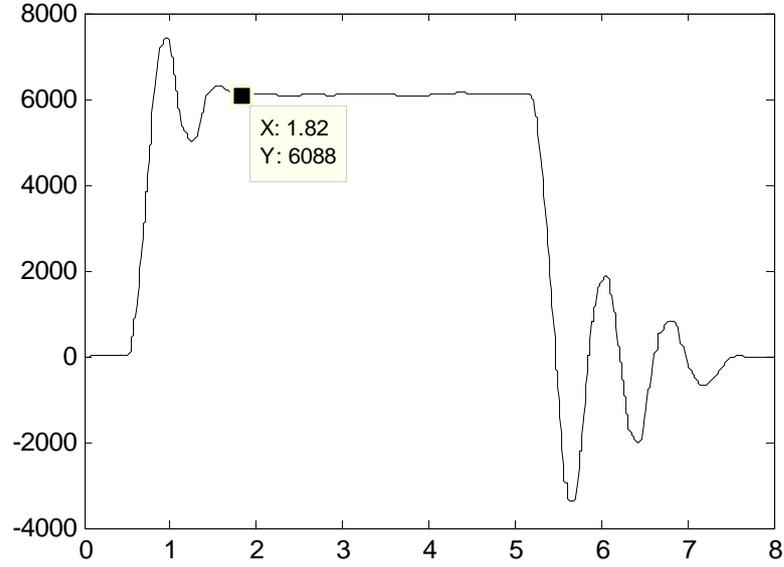
Figure 3.2.1.2: Settling time of the System in 50 Hz

According to the step response of the system in 50 Hz, the settling time of the system is 1.82 s. So, from the equation (3.2.1.10), ω_n can be calculated such as the following:

$$\omega_n = \frac{4}{\zeta T_s} \qquad (3.2.1.11)$$

$$\omega_n = \frac{4}{0.435796584 \bullet 1.82} \qquad (3.2.1.12)$$

$$\omega_n = 5.043183929 \qquad (3.2.1.13)$$

All the values that are needed to model a system are found. So, the transfer function of the system can be written according to the equation (3.2.1.14).

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} e^{-hs} \qquad (3.2.1.14)$$

By the way, the model of the flapper position control system for 50 Hz is given below:

$$P(s) = \frac{4.838762213}{s^2 + 4.395604657s + 25.43370414} e^{-0.5s} \quad (3.2.1.15)$$

The simulation of the system in Simulink can be shown as below:
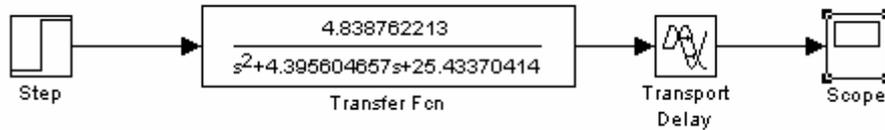
Figure 3.2.1.3: The Simulation Model of the system in 50 Hz

The response of this model can be shown as below:
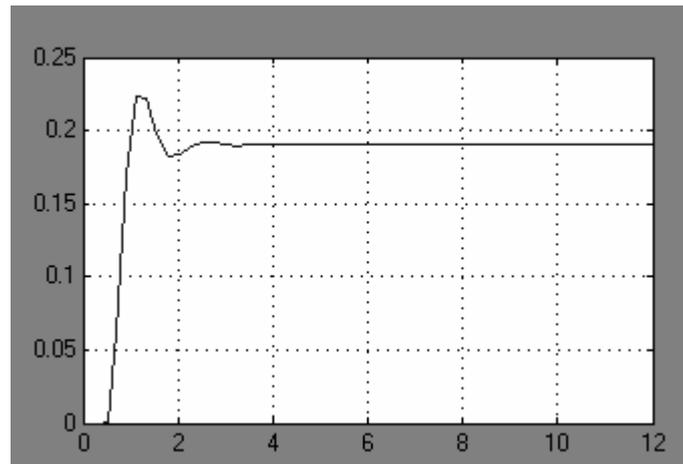
Figure 3.2.1.4: The Response of the Simulation Model in 50 Hz

3.2.2 Modeling of the Flapper Control System in 35 Hz

For modeling the flapper control system in 35 Hz, the step response must be obtained initially. 35 Hz equals to 22400 as integer value data. Thus, the 22400 integer value was applied as the input signal to the system. As a result, the following step response was obtained:
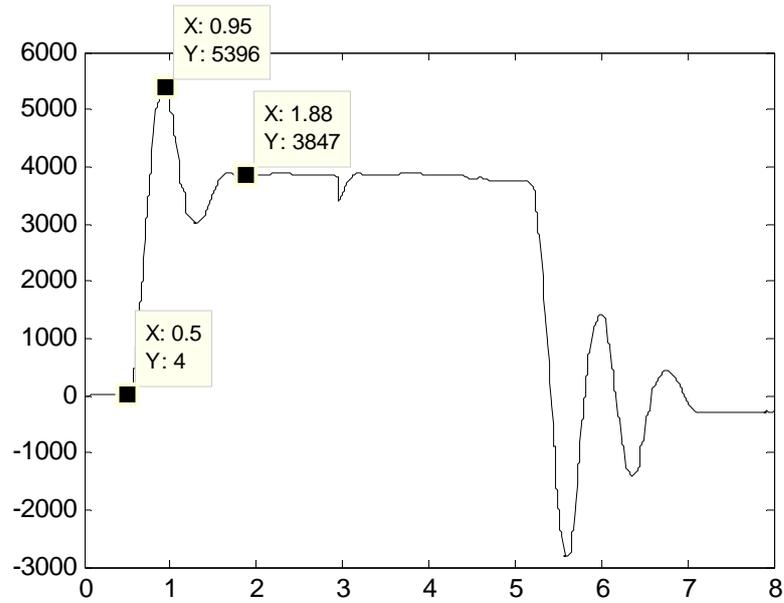
Figure 3.2.2.1: The Response of the System in 35 Hz

The system has a dead time that equals to 0.5 s.

For modeling a system from a step response, the steady state value (y_{ss}), the peak value (y_p), natural frequency (ω_n), damping ratio are all needed. All these values can be found and calculated from the step response.

A general transfer function for a second order system with a dead time can be written as the following:

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} e^{-hs} \qquad (3.2.2.1)$$

In this equation, K demonstrates the 'Gain', ω_n demonstrates the 'Natural Frequency' and ζ demonstrates the damping ratio, h demonstrates the 'Dead Time' of the system.

According to the step response of the system in 35 Hz frequency, the peak value (y_p) of the step response is 5396. Moreover the steady state value (y_{ss}) is3847. But all these value are not normalized values. Therefore, it is not true to model the system with these data. So, all these values must be normalized.

For normalizing these values, all these values must be divided by 22400, because these values were obtained by sending 22400 integer value data. Thus, the peak value and the steady state value of the system are obtained such as the following:

26

$$y_p = 5396/22400 = 0.240892857 \qquad (3.2.2.2)$$

$$y_{ss} = 3847/22400 = 0.171741071 \qquad (3.2.2.3)$$

The steady state value also equals to the gain of the system. So, the gain (K) of the system in 35 Hz is 0.171741071.

For finding ω_n and ζ the overshoot must be calculated. The overshoot can be calculated such as the following equations:

$$Overshoot\% = \frac{y_p - y_{ss}}{y_{ss}} \times 100 \qquad (3.2.2.4)$$

$$Overshoot\% = \frac{0.240892857 - 0.171741071}{0.171741071} \times 100 \quad (3.2.2.5)$$

$$Overshoot\% = 40.2651416\% \qquad (3.2.2.6)$$

The damp ratio (ζ) can be calculated from the overshoot such as the following equations:

$$\zeta = \frac{-\ln(Overshoot)}{\sqrt{\pi^2 + \ln^2(Overshoot)}} \qquad (3.2.2.7)$$

$$\zeta = \frac{-\ln(0.402651416)}{\sqrt{(3.14)^2 + \ln^2(0.402651416)}} \qquad (3.2.2.8)$$

$$\zeta = 0.278135843 \qquad (3.2.2.9)$$

After finding the damp ratio, the natural frequency is needed to complete the transfer function of the flapper position control system. Natural frequency can be found from the settling time. The formula for the settling time can be given for the 2 % band such as the following equation:

$$T_s = \frac{4}{\zeta\omega_n} \qquad (3.2.2.10)$$

The settling time of the system can be read from the step response of the system in 35 Hz frequency.
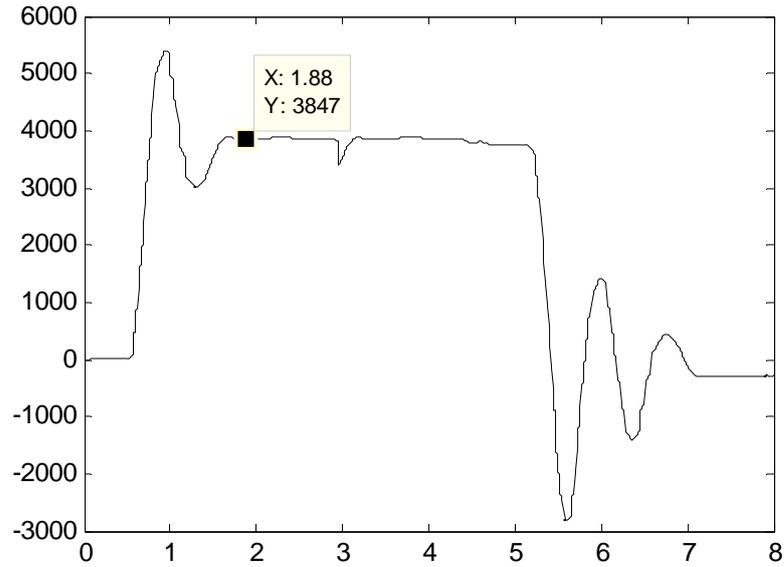
Figure 3.2.2.2: The Settling Time of the System in 35 Hz

According to the step response of the system in 35 Hz, the settling time of the system is 1.88 s. So, from the equation (3.2.2.10), ω_n can be calculated such as the following:

$$\omega_n = \frac{4}{\zeta T_s} \tag{3.2.2.11}$$

$$\omega_n = \frac{4}{0.278135843 \bullet 1.88} \tag{3.2.2.12}$$

$$\omega_n = 7.649713721 \tag{3.2.2.13}$$

All the values that are needed to model a system are found. So, the transfer function of the system can be written according to the equation (3.2.2.14)

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} e^{-hs} \tag{3.2.2.14}$$

By the way, the model of the flapper position control for system is given below:

$$P(s) = \frac{10.0499646}{s^2 + 4.255319149s + 58.51812001} e^{-0.5s} \tag{3.2.2.15}$$

28

The simulation of the system in Simulink can be shown as below:
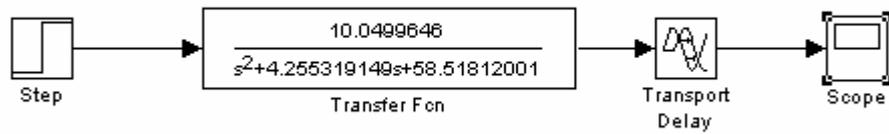
Figure 3.2.2.3: The Simulation Model of the System in 35 Hz

The response of this model can be shown as below:
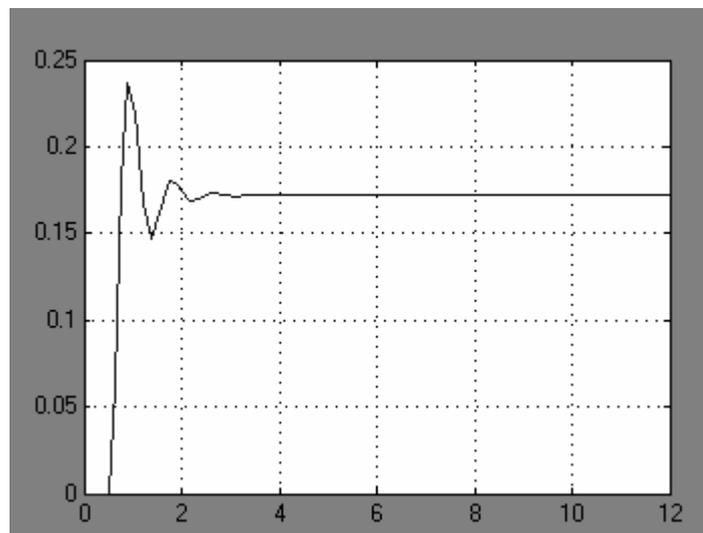
Figure 3.2.2.4: The Response of the Simulation Model in 35 Hz

3.2.3 Modeling of the Flapper Control System in 25 Hz

For modeling the flapper control system in 25 Hz, the step response must be obtained initially. 25 Hz equals to 16000 as integer value data. Thus, the 16000 integer value was applied as the input signal to the system. As a result, the following step response was obtained:
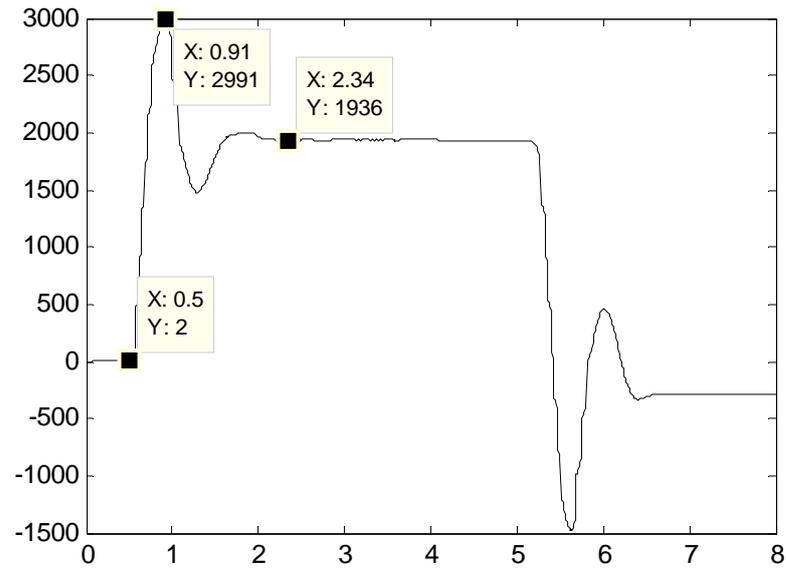
Figure 3.2.3.1: The Response of the System in 25 Hz

The system has a dead time that equals to 0.5s

For modeling a system from a step response, the steady state value (y_{ss}), the peak value (y_p), natural frequency (ω_n), damping ratio are all needed. All these values can be found and calculated from the step response.

A general transfer function for a second order system with a dead time can be written as the following:

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} e^{-hs} \qquad (3.2.3.1)$$

In this equation, K demonstrates the 'Gain', ω_n demonstrates the 'Natural Frequency' and ζ demonstrates the damping ratio, h demonstrates the 'Dead Time' of the system.

According to the step response of the system in 25 Hz frequency, the peak value (y_p) of the step response is 2991. Moreover the steady state value (y_{ss}) is 1936. But all these value are not normalized values. Therefore, it is not true to model the system with these data. So, all these values must be normalized.

For normalizing these values, all these values must be divided by 16000, because these values were obtained by sending 16000 integer value data. Thus, the peak value and the steady state value of the system are obtained such as the following:

30

$$y_p = 2991/16000 = 0.1869375 \qquad (3.2.3.2)$$

$$y_{ss} = 1936/16000 = 0.121 \qquad (3.2.3.3)$$

The steady state value also equals to the gain of the system. So, the gain (K) of the system in 25 Hz is 0.121.

For finding ω_n and ζ the overshoot must be calculated. The overshoot can be calculated such as the following equations:

$$Overshoot\% = \frac{y_p - y_{ss}}{y_{ss}} \times 100 \qquad (3.2.3.4)$$

$$Overshoot\% = \frac{0.1869375 - 0.121}{0.121} \times 100 \qquad (3.2.3.5)$$

$$Overshoot\% = 54.49380165\% \qquad (3.2.3.6)$$

The damp ratio (ζ) can be calculated from the overshoot such as the following equations:

$$\zeta = \frac{-\ln(Overshoot)}{\sqrt{\pi^2 + \ln^2(Overshoot)}} \qquad (3.2.3.7)$$

$$\zeta = \frac{-\ln(0.5449380165)}{\sqrt{(3.14)^2 + \ln^2(0.5449380165)}} \qquad (3.2.3.8)$$

$$\zeta = 0.189730605 \qquad (3.2.3.9)$$

After finding the damp ratio, the natural frequency is needed to complete the transfer function of the flapper position control system. Natural frequency can be found from the settling time. The formula for the settling time can be given for the 2 % band such as the following equation:

$$T_s = \frac{4}{\zeta \omega_n} \qquad (3.2.3.10)$$

The settling time of the system can be read from the step response of the system in 25 Hz frequency.
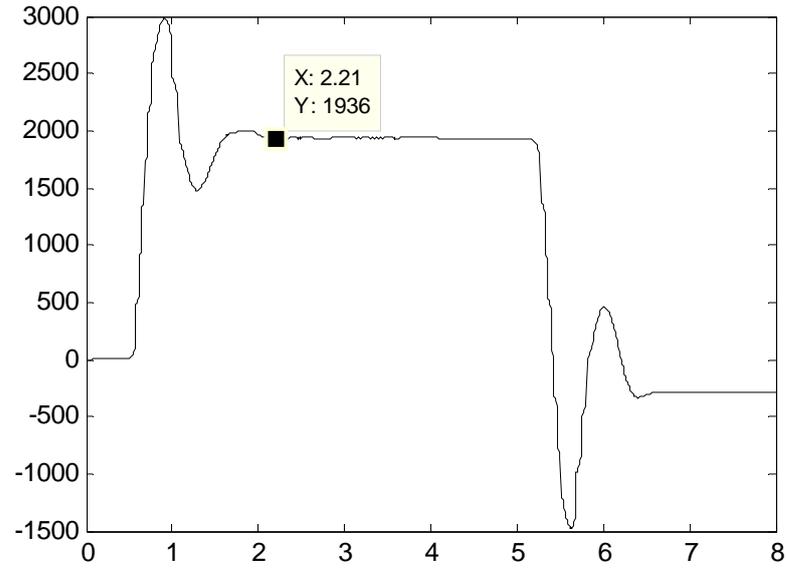
Figure 3.2.3.2: The Settling Time of the System in 25 Hz

According to the step response of the system in 25 Hz, the settling time of the system is 2.21s. So, from the equation (3.2.3.10), ω_n can be calculated such as the following:

$$\omega_n = \frac{4}{\zeta T_s} \qquad (3.2.3.11)$$

$$\omega_n = \frac{4}{0.189730605 \bullet 2.21} \qquad (3.2.3.12)$$

$$\omega_n = 9.539603512 \qquad (3.2.3.13)$$

All the values that are needed to model a system are found. So, the transfer function of the system can be written according to the equation (3.2.3.14)

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \qquad (3.2.3.14)$$

By the way, the model of the flapper position control for system is given below:

$$P(s) = \frac{11.01148825}{s^2 + 3.619909492s + 91.00403517} \qquad (3.2.3.15)$$

The simulation of the system in Simulink can be shown as below:
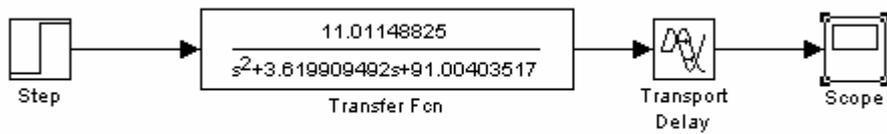
Figure 3.2.3.3: The Simulation Model of the System in 25 Hz

The response of this model can be shown as below:
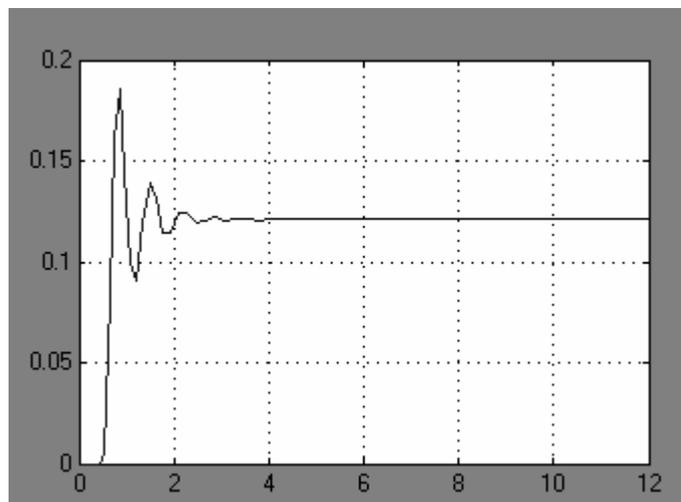
Figure 3.2.3.4: The response of the Simulation Model in 25 Hz

3.2.4 Modeling of the Flapper Control System in 15 Hz

For modeling the flapper control system in 15 Hz, the step response must be obtained initially. 15 Hz equals to 9600 as integer value data. Thus, the 9600 integer value was applied as the input signal to the system. As a result, the following step response was obtained:
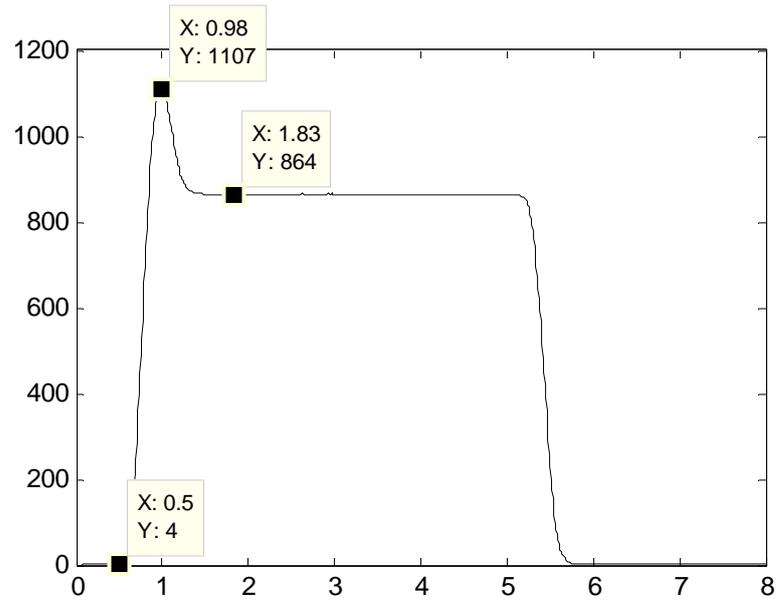
Figure 3.2.4.1: The Response of the System in 15 Hz

The system has a dead time that equals to 0.5s

For modeling a system from a step response, the steady state value (y_{ss}), the peak value (y_p), natural frequency (ω_n), damping ratio are all needed. All these values can be found and calculated from the step response.

A general transfer function for a second order system with a dead time can be written as the following:

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} e^{-hs} \qquad (3.2.4.1)$$

In this equation, K demonstrates the 'Gain', ω_n demonstrates the 'Natural Frequency' and ζ demonstrates the damping ratio, h demonstrates the 'Dead Time' of the system.

According to the step response of the system in 15 Hz frequency, the peak value (y_p) of the step response is 1107. Moreover the steady state value (y_{ss}) is 864. But all these value are not normalized values. Therefore, it is not true to model the system with these data. So, all these values must be normalized.

For normalizing these values, all these values must be divided by 9600, because these values were obtained by sending 9600 integer value data. Thus, the peak value and the steady state value of the system are obtained such as the following:

$$y_p = 1107/9600 = 0.1153125 \qquad (3.2.4.2)$$

$$y_{ss} = 864/9600 = 0.09 \qquad (3.2.4.3)$$

The steady state value also equals to the gain of the system. So, the gain (K) of the system in 15 Hz is 0.09.

For finding ω_n and ζ the overshoot must be calculated. The overshoot can be calculated such as the following equations:

$$Overshoot\% = \frac{y_p - y_{ss}}{y_{ss}} \times 100 \qquad (3.2.4.4)$$

$$Overshoot\% = \frac{0.1153125 - 0.09}{0.09} \times 100 \qquad (3.2.4.5)$$

$$Overshoot\% = 28.125\% \qquad (3.2.4.6)$$

The damp ratio (ζ) can be calculated from the overshoot such as the following equations:

$$\zeta = \frac{-\ln(Overshoot)}{\sqrt{\pi^2 + \ln^2(Overshoot)}} \qquad (3.2.4.7)$$

$$\zeta = \frac{-\ln(0.28125)}{\sqrt{(3.14)^2 + \ln^2(0.28125)}} \qquad (3.2.4.8)$$

$$\zeta = 0.374410065 \qquad (3.2.4.9)$$

After finding the damp ratio, the natural frequency is needed to complete the transfer function of the flapper position control system. Natural frequency can be found from the settling time. The formula for the settling time can be given for the 2 % band such as equation of (3.2.4.10):

$$T_s = \frac{4}{\zeta \omega_n} \qquad (3.2.4.10)$$

The settling time of the system can be read from the step response of the system in 15 Hz frequency.
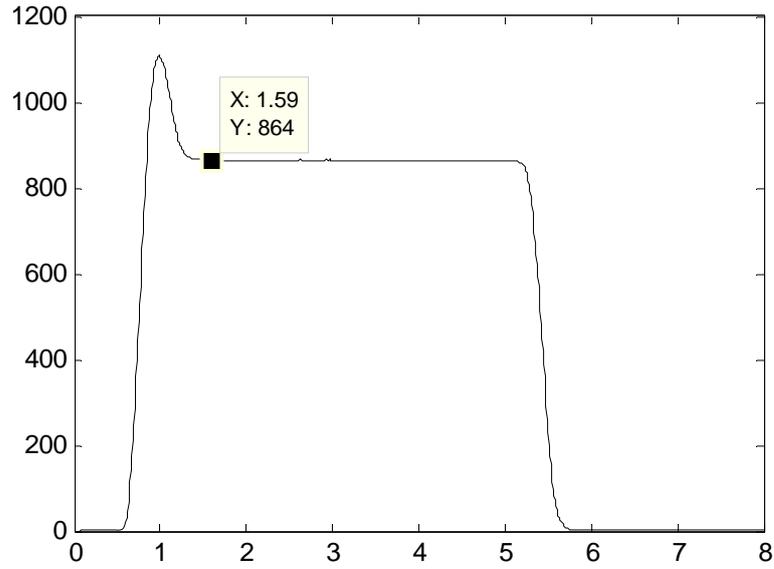
Figure 3.2.4.2: The Settling Time of the System in 15 Hz

According to the step response of the system in 15 Hz, the settling time of the system is 1.59s. So, from the equation (3.2.4.10), ω_n can be calculated such as the following:

$$\omega_n = \frac{4}{\zeta T_s} \tag{3.2.4.11}$$

$$\omega_n = \frac{4}{0.374410065 \bullet 1.59} \tag{3.2.4.12}$$

$$\omega_n = 6.71916571 \tag{3.2.4.13}$$

All the values that are needed to model a system are found. So, the transfer function of the system can be written according to the equation (3.2.4.14).

$$P(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} e^{-hs} \tag{3.2.4.14}$$

By the way, the model of the flapper position control for system is given below:

$$P(s) = \frac{4.063246906}{s^2 + 5.03144654s + 45.14718784} e^{-0.5s} \tag{3.2.4.15}$$

36

The simulation of the system in Simulink can be shown as below:
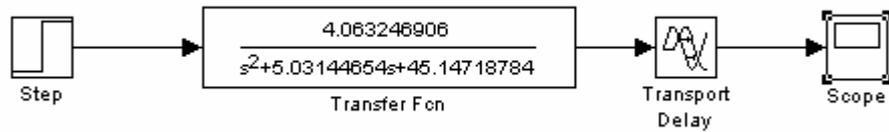
Figure 3.2.4.3: The Simulation Model of the System in 15 Hz

The response of this model can be shown as below:
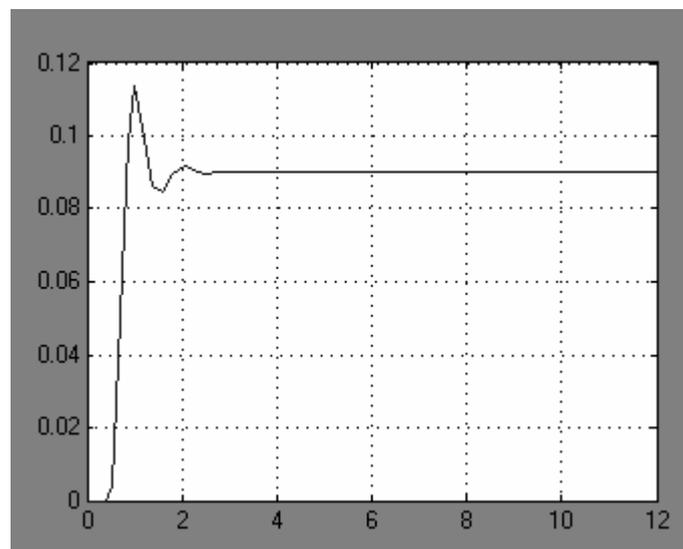
Figure 3.2.4.4: The Response of the Simulation Model in 15 Hz

3.2.5 Modeling of the Flapper Control System in 10 Hz

For modeling the flapper control system in 10 Hz, the step response must be obtained initially. 10 Hz equals to 6400 as integer value data. Thus, the 6400 integer value was applied as the input signal to the system. As a result, the following step response was obtained:
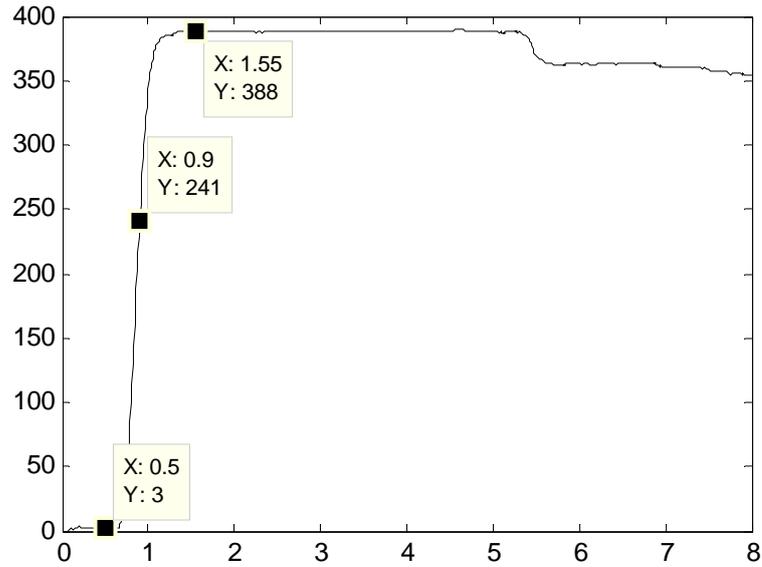
Figure 3.2.5.1: The Response of the System in 10 Hz

It is obvious that, the system behaves as a first order system when the 15 Hz input signal is applied. Therefore, the type of modeling is changed.

A general transfer function for a first order system with a dead time can be written as the following:

$$P(s) = \frac{K}{\tau s + 1} \qquad (3.2.5.1)$$

In this equation, K demonstrates the 'Gain', τ demonstrates the 'Time Constant' of the system.

According to the step response of the system in 10 Hz frequency, the gain (K) of the system is 388. But this value is not normalized. Therefore, it is not true to model the system with these data. So, this value must be normalized.

Moreover the time constant (τ) of the system is 244. The time constant equals to the value of time when system is at 63% of the step response. For normalizing these values, all these values must be divided by 6400, because these values were obtained by sending 6400 integer value data. Thus, the gain of the system is obtained such as the following:

$$K = 388 / 6400 = 0.060625 \qquad (3.2.5.2)$$

$$\tau = 0.63 \bullet 388 = 244 \qquad (3.2.5.3)$$

According to the step response of the system in 10 Hz frequency, the system reaches to 244 at 0.9 s.

All the values that are needed to model a system are found. So, the transfer function of the system can be written according to the equation (3.2.5.4)

$$P(s) = \frac{K}{\tau s + 1} e^{-hs} \qquad (3.2.5.4)$$

By the way, the model of the flapper position control for system is given below:

$$P(s) = \frac{0.060625}{0.9s + 1} e^{-0.5s} \qquad (3.2.5.5)$$

The simulation of the system in Simulink can be shown as below:
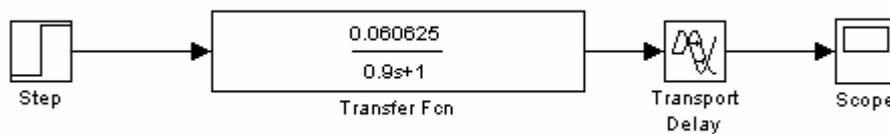
Figure 3.2.5.2: The Simulation Model of the System in 10 Hz

The response of this model can be shown as below:
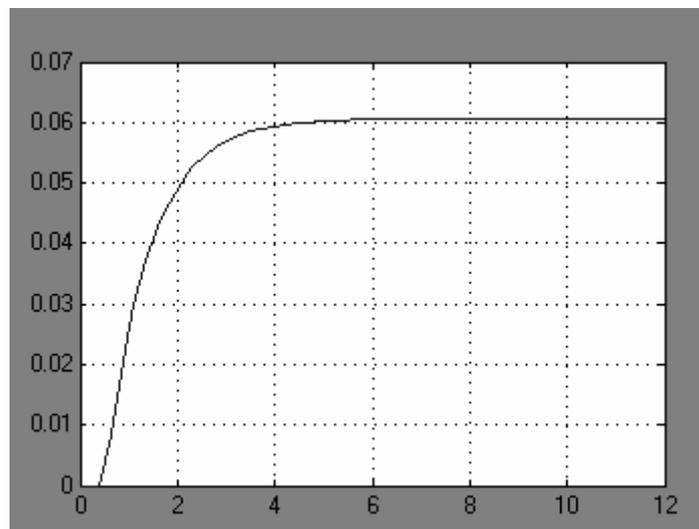
Figure 3.2.5.3: The Response of the Simulation Model in 10 Hz

3.3 IMC Based PID Controller Design for the Flapper Position Control System

The flapper position control system can be controlled with a PID controller. As it is told above, the constant of PID controller can be found by Internal Model Control procedure.

Initially, the process model must be defined for designing the IMC based PID controller. In the flapper control system, the transfer function of the system and

model of the system are different. Because of not having the transfer function of the system, the transfer function of the system and the model of the system are accepted as equal. This creates the perfect matching situation. As it is written at the formula of (2.3.6) , if the perfect matching occurs, the output simplifies to:

$$Y(s) = \frac{P(s)q}{1 + \underbrace{q(P(s) - \tilde{P}(s))}_{0}} R(s) + \frac{1 - \tilde{P}(s)q}{1 + \underbrace{q(P(s) - \tilde{P}(s))}_{0}} d \qquad (3.3.1)$$

$$Y(s) = \tilde{P}(s)qR(s) + (1 - \tilde{P}(s)q)d(s) \qquad (3.3.2)$$

In this equations, $\tilde{P}(s)$ demonstrates the model of the process, q demonstrates the IMC controller, R(s) demonstrates the reference signal and the d(s) demonstrates the disturbance signal.

Moreover, as it is written in the equation (2.4.4) general feedback structure controller (C(s)) is calculated as:

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} \qquad (3.3.3)$$

As it is written in the equation….the IMC controller is calculated as in the following equations:

$$\tilde{q}(s) = \tilde{P}_{-}^{-1}(s) \qquad (3.3.4)$$

$$q(s) = \tilde{q}(s)f(s) \qquad (3.3.5)$$

$$f(s) = \frac{1}{(\lambda s + 1)^{n}} \text{ (low band pass filter)} \qquad (3.3.6)$$

3.3.1 Designing of IMC Based PID Controller in 50 Hz Frequency for the Flapper Position Control System

For designing IMC based PID controller for the system, the system model must be defined. The models have been already defined above. For the flapper control system, the transfer function of the system and the model of the system can be accepted equal. This situation makes easier to find constants of PID controller.

The model of the system in 50 Hz frequency has been already found as:

$$\tilde{P}(s) = \frac{4.838762213}{s^{2} + 4.395604657s + 25.43370414} e^{-0.5s} \qquad (3.3.1.1)$$

The dead time in the transfer function can be written in the Taylor series such as the following equation:

$$e^{-hs} = (1 - hs) \qquad (3.3.1.2)$$

$$\tilde{P}(s) = \frac{4.838762213}{s^2 + 4.395604657s + 25.43370414} e^{-0.5s} = \frac{4.838762213(1 - 0.5s)}{s^2 + 4.395604657s + 25.43370414}$$

$$(3.3.1.3)$$

By the way, $e^{-0.5s}$ is eliminated.

The model of the system and the transfer function of the system are accepted perfectly matched. So, the transfer function of the system can be written as:

$$\tilde{P}(s) = P(s) = \frac{4.838762213}{s^2 + 4.395604657s + 25.43370414} e^{-0.5s} = \frac{4.838762213(1 - 0.5s)}{s^2 + 4.395604657s + 25.43370414}$$

$$(3.3.1.4)$$

The IMC controller (q(s)) must be calculated to find the general feedback structure controller (C(s)). q(s) can be found as in the following equations:

$$\tilde{q}(s) = \tilde{P}_-^{-1}(s) \qquad (3.3.1.5)$$

$$\tilde{P}_- = \frac{4.838762213}{s^2 + 4.395604657s + 25.43370414} \qquad (3.3.1.6)$$

$$\tilde{q}(s) = \frac{s^2 + 4.395604657s + 25.43370414}{4.838762213} \qquad (3.3.1.7)$$

All the transfer functions for creating the general feedback structure controller is obtained now. So, the general feedback structure controller can for designing the IMC controller (q(s)), a low pass filter must be designed. For the flapper control system, the filter can be designed as below:

$$f(s) = \frac{1}{(\lambda s + 1)^n} \qquad (3.3.1.8)$$

$$f(s) = \frac{1}{\lambda s + 1} \qquad (3.3.1.9)$$

The low pass filter can be chosen as a first order filter.

The IMC Controller, q(s), can be written as:

$$q(s) = \tilde{q}(s)f(s) = \frac{s^2 + 4.395604657s + 25.43370414}{4.838762213} \bullet \frac{1}{\lambda s + 1} \qquad (3.3.1.10)$$

be calculated as in the following equations:

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} \qquad (3.3.1.11)$$

$$C(s) = \frac{\dfrac{s^2 + 4.395604657s + 25.43370414}{4.838762213} \bullet \dfrac{1}{\lambda s + 1}}{1 - \dfrac{4.838762213(1 - 0.5s)}{s^2 + 4.395604657s + 25.43370414} \bullet \dfrac{s^2 + 4.395604657s + 25.43370414}{4.838762213(\lambda s + 1)}}$$

$$(3.3.1.12)$$

This equation simplifies to:

$$C(s) = \frac{s^2 + 4.395604657s + 25.4337044}{4.838762213(\lambda + 0.5)s} \qquad (3.3.1.13)$$

This transfer function also equals a PID controller transfer function. To show the equality of it, writing the general PID controller transfer function may be beneficial. The general transfer function of a PID controller can be written such as in the following:

$$K_c\left(1 + \frac{1}{T_i s} + T_D s\right) \qquad (3.3.1.14)$$

This equation can be formed as below:

$$K_c\left(1 + \frac{1}{T_i s} + T_D s\right) = K_C\left(\frac{T_i s + 1 + T_i T_D s^2}{T_i s}\right) \qquad (3.3.1.15)$$

$$K_C\left(\frac{T_i s + 1 + T_i T_D s^2}{T_i s}\right) = K_C\left(\frac{T_D s^2 + s + \dfrac{1}{T_i}}{s}\right) \qquad (3.3.1.16)$$

$$K_C \left(\frac{T_D s^2 + s + \frac{1}{T_i}}{s} \right) = K_C \left(\frac{s^2 + \frac{1}{T_D} s + \frac{1}{T_i T_D}}{T_D s} \right) \qquad (3.3.1.17)$$

So, a general PID controller transfer function can be also written as:

$$K_C \left(\frac{s^2 + \frac{1}{T_D} s + \frac{1}{T_i T_D}}{T_D s} \right) \qquad (3.3.1.18)$$

The general feedback structure controller transfer function of the system has already been found as:

$$C(s) = \frac{s^2 + 4.395604657 s + 25.4337044}{4.838762213(\lambda + 0.5)s} \qquad (3.3.1.19)$$

Comparing with the transfer function of a PID controller, it is obvious that the transfer function of the general feedback structure controller also leads to a PID controller. By the way the PID controller constants can be found. So, The PID controller constant can be written as the followings:

$$\frac{1}{T_D} = 4.395604657 \Rightarrow T_D = 0.227499986 \qquad (3.3.1.20)$$

$$\frac{1}{T_i T_D} = 25.4337044 \Rightarrow T_i = 0.17282597 \qquad (3.3.1.21)$$

$$\frac{K_C}{T_D} = \frac{1}{4.838762213(\lambda + 0.5)} \Rightarrow K_C(\lambda + 0.5) = 0.047016153 \qquad (3.3.1.22)$$

For the flapper control system, the filter constant λ can be chosen the same value as dead time. The following simulation was made when λ is 0.5.
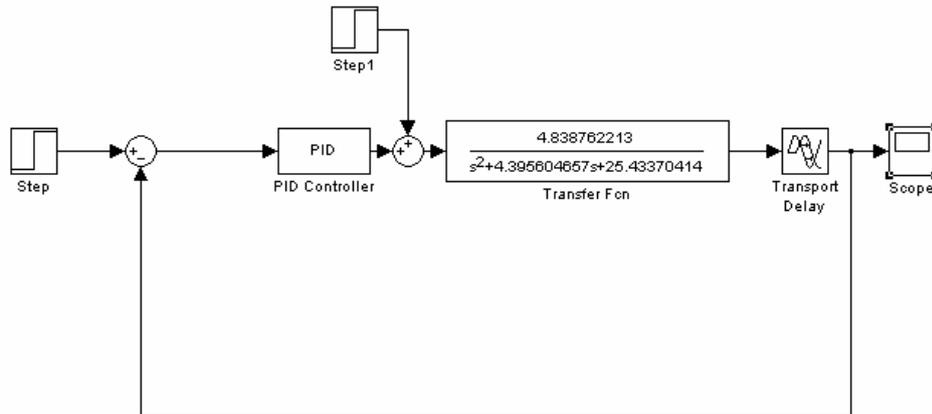
Figure 3.3.1.1: The Simulation Model of IMC Based PID for the System in 50 Hz

The result of the simulation:
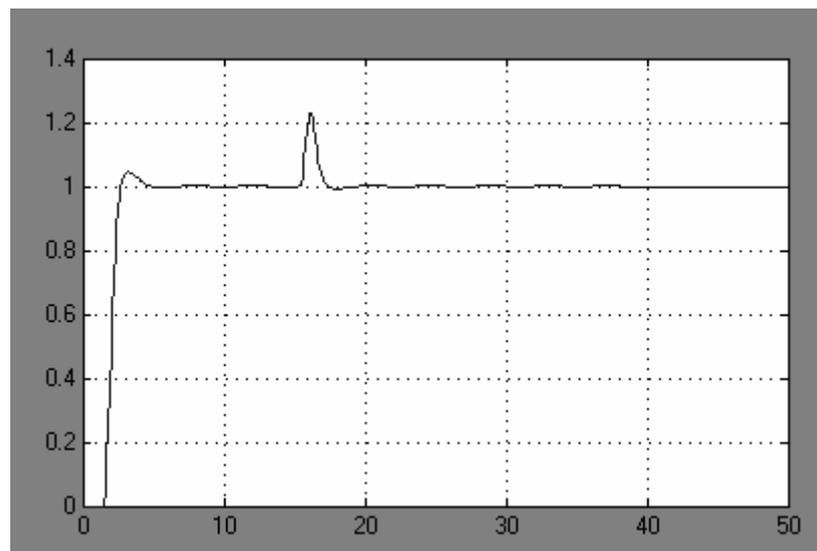
Figure 3.3.1.2: The Response of the Simulation Model of IMC Based PID for the System in 50 Hz

3.3.2 Designing of IMC Based PID Controller in 35 Hz Frequency for the Flapper Position Control System

For the flapper control system, the transfer function of the system and the model of the system can be accepted equal. This situation makes easier to find constants of PID controller.

The model of the system in 35 Hz frequency has been already found as:

$$\tilde{P}(s) = \frac{10.0499646}{s^2 + 4.255319149s + 58.51812001} e^{-0.5s} \qquad (3.3.2.1)$$

The dead time in the transfer function can be written in the Taylor series such as the following equation:

$$e^{-hs} = (1 - hs) \qquad (3.3.2.2)$$

$$\tilde{P}(s) = \frac{10.0499646}{s^2 + 4.255319149s + 58.51812001} e^{-0.5s} = \frac{10.0499646(1 - 0.5s)}{s^2 + 4.255319149s + 58.51812001}$$

$$(3.3.2.3)$$

By the way, $e^{-0.5s}$ is eliminated.

The model of the system and the transfer function of the system are accepted perfectly matched. So, the transfer function of the system can be written as:

$$\tilde{P}(s) = P(s) = \frac{10.0499646}{s^2 + 4.255319149s + 58.51812001} e^{-0.5s} = \frac{10.0499646(1 - 0.5s)}{s^2 + 4.255319149s + 58.51812001}$$

$$(3.3.2.4)$$

The IMC controller (q(s)) must be calculated to find the general feedback structure controller (C(s)). q(s) can be found as in the following equations:

$$\tilde{q}(s) = \tilde{P}_-^{-1}(s) \qquad (3.3.2.5)$$

$$\tilde{P}_- = \frac{10.0499646}{s^2 + 4.255319149s + 58.51812001} \qquad (3.3.2.6)$$

$$\tilde{q}(s) = \frac{s^2 + 4.255319149s + 58.51812001}{10.0499646} \qquad (3.3.2.7)$$

All the transfer functions for creating the general feedback structure controller is obtained now. So, the general feedback structure controller can for designing the IMC controller (q(s)), a low pass filter must be designed. For the flapper control system, the filter can be designed as below:

$$f(s) = \frac{1}{(\lambda s + 1)^n} \qquad (3.3.2.8)$$

The low pass filter can be chosen as a first order filter.

The IMC Controller, q(s), can be written as:

$$q(s) = \tilde{q}(s)f(s) = \frac{s^2 + 4.255319149s + 58.51812001}{10.0499646} \bullet \frac{1}{\lambda s + 1} \qquad (3.3.2.9)$$

be calculated as in the following equations:

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} \qquad (3.3.2.10)$$

$$C(s) = \frac{\dfrac{s^2 + 4.255319149s + 58.51812001}{10.0499646} \bullet \dfrac{1}{\lambda s + 1}}{1 - \dfrac{10.0499646(1 - 0.5s)}{s^2 + 4.255319149s + 58.51812001} \bullet \dfrac{s^2 + 4.255319149s + 58.51812001}{10.0499646(\lambda s + 1)}} \qquad (3.3.2.11)$$

This equation simplifies to:

$$C(s) = \frac{s^2 + 4.2553119149s + 58.51812001}{10.0499646(\lambda + 0.5)s} \qquad (3.3.2.12)$$

The general transfer function of a PID controller can be written such as in the following:

$$K_C \left(\frac{s^2 + \dfrac{1}{T_D}s + \dfrac{1}{T_i T_D}}{T_D s} \right) \qquad (3.3.2.13)$$

The general feedback structure controller transfer function of the system has already been found as:

$$C(s) = \frac{s^2 + 4.2553119149s + 58.51812001}{10.0499646(\lambda + 0.5)s} \qquad (3.3.2.14)$$

Comparing with the transfer function of a PID controller, it is obvious that the transfer function of the general feedback structure controller also leads to a PID controller. By the way the PID controller constants can be found. So, The PID controller constant can be written as the followings:

$$\frac{1}{T_D} = 4.255319149 \Rightarrow T_D = 0.235 \qquad (3.3.2.15)$$

$$\frac{1}{T_i T_D} = 58.51812001 \Rightarrow T_i = 0.0772717974 \qquad (3.3.2.16)$$

$$\frac{K_C}{T_D} = \frac{1}{10.0499646(\lambda + 0.5)} \Rightarrow K_C(\lambda + 0.5) = 0.023383166 \qquad (3.3.2.17)$$

46

For the flapper control system, the filter constant λ can be chosen the same value as dead time. The following simulation was made when λ is 0.5.
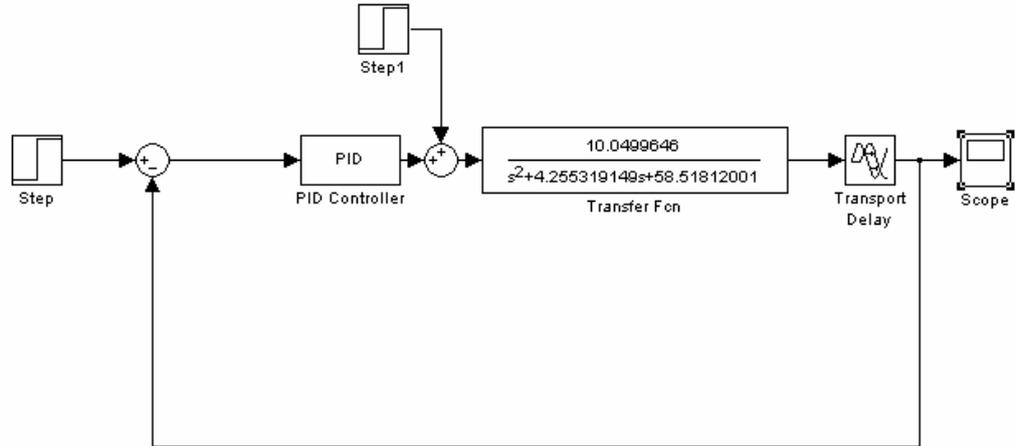
Figure 3.3.2.1: The Simulation Model of IMC Based PID for the System in 35 Hz

The result of simulation:
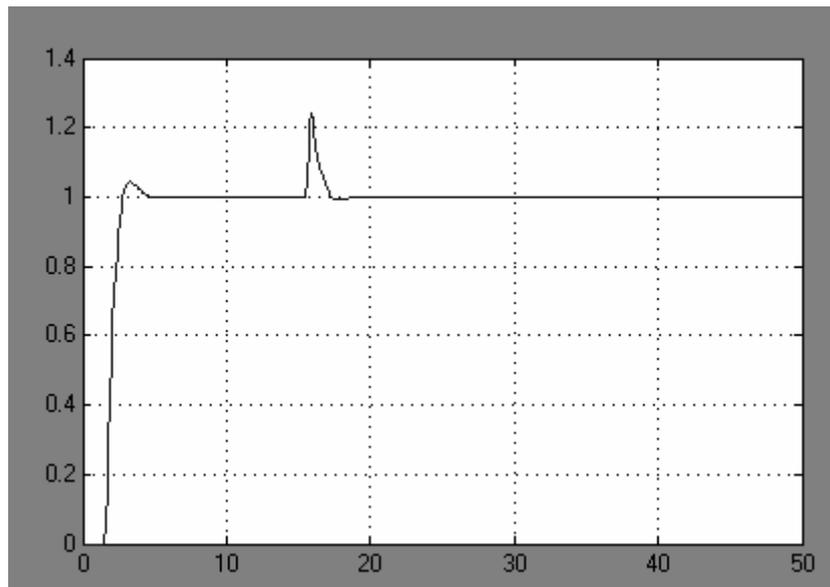
Figure 3.3.2.2: The Response of the Simulation Model of IMC Based PID for the System in 35 Hz

3.3.3 Designing of IMC Based PID Controller in 25 Hz Frequency for the Flapper Position Control System

For the flapper control system, the transfer function of the system and the model of the system can be accepted equal. This situation makes easier to find constants of PID controller.

The model of the system in 25 Hz frequency has been already found as:

$$\tilde{P}(s) = \frac{11.01148825}{s^2 + 3.619909492s + 91.00403517} e^{-0.5s} \qquad (3.3.3.1)$$

The dead time in the transfer function can be written in the Taylor series such as the following equation:

$$e^{-hs} = (1 - hs) \qquad (3.3.3.2)$$

$$\tilde{P}(s) = \frac{11.01148825}{s^2 + 3.619909492s + 91.00403517} e^{-0.5s} = \frac{11.01148825(1 - 0.5s)}{s^2 + 3.619909492s + 91.00403517}$$

$$(3.3.3.3)$$

By the way, $e^{-0.5s}$ is eliminated.

The model of the system and the transfer function of the system are accepted perfectly matched. So, the transfer function of the system can be written as:

$$\tilde{P}(s) = P(s) = \frac{11.01148825}{s^2 + 3.619909492s + 91.00403517} e^{-0.5s} = \frac{11.01148825(1 - 0.5s)}{s^2 + 3.619909492s + 91.00403517}$$

$$(3.3.3.4)$$

The IMC controller (q(s)) must be calculated to find the general feedback structure controller (C(s)). q(s) can be found as in the following equations:

$$\tilde{q}(s) = \tilde{P}_-^{-1}(s) \qquad (3.3.3.5)$$

$$\tilde{P}_- = \frac{11.01148825}{s^2 + 3.619909492s + 91.00403517} \qquad (3.3.3.6)$$

$$\tilde{q}(s) = \frac{s^2 + 3.619909492s + 91.0403517}{11.01148825} \qquad (3.3.3.7)$$

All the transfer functions for creating the general feedback structure controller is obtained now. So, the general feedback structure controller can for designing the IMC controller (q(s)), a low pass filter must be designed. For the flapper control system, the filter can be designed as below:

$$f(s) = \frac{1}{(\lambda s + 1)^n} \qquad (3.3.3.8)$$

The low pass filter can be chosen as a first order filter.

The IMC Controller, q(s), can be written as:

$$q(s) = \tilde{q}(s)f(s) = \frac{s^2 + 3.619909492s + 91.0403517}{11.01148825} \cdot \frac{1}{\lambda s + 1} \qquad (3.3.3.9)$$

be calculated as in the following equations:

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} \qquad (3.3.3.10)$$

$$C(s) = \frac{\dfrac{s^2 + 3.619909492s + 91.0403517}{11.01148825} \cdot \dfrac{1}{\lambda s + 1}}{1 - \dfrac{11.01148825(1 - 0.5s)}{s^2 + 3.619909492s + 91.0403517} \cdot \dfrac{s^2 + 3.619909492s + 91.0403517}{11.01148825(\lambda s + 1)}} \qquad (3.3.3.11)$$

This equation simplifies to:

$$C(s) = \frac{s^2 + 3.619909492s + 91.0403517}{11.01148825(\lambda + 0.5)s} \qquad (3.3.3.12)$$

The general transfer function of a PID controller can be written such as in the following:

$$K_C \left(\frac{s^2 + \dfrac{1}{T_D}s + \dfrac{1}{T_i T_D}}{T_D s} \right) \qquad (3.3.3.13)$$

The general feedback structure controller transfer function of the system has already been found as:

$$C(s) = \frac{s^2 + 3.619909492s + 91.0403517}{11.01148825(\lambda + 0.5)s} \qquad (3.3.3.14)$$

Comparing with the transfer function of a PID controller, it is obvious that the transfer function of the general feedback structure controller also leads to a PID controller. By the way the PID controller constants can be found. So, The PID controller constant can be written as the followings:

$$\frac{1}{T_D} = 3.619909492 \Rightarrow T_D = 0.27625 \qquad (3.3.3.15)$$

$$\frac{1}{T_i T_D} = 91.00403517 \Rightarrow T_i = 0.039777461 \qquad (3.3.3.16)$$

$$\frac{K_C}{T_D} = \frac{1}{11.01148825(\lambda + 0.5)} \Rightarrow K_C(\lambda + 0.5) = 0.025087435 \qquad (3.3.3.17)$$

For the flapper control system, the filter constant λ can be chosen the same value as dead time. The following simulation was made when λ is 0.5.
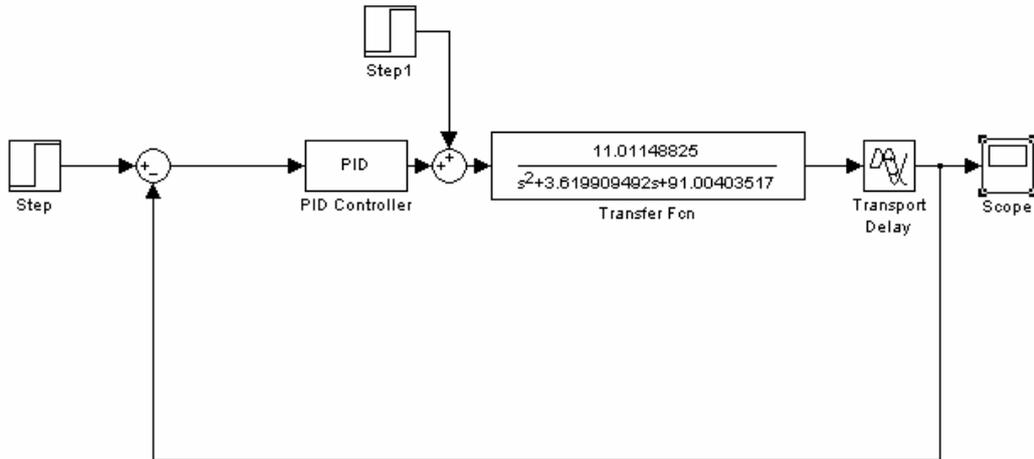
Figure 3.3.3.1: The Simulation Model of IMC Based PID for the System in 25 Hz

The result of the simulation is
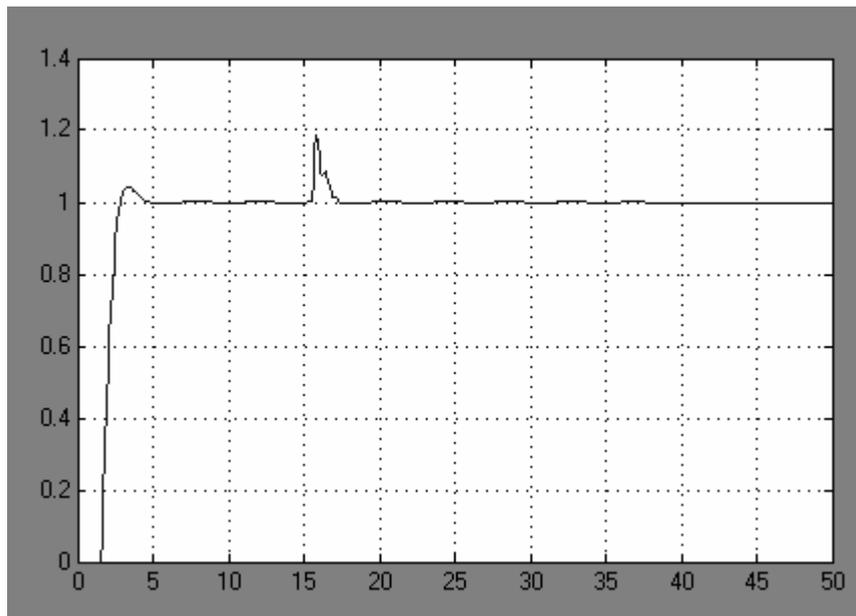
Figure 3.3.3.2: The Response of the Simulation Model of IMC Based PID for the System in 25 Hz

3.3.4 Designing of IMC Based PID Controller in 15 Hz Frequency for the Flapper Position Control System

For the flapper control system, the transfer function of the system and the model of the system can be accepted equal. This situation makes easier to find constants of PID controller.

The model of the system in 15 Hz frequency has been already found as:

$$\tilde{P}(s) = \frac{4.063246906}{s^2 + 5.03144654s + 45.14718784} e^{-0.5s} \qquad (3.3.4.1)$$

The dead time in the transfer function can be written in the Taylor series such as the following equation:

$$e^{-hs} = (1 - hs) \qquad (3.3.4.2)$$

$$\tilde{P}(s) = \frac{4.063246906}{s^2 + 5.03144654s + 45.14718784} e^{-0.5s} = \frac{4.063246906(1 - 0.5s)}{s^2 + 5.03144654s + 45.14718784}$$

$$(3.3.4.3)$$

By the way, $e^{-0.5s}$ is eliminated.

The model of the system and the transfer function of the system are accepted perfectly matched. So, the transfer function of the system can be written as:

$$\tilde{P}(s) = P(s) = \frac{4.063246906}{s^2 + 5.03144654s + 45.14718784} e^{-0.5s} = \frac{4.063246906(1 - 0.5s)}{s^2 + 5.03144654s + 45.14718784}$$

$$(3.3.4.4)$$

The IMC controller (q(s)) must be calculated to find the general feedback structure controller (C(s)). q(s) can be found as in the following equations:

$$\tilde{q}(s) = \tilde{P}_-^{-1}(s) \qquad (3.3.4.5)$$

$$\tilde{P}_- = \frac{4.063246906}{s^2 + 5.03144654s + 45.14718784} \qquad (3.3.4.6)$$

$$\tilde{q}(s) = \frac{s^2 + 5.03144654s + 45.14718784}{4.063246906} \qquad (3.3.4.7)$$

All the transfer functions for creating the general feedback structure controller is obtained now. So, the general feedback structure controller can for designing the IMC controller (q(s)), a low pass filter must be designed. For the flapper control system, the filter can be designed as below:

$$f(s) = \frac{1}{(\lambda s + 1)^n} \qquad (3.3.4.8)$$

The low pass filter can be chosen as a first order filter.

The IMC Controller, q(s), can be written as:

$$q(s) = \tilde{q}(s)f(s) = \frac{s^2 + 5.03144654s + 45.14718784}{4.063246906} \bullet \frac{1}{\lambda s + 1} \qquad (3.3.4.9)$$

be calculated as in the following equations:

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} \qquad (3.3.4.10)$$

$$C(s) = \frac{\dfrac{s^2 + 5.03144654s + 45.14718784}{4.063246906} \bullet \dfrac{1}{\lambda s + 1}}{1 - \dfrac{4.063246906(1 - 0.5s)}{s^2 + 5.03144654s + 45.14718784} \bullet \dfrac{s^2 + 5.03144654s + 45.14718784}{4.063246906(\lambda s + 1)}}$$

$$(3.3.4.11)$$

This equation simplifies to:

$$C(s) = \frac{s^2 + 5.03144654s + 45.14718784}{4.063246906(\lambda + 0.5)s} \qquad (3.3.4.12)$$

The general transfer function of a PID controller can be written such as in the following:

$$K_C \left(\frac{s^2 + \dfrac{1}{T_D}s + \dfrac{1}{T_i T_D}}{T_D s} \right) \qquad (3.3.4.13)$$

The general feedback structure controller transfer function of the system has already been found as:

$$C(s) = \frac{s^2 + 5.03144654s + 45.14718784}{4.063246906(\lambda + 0.5)s} \qquad (3.3.4.14)$$

Comparing with the transfer function of a PID controller, it is obvious that the transfer function of the general feedback structure controller also leads to a PID controller. By the way the PID controller constants can be found. So, The PID controller constant can be written as the followings:

$$\frac{1}{T_D} = 5.03144654 \Rightarrow T_D = 0.19875 \qquad (3.3.4.15)$$

$$\frac{1}{T_i T_D} = 45.14718784 \Rightarrow T_i = 0.111445402 \qquad (3.3.4.16)$$

$$\frac{K_C}{T_D} = \frac{1}{4.063246906(\lambda + 0.5)} \Rightarrow K_C(\lambda + 0.5) = 0.048914083 \qquad (3.3.4.17)$$

For the flapper control system, the filter constant λ can be chosen the same value as dead time. The following simulation was made when λ is 0.5.
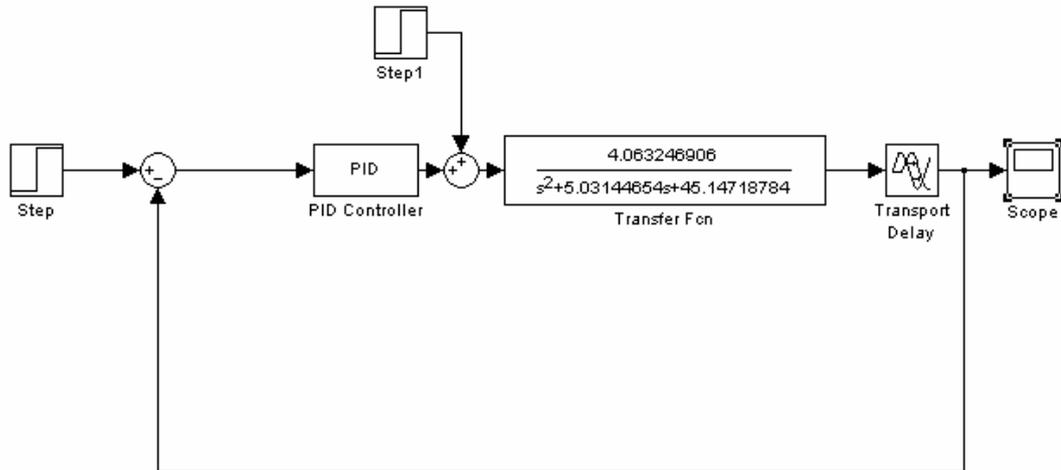
Figure 3.3.4.1: The Simulation Model of IMC Based PID for the System in 15 Hz

The result of the simulation can be shown as below:
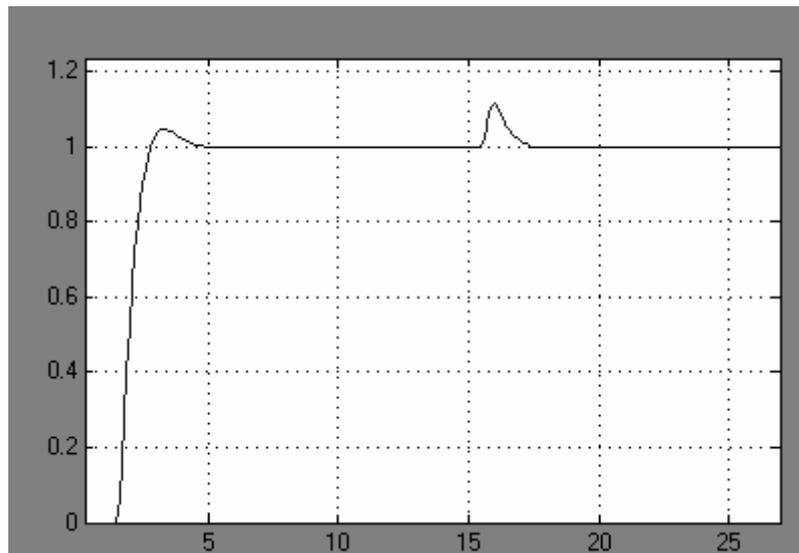
Figure 3.3.4.2: The Response of the Simulation Model of IMC Based PID for the System in 15 Hz

3.3.5 Designing of IMC Based PID Controller in 10 Hz Frequency for the Flapper Position Control System

For the flapper control system, the transfer function of the system and the model of the system can be accepted equal. This situation makes easier to find constants of PID controller.

The model of the system in 10 Hz frequency has been already found as:

$$\tilde{P}(s) = \frac{0.060625}{0.9s + 1} e^{-0.5s} \qquad (3.3.5.1)$$

The dead time in the transfer function can be written in the Taylor series such as the following equation:

$$e^{-hs} = (1 - hs) \qquad (3.3.5.2)$$

$$\tilde{P}(s) = \frac{0.060625}{0.9s + 1} e^{-0.5s} = \frac{0.060625(1 - 0.5s)}{0.9s + 1} \qquad (3.3.5.3)$$

By the way, $e^{-0.5s}$ is eliminated.

The model of the system and the transfer function of the system are accepted perfectly matched. So, the transfer function of the system can be written as:

$$\tilde{P}(s) = P(s) = \frac{0.060625}{0.9s + 1} e^{-0.5s} = \frac{0.060625(1 - 0.5s)}{0.9s + 1} \qquad (3.3.5.4)$$

The IMC controller (q(s)) must be calculated to find the general feedback structure controller (C(s)). q(s) can be found as in the following equations:

$$\tilde{q}(s) = \tilde{P}_-^{-1}(s) \qquad (3.3.5.5)$$

$$\tilde{P}_- = \frac{0.060625}{0.9s + 1} \qquad (3.3.5.6)$$

$$\tilde{q}(s) = \frac{0.9s + 1}{0.060625} \qquad (3.3.5.7)$$

All the transfer functions for creating the general feedback structure controller is obtained now. So, the general feedback structure controller can for designing the IMC controller (q(s)), a low pass filter must be designed. For the flapper control system, the filter can be designed as below:

$$f(s) = \frac{1}{(\lambda s + 1)^n} \qquad (3.3.5.8)$$

The low pass filter can be chosen as a first order filter.

The IMC Controller, q(s), can be written as:

$$q(s) = \tilde{q}(s)f(s) = \frac{0.9s + 1}{0.060625} \bullet \frac{1}{\lambda s + 1} \qquad (3.3.5.9)$$

be calculated as in the following equations:

$$C(s) = \frac{q(s)}{1 - \tilde{P}(s)q(s)} \qquad (3.3.5.10)$$

$$C(s) = \frac{\dfrac{0.9s+1}{0.060625} \bullet \dfrac{1}{\lambda s+1}}{1 - \dfrac{0.060625(1-0.5s)}{0.9s+1} \bullet \dfrac{0.9s+1}{0.060625(\lambda s+1)}} \qquad (3.3.5.11)$$

This equation simplifies to:

$$C(s) = \frac{0.9s+1}{0.060625(\lambda+0.5)s} \qquad (3.3.5.12)$$

This controller can be written as a PI controller. The general transfer function of a PI controller can be written such as in the following:

$$K_C\left(1 + \frac{1}{T_i s}\right) = \left(\frac{T_i s + 1}{T_i s}\right) \qquad (3.3.5.13)$$

The general feedback structure controller transfer function of the system has already been found as:

$$C(s) = \frac{0.9s+1}{0.060625(\lambda+0.5)s} \qquad (3.3.5.14)$$

Comparing with the transfer function of a PI controller, it is obvious that the transfer function of the general feedback structure controller also leads to a PI controller. By the way the PI controller constants can be found. So, The PI controller constant can be written as the followings:

$$T_i = 0.9 \qquad (3.3.5.15)$$

$$\frac{K_C}{T_i} = \frac{1}{0.060625(\lambda+0.5)} \Rightarrow K_C(\lambda+0.5) = 14.84536082 \qquad (3.3.5.16)$$

For the flapper control system, the filter constant λ can be chosen the same value as dead time. The following simulation was made when λ is 0.5.
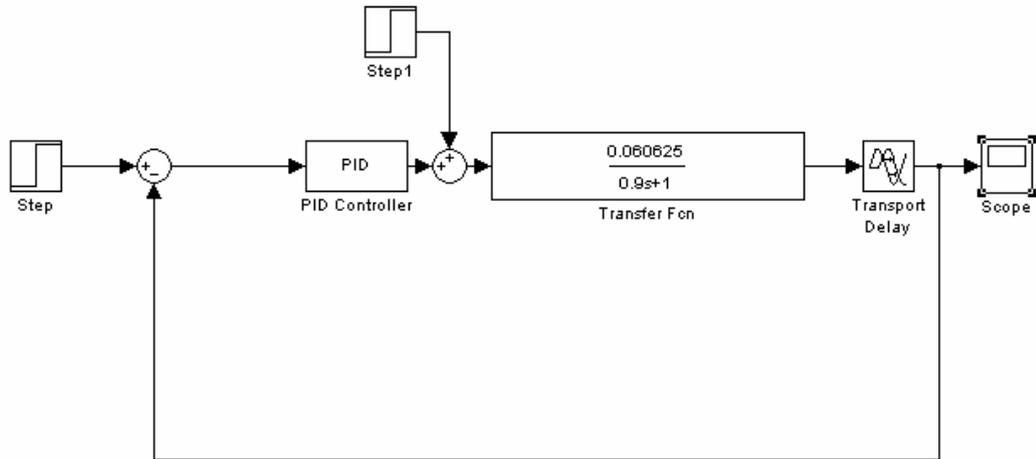
Figure 3.3.5.1: The Simulation Model of IMC Based PID for the System in 10 Hz

The result of simulation can be shown as below:
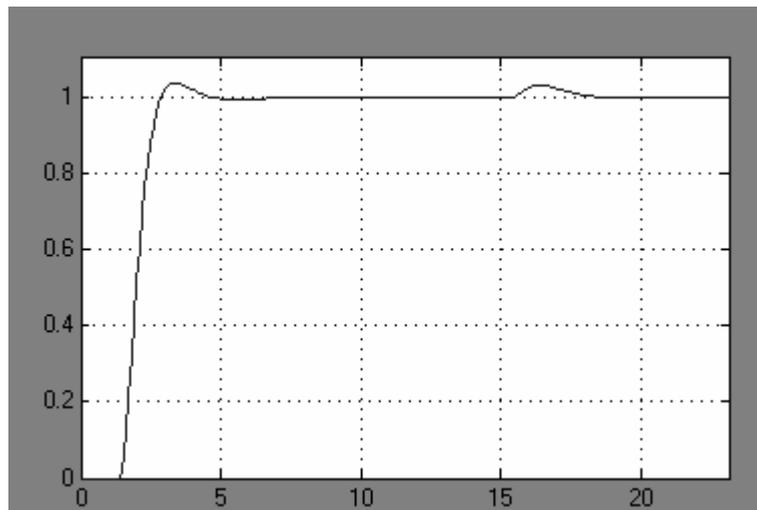
Figure 3.3.5.2: The Response of the Simulation Model of IMC Based PID for the System in 10 Hz

3.4 λ Tuning for Calculating the PID Constants on the Flapper Position Control System

As it is seen from the PID constants that are found at the above calculations, the system gain depends on the filter constant. Thus, if the filter constant changes, the gain of the system will change.

The filter order can be selected large enough to make q proper. λ, which is an flexible parameter, determines the speed of response. If λ increases, the speed response of the control system becomes slower. On the other hand, if λ decreases,

the speed response of the control systems becomes faster. More over, λ affects the robustness of the control system. The higher values of the λ, the higher robustness are got for the system. (Rivera, 1999)

λ tuning can be applied to the gain of the system. When the error signal is high, the filter constant will be low. So, the response of the system will be fast. But then, for avoiding overshoot, the filter constant will be high enough may be higher than 1 or 2, which depends the systems' dynamic. Then, to for making the system stable, when the error signal is zero, the filter constant will be not so low but not so high.

This kind of a λ tuning can be applied to the flapper position control system. For applying it, the PID transfer function should be transformed to the discrete z domain. The PID constant needn't to be changed because of being designed in the continuous time. Only the transfer function of the system will be changed. The proportional is same in the discrete time changed. The derivative function of the PID in the discrete time is $\dfrac{z-1}{z}$. And the integral function of the PID of the PID in the discrete time is $\dfrac{z+1}{z-1}$. And the sampling time can be chosen as 50 ms.

The transfer function of the system was chosen as the transfer function of the system in 25 Hz, because the highest reference value that can be applied to the system is 18000 as an integer value data. In this reference, the system behaves as in the 25 Hz frequency. Thus, using the transfer function of the system in the 25 Hz results best.

The simulation of the system can be done as it is shown below:
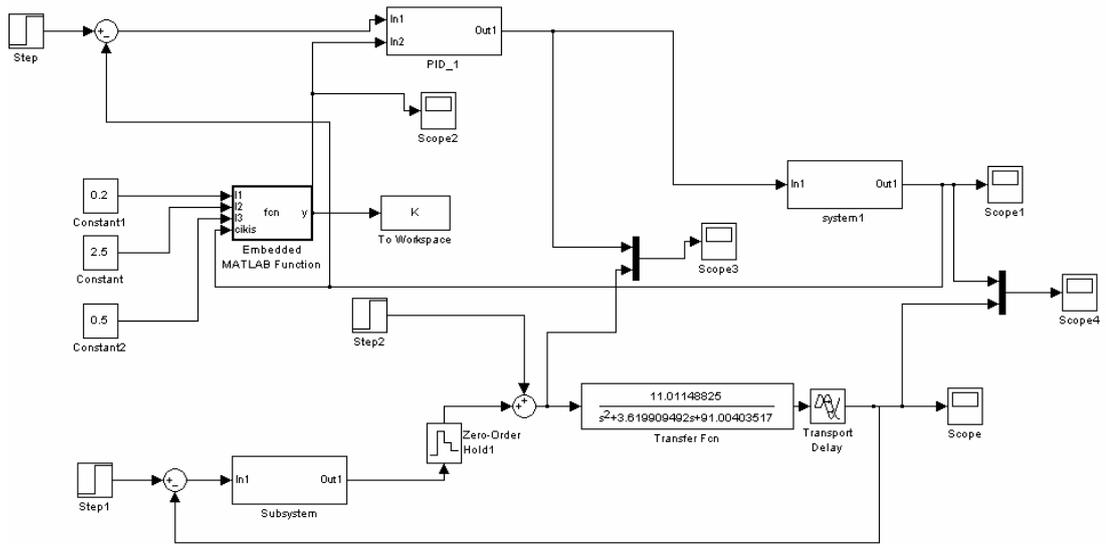
Figure 3.4.1: The Simulation of the System for λ Tuning

In the simulation, the filter constant will be 0.2 during the 20% of the response of the system which means that the error signal will be high. Then, the filter constant will be 2.5 for avoiding the high overshoot. Finally, the filter constant will be 0.5 when the response of the system reaches to 1.

When these rules were applied to the system the results at the Figure 3.4.2, 3.4.3 and 3.4.4 were obtained respectively.
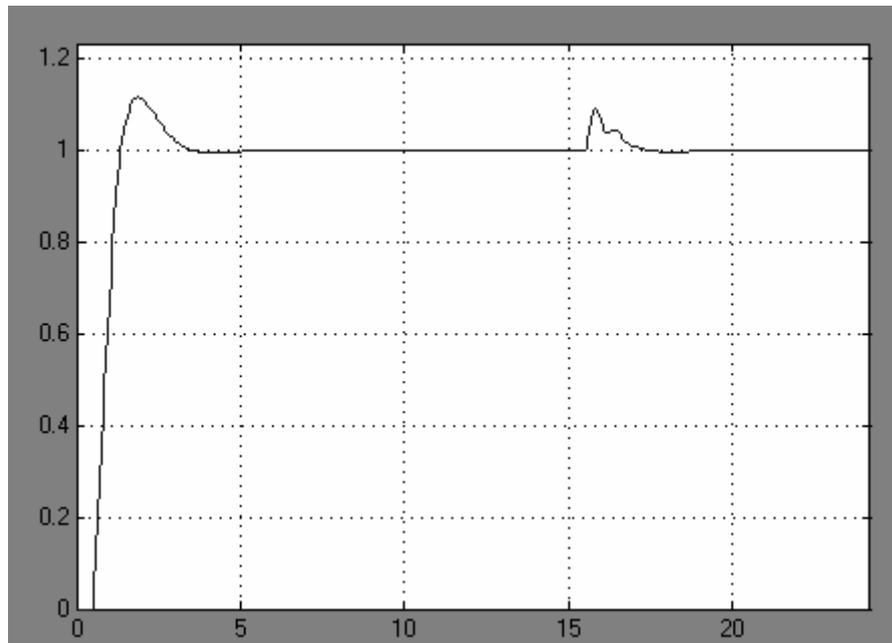
Figure 3.4.2: The Response of the System with Using λ Tuning
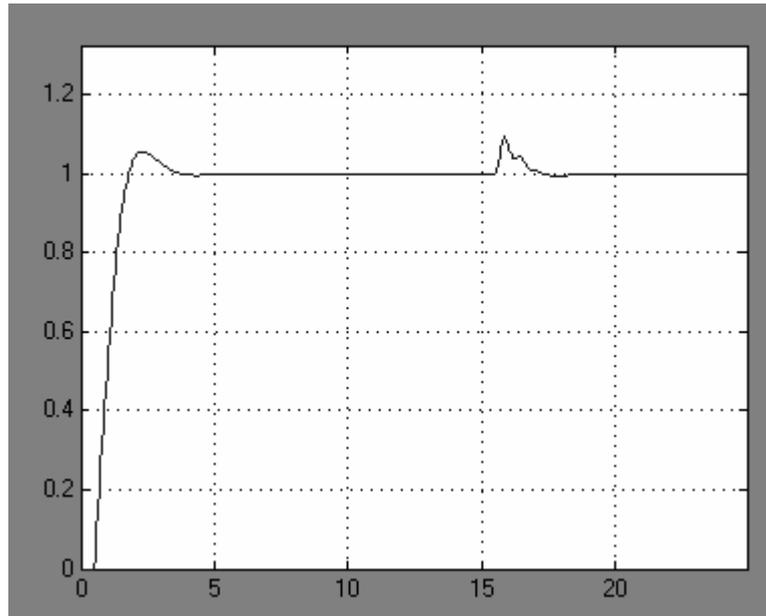
Figure 3.4.3: The Response of the System without Using λ Tuning
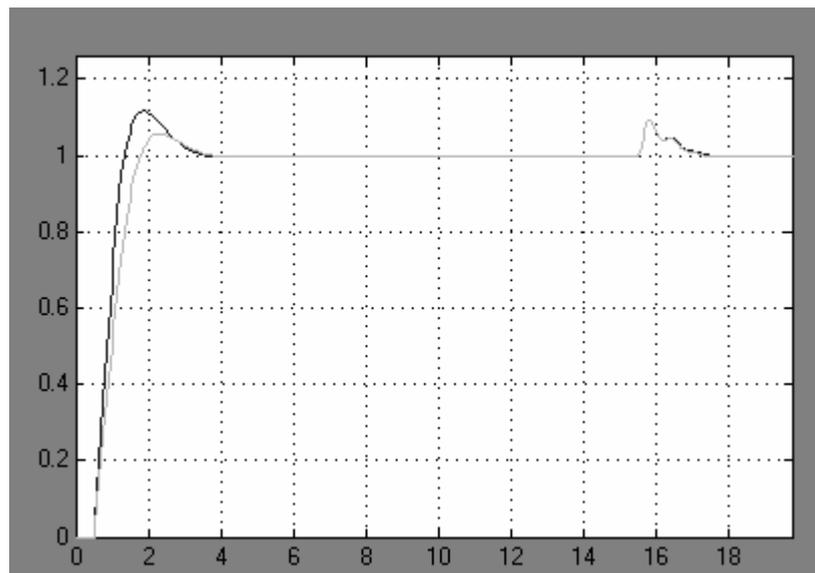
Figure 3.4.4: The Responses of the System

The lighter line represents the response of the system when tuning is applied. It is obvious that, the system response is better when the tuning algorithm is applied to the system to find the gain of the system.

3.5 λ Tuning for Calculating the PID Constants on the Flapper Position Control System by Using Fuzzy Algorithm

The λ tuning can be applied by using the fuzzy algorithm to find the gain of the system. By this way, the PID constants will be set by the fuzzy algorithm. Before applying the fuzzy algorithm to the system, the membership functions must be defined. The membership functions can be changed according to the response of the system. But the logic for fixing the PID constants will not change. Only filter constant will be fixed according to the error signal. When the error signal is high, the filter constant will be low. So, the response of the system will be fast. But then, for avoiding overshoot, the filter constant will be high enough may be higher than 1 or 2, which depends the systems' dynamic. Then, to for making the system stable, when the error signal is zero, the filter constant will be not so low but not so high.

The membership function of the fuzzy algorithm can be defined as the following for the flapper position control system:
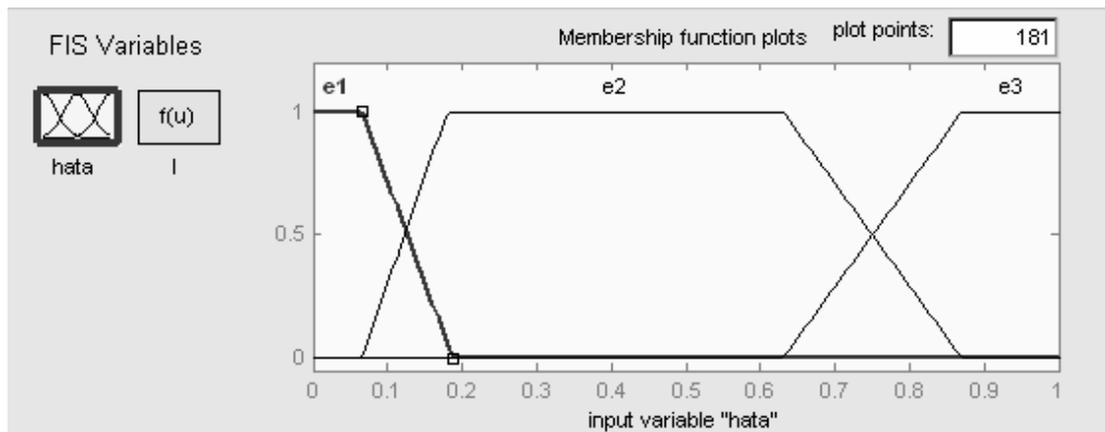
Figure 3.5.1: Membership Functions

The rules of the fuzzy algorithm were created as the following:
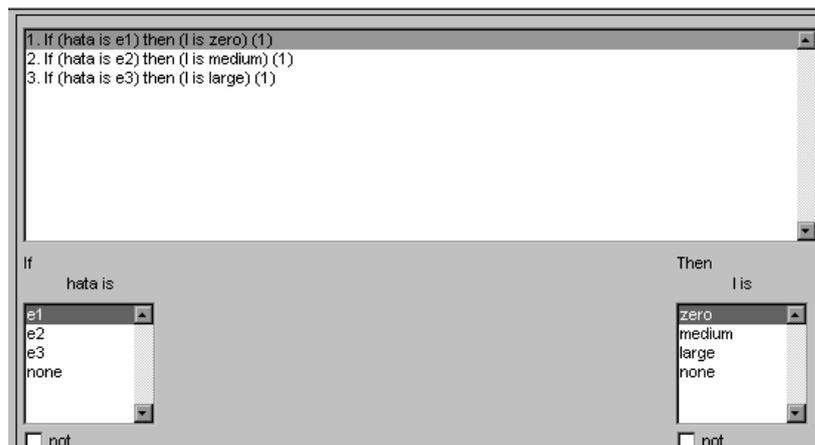
Figure 3.5.2: Fuzzy Algorithm Rules

The output function was created as the following:
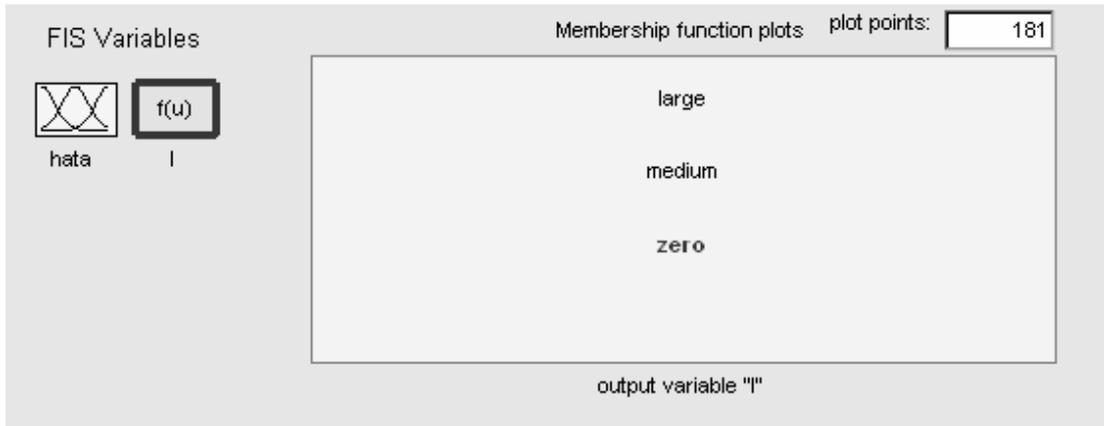
Figure 3.5.3: Output Functions of the Fuzzy Algorithm

In the output function zero equals to 0.5, medium equals to 2.5 and large equals to 0.5. So, when the absolute value of error signal is high, the filter constant will be 0.2. For avoiding the overshoot, the filter constant will be 2.5. Finally, the filter constant will be 0.5 when the response of the system reaches to 1.

The simulation of the system can be done as it is shown below:
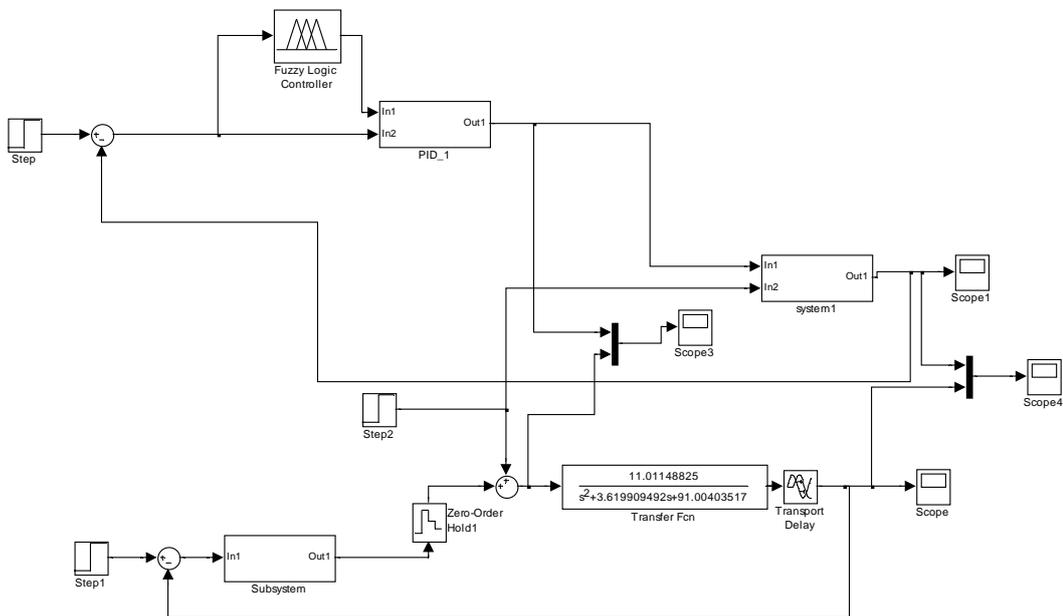
Figure 3.5.4: Simulation Model of Fuzzy Algorithm

The responses of the system with and without using fuzzy algorithm are given in Figure 3.5.5 and 3.5.6 respectively.
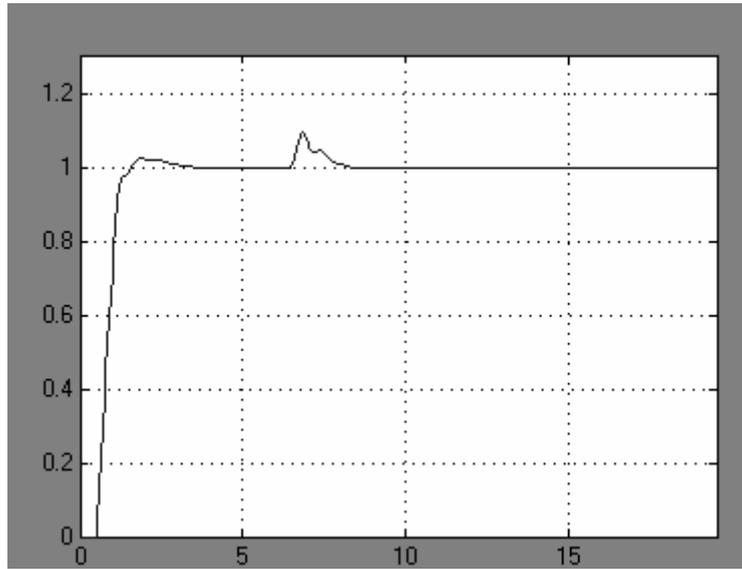
Figure 3.5.5: The Response of the Simulation Model of Fuzzy Algorithm
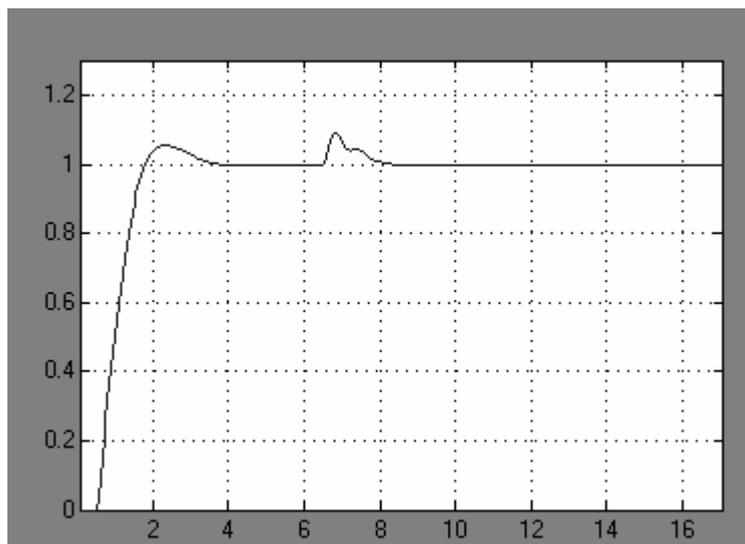
Figure 3.5.6: The Response of the Simulation Model without Fuzzy Algorithm

The responses, which are drawn together, are given in Figure 3.5.7.
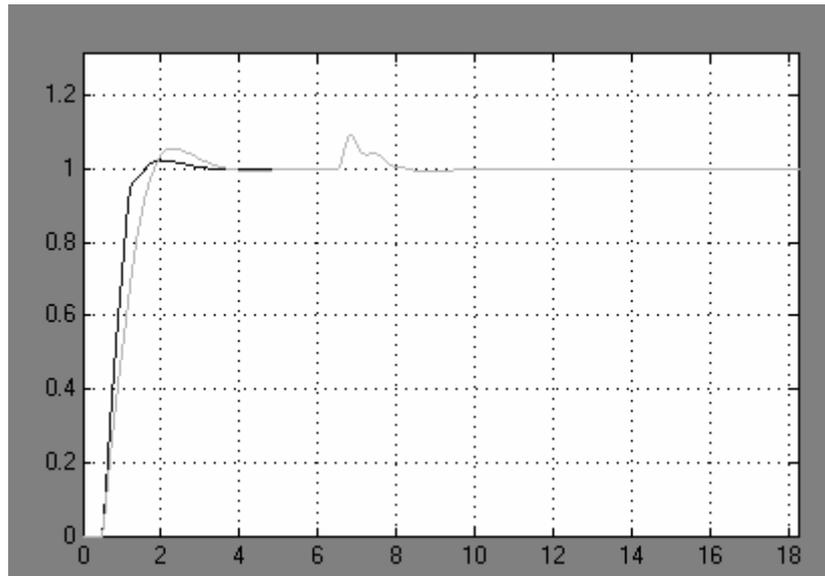
Figure 3.5.7: The Responses of the System

The lighter line represents the response of the system when fuzzy algorithm is applied. It is obvious that, the system response is better when the fuzzy algorithm is applied to the system to find the gain of the system.

3.6 Application of IMC Based PID Controller no the Flapper Position Control System with Siemens S7 300 PLC

All the calculations, which are made to find the PID constants at the above, were applied to the system by using the Siemens S7 300 PLC. All the responses, which are obtained for the references from 15000 to 18000, are drawn at the below figures. The PLC code of the controller will be given at the 'Appendix A'. The sampling time can be chosen 50 ms.

The system response for the reference signal of 15000, 16000, 17000 and 18000 are given in Figure 3.6.1, Figure 3.6.2, Figure 3.6.3 and 3.6.4 respectively.
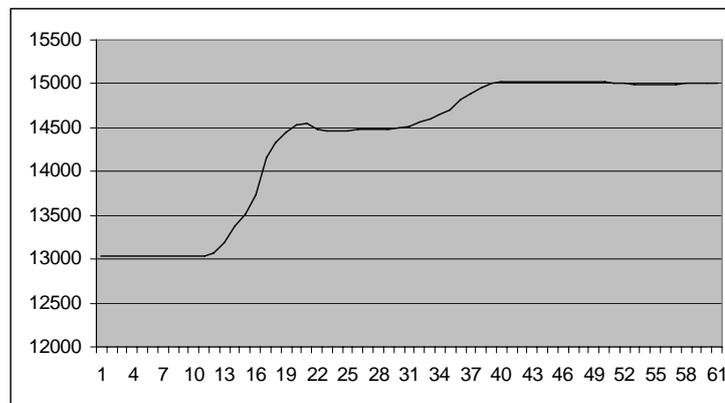
Figure 3.6.1: The system response for the reference signal of 15000
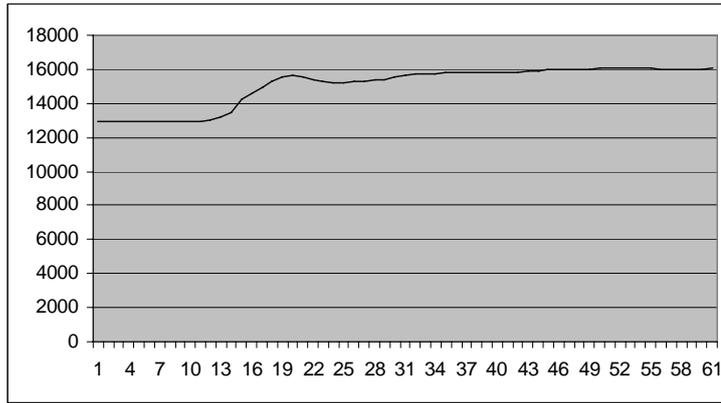
Figure 3.6.2: The system response for the reference signal of 16000
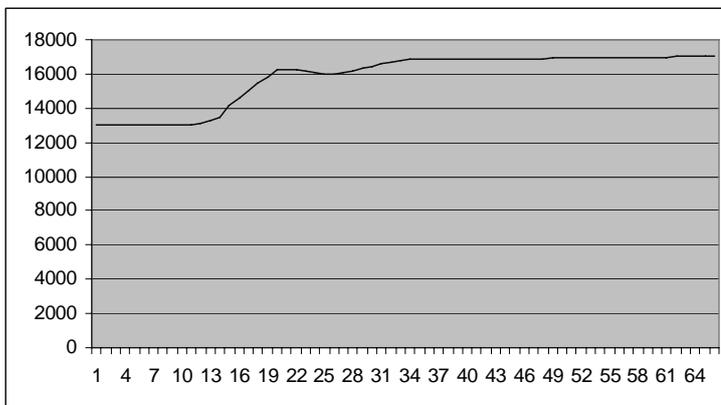
Figure 3.6.3: The system response for the reference signal of 17000
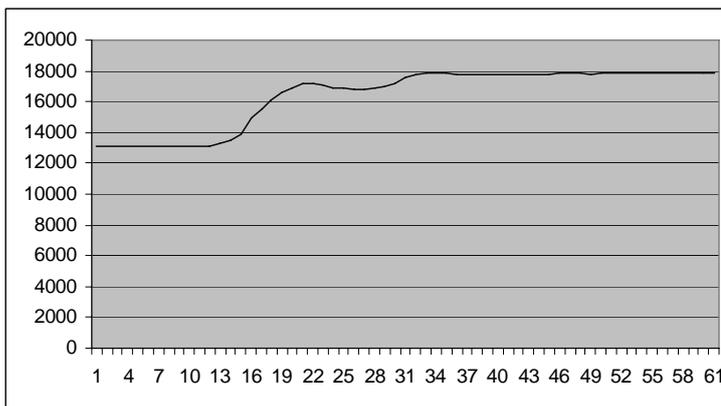
Figure 3.6.4: The system response for the reference signal of 18000

According to the responses, IMC based PID results good in the flapper position control system. Furthermore, the system responses are better for higher frequencies, because the model of the system matches to the transfer function of the system in the higher frequencies.

3.7 Application of λ **Tuning for Calculating the PID Constants on the Flapper Position Control System by Using Fuzzy Algorithm with Siemens S7 300 PLC**

All the calculations of the previous section, which are made to find the PID constants, were applied to the system by using the Siemens S7 300 PLC. All the responses, which are obtained for the references from 15000 to 18000, are drawn at the below figures. The PLC code of the controller will be given at the 'Appendix B'. The sampling time can be chosen 50 ms.

The aim, creating the fuzzy λ tuning algorithm theoretically, is to accelerate the output signal while the error signal is high. The rules, created for the fuzzy λ tuning algorithm, resulted well at the simulations. On the other hand, when the rules applied to the system, it was observed that the robustness of the system went bad. Besides, the system made oscillatory when a disturbance occurred.

In the output function zero equals to 1, medium equals to 0.2 and large equals to3. So, when the absolute value of error signal is high, the filter constant will be1. Then, the filter constant will be 0.2 when the system response is close to 1. And finally, when it reaches the 1 the filter constant will be 3.

Because of the nonlinearities, that could not be inserted to the models used in simulation, the results obtained in simulation studies could not be obtained in real time application.

The system response for the reference signal of 15000, 16000, 17000 and 18000 are given in Figure 3.7.1, Figure 3.7.2, Figure 3.7.3 and 3.7.4 respectively.
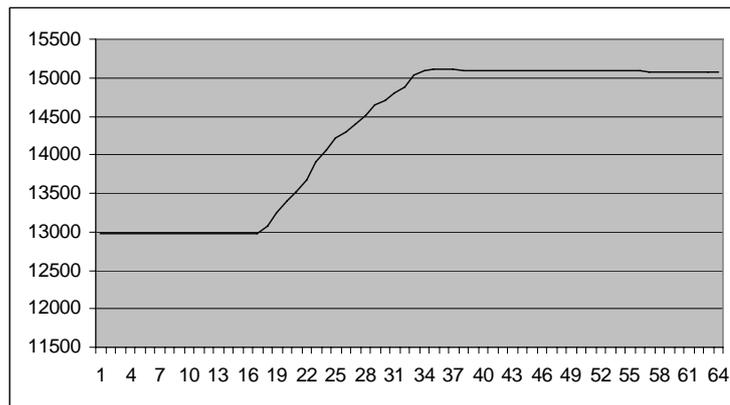
Figure 3.7.1: The system response for the reference signal of 15000
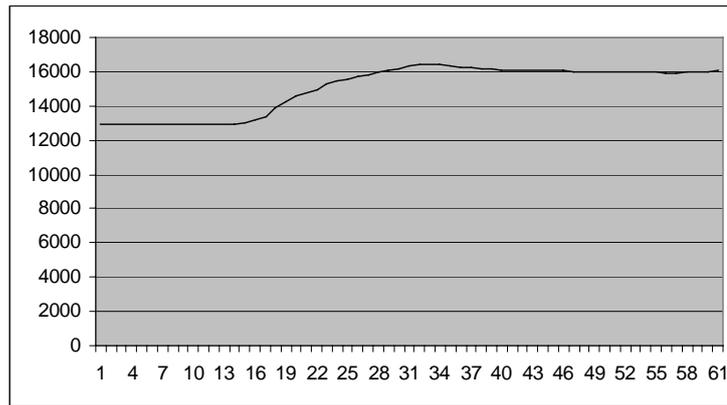
Figure 3.7.2: The system response for the reference signal of 16000
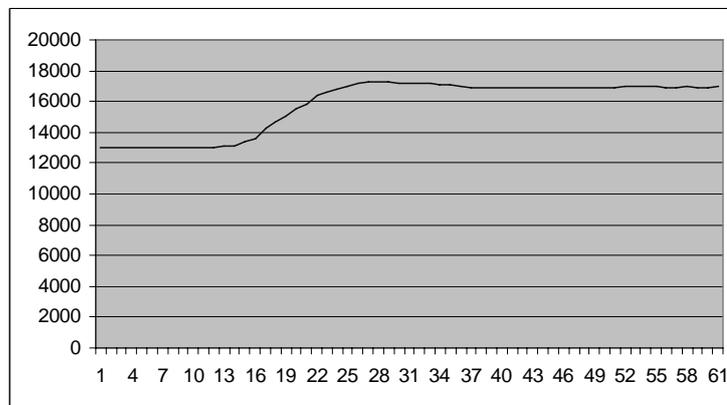
Figure 3.7.3: The system response for the reference signal of 17000
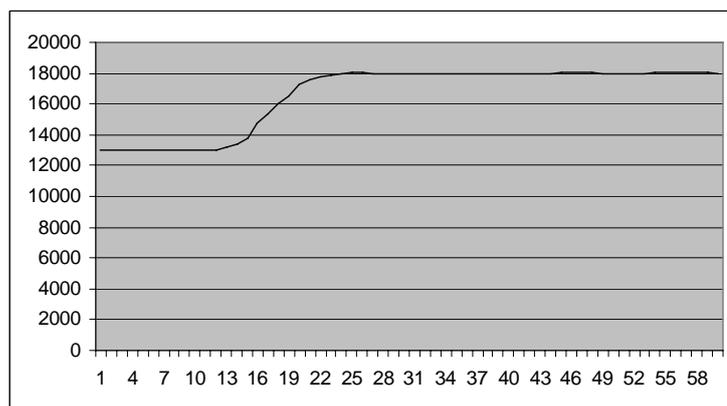
Figure 3.7.4: The system response for the reference signal of 18000

4. CONCLUSION

There are some advantageous of IMC based PID controller such as the followings:

- The IMC based PID controller depends on the system model. The PID constants change according to the system model. By the way, calculating only one parameter is enough for creating the IMC based PID controller. But for the classical PID controllers, three controller parameters should be found when creating the controller.

- The IMC based PID control is sensitive for the change of models by tuning the filter constant (λ) of the controller. For the higher values of the λ, the system will be more robust. Besides, for the lower values of λ, the system response will be faster.

Tuning of the filter constant with a fuzzy algorithm has also some benefits. Tuning λ with a fuzzy algorithm can make system response faster. Tuning λ with a fuzzy algorithm can also make system more robust. Moreover, the filter constant can be changed anytime according to the system response. So the overshoot and disturbance can be eliminated.

When comparing the results with the IMC based PID control, the tuning with a fuzzy algorithm' results are really good. By using the fuzzy algorithm, the settling time and the overshoot of the system are shortened. Moreover, the system is more robust when the fuzzy algorithm is applied.

The aim was applying the fuzzy algorithm to the real system was to make the system faster and more robust. Thus, the filter constant was chosen small enough to make the system faster. Then, the filter constant was chosen big enough to avoid the overshoot. Finally, the filter constant was chosen the same as the system's dead time, which was also used as filter constant in the IMC based PID control, to make the system stable. This algorithm was successful in the simulation.

Applying to the system the fuzzy algorithm that was used to the flapper control system with PLC, didn't result good. Then, another algorithm was applied to the system. Firstly, the filter constant was 1, and then the filter constant was chosen 0.2. Finally, it is chosen 3. This algorithm was successful when it was applied to the system. On the other hand, the results in the simulation were not good.

The flapper position control is a nonlinear system. The non linearity of the system makes hard apply fuzzy algorithm to the system. So, the fuzzy algorithm that was created was unsuccessful even it resulted well in the simulations.

REFERENCES

Äström, K.J. and Hägglund, T., 1995. PID Controllers**,** Instrument Society of America, USA

Bequette, B.W., 1994. Chapter 7: The IMC – Based PID Procedure, *http://citeseer.ist.psu.edu/430790.html* , USA

Górecki, J.E.M and Korytowski K.W.A., 1992. Time Delay Systems – Stability and Performance Criteria with Applications, Ellis Horwood Limited: Redwood Press, Great Britain

Kaya, İ., 2004. IMC Based Automatic Tuning Method for PID Controllers in a Smith Predictor Configuration, *Elsevier*, 281-290

Levine, W.S., 2000. Control System Fundamentals, CRC Press, Danvers, USA

Rivera, D.E., 1999. Internal Model Control: A Comprehensive View, Arizona State University, Tempe, Arizona.

Yamanaka, K. and Abe, N., 2003. Smith Predictor Control and Internal Model Control, *SICE Annual Conference in Fukui,* Fukui University, Japan, August 4-6

Zhang, L., Liu, J., Sun, J., Jin, X., 2004. Analysis and Design of Adaptive Internal Model Control for Unstable Process with Dead Time, *The Third International Conference on Machine Learning and Cybernetics, Shanghai,* China, August 26-29

APPENDIX A

```
FB1 : Title:
```

Comment:

Network 1: Hatanin hesaplanmasi

Comment:

```
        L       #Ref
        L       #Y
        -I
        T       #hata
        ITD
        DTR
        T       #e
```

Network 2: P nin hesaplanmasi

Comment:

```
        L       #K
        L       #Td
        /R
        L       #Td
        /R
        L       #e
        *R
        T       #Yp
```

Network 3: Turevin hesaplanmasi (K/Ti*Ts)*(e(k)-e(k-1))=Yd

Comment:

```
        L       #K
        L       #Td
        /R
        L       #Ts
        /R
        T       #temp_d
        L       #e
        L       #e_1
        -R
        L       #temp_d
        *R
        T       #Yd
```

Network 4 : Turev icin sinirlandirma   Dmax

Comment:

```
        L     #Yd
        L     #Umax
        >=R
        JNB   _005
        L     #Umax
        T     #Yd
_005: NOP   0
```

Network 5 : Turev icin sinirlandirma   Dmin

Comment:

```
        L     #Yd
        L     #Umin
        <=R
        JNB   _006
        L     #Umin
        T     #Yd
_006: NOP   0
```

Network 6 : Integralin hesaplanmasi

Comment:

```
        L     #K
        L     #Ts
        *R
        L     #x
        *R
        L     5.000000e-001
        *R
        L     #Ti
        /R
        L     #Td
        /R
        L     #Td
        /R

        T     #temp_i
        L     #e
        L     #e_1
        +R
        L     #temp_i
        *R
        L     #Yi_1
        +R
        T     #Yi
```

Network 7 : Integral icin sinirlandirma   Imax

Comment:

```
        L     #Yi
        L     #Umax_sinir
        >=R
        JNB   _003
        L     #Umax_sinir
        T     #Yi
  _003: NOP   0
```

Network 8 : Integral icin sinirlandirma

Comment:

```
        L     #Yi
        L     #Umin_sinir
        <=R
        JNB   _004
        L     #Umin_sinir
        T     #Yi
  _004: NOP   0
```

Network 9 : Kontrolor

Comment:

```
        L     #Yp
        L     #Yd
        +R
        L     #Yi
        +R
        T     #Ytop
```

Network 10 : Kontrolor icin sinirlandirma

Comment:

```
        L     1.000000e+000
        T     #x
        L     #Ytop
        L     #Umax_sinir
        >=R
        JNB   _007
        L     0.000000e+000
        T     #x
        L     #Umax_sinir
        T     #Ytop
  _007: NOP   0
```

Network 11: Kontrolor icin sinirlandirma

Comment:

```
        L       #Ytop
        L       #Umin_sinir
        <=R
        JNB     _008
        L       0.000000e+000
        T       #x
        L       #Umin_sinir
        T       #Ytop
  _008: NOP     0
```

Network 12: Title:

Comment:

```
        L       #e
        T       #e_1
        L       #Yi
        T       #Yi_1
        TRUNC
        T       #u_i
        L       #Yp
        TRUNC
        T       #u_p
        L       #Yd
        TRUNC
        T       #u_d
        L       #Ytop
        TRUNC
        T       #u
```

Network 13: Title:

Comment:

```
        AN      OS
        SAVE
```

APPENDIX B

```
FUNCTION_BLOCK FB40

VAR_INPUT
REF,Y:INT;
Ti,Td,Ts:REAL;
Umax,Umin,Umax_sinir,Umin_sinir:REAL;
END_VAR

VAR_OUTPUT
U,U_I,U_P,U_D:INT;
HATA:INT;
END_VAR

VAR_TEMP
TEMP_D,TEMP_I,K,l,B1,B2,B3,TEMP_ABS:REAL;
END_VAR

VAR
e,e_1,Yi,Yi_1,Yd,Yp,Y_top,X:REAL;
END_VAR

CONST
l1:=1.0;
l2:=0.2;
l3:=3.0;
END_CONST

BEGIN
HATA:=REF-Y;
e:=INT_TO_REAL(HATA);
TEMP_ABS:=ABS(e)/(INT_TO_REAL(REF-13000));

//FUZZY
  //e1
IF TEMP_ABS<=0.06524 THEN
    B1:=1;
ELSE
    IF TEMP_ABS<=0.1871 THEN
        B1:=-8.2061*(TEMP_ABS-0.1871);
    ELSE
        B1:=0;
    END_IF;
```

```
//e2
IF TEMP_ABS<=0.06445 THEN
    B2:=0.0;
ELSE
    IF TEMP_ABS<=0.18 THEN
        B2:=8.6542*(TEMP_ABS-0.06445);
    ELSE
        IF TEMP_ABS<=0.63 THEN
            B2:=1.0;
        ELSE
            IF TEMP_ABS<=0.87 THEN
                B2:=-4.1666*(TEMP_ABS-0.87);
            ELSE
                B2:=0.0;
            END_IF;

        END_IF;
    END_IF;
END_IF;

//e3
IF TEMP_ABS<=0.63 THEN
    B3:=0.0;
ELSE
    IF TEMP_ABS<=0.87 THEN
        B3:=4.1666*(TEMP_ABS-0.63);
    ELSE
        B3:=1;
    END_IF;
END_IF;

//LAMDA HESABI
l:=((B1*l3)+(B2*l2)+(B3*l1))/(B1+B2+B3);

K:=0.025087435/(l+0.5);

//P NIN HESAPLANMASI
Yp:=(K/(Td*Td))*e;

//TUREVIN HESAPLANMASI
Yd:=(K/(Td*Ts))*(e-e_1);
```

```
//turev sinirlandirmasi
IF Yd>=Umax THEN
    Yd:=Umax;
END_IF;

IF Yd<=Umin THEN
    Yd:=Umin;
END_IF;

//INTEGRALIN HESAPLANMASI
Yi:=(((K*Ts*X)/(2.0*(Ti*(Td*Td))))*(e+e_1))+Yi_1;

//integralin sinirlandirilmasi
IF Yi>=Umax_sinir THEN
    Yi:=Umax_sinir;
END_IF;

IF Yi<=Umin_sinir THEN
    Yi:=Umin_sinir;
END_IF;

//kontrolor icin sinirlandirma
X:=1.0;
IF Y_top>=Umax_sinir THEN
    X:=0.0;
    Y_top:=Umax_sinir;
END_IF;

IF Y_top<=Umin_sinir THEN
    X:=0.0;
    Y_top:=Umin_sinir;
END_IF;

e_1:=e;
Yi_1:=Yi;
U_I:=REAL_TO_INT(Yi_1);
U_P:=REAL_TO_INT(Yp);
U_D:=REAL_TO_INT(Yd);
U:=REAL_TO_INT(Y_top);

END_FUNCTION_BLOCK
```

CONTENTS