MONITORING & CONTROL OF PRESSURE IN A GAS PLANT VIA PID PLUS FEEDFORWARD CONTROLLER

EZYAN HARIS BINTI SULAIMAN

ELECTRICAL & ELECTRONICS ENGINEERING
UNIVERSITI TEKNOLOGI PETRONAS
JUNE 2009

CERTIFICATION OF APPROVAL

Monitoring and Control of Pressure in a Gas Plant via PID plus Feedforward Controller

by

Ezyan Hanis Binti Sulaiman

A project interim submitted to the

Electrical & Electronics Engineering Programme

Universiti Teknologi PETRONAS

in partial fulfilment of the requirement for the

Bachelor of Engineering (Hons)

(Electrical & Electronics Engineering)

Approved:

____

AP Dr. Nordin bin Saad

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

June 2009

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

EZYAN HANIS BINTI SULAIMAN

ABSTRACT

This project is about the development of a controller for a process using well-established advanced process control (APC) algorithm; mainly PID and Feedforward controls. This work focuses on a model of a process that would be used for investigation of the effectiveness of several control strategies towards effective control in overcoming disturbances in the plant. The controller is observed to see how well a variable can be manipulated and controlled in real-time implementation. However, it is well known that the performances of these controllers much depend on the appropriate implementation of additional functionalities such as anti-windup and feedforward, for example, in addition to the tuning of PID parameters. The process targeted is a gas process and it mainly focused on pressure and flow control of a gaseous pilot plant. To execute the overall simulation, the controller is built on MATLAB/Simulink/LabVIEW which is a technical computing program that has easily adaptable structure where control strategies and model variables can be modified. It is shown in the results of simulation and performance analysis of both controller and process that the PID plus Feedforward control could substantially improve control performance with implementation of a model error. The PID controller provides the needed reaction to the process variable to reach steady state during setpoint changes and disturbances while the feedforward element manages to fully eliminate the effects of disturbance injection without causing too much disruption to the process response.

ACKNOWLEDGEMENTS

I would like to start by expressing my gratitude towards Universiti Teknologi PETRONAS for giving me the opportunity to carry out an interesting and challenging final project. Being involved in this project has had me exposed to the field of process control engineering and a whole new meaning to the engineering course I have taken. My heartfelt appreciation to Assoc. Prof. Dr. Nordin bin Saad for his guidance and teachings as my supervisor. I have learned so much from him and this project would be less meaningful without his involvement.

Secondly, I would like to thank all Final Year Project (FYP) Committee, especially Electrical and Electronics Engineering Department that has equipped students with essential skills for self-learning. Its well-rounded graduate philosophy has proven to be useful for carrying out this project. Also, my acknowledgement would be incomplete without giving credit to Mr. Tri Chandra as the graduate assistant who has guided me along with two other final year students on the basic theory and development of this project.

Furthermore, I would like to thank my family and close friends for all the help and support they have given me throughout completing this project. Their guidance, kindness, and sincerity are very much appreciated. Finally, to each and every one that has been helping me directly or indirectly, thank you.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

PID	proportional-integral-derivative
FODT	first order plus dead time
PV	process variable
barg	unit of pressure
m^3/s	volumetric flow rate

CHAPTER 1

INTRODUCTION

1.1 Background Study

To maintain the desired set point for a control variable in a plant where the process variable is constantly being monitored, an engineer has to be aware of the disturbances that may occur during a process. The type and magnitude of disturbances affecting a gas plant can have a direct effect on the resulting product variability. Therefore, a study on a control process for an improved disturbance rejection is done to design the controller of the proposed control loop.

One of the control strategies to improve the regulatory performance of a process is feedforward control. This strategy trades off additional complexity in the form of instrumentation and engineering time in return for a controller better able to reject the impact of disturbances on the measured process variable. All of the elements used in the design of the controller refer to a particular action done in the control loop. The control actions would have an effect on the control loop performance [1].

1.2 Problem Statement

To achieve the best control strategy would involve performance requirements from the process control design such as process variable measurements, final control element characteristics, control structure in MATLAB/Simulink/LabVIEW and control calculations to achieve the best performance. A new design method is proposed in this report to further refine the concept for modeling a controller to monitor and control the pressure in the gas plant, which is PID plus Feedforward Controller.

For the experiment, the control loop of the test rig was modified to better suit the chosen method in terms of dynamic response and sensitivity. Modifications are made to reduce the complications in calculations of the variables involved.

1.3 Objectives and Scope of Study

This study investigates the monitoring and controlling of a pressure vessel in a gas plant. The open loop control system comprises a gas vessel, transmitters, controllers and control valves.

The main objective of this project is to create and develop a controller for maintaining desired set point of pressure in gas tank while at the same time, controlling the flow of input in a gaseous pilot plant. In order to do this, a new design method is proposed, which is using PID plus Feedforward controller. To proceed with this, the knowledge of MATLAB/Simulink/LabVIEW is essential in order to create the plant controller. Although regulation is often of primary concern for a PID controller, achieving a high performance in following set point is important in this application. The feedforward control structure and design would be applied in this project.

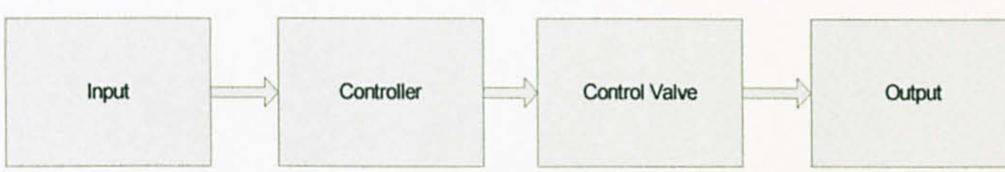


Figure 1: Block Diagram of Process Control

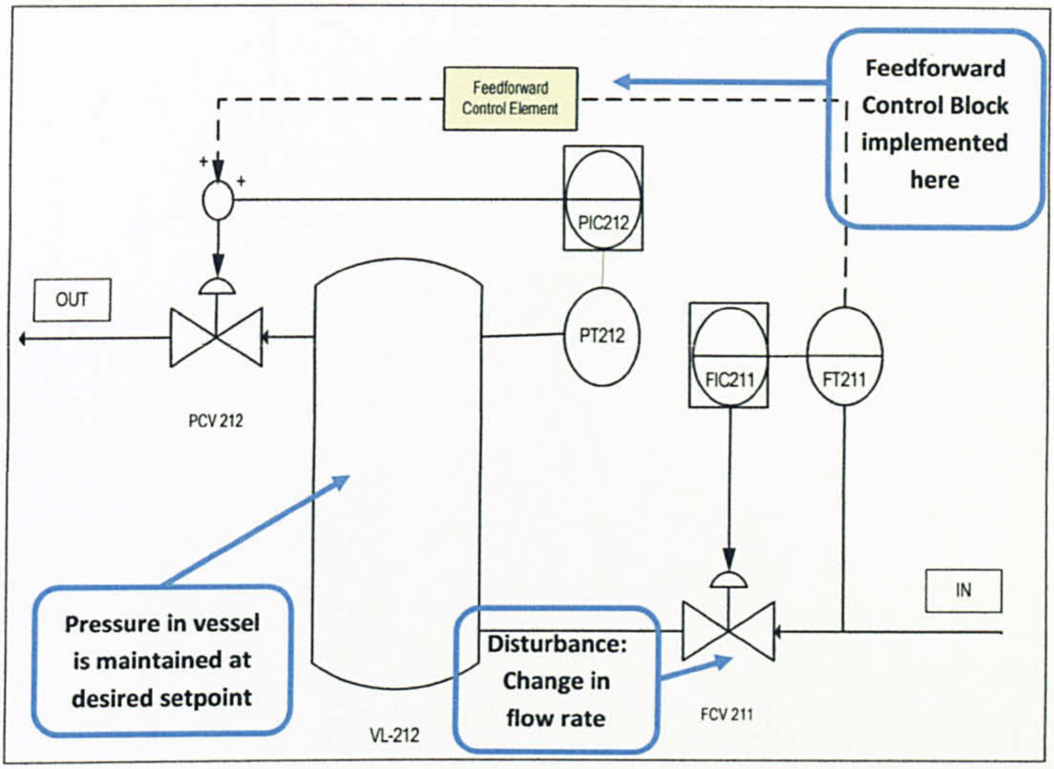


Figure 2: The Control Loop of Test Rig

CHAPTER 2

LITERATURE REVIEW

2.1 The PID Controller

A PID controller is a generic control loop feedback mechanism widely used in industrial control systems. PID stands for proportional–integral–derivative. A PID controller attempts to correct the error between a measured process variable and a desired setpoint by calculating and then outputting a corrective action that can adjust the process accordingly. The PID controller algorithm involves three separate parameters; the Proportional, the Integral and Derivative values. The Proportional value determines the reaction to the current error, the Integral determines the reaction based on the sum of recent errors and the Derivative determines the reaction to the rate at which the error has been changing.

By adjusting the three constants in the PID controller algorithm, the controller can provide control action designed for specific process requirements such as the position of a control valve or the power supply of a heating element. The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the setpoint and the degree of system oscillation. It is noted that the use of the PID algorithm for control does not guarantee optimal control of the system or system stability [3].

The controller algorithm is shown as below;

$$MV(t) = Kc\left(E(t) + \frac{1}{Ti}\int_0^t E(t')\,dt' + Td\,\frac{d\,CV(t)}{dt}\right) + I$$

where,

MV (t) = Manipulated Variable

Kc = Controller Gain

Ti = Integral Time

Td = Derivative Time

E (t) = Error

CV (t) = Controlled Variable or Process Variable

I = Initial Constant

The proportional mode provides a rapid adjustment of the manipulated variable, does not provide zero offset although it reduces the error, speeds the dynamic response and can cause instability if tuned improperly. The integral mode achieves zero offset, adjusts the manipulated variable in a slower manner than the proportional mode, thus giving poor dynamic performance and can cause instability if tuned improperly. The derivative mode does not influence the final steady-state value of error, provides rapid correction based on the rate of change of controlled variable and can cause undesirable high-frequency variation in the manipulated variable [1].

2.2 The Feedforward Theory

Feedforward control is designed using the five design rules which are the feedforward design criteria. They are;

- Single loop control is not acceptable
- Variable is measured
- Indicates a key disturbance
- No causal relationship with valve
- Variable dynamics due to disturbance not much faster than feedback loop

Below is an example of the process reactions when feedforward control is implemented [6].

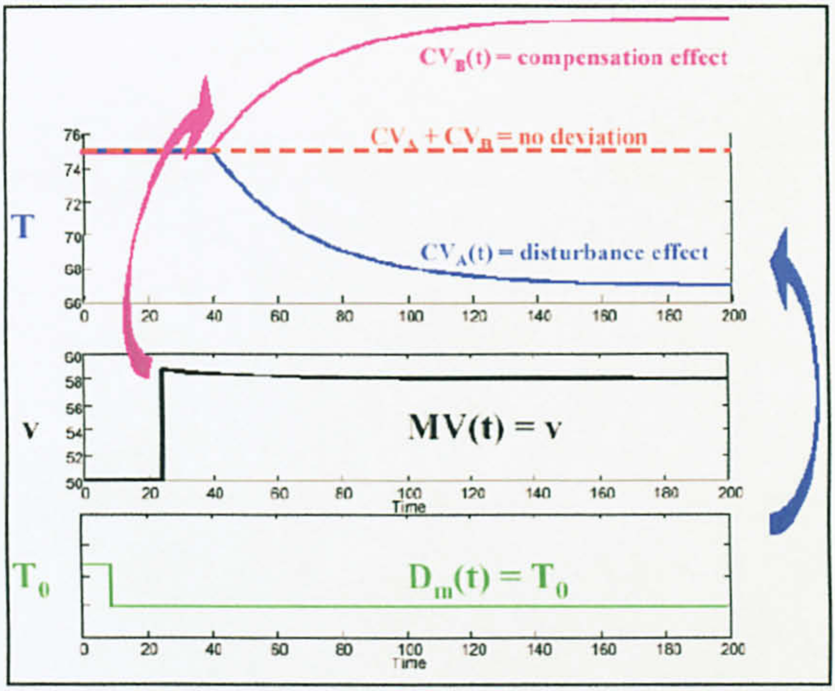


Figure 3: Example of Feedforward Control Process Reactions

To achieve the desired performance, block diagram algebra is implemented to determine the controller design;

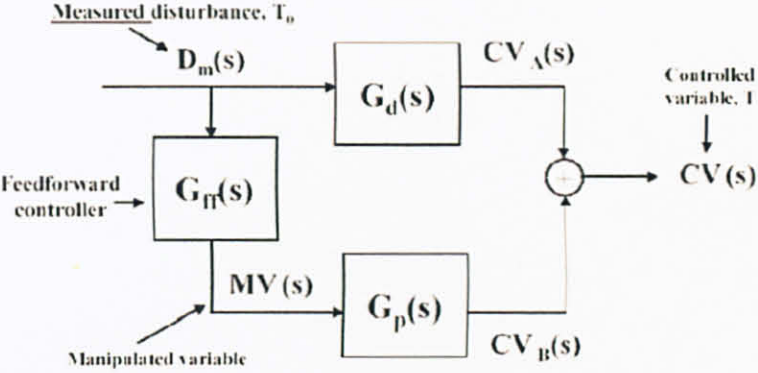


Figure 4: The Block Diagram for Feedforward Implementation

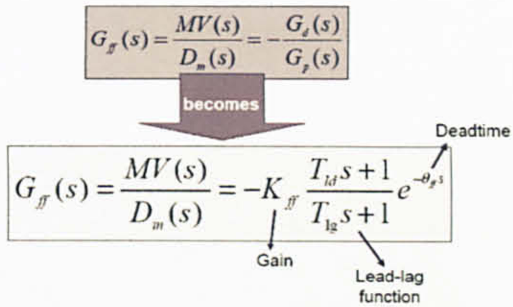
$$\begin{aligned} CV'(s) &= CV_A'(s) + CV_B'(s) = 0 \\ &= [G_d(s) + G_{ff}(s)G_p(s)]\, D_m(s) = 0 \end{aligned}$$

$$G_{ff}(s) = \frac{MV(s)}{D_m(s)} = -\frac{G_d(s)}{G_p(s)}$$

The relationship between $G_{ff}(s)$, $G_d(s)$ and $G_p(s)$

$D_m(s)$	= Measured disturbance
$G_{ff}(s)$	= Feedforward controller
$G_d(s)$	= Disturbance
$G_p(s)$	= Process
$CV(s)$	= Controlled Variables

If $G_p(s)$ and $G_d(s)$ are both first order with dead time, then it is proven that the equation of feedforward control is;



The Feedforward Control Equation

Thus, the transfer function for Gff(s) is;

$$G_{ff}(s) = \frac{MV(s)}{D_m(s)} = -K_{ff}\frac{T_{ld}s+1}{T_{lg}s+1}e^{-\theta_{ff}s}$$

where

Lead – lag	$=\frac{T_{ld}s+1}{T_{lg}s+1}e^{-\theta_{ff}s}$
FF controller gain	$=K_{ff}=-\frac{K_d}{K_p}$
Controller dead time	$=\theta_{ff}=\theta_d-\theta_p\geq 0$
Lead time	$=T_{ld}=\tau_p$
Lag time	$=T_{lg}=\tau_d$

There are pros and cons when using feedforward control. This is discussed in the table below;

Pros	**Cons**
Compensates for disturbance before the controlled variable is affected.	Cannot eliminate the steady state offset
Does not affect the stability of system	Requires a sensor and model for each disturbance.

Table 1: Pros and Cons of using Feedforward Control

The feedforward control is desired when a single loop performance is unacceptable. Apart from that, a measured variable should be available for it to work. A measured disturbance variable must;

- Indicate the occurrence of an important disturbance
- Not have a causal relationship from valve to measured disturbance sensor
- Not be significantly faster than the manipulated variable output dynamics (feedback)

CHAPTER 3

METHODOLOGY

3.1 Process Description

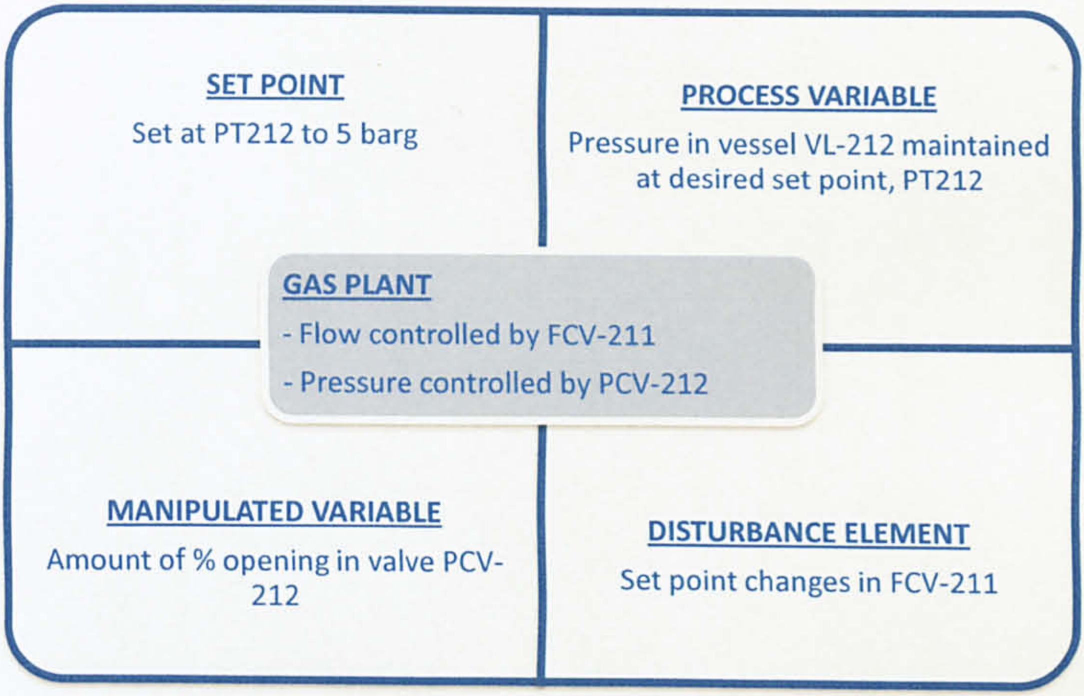
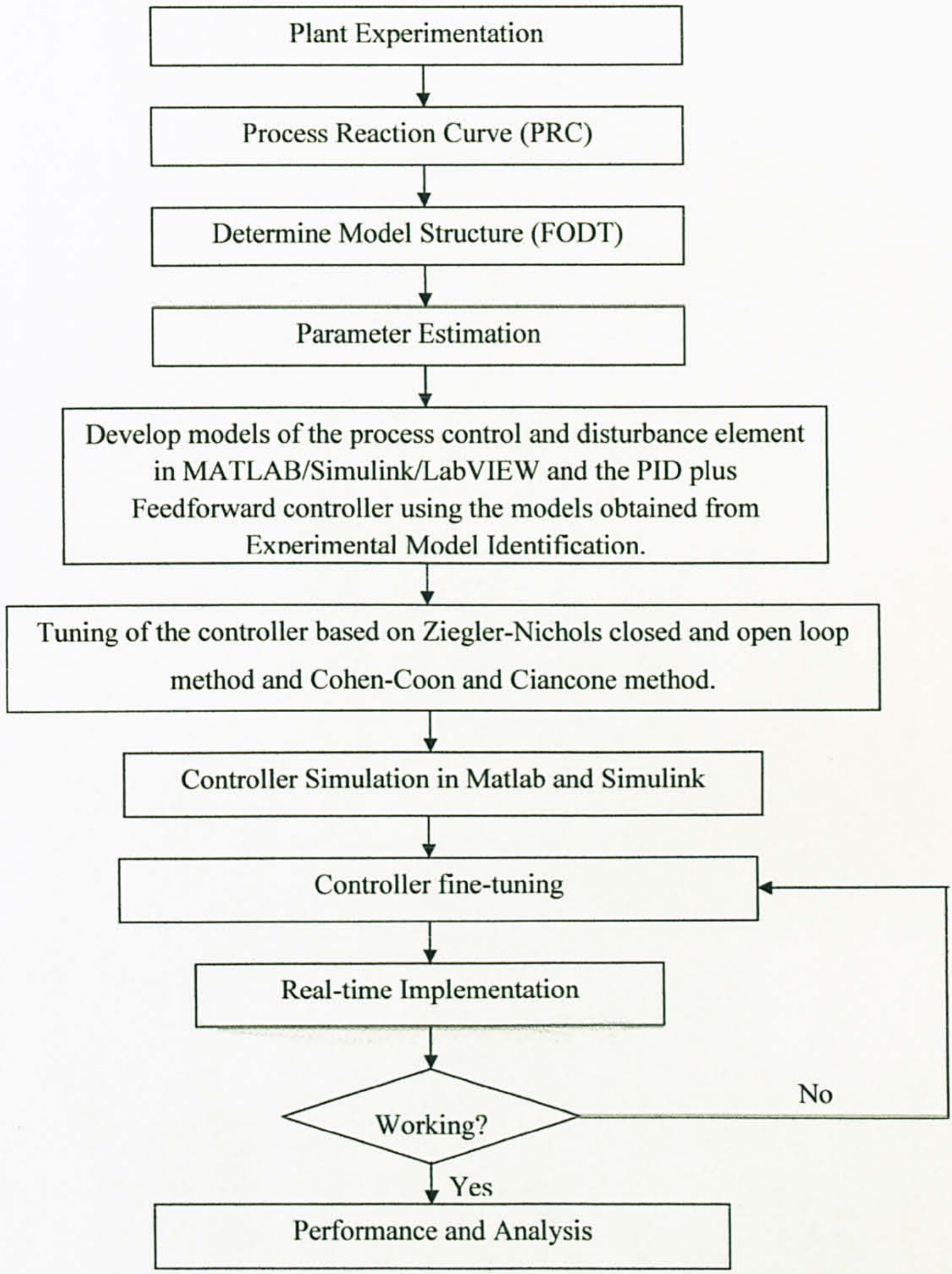


Figure 5: Variables chosen for the process control

3.2 Procedure Identification



3.3 Tools Required

The applications and plant used in this project will be;

a. MATLAB

MATLAB is a numerical computing environment and programming language. Created by The MathWorks, MATLAB allows easy matrix manipulation, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs in other languages.

b. Simulink

Simulink, developed by The MathWorks, is a commercial tool for modeling, simulating and analyzing multidomain dynamic systems. Its primary interface is a graphical block diagramming tool and a customizable set of block libraries. It offers tight integration with the rest of the MATLAB environment and can either drive MATLAB or be scripted from it. Simulink is widely used in control theory and digital signal processing for multidomain simulation and design.

c. LabVIEW

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) is a platform and development environment for a visual programming language from National Instruments.

d. Gas Pilot Plant

The process plant that will be used is the Gas Pilot Plant located in the Plant Process Control Systems Laboratory at Block 23, Universiti Teknologi PETRONAS. It consists of equipments that can be found in any industrial process plants such as controllers and valves.

3.4 Empirical Model Identification

Empirical model identification is a very efficient modeling method specifically designed for process control. The objective of the empirical model identification is to do parameter estimation based on the obtained Process Reaction Curve (PRC) which will lead to the identification of the process and disturbance models of the control system. Another one is to determine the initial values of controller parameters (Kc, T_I and T_D) through several tuning methods such as the Ziegler-Nichols Open-Loop Tuning method and Cohen-Coon Tuning correlations. Furthermore, it is done to test out the performance of the controller mode chosen (PID).

To start off the modeling, a proper experimental design is done to determine process shape and duration and to also determine the base operating conditions for the process, which essentially determine the conditions on which process model is accurate. The steps to starting the experimental design are describing the base operating conditions, defining the perturbations, defining the variables to be measured and estimating the duration of experiment [1].

3.4.1 Plant Experimentation

The experiment is designed to establish the relationship between one input and output. Plant operation is monitored during the experiment, using devices such as transmitters, controllers and valves. In this study, the experiment makes use of valves PCV212 and FCV211, controllers PIC212 and FIC211, and transmitters PT212 and FT211. There are two models that needed to be identified which are the process model and the disturbance model. The assumptions for the model structures would be first-order-with-dead-time (FODT) models, which are based on prior knowledge on unit operation and patterns of experimental data.

The first model is the process model which involves PCV212 and the process response is monitored at PT212. A step change input is introduced in PCV212 at a specified time to analyze the result of the experiment. The second model is the disturbance model which followed the same procedures as the first model and involving FCV211 and transmitter FT211. Starting with the process model, the experiment is started by maintaining the valve's opening PCV212 at 50% and PT212 is monitored. At time 120.2 seconds, the valve opening is changed to 70% and the change in process response is recorded. The initial value of pressure at PT212 is 4.68 barg.

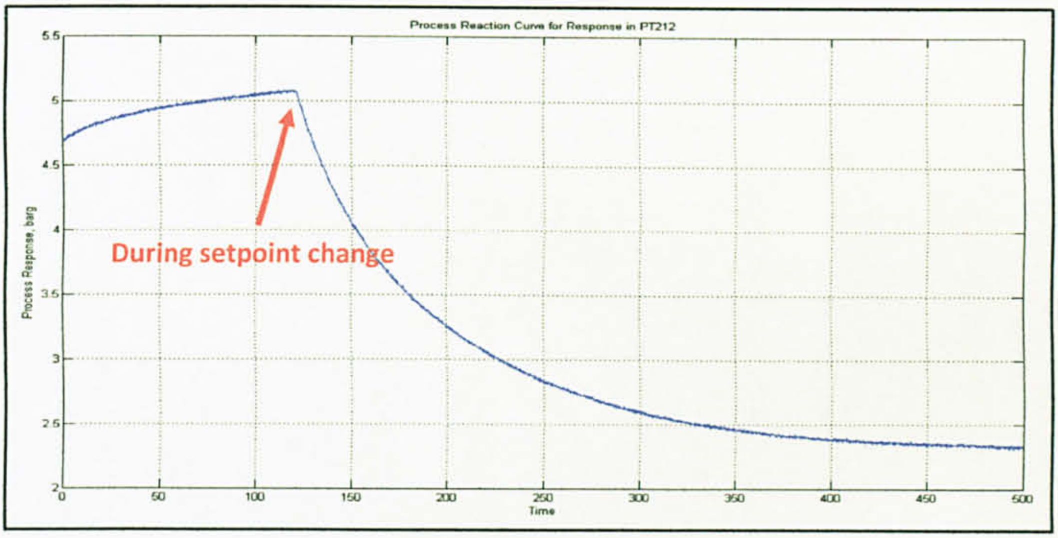


Figure 6: The Process Reaction Curve for Response at PT212

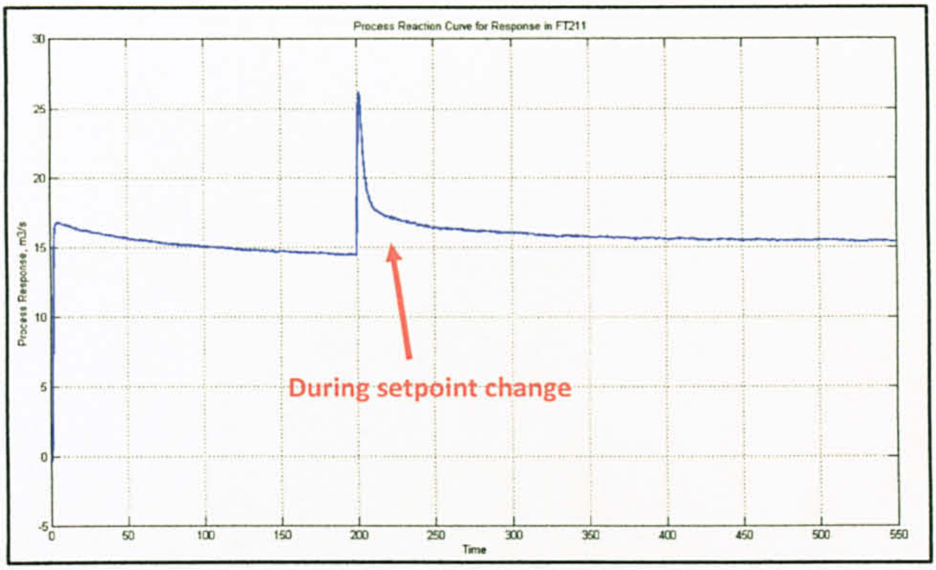


Figure 7: The Process Reaction Curve for Response at FT211

The experiment for the disturbance model is started by maintaining valve FCV211's opening at 50% and FT211 is monitored. At time 198.9 seconds, the valve opening is changed to 70% and the change in process response is recorded. The initial value of flow at FT211 is 14.48 m^3 /s.

3.4.2 Determine Model Structure

From the patterns in the experimental data and study of the plant's operation, the initial model structure can be chosen. In this study, the models for both process and disturbance are determined to be first-order-with-dead-time models which are adequate for process control analysis and design. The form of the model is expressed below, with X(s) referring to the input and Y(s) referring to the output [1].

$$\frac{Y(s)}{X(s)} = \frac{Kp\ e^{-\theta s}}{\tau s + 1}$$

Kp = Steady State Process Gain

τ = Apparent Time Constant

θ = Apparent Dead Time

3.4.3 Parameter Estimation

After a model structure has been selected and data collected, the values for the model parameters is determined to observe whether the model provided a good fit for the experimental data. The method used is the graphical process curve calculation which involved Method I and Method II. The calculations for process model PT212 are as below;

Measurement	Value
Change in Perturbation (MV), σ	20% opening
Change in output (PV), Δ	2.609 barg
Maximum slope, S	-0.0536 barg/s
Apparent Dead Time, θ	7 s

Table 2: Calculations from Process Reaction Curve of PT212

Calculations	Value	
Steady State Process Gain, K_p	0.13045 barg/% opening	
Apparent Time Constant, τ	Method I	Method II
Method I : $\tau = \Delta / S$ Method II: $\tau = 1.5(t_{0.63\Delta} - t_{0.28\Delta})$	-48.66 s	-65.55 s

Table 3: Model Parameters for Transfer Function PT212

The methods and calculations for disturbance model FT211 are done similar to the process model and tabulated as below;

Measurement	Value
Change in Perturbation (MV), σ	20% opening
Change in output (PV), Δ	0.9582 m^3/s
Maximum slope, S	-0.0536 m^3/s^2
Apparent Dead Time, θ	0.8 s

Table 4: Calculations from Process Reaction Curve of FT211

Calculations	Value	
Steady State Process Gain, K_p	0.04791 (m^3/s)/% opening	
Apparent Time Constant, τ	Method I	Method II
Method I : $\tau = \Delta / S$ Method II: $\tau = 1.5(t_{0.63\Delta} - t_{0.28\Delta})$	12.22 s	582.15 s

Table 5: Model Parameters for Transfer Function FT211

3.4.4 Diagnostic Evaluation

Evaluation is required before the model is used for control, which could determine how well the model fits the data used for parameters estimation. In this study, the approach used is the comparison of the model prediction with the measured data.

Starting with the process model for PT212, a comparison of measured data and simulated models of both Methods I and II is done to select the process model that would be used in this project to further construct the proposed controller.

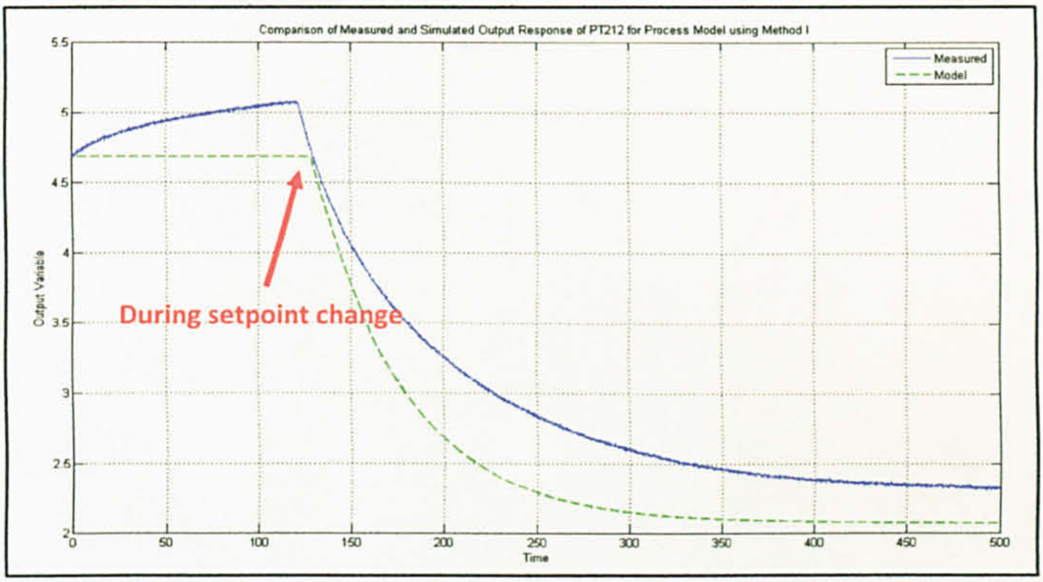


Figure 8: Comparison of Measured and Simulated Response of PT212 for Process Model using Method I

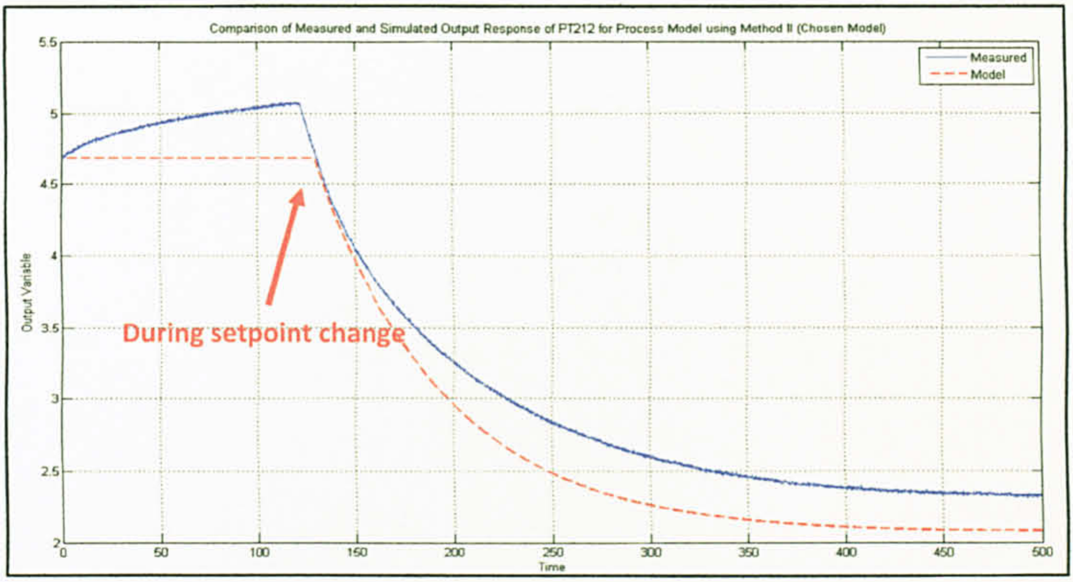


Figure 9: Comparison of Measured and Simulated Response of PT212 for Process Model using Method II

From the comparison of simulated response using Method I and II, the model calculated using Method II generated a much closer resemblance to the measured data. Thus, the chosen model for the process, G_p is as below;

$$Gp = \frac{Kp\, e^{-\theta ps}}{\tau s + 1}$$

$$Gp = \frac{-0.13045\, e^{-7s}}{65.55s + 1}$$

For the disturbance model FT211, the same comparison of measured data and simulated models of both Methods I and II are done to select the disturbance model that would be used in this project to further construct the proposed controller.

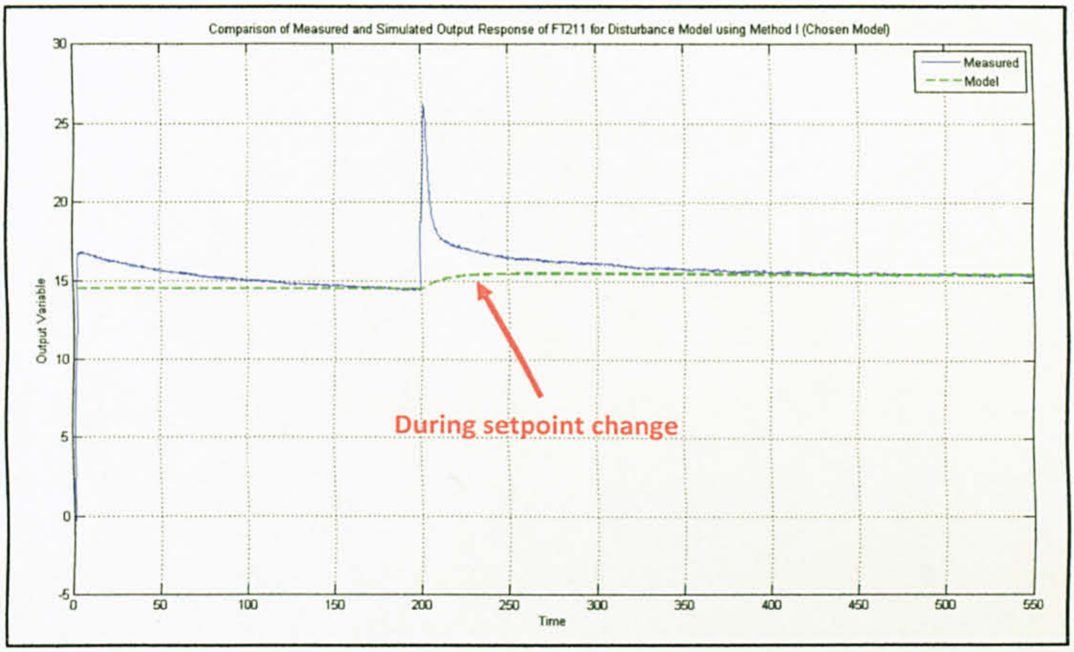


Figure 10: Comparison of Measured and Simulated Response of FT211 for Disturbance Model using Method I

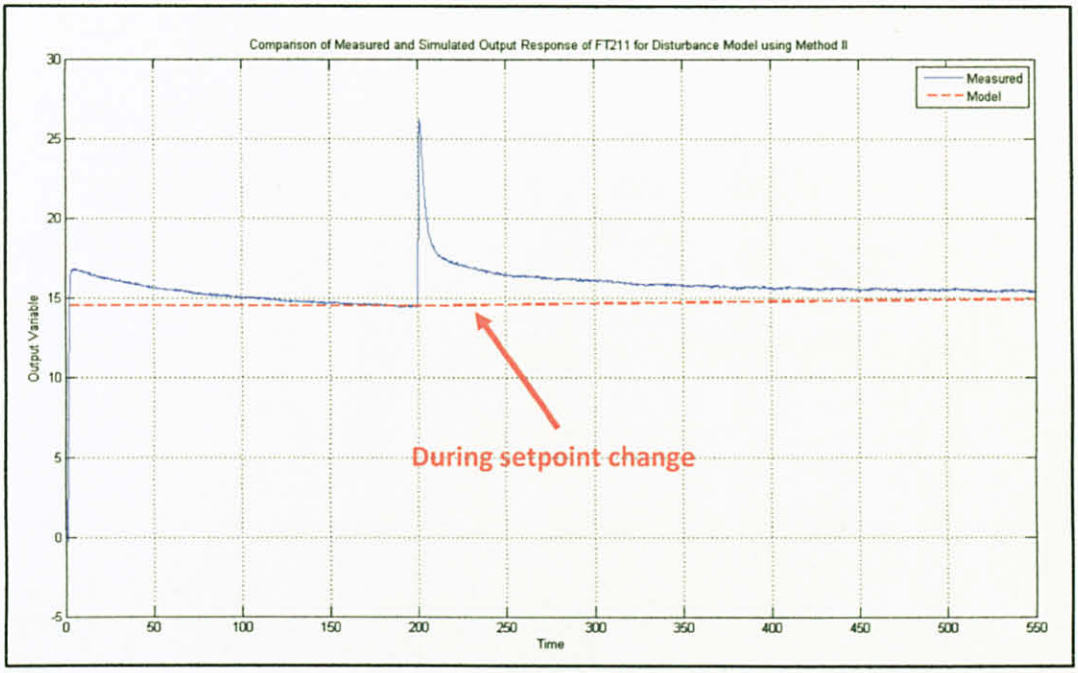


Figure 11: Comparison of Measured and Simulated Response of FT211 for Disturbance Model using Method II

From the comparison of simulated response using Method I and II, the model calculated using Method I generated a much closer resemblance to the measured data. Thus, the chosen model for the disturbance, G_d is as below;

$$Gd = \frac{Kd\, e^{-\theta ds}}{\tau s + 1}$$

$$Gd = \frac{0.04791\, e^{-0.8s}}{12.22s + 1}$$

3.5 Controller Design

The aim of this study is to implement the feedforward control strategy with PID control scheme and achieve a newly-devised controller. By testing each control scheme and putting them together, a new control strategy is obtained. Figure below shows a conceptual block diagram of the control scheme that needed to be achieved [2].

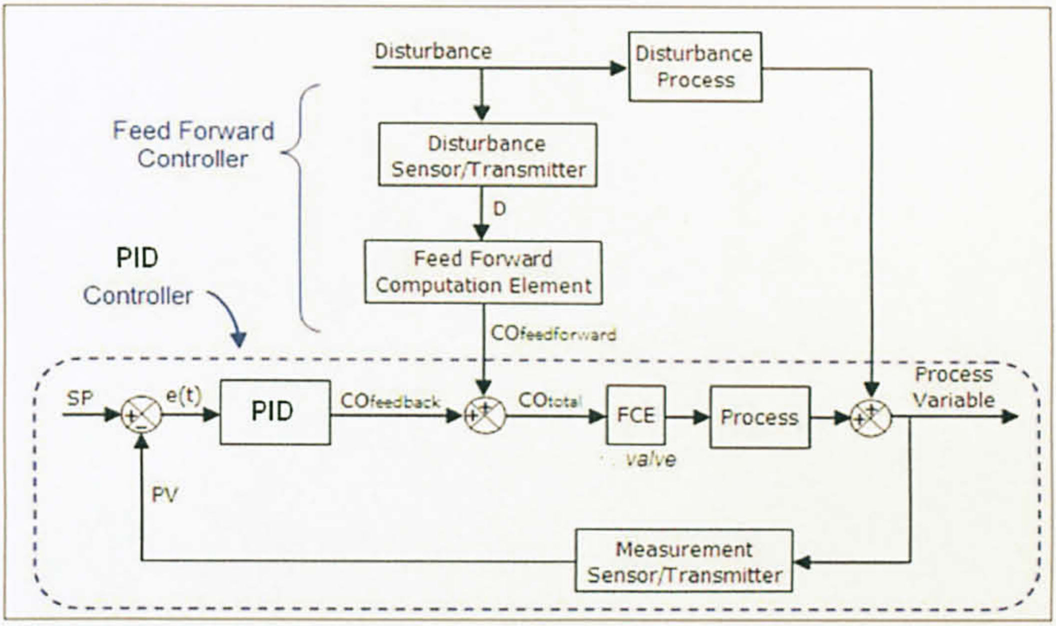


Figure 12: The PID plus Feedforward Block Diagram

3.5.1 Building the PID Controller

Since one of the aims of this methodology is to design a control scheme based on PID Controller, the controller mode that needed to be concentrated on is PID mode in reverse-acting since the response goes to the opposite direction from the change in input and the process gain is negative [3]. Thus, the controller algorithm used is shown below;

$$MV(t) = Kc\left(E(t) + \frac{1}{Ti}\int_{0}^{t} E(t')\,dt' + Td\,\frac{d\,CV(t)}{dt}\right) + I$$

where,

MV (t) = Manipulated Variable

Kc = Controller Gain

Ti = Integral Time

Td = Derivative Time

E (t) = Error

CV (t) = Controlled Variable or Process Variable

I = Initial Constant

In order to build the PID controller, the parameters would need to be tuned using three controller tuning methods and the responses analyzed to choose the most suitable controller tuning method for the PID controller and further fine-tuning [1]. The three controller tuning methods used are;

i) Ziegler-Nichols Closed Loop Tuning Method

This method used the plotting of amplitude ratio and phase angle in the form of Bode plot. The critical frequency w_c and the amplitude ratio *AR* are determined to calculate the ultimate gain, *Ku*, which brings the system to margin of stability at the critical frequency, and the ultimate period, *Pu*, which is the period of oscillation of system at the margin of stability. From here, the parameters of PID controller mode are calculated and tabulated.

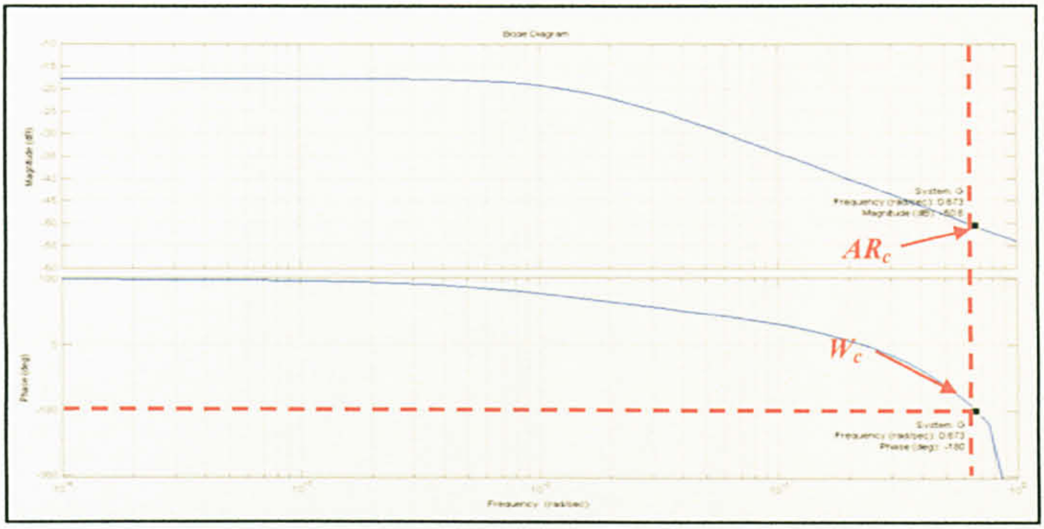


Figure 13: Bode plot for Process Model, Gp for PT212

Ultimate gain, *Ku,* is the value of the proportional gain that brings the system to the boundary of stability at critical frequency.

$$K_u = \frac{1}{|G_{OL}(j\omega_c)|} = \frac{1}{AR_c}$$

Ultimate period, *Pu*, is the period of oscillation of the system at the margin of stability.

$$P_u = \frac{2\pi}{\omega_c}$$

$\textit{Ultimate gain}, Ku = 338.8 \qquad \textit{Ultimate period}, Pu = 9.336\ s$

Controller	*Kc*	*Ti*	*Td*
PID	*Ku / 1.7 = 199.3*	*Pu / 2.0 = 4.7*	*Pu / 8 = 1.17*

Table 6: Ziegler-Nichols Closed Loop Tuning Correlations

The controller is set to manual mode when the opening in valve is changed. PID tuning correlations are included in the controller. The PID controller's mode is then changed to auto and the process variable response is observed.

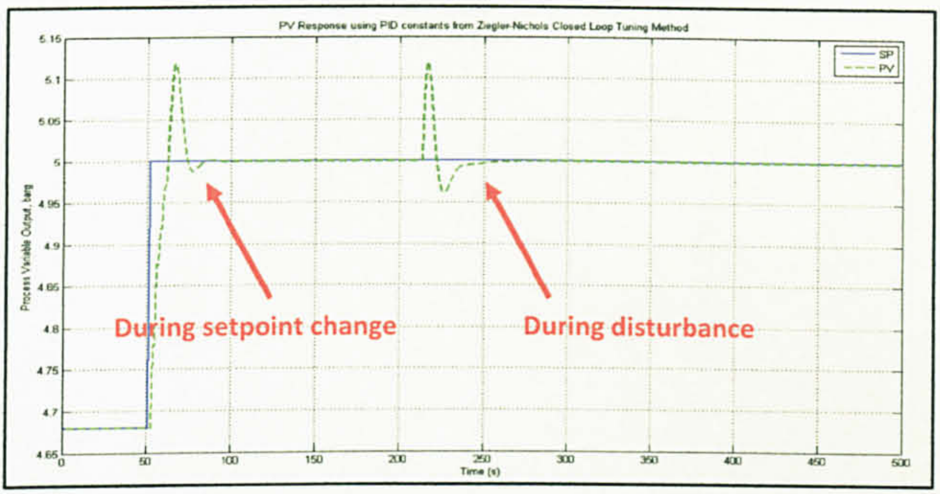


Figure 14: PV Response using PID Constants from Ziegler-Nichols Closed Loop Tuning Method

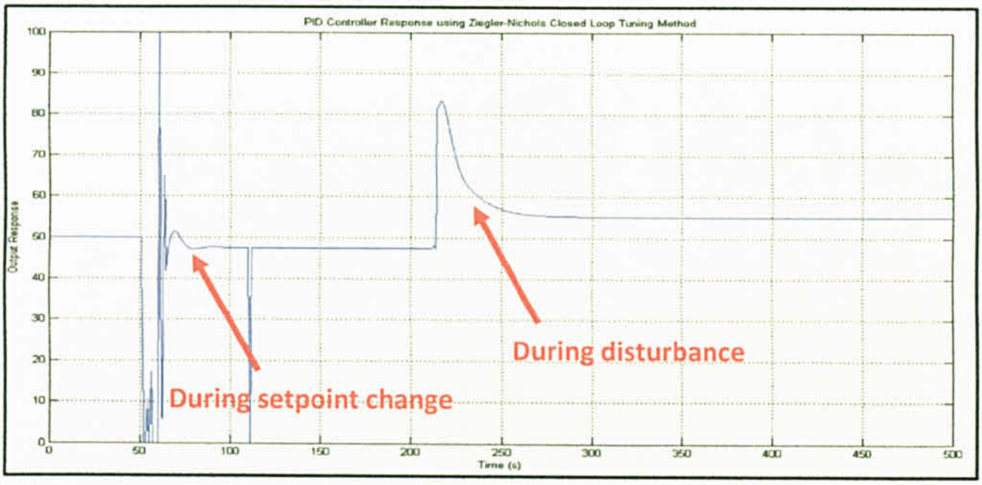


Figure 15: PID Controller Response using Ziegler-Nichols Closed Loop Tuning Method

From the figures above, it is observed that the process variable response had quite an overshoot, quick rise time and settling time. The controller response was aggressive at setpoint change but that could be fixed with fine-tuning.

ii) Ziegler Nichols Open Loop Tuning Method

This method provided correlations that are used with simplified process models developed from open loop process reaction curves. The parameters used are taken from the process model, *Gp*.

Controller	*Kc*	*Ti*	*Td*
PID	$(1.2/Kp)\ (\tau/\theta) = 86$	$2.0\ \theta = 14$	$0.5\ \theta = 3.5$

Table 7: Ziegler-Nichols Open Loop Tuning Correlations

The controller is set to manual mode when the opening in valve is changed. PID tuning correlations are included in the controller. The PID controller's mode is then changed to auto and the process variable response is observed.

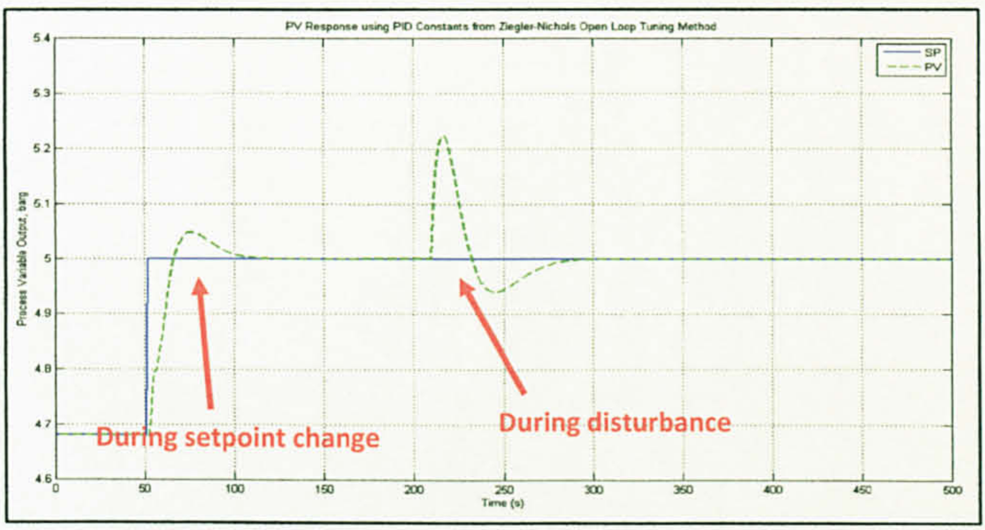


Figure 16: PV Response using PID Constants from Ziegler-Nichols Open Loop Tuning Method

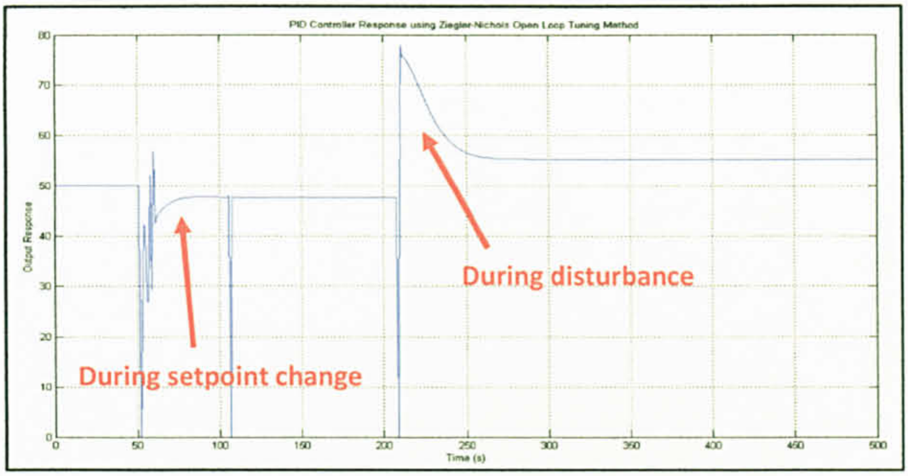


Figure 17: PID Controller Response using Ziegler-Nichols Open Loop Tuning Method

From the figures above, it is observed that the process variable response had a smaller overshoot during setpoint change compared to Ziegler-Nichols Closed Loop Tuning Method but a slower rise time. The system also showed a slower settling time and a much higher overshoot during the disturbance in comparison with previous method. The controller response was less aggressive at setpoint change but that could be fixed with fine-tuning.

iii) Cohen-Coon Tuning Correlations

This method of controller tuning corrects the slow, steady-state response given by the Ziegler-Nichols method when there is a large dead time relative to the open loop time constant. This method is only used for first-order models with time delay.

Controller	*Kc*	*Ti*	*Td*
PID	$\frac{1}{Kp}\frac{\tau}{\theta}\left(1+\frac{\theta}{3\tau}\right)$ $= 97.6$	$\theta\frac{\left(32+6\frac{\theta}{\tau}\right)}{13+8\frac{\theta}{\tau}}$ $= 16.5$	$\frac{4\theta}{11+2\frac{\theta}{\tau}}$ $= 2.5$

Table 8: Cohen-Coon Tuning Correlations

The controller is set to manual mode when the opening in valve is changed. PID tuning correlations are included in the controller. The PID controller's mode is then changed to auto and the process variable response is observed.

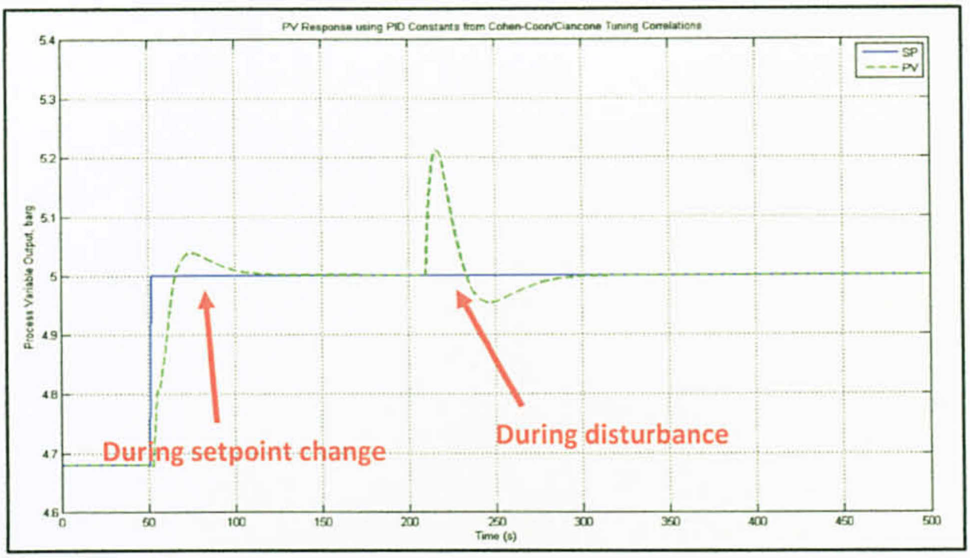


Figure 18: PV Response using PID Constants using Cohen-Coon/Ciancone Tuning Correlations

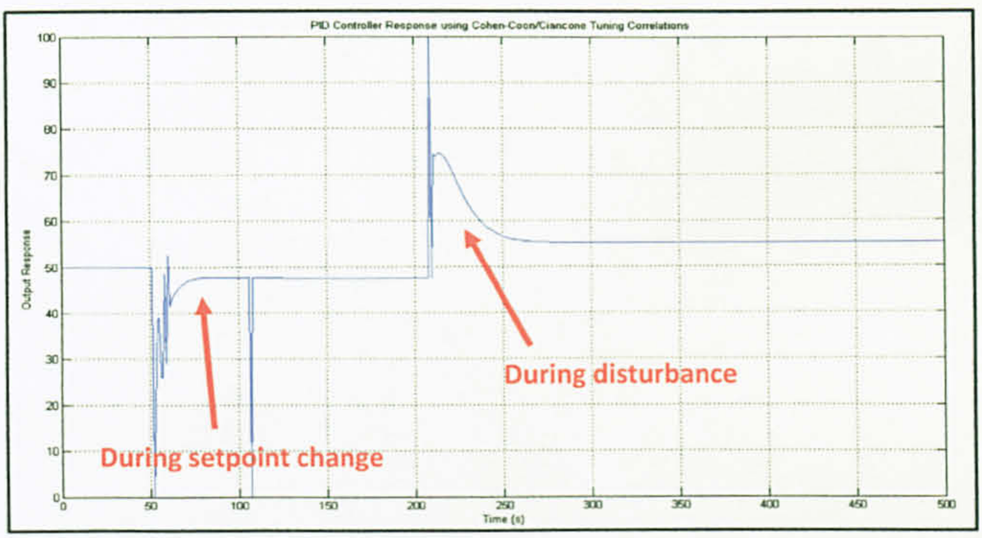


Figure 19: PID Controller Response using Cohen-Coon/Ciancone Tuning Correlations

From the figures above, it is observed that, similar with the results from Ziegler-Nichols Open Loop Tuning Method, the process variable response had a smaller overshoot during setpoint change compared to Ziegler-Nichols Closed Loop Tuning Method but a slower rise time. The system also showed a slower settling time and a much higher overshoot during the disturbance in comparison with the first method. The controller response was less aggressive at setpoint change but that could be fixed with fine-tuning.

Comparing all three tuning methods, the most suitable system needed to be chosen. Each method achieved zero steady state offset, which is the aim of this control system, but the responses from Ziegler-Nichols Closed Loop Tuning method has the shortest rise time, settling time and smaller overall overshoot in contrast to the other two methods.

Thus, the most suitable tuning method for the PID Controller is chosen to be Ziegler-Nichols Closed Loop Tuning Method. Shown below is the PID structure built inside Simulink to build the overall controller.

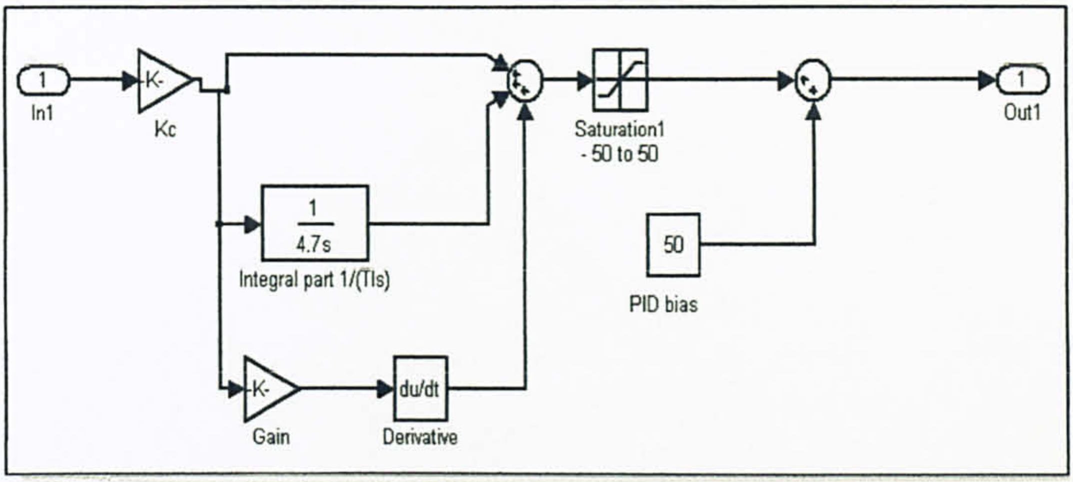


Figure 20: The PID Structure

From the PID structure above, the PID controller is then built in MATLAB and Simulink and shown below;

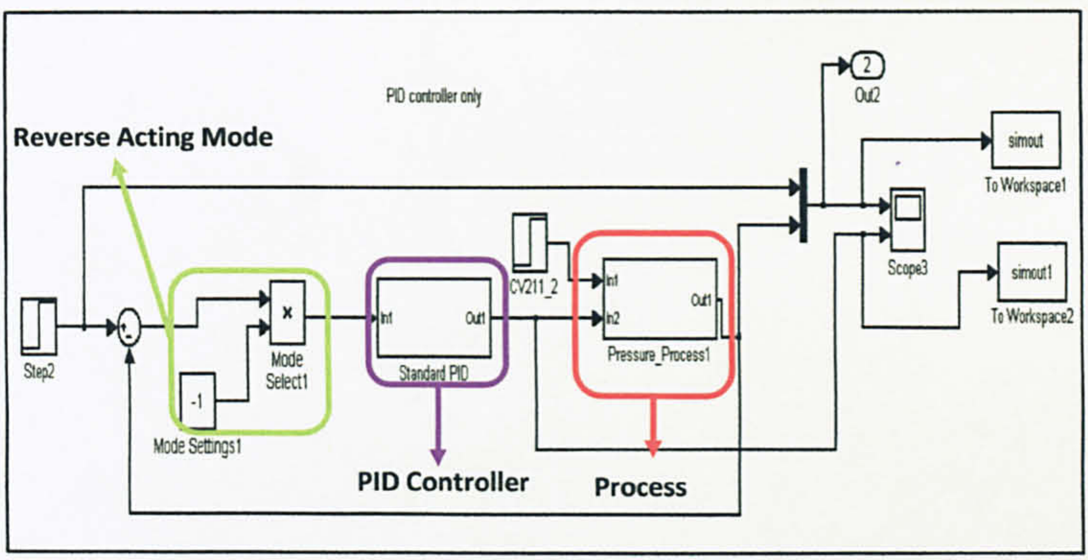


Figure 21: The PID Controller built in Simulink

3.5.2 Building the Feedforward Controller

After selecting the suitable transfer functions to present the process and disturbance models, thus the feedforward element could be constructed. The feedforward algorithm that has been studied is structured as below;

$$G_{ff}(s) = \frac{MV(s)}{D_m(s)} = -K_{ff}\,\frac{T_{ld}s+1}{T_{lg}s+1}e^{-\theta_{ff}s}$$

where

$$\begin{aligned}
\text{Lead-lag} &= \frac{T_{ld}s+1}{T_{lg}s+1}e^{-\theta_{ff}s} \\
\text{FF controller gain} &= K_{ff} = -\frac{K_d}{K_p} \\
\text{Controller dead time} &= \theta_{ff} = \theta_d - \theta_p \geq 0 \\
\text{Lead time} &= T_{ld} = \tau_p \\
\text{Lag time} &= T_{lg} = \tau_d
\end{aligned},$$

The process and disturbance models that have been selected are shown as below;

$$Process\ Model, Gp = \frac{-0.13045\ e^{-7s}}{65.55s+1}$$

and

$$Disturbance\ Model, Gd = \frac{0.04791\ e^{-0.8s}}{12.22s+1}$$

Hence, the feedforward model, Gff structured using the disturbance and process models and algorithms above are obtained as below. The feedforward dead time is set to 0 to make it logical, as the latter dead time resulted in negative value [6].

$$Feedforward\ Model, Gff = 0.3678\left(\frac{65.55s+1}{12.22s+1}\right)e^{-0s}$$

The feedforward model needed to be included in the PID Controller design, thus the completed controller design which has been built in Simulink is shown as below;

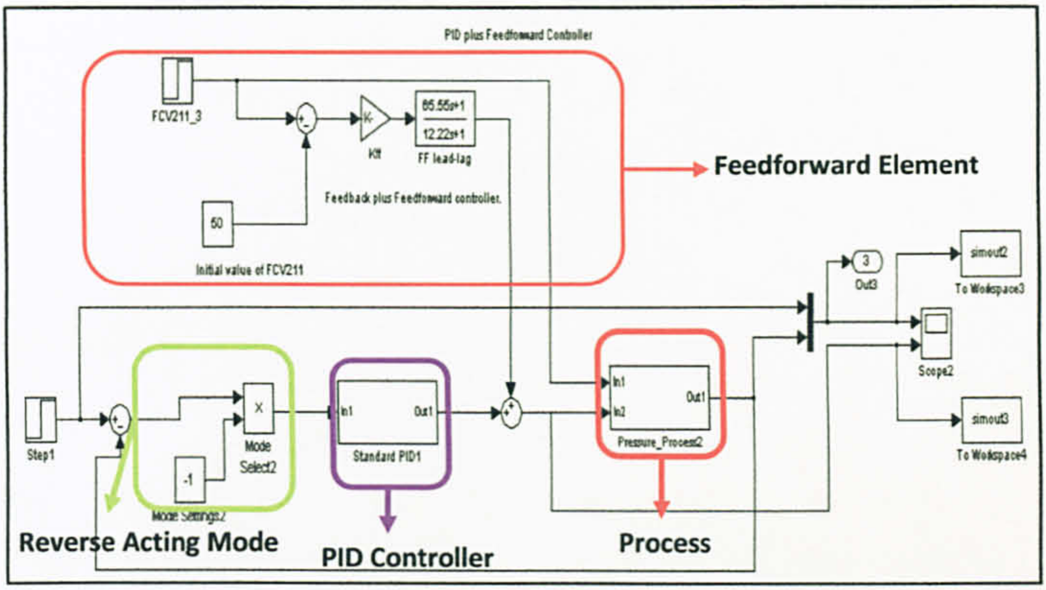


Figure 22: The PID plus Feedforward Controller built in Simulink

The controller has been built using the traditional feedback system, PID control system and Feedforward model. As shown in the figure above, these control schemes has all been put together to build the controller structure as studied in this project.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Performance Analysis

Typically, after designing the controller structure, the controller and process performance are subsequently evaluated. In this study, simulations are done repeatedly so analysis can be made to further investigate the characteristics of the controller and responses.

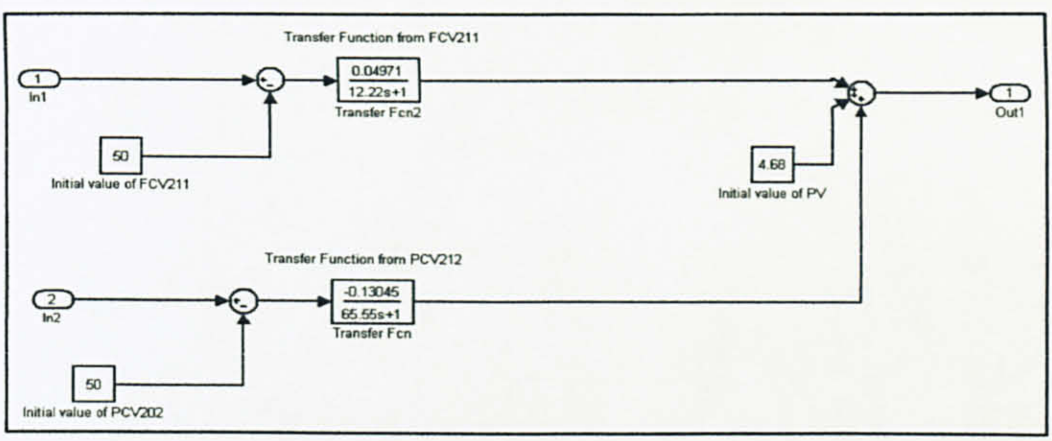


Figure 23: Open Loop Simulation of Pressure Process as the Responses changed in FCV211 and PCV212

4.1.1 Controller Performance and Process Response using PID Controller only

For this controller's simulation, the step change is introduced at 50 seconds and the setpoint is changed from 4.68 barg to 5 barg. The process variable reached steady state at 78.5 seconds at the first setpoint change and the disturbance step input is introduced at 200 seconds. The controller has been fine-tuned to get a better process response.

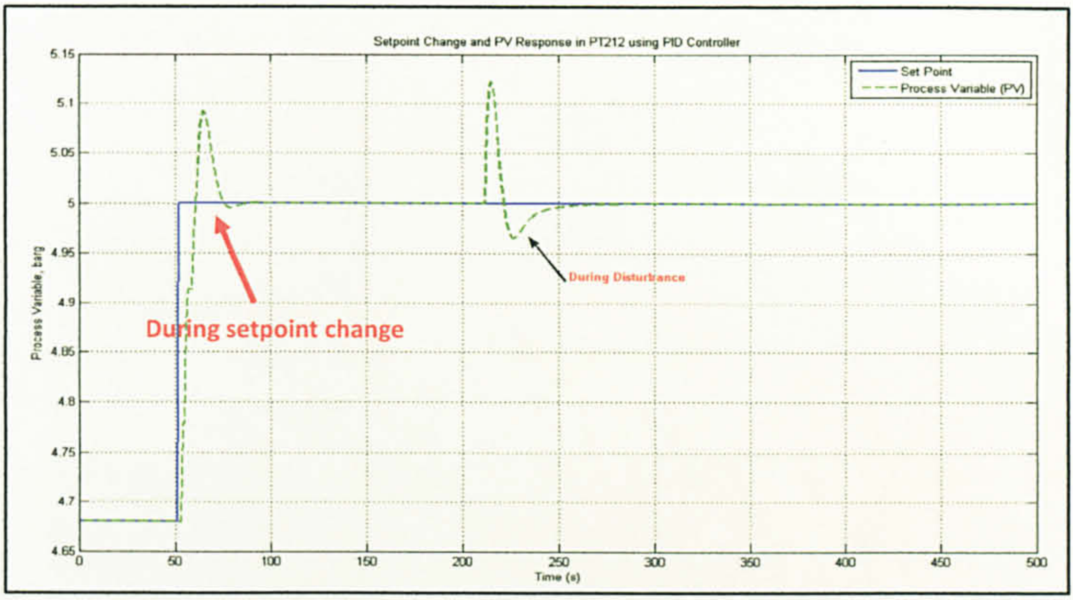


Figure 24: Setpoint Change and Process Variable Response using PID Controller

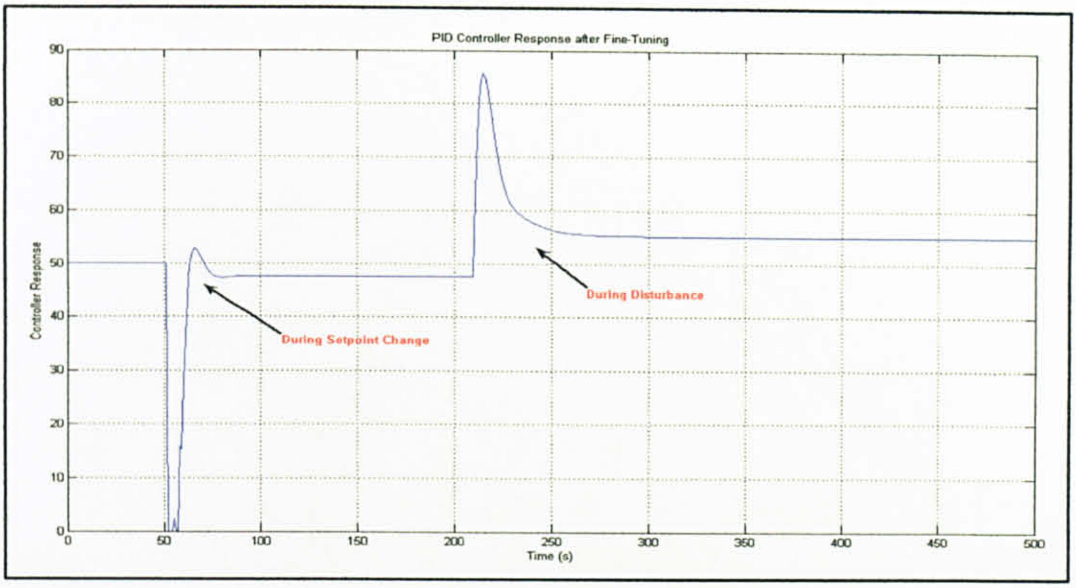


Figure 25: The PID Controller Response

During setpoint change, a small percentage of overshoot occurred in the process response. The system had a short rise time and took approximately 28.5 seconds to reach steady state. It also had zero steady state offset which is the aim of the control system. After disturbance is injected, the response showed minor overshoot before PID control corrected the error to reach steady state at 246 seconds.

4.1.2 Controller Performance and Process Response using PID plus Feedforward Controller

For this controller's simulation, the conditions have been set the same with the previous controller. The step change is introduced at 50 seconds and the setpoint is changed from 4.68 barg to 5 barg. The process variable reached steady state at 80 seconds at the first setpoint change and the disturbance step input is introduced at 200 seconds. The controller has been fine-tuned to get a better process response.

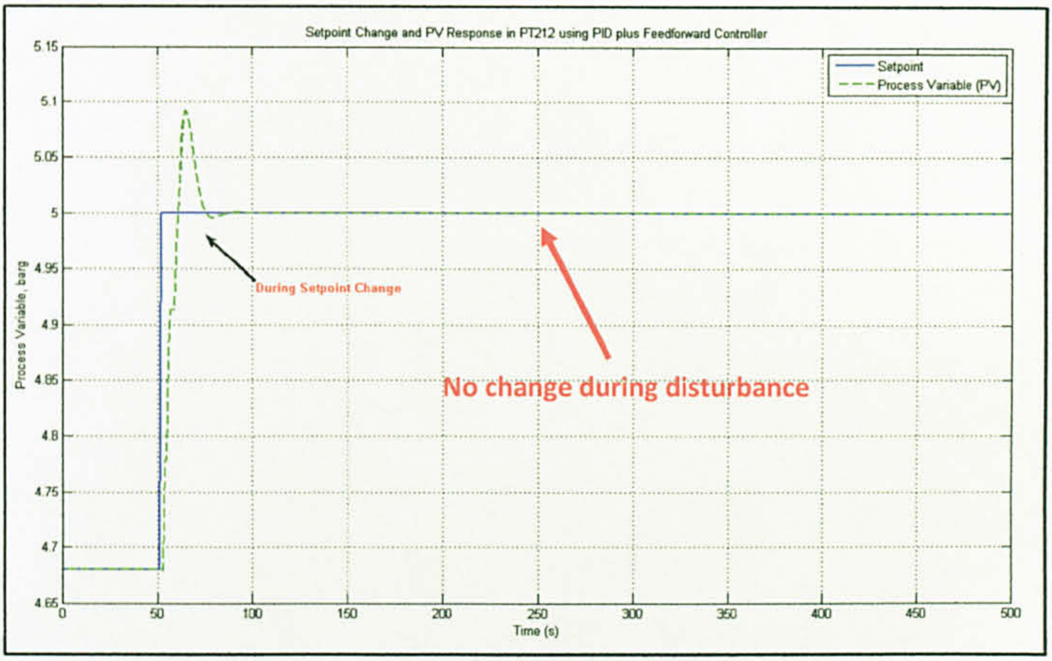


Figure 26: Setpoint Change and Process Variable Response using PID plus Feedforward Controller

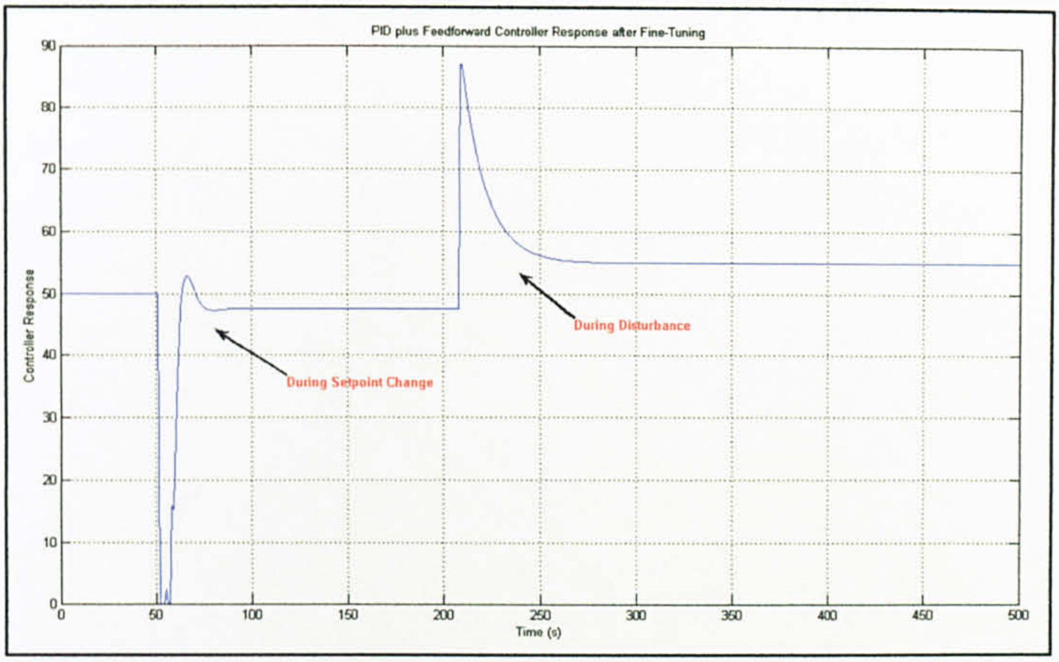


Figure 27: The PID plus Feedforward Controller Response

During setpoint change, a small percentage of overshoot occurred in the process response. The system had a short rise time and took approximately 30 seconds to reach steady state. It also had zero steady state offset which is the aim of the control system. After disturbance is injected, the response did not show any change which demonstrates that PID plus Feedforward control could substantially improve control performance with implementation of a model error.

4.2 Fine-Tuning

The tuning constants calculated previously are considered initial values to be applied to the process to obtain empirical information on closed-loop performance and modified until acceptable control performance is obtained. Fine-tuning had been necessary because of errors in the process model and simplifications in the tuning method [1].

4.2.1 Comparisons of Controllers before and after PID Fine-tuning

The PID and PID plus Feedforward controllers has been fine-tuned to get improved control performances. Changes for both controllers in the controller gain, *Kc*, increased around 0.5%, and derivative time, *Td*, decreased around 50% are implemented. The controller responses before and after fine-tuning are compared and analyzed.

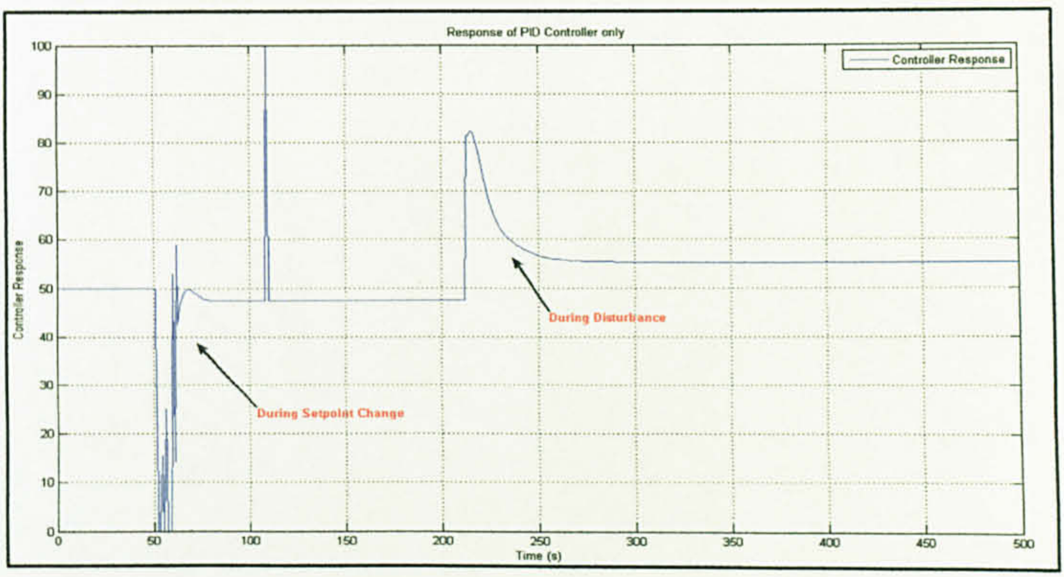


Figure 28: PID Controller Response before Fine-Tuning

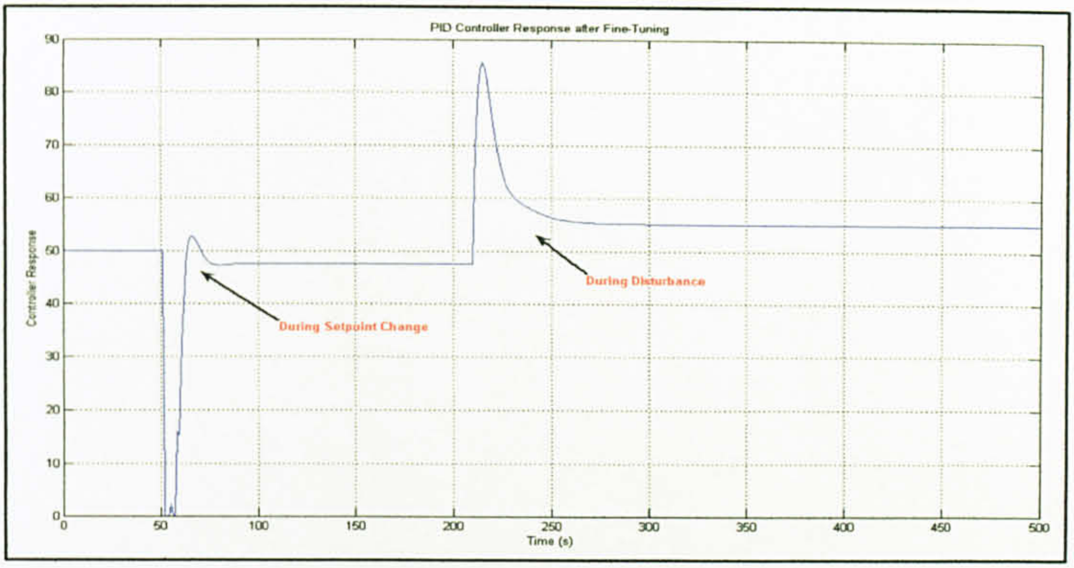


Figure 29: PID Controller Response after Fine-Tuning

From the figures above, it is observed that the PID controller gave quite an aggressive performance using the tuning constants earlier calculated. By fine-tuning the constants, the controller's performance improved and showed a much stable characteristic.

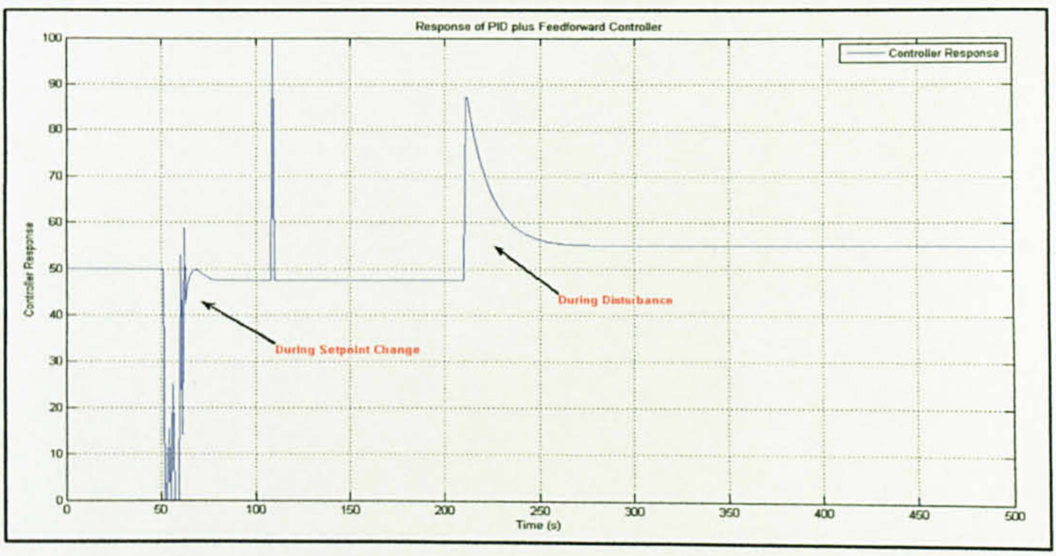


Figure 30: PID plus Feedforward Controller Response before Fine-Tuning

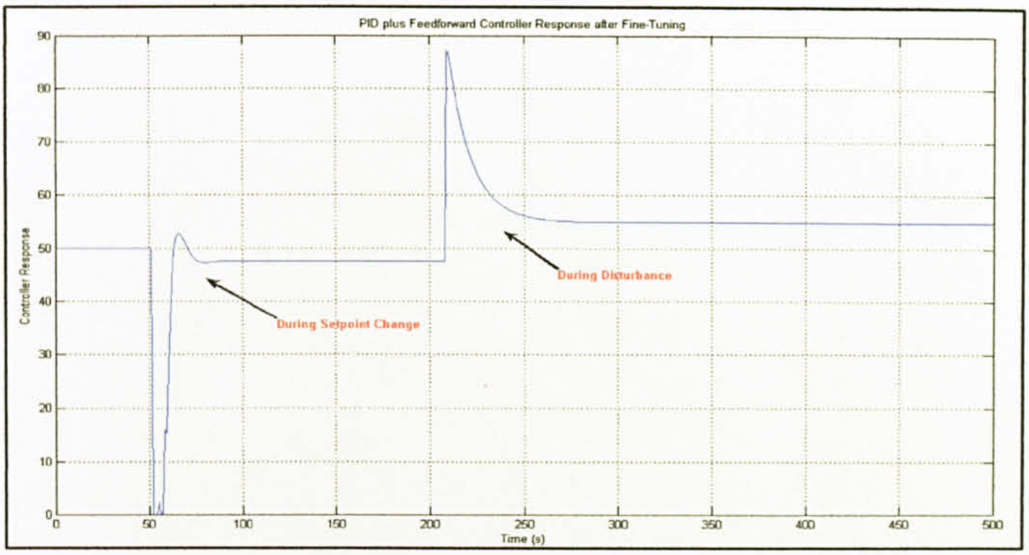


Figure 31: PID plus Feedforward Controller Response after Fine-Tuning

From the figures above, it is observed that the PID plus Feedforward controller gave quite an aggressive performance using the tuning constants earlier calculated. By fine-tuning the constants, the controller's performance improved and showed a much stable characteristic.

4.2.2 Feedforward Controller Gain Fine-tuning

In this study, the feedforward controller gain, *Kff*, is adjusted to analyze the difference it had in overcoming disturbance in the process variable (PV) response. The calculated *Kff* is 0.3678 and the other controller gain values tested vary from 0.2 and 0.4.

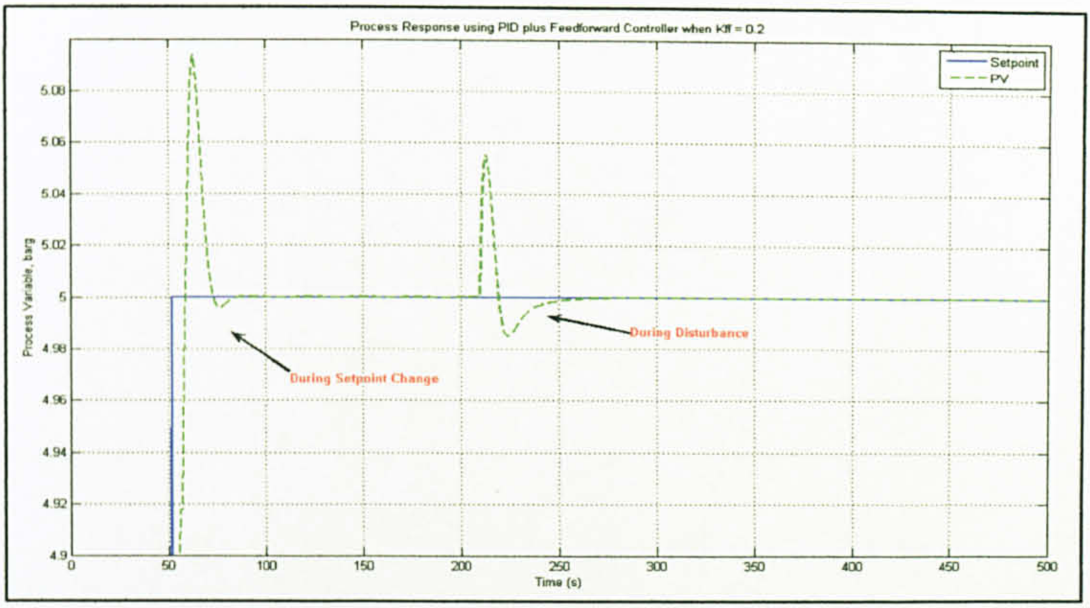


Figure 32: Process Variable Response when Kff = 0.2

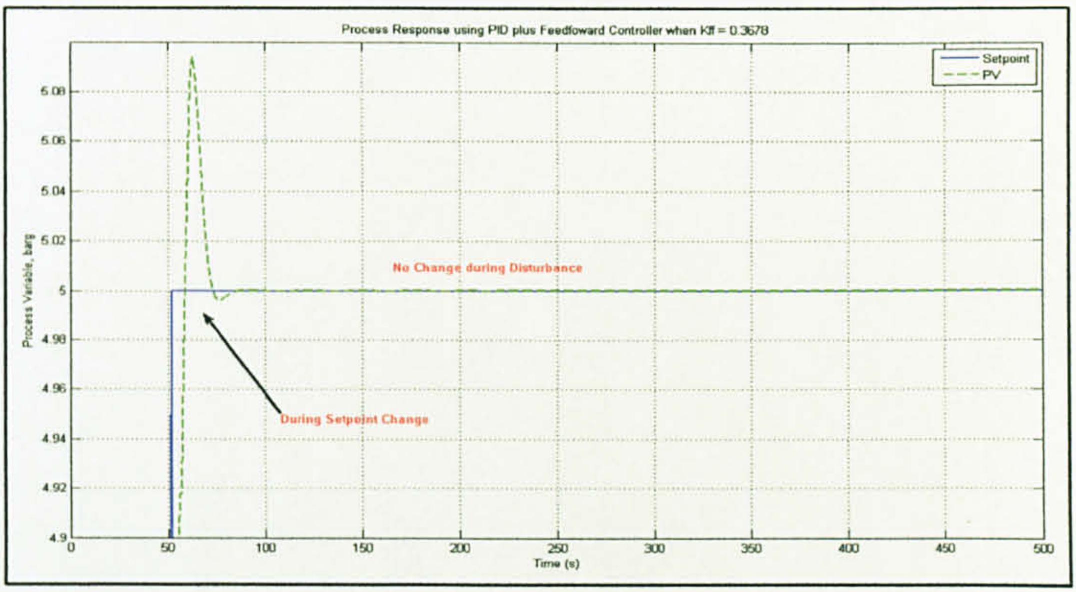


Figure 33: Process Variable Response when Kff = 0.3678

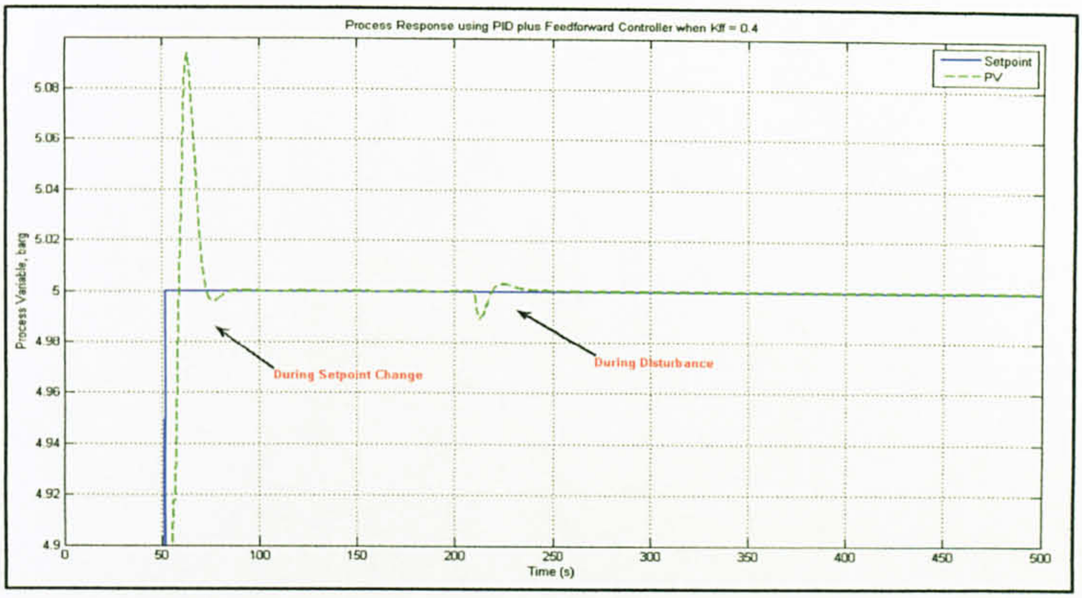


Figure 34: Process Variable Response when Kff = 0.4

From the figures above, it is observed that at *Kff* = 0.2, the PV response showed that when the disturbance is injected, the feedforward controller manage to eliminate the disturbance error by a small percent compared to when *Kff* = 0.4, which showed that the disturbance error managed to disrupt the PV response a little before response went back to steady state. When *Kff* = 0.3678, the feedforward controller has managed to fully eliminate the effects of disturbance injection without bringing any disruptions to the PV response.

This proved the importance of approximate calculations of process and disturbance models since the feedforward algorithm depended solely on the models to get the most desired controller performance.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The PID plus Feedforward controller is one of the choice used in an industrial process plant, however, a thorough study to improve this control scheme and strategy still need to be explored. Understanding how to apply it on a plant process apart from studying its features, design and improving control performance is essential. The process targeted is a gas process and it mainly focused on pressure control on a gaseous pilot plant. The understanding of the controllers and the protocol eases the understanding on how a process control work and how it can be manipulated for various purposes. The methodology covers the empirical model identification to select the suitable parameters for the PID plus Feedforward controller which involved calculation of process and disturbance models, data analysis, controller tuning and performance. Furthermore, real-time implementation could be conducted to evaluate the performance and viability of the approach by making a few adjustments and tuning on the controller.

From the analysis of controller performance and process response, it is evaluated that the PID plus Feedforward control could substantially improve control performance with model accuracy. The implementation of fine-tuning had been necessary because of errors in the process model and simplifications in the tuning method. Furthermore, this study has shown that feedforward is applied when feedback control does not provide satisfactory control performance.

5.2 Recommendations for Future Work

Future work should be as follows;

- Feedforward control involves a new algorithm for which there is no accepted standard display used in commercial equipment. Since the feedforward controller responds to disturbances, it has no set point – a factor that changes the display significantly. One feature that should be provided in the display is the ability to turn the feedforward and feedback on PID on and off separately.
- The operator should have a display of the result after the feedforward and feedback signal have been combined, because the operator would always want to know the signal sent to the final control element.
- The work presented in this report could still be improved where it is recommended to further refine the approach of the studied controller that can later be implemented in real time.

REFERENCES

[1] Thomas E. Marlin, *Process Control: Designing Processes and Control Systems for Dynamic Performance, 2nd Edition,* McGraw Hill, 2000.

[2] Antonio Visioli, *Practical PID Control - Advances in Industrial Control, Use of a Feedforward Action*, Springer, 2006.

[3] K. Åström and T. Hägglund, *PID Controllers: Theory, Design, and Tuning, 2nd Edition*, Instrument Society of America, 1995.

[4] Antonio Visioli (2003), *A new design for a PID plus Feedforward controller,* Dipartimento di Elettronica per l'Automazione, University of Brescia, Italy.

[5] Antonio Visioli, *Practical PID Control - Advances in Industrial Control, Basics of PID Control*, Springer, 2006.

[6] Plant Process Control Systems Lecture, *Topic on Feedforward and Flow Ratio Control*, Ms. Suhaila Badarol Hisham, 2008

APPENDIX: GANTT CHART FOR SECOND SEMESTER OF FINAL YEAR PROJECT (JAN 2009)