# CONTROLLING A GAS PLANT USING STATE SPACE APPROACH

By

DANIEL YEO REN WEI

DISSERTATION

Submitted to the Electrical & Electronics Engineering Programme
in Partial Fulfilment of the Requirements
for the Degree
Bachelor of Engineering (Hons.)
(Electrical & Electronics Engineering)

DECEMBER 2009

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

## CERTIFICATION OF APPROVAL

**CONTROLLING A GAS PLANT**
**USING STATE SPACE APPROACH**

by

Daniel Yeo Ren Wei

A project dissertation 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:

_____________________________

Assoc. Prof Dr. Nordin Saad
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

December 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.

---

Daniel Yeo Ren Wei

# ABSTRACT

Monitoring and controlling of a plant's process variables can be carried out using the computer via software in the instrumentation and control industry. Control systems oil and gas industries are designed using the conventional approach, as opposed to the modern control approach commonly used in aerospace industries, thus controller qualities such as robustness, optimality and adaptivity could have been overlooked. This project aims to theoretically apply the concepts of modern control engineering in plant process control systems. The entire project involves understanding process control and state space, grasping the concept of system identification and mastering the functions of MATLAB and Simulink for controller and observer design and simulation. Extensive utilization of MATLAB and Simulink were involved in the several experiments and simulations that were carried out and run. Results from this project indicate the practicality of modern control in plant process control systems. This project successfully achieved the theoretical implementation of modern control engineering in plant process control systems, paving way for a possible design of a new controller and observer strategy that are robust, optimal and adaptive via modern control approach.

## ACKNOWLEDGMENTS

My appreciation to my project supervisor Assoc. Prof. Dr. Nordin Saad, for taking me under his wing and providing me all the necessary assets and resources, not only to accomplish this project, but to enrich my character and knowledge further. My gratitude towards En. Azhar bin Zainal Abidin, for always keeping his door open. My utmost appreciation and gratitude is also extended to Mr. Tri Chandra S. Wibowo, for the dedication of his time and effort, relentlessly teaching and guiding me despite his many other obligations. Many thanks to my family back home for their sacrifices coupled with their continuous encouragement and support and heading me towards the stars. Special thanks to all the members of the Electrical and Electronics Engineering Department, for establishing continuous support and backing me up. My appreciation is also extended to my friends and everyone who encouraged and supported me throughout the successful completion of this project.

## TABLE OF CONTENTS

**ABSTRACT** ........................................ **iv**

**ACKNOWLEDGMENTS** ........................................ **v**

**LIST OF TABLES** ........................................ **viii**

**LIST OF FIGURES** ........................................ **ix**

**LIST OF ABBREVIATIONS** ........................................ **xi**

**CHAPTER 1 INTRODUCTION** ........................................ **1**

1.1 Background of Study ........................................ 1

1.2 Problem Statement ........................................ 3

1.3 Objectives ........................................ 4

1.4 Scope of Study ........................................ 4

**CHAPTER 2 LITERATURE REVIEW** ........................................ **5**

2.1 Process Control ........................................ 5

2.2 Mechanism of Process Control ........................................ 7

2.3 State-Space Representation ........................................ 9

2.4 System Identification ........................................ 10

2.5 System Identification Toolbox ........................................ 15

**CHAPTER 3 METHODOLOGY** ........................................ **20**

3.1 Procedure Identification ........................................ 20

3.2 Tools and Equipment ........................................ 21

**CHAPTER 4 RESULTS AND DISCUSSION** ........................................ **22**

4.1 Experimental Data ........................................ 22

4.2 Model Estimation and Validation ........................................ 24

4.3 Controller and Observers Design ........................................ 26

4.4 Simulation of Controller and Observers.................................. 28
4.4.1 Open-Loop System ........................................................ 30
4.4.2 Open-Loop System with Disturbance............................. 32
4.4.3 Closed-Loop System with Disturbance ........................... 33
4.4.4 Closed-Loop System with Disturbance and Set Point..... 37
4.4.5 Closed-Loop System with Disturbance, SP & Integrator 42

**CHAPTER 5 CONCLUSION AND RECOMMENDATIONS ...................... 48**
5.1 Recommendations .................................................................. 48
5.2 Conclusion................................................................................ 49

**REFERENCES....................................................................................... 50**

**APPENDICES ......................................................................................... 52**
Appendix A Project Flowchart...................................................... 53
Appendix B Project Gantt Chart ................................................... 54
Appendix C Block Diagram of Gas Plant ..................................... 55
Appendix D Simulink Model for Experiments .............................. 56
Appendix E M-File: daniel_datalog_1.m...................................... 57
Appendix F M-File: daniel_datalog_2.m ...................................... 58
Appendix G Computing Gains for Controller & FS Observer....... 59
Appendix H Computing Gain for Reduced-Order Observer ......... 60

# LIST OF TABLES

**Table 1** Actions to be taken in each session of system identification [12] .......... 11

**Table 2** Phases of the black box modelling technique [2].................................... 13

**Table 3** Transfer function, poles and state space parameters of estimated model 24

**Table 4** Assumed and chosen poles..................................................................... 26

**Table 5** Controller and observer gains.................................................................. 26

# LIST OF FIGURES

**Figure 1** Block diagram of a plant process control system .................................. 1
**Figure 2** Valves and sensors in plant control systems.......................................... 2
**Figure 3** Flow diagram of a process control in a plant.......................................... 7
**Figure 4** Example of feedback control in a process plant .................................... 8
**Figure 5** State-space representation of a plant control system ............................. 9
**Figure 6** Response of a working control system .................................................. 9
**Figure 7** Modelling steps of the black box model [2] ......................................... 14
**Figure 8** The System Identification Toolbox ...................................................... 16
**Figure 9** Plot of input data (from PCV212) ....................................................... 23
**Figure 10** Plot of output data (from PT212)....................................................... 23
**Figure 11** Measured and simulated model output ............................................... 25
**Figure 12** Pole selections for closed-loop systems ............................................. 27
**Figure 13** Full-state observer (constructed in subsystem)................................... 28
**Figure 14** Reduced-order observer (constructed in subsystem)........................... 28
**Figure 15** Open-loop system ................................................................................ 30
**Figure 16** Output of *PT212* ................................................................................ 31
**Figure 17** Output of *Scope6* ................................................................................ 31
**Figure 18** Open-loop system with disturbance.................................................... 32
**Figure 19** Output of *Scope2* ................................................................................ 32
**Figure 20** Closed-loop system with disturbance ................................................. 33
**Figure 21** Output of *Scope1* (full-state observer used) ....................................... 34
**Figure 22** Output of *Scope1* (reduced-order observer used) ............................... 35
**Figure 23** Open-loop vs. closed-loop .................................................................. 36
**Figure 24** Closed-loop system with disturbance and set point............................. 37
**Figure 25** Output of *Scope5* (full-state observer used) ....................................... 38
**Figure 26** Output of *Scope5* (reduced-order observer used) ............................... 39
**Figure 27** Output of *Scope4* (full-state observer used) ....................................... 40
**Figure 28** Output of *Scope4* (reduced-order observer used) ............................... 41
**Figure 29** Closed-loop system with disturbance, set point and integrator ........... 42

**Figure 30** Output of *Scope9* (full-state observer used) ........................................ 43
**Figure 31** Output of *Scope9* (reduced-order observer used) ................................ 44
**Figure 32** Output of *Scope8* (full-state observer used) ........................................ 45
**Figure 33** Output of *Scope8* (reduced-order observer used) ................................ 46
**Figure 34** System without integrator vs. system with integrator ......................... 47

# LIST OF ABBREVIATIONS

| | |
|---|---|
| **ARX** | Autoregression with Exogenous Signal |
| **CVA** | Concurrent Viterbi with Association |
| **DCS** | Distributed Control System |
| **FORTRAN** | Formula Translating |
| **GUI** | Graphical User Interface |
| **LTI** | Linear Time-Invariant |
| **MOESP** | MIMO Output-Error State Space |
| **OE** | Output-Error |
| **PCV** | Pressure Control Valve |
| **PLC** | Programmable Logic Controller |
| **PT** | Pressure Transmitter |
| **SCADA** | Supervisory Control and Data Acquisition |
| **UTP** | Universiti Teknologi Petronas |
| **VL** | Vessel |

# CHAPTER 1
# INTRODUCTION

This chapter introduces and explains the project topic, "Controlling a Gas Plant Using State Space Approach". A background study on this topic is highlighted followed by the problem statement, objectives and finally the scope of study.

## 1.1 Background of Study

In plant process control systems, a process plant is controlled from a computer via an xPC target communications interface. Thus, monitoring and controlling of the plant's process variables such as pressure, temperature, flow, etc. can be carried out using the computer via software such as MATLAB and Simulink.

The block diagram of a typical plant process control system is shown in **Figure 1**.


**Figure 1** Block diagram of a plant process control system

Control is only made possible if the essential and appropriate equipment e.g. sensors and valves are present within the process plant. These equipment must be ensured to possess a large enough maximum capacity to respond to all possible disturbances e.g. noise.

The involvement of the equipment in plant process control systems is shown in **Figure 2**.

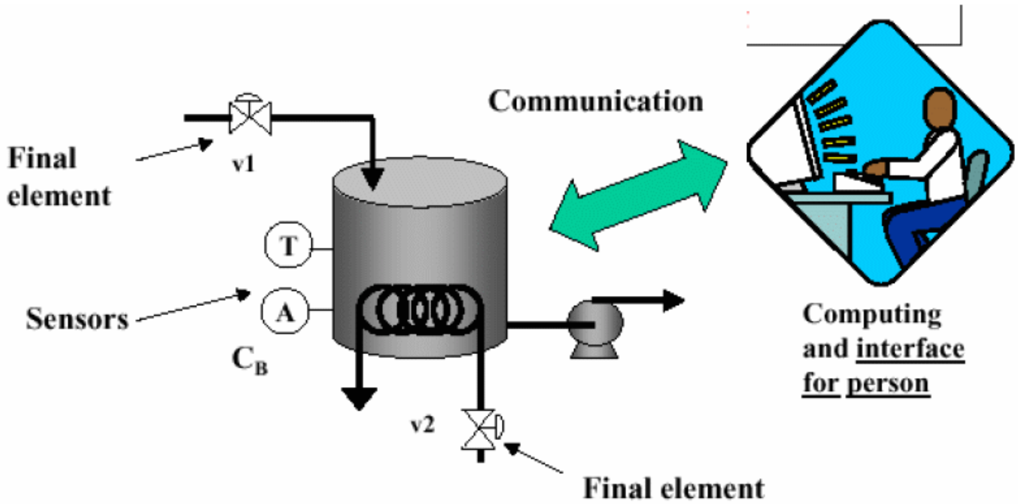



**Figure 2** Valves and sensors in plant control systems

## 1.2 Problem Statement

Today’s oil and gas industries focus only on stability in plant control systems. Other qualities such as robustness, optimality and adaptivity are often overlooked. However, growing demands for consistency in these qualities have urged the development of systems that are not only stable, but also robust, optimal and adaptive.

Furthermore, control systems oil and gas industries are designed using the conventional approach. Thus, the design of a control system using modern control approach, which is commonly used in aerospace industries, is experimented on plant process control systems in this project.

Apart from that, the modelling of plant control systems are presently carried out only using first-order state space equations, providing only limited capabilities in modern plant control. A study on modelling using second-order state space equations is carried out in this project to enable more flexibility in plant control, whereby more states can be controlled.

## 1.3 Objectives

The objectives of this project are:

- To model the control system of a process plant in second-order state space representations
- To design a controller and observer strategy that are robust, optimal and adaptive
- To theoretically apply the concepts of modern control engineering in plant process control systems

## 1.4 Scope of Study

The scope of study of the project covers the following:

- Modelling a plant process control system on MATLAB and Simulink
- Designing and implementing observer and controller strategies for plant processes

# CHAPTER 2
# LITERATURE REVIEW

This section reviews the critical points and theories covered in this project.

## 2.1 Process Control

Process control is a statistics and engineering discipline that deals with architectures, mechanisms, and algorithms for controlling the output of a specific process [18].

For example, increasing the pressure in a gas vessel is a process that has the specific, desired outcome to reach and maintain a defined pressure, kept constant over time. Here, the pressure is the controlled variable. At the same time, it is the input variable since it is measured by a pressure sensor and used to decide whether to increase or decrease the opening of the valves connected to the vessel. The desired pressure is the set point. The state of the valves (e.g. the setting of the valves allowing more gas to flow through it) is called the manipulated variable since it is subject to control actions.

A commonly used control device called a programmable logic controller, or a PLC is used to read a set of digital and analogue inputs, apply a set of logic statements, and generate a set of analogue and digital outputs. Using the example in the previous paragraph, the room temperature would be an input to the PLC. The logical statements would compare the set point to the input temperature and determine whether more or less heating was necessary to keep the temperature constant. A PLC output would then open or close the hot water valve, an incremental amount, depending on whether more or less hot water was needed. Larger, more complex systems can be controlled by a DCS or SCADA system.

Process control systems can be characterized as one or more of the following forms:

- **Batch**: Some applications require that specific quantities of raw materials be combined in specific ways for particular durations to produce an intermediate or end result. An example is the production of adhesives and glues, which normally require the mixing of raw materials in a heated vessel for a period to form a quantity of product. Other important examples are the production of food, beverages and medicine. Batch processes are generally used to produce a relatively low to intermediate quantity of product per year (a few pounds to millions of pounds).

- **Continuous**: Often, a physical system is represented though variables that are smooth and uninterrupted in time. The control of the water temperature in a heating jacket is an example of continuous process control. Some important continuous processes are the production of fuels, chemicals and plastics. Continuous processes, in manufacturing, are used to produce very large quantities of product per year (millions to billions of pounds).

- **Discrete**: Found in many manufacturing, motion and packaging applications. Robotic assembly can be characterized as discrete process control. Most discrete manufacturing involves the production of discrete pieces of product, such as metal stamping.

Applications having elements of discrete, batch and continuous process control are often called hybrid applications.

## 2.2 Mechanism of Process Control

Measurements are made via sensors, in which the analogue signals are converted into digital for plant control. A controller then examines the error present to determine the amount of actions required to be taken. The inputs of measured variable and desired value for the variable are necessary, in which all these control operations are performed via computers using software such as MATLAB and Simulink.

The final element e.g. valves, pumps and motors exert direct influence on the process and bring the controlled variable into the desired value by accepting inputs from the controller before converting and performing proportional operation on the process plant.

The flow diagram of the entire process is shown in **Figure 3**.

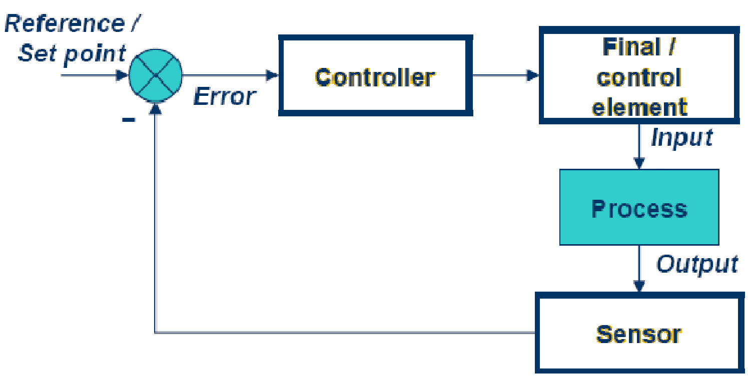



**Figure 3** Flow diagram of a process control in a plant

The desired conditions e.g. pressure and temperature in the process plant is maintained by adjusting selected variables in the system. Such feedback control is carried out by deviating outputs of the system in order to influence an input (correction) back into the system.

An example of a feedback control that can be carried out in maintaining the pressure in a process plant is shown in **Figure 4**.

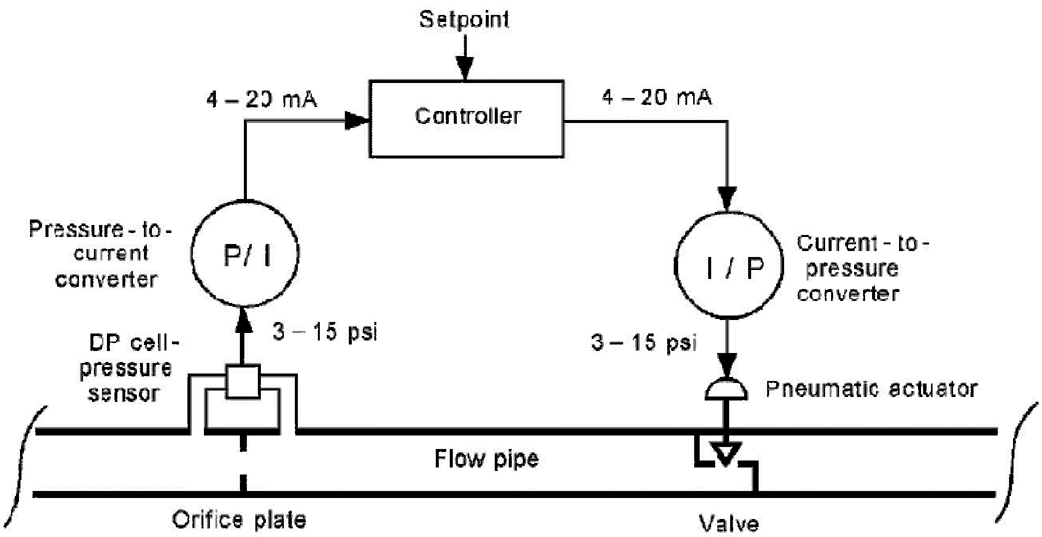



**Figure 4** Example of feedback control in a process plant

## 2.3 State-Space Representation

In the state-space representation of a plant control system, the entire system is expressed in the form shown in **Figure 5**.

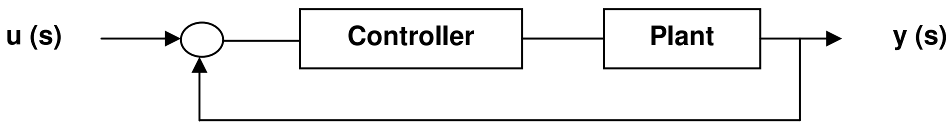



**Figure 5** State-space representation of a plant control system

To model a plant control system using state-space approach, the transfer function of the process plant must first be obtained via the empirical modelling method (using system identification). In this method, the output, *y* is known and experiments are carried out to obtain the transfer function.

With the transfer function obtained, the states are then estimated and finally the state-space equations can be obtained in the following form:

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

A working control system using state-space approach shall result in the following response shown in **Figure 6**.

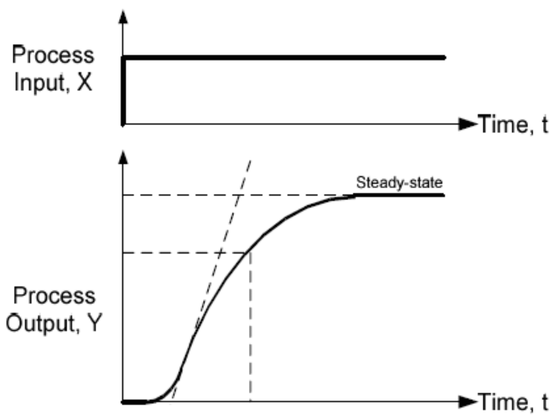



**Figure 6** Response of a working control system

## 2.4 System Identification

System identification is a general term to describe mathematical tools and algorithms that build dynamical models from measured data [19]. A dynamical mathematical model in this context refers to a mathematical description of the dynamic behaviour of a system or process in either the time or the frequency domain. Examples include:

- economic processes such as price hikes that react to external influences.
- physical processes such as the movement of a falling object under the influence of gravity;

A so-called white-box model can be built based on principles, e.g. a model for a physical process from the Newton equations, but such models will be overly complex and likely impossible to obtain in reasonable time due to the complex nature of many systems and processes.

A much more common method is therefore to start from measurements of the behaviour of the system and the external influences (inputs to the system) and try to determine a mathematical relation between them without going into the details of what is actually happening inside the system. This technique is called system identification.

Two types of models are common in the field of system identification:

- **grey box model**: although the peculiarities of what is going on inside the system are not entirely known, a certain model based on both insight into the system and experimental data is constructed. This model does however still have a number of unknown free parameters, which can be estimated using system identification. Grey box modelling is also known as semi-physical modelling.
- **black box model**: No prior model is available. Most system identification algorithms are of this type.

To apply the System Identification method in modelling a system, each identification session will consist of a series of basic steps. Some of them may be hidden or selected without the user being aware of his choice, resulting in poor or suboptimal results.

The actions to be taken in each session are stated in **Table 1**.

**Table 1** Actions to be taken in each session of system identification [12]

| Session | Actions |
|---|---|
| Collect Information about the System | This can be done by observing the natural fluctuations, but it is more efficient to set up dedicated experiments that actively excite the system in which the user has to select an excitation that optimizes his own goal within the operator constraints. The quality of the result can depend heavily on the choices that are made. |
| Select a Model Structure to Represent the System | A choice should be made within all the possible mathematical models that can be used to represent the system. A wide variety of possibilities exist e.g. parametric versus nonparametric models, white box models versus black box models, linear models versus nonlinear models or Linear-in-the-parameters versus nonlinear-in-the-parameters. |
| Match the Selected Model Structure to the Measurements | Once a model structure is chosen, it should be matched with the available information about the system. This is done by minimizing a criterion that measures a goodness of the fit. The choice of this criterion is extremely important because it determines the stochastic properties of the final estimator. Many choices are possible and each of them can lead to a different estimator with its own properties. The cost function defines a distance between the experimental |

| | |
|---|---|
| | data and the model. The cost function can be chosen on an ad hoc basis using intuitive insight, but there also exists a more systematic approach based on stochastic arguments. Simple tests on the cost function exist (necessary conditions) to check even before deriving the estimator whether it can be consistent. |
| Validate the Selected Model | The validity of the selected model should be tested. The best model with the smallest errors is not always preferred. A simpler model that describes the system within user-specified error bounds is preferred. Tools will be provided that guide the user through this process by separating the remaining errors into different classes e.g. unmodeled linear dynamics and nonlinear distortions. Further improvements of the model can be proposed from this information. It is important to keep the application in mind during the validation tests. The model should be tested under the same conditions, as it will be used later. Extrapolation should be avoided as much as possible. The application also determines what properties are critical. |

The black box modelling technique will be implemented in obtaining the equations. The phases are stated in **Table 2**.

**Table 2** Phases of the black box modelling technique [2]

| Phase | Description |
|---|---|
| System analysis | The goals and the requirements of the model are formulated and the boundaries are determined during this phase. It is already necessary to consider the three most important characteristics of a black-box model i.e. level of detail, degree of non-linearity of the process and the order of the dynamics. |
| Data conditioning | The models are based on the data available and form the starting point of the design, thus the data mining and conditioning is a crucial initial step. |
| Key variable analysis | The choice of the input and output variables can be based on process insight or determined by a sensitivity analysis. The input space must not be too large and the variables should be as mutually independent as possible. The number of variables can be reduced by othogonalization. |
| Model structure design | The number of parameters is a good measure of the level of detail that will be obtained. The level of detail should be balanced with the amount and the quality of the data available. |
| Model identification | In this step, the model is fitted to the measured data. The error between model and reality is usually minimized |
| Model evaluation | The model is tested by means of special test data sets to determine whether the model has sufficient capacity to predict stationary and dynamic behaviour. Most black-box models show good interpolation properties. However, their extrapolation qualities are mostly limited and should be tested if required |

The modelling steps of the black box model design are shown in **Figure 7**.

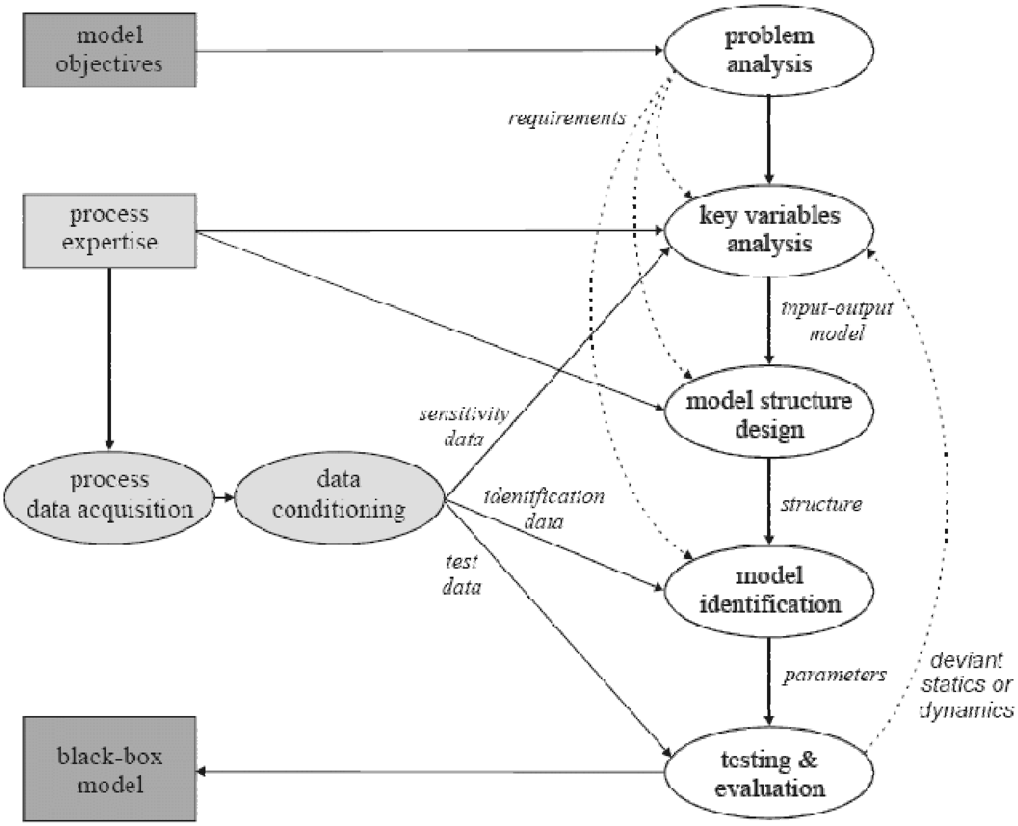



**Figure 7** Modelling steps of the black box model [2]

## 2.5 System Identification Toolbox

MATLAB and Simulink's System Identification Toolbox allows the construction of mathematical models of dynamic systems from measured input-output data [17]. This data-driven approach helps describe systems that are not easily modelled from first principles or specifications, such as chemical processes and engine dynamics. It also helps simplify detailed first-principle models, such as finite-element models of structures and flight dynamics models, by fitting simpler models to their simulated responses.

Models obtained with System Identification Toolbox are well suited for simulation, prediction, and control system design using products such as Simulink, Control System Toolbox, and Model Predictive Control Toolbox.

System Identification Toolbox enables the fitting of linear and nonlinear models to data, a process known as black box modelling. Available model structures include low-order process models, transfer functions, state-space models, linear models with static nonlinearities at the inputs or outputs, and nonlinear autoregressive models. If a mathematical model of the system dynamics is obtained, its parameters can be tuned to better match experimental data, a process known as grey-box modelling.

The principal architect of the toolbox is Professor Lennart Ljung, a recognized leader in the field of system identification.

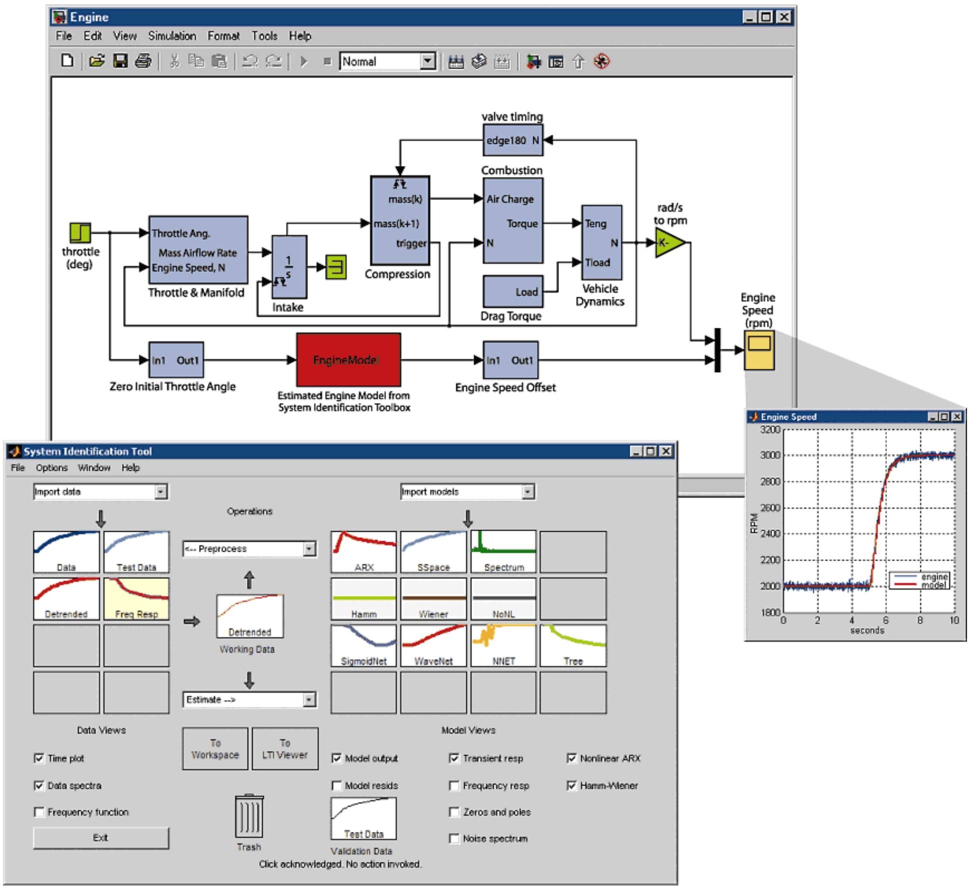



**Figure 8** The System Identification Toolbox

The key features of the System Identification toolbox are as follows:

- Analyzes measured data and advises on data quality, required pre-processing, and presence of feedback or nonlinearities
- Identifies linear and nonlinear models from time- and frequency-domain data
- Provides an interface for using linear models in Control System Toolbox (available separately)
- Provides data processing tools for detrending, filtering, and reconstructing missing data
- Provides Simulink blocks for simulating identified models and transferring data to and from the MATLAB workspace
- Simplifies identification of first-, second-, and third-order continuous-time models

System Identification Toolbox facilitates the multistep process of identifying models from data. The toolbox enables:

- Analysis and processing of data
- Determination of suitable model structure and order
- Estimation of model parameters and validation of model accuracy
- Viewing of the model responses and their uncertainties

These tasks can be performed by using either command-line functions or a graphical user interface (GUI). The models can be converted into linear time-invariant (LTI) objects for use with Control System Toolbox. Most identified models can be incorporated into Simulink models using blocks provided by the toolbox.

When preparing data for identification of models, information such as input-output channel names, sampling interval and intersample behaviour must be specified. The toolbox allowed the attachment of this information to the data using data objects. The data objects facilitate easy visualization of data, domain conversion, and various pre-processing tasks.

Measured data often has offsets, slow drifts, outliers, missing values, and other anomalies. The toolbox removes such anomalies by performing operations such as detrending, filtering, resampling, and reconstruction of missing data. The toolbox can analyze the suitability of data for identification and provide diagnostics regarding persistence of excitation, existence of feedback loops, intersample behaviour, and presence of nonlinearities.

The toolbox produces estimates of step and frequency responses of the system directly from measured data. Using these responses, the system characteristics, such as time constants, input delays, and resonant frequencies, can be analysed. This information can be used to configure the parametric models during estimation.

Parametric models, such as transfer functions or state-space models use a small number of parameters to capture system dynamics. The toolbox estimates model parameters and their uncertainties. Analysis of these models, or their linear equivalents, can be carried out using time- and frequency-response plots such as step, impulse, bode plots, and pole-zero maps.

Polynomial and state-space models can be identified using various estimation routines offered by the toolbox. These routines include autoregressive models (ARX, ARMAX), Box-Jenkins (BJ) models, Output-Error (OE) models, and state-space parameterizations. Estimation techniques include maximum likelihood, prediction error minimization schemes, and such subspace methods as CVA, MOESP, and N4SID. Estimation of a model of the noise affecting the observed system can also be carried out.

In cases where only a low-order continuous-time model is needed, the toolbox provides special capabilities to simplify the estimation process and obtain results quickly. These models are expressed as simple transfer functions involving three or fewer poles, and optionally, a zero, a time-delay, or an integrator.

When linear models are not sufficient to capture system dynamics, nonlinear models, such as nonlinear ARX and Hammerstein-Wiener models can be estimated. Nonlinear ARX models enables the modelling of nonlinearities using wavelet networks, tree-partitioning, sigmoid networks, and neural networks (with Neural Network Toolbox, available separately). Using Hammerstein-Wiener models, static nonlinear distortions present at the input and/or output of an otherwise linear system can be estimated. For example, the user can estimate the saturation levels affecting the input current into a DC motor, or capture a complex nonlinearity at the output using a piecewise linear nonlinearity.

A user-defined (grey-box) model is a set of differential or difference equations with some unknown parameters. The toolbox allows the user to specify the model structure and estimate its parameters using nonlinear optimization techniques. For linear models, the structure of state-space matrices can be specified and constraints on identified parameters can be imposed. For nonlinear models, differential equations as M, C, or FORTRAN code can be specified.

System Identification Toolbox helps validate the accuracy of identified models using independent sets of measured data from a real system. For a given set of input data, comparison between the simulated output of the identified model with the measured output from the real system can be performed. The user can also view the prediction error and produce time- and frequency-response plots with confidence bounds to visualize the effect of parameter uncertainties on model responses.

# CHAPTER 3
# METHODOLOGY

This chapter discusses the project's procedure identification as well as the tools and equipment utilized throughout the course of completing this project.

## 3.1 Procedure Identification

The flowchart and Gantt Chart of the entire project can be found in **Appendices A** and **B**. The project is generally divided into four main phases:

- **Phase 1:** Conduct experiments on gas plant

  Experiment are carried out on the gas plant and all input-output data are recorded to be used to model the system.

- **Phase 2:** Estimate and validate model

  The model of the gas plant's system is then modelled via MATLAB's System Identification Toolbox using the data from the experiments.

- **Phase 3:** Design controller and observers

  The controller and observer poles and gains are determined based on the model parameters.

- **Phase 4:** Simulate controller and observers on Simulink

  The designed controller and observers are then simulated on different system types on Simulink to observe their effects.

## 3.2 Tools and Equipment

The tools and equipment required for this project are:

- MATLAB and Simulink
- Process Plant
- xPC Target Interface

# CHAPTER 4

# RESULTS AND DISCUSSION

This chapter discusses the outcomes of every stage and phase of the project.

## 4.1 Experimental Data

Experiments are carried out on the gas plant located in UTP's Building 23. The entire gas plant is described in the block diagram shown in **Appendix C**.

The pressure of the main vessel (*VL-212* in **Appendix C**) is chosen to be the controlled variable. Therefore, the sensor PT212 is monitored for output data. The opening of outlet valve PCV212 is chosen to be the manipulated variable, and thus monitored for input data.

MATLAB and Simulink are run. The Simulink model for the experiments is constructed as shown in **Appendix D**.

Two M-files (*daniel_datalog_1.m* and *daniel_datalog_2.m*) are run to obtain input and output data for a second-order system. The opening of PCV212 is varied between 30% to 70% while observing and recording the changes in PT212 for a period of 7000 seconds. These M-files can be found in **Appendices E** and **F**.

The plots of the input (PCV212) and the output (PT212) are shown in **Figures 9** and **10**.

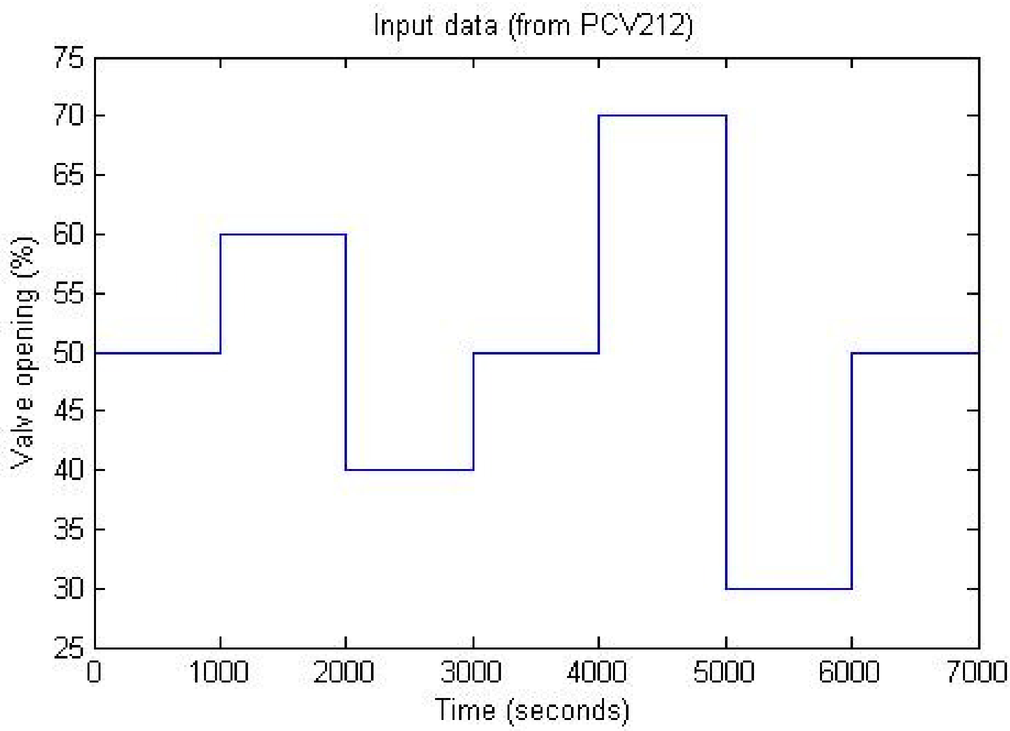



**Figure 9** Plot of input data (from PCV212)

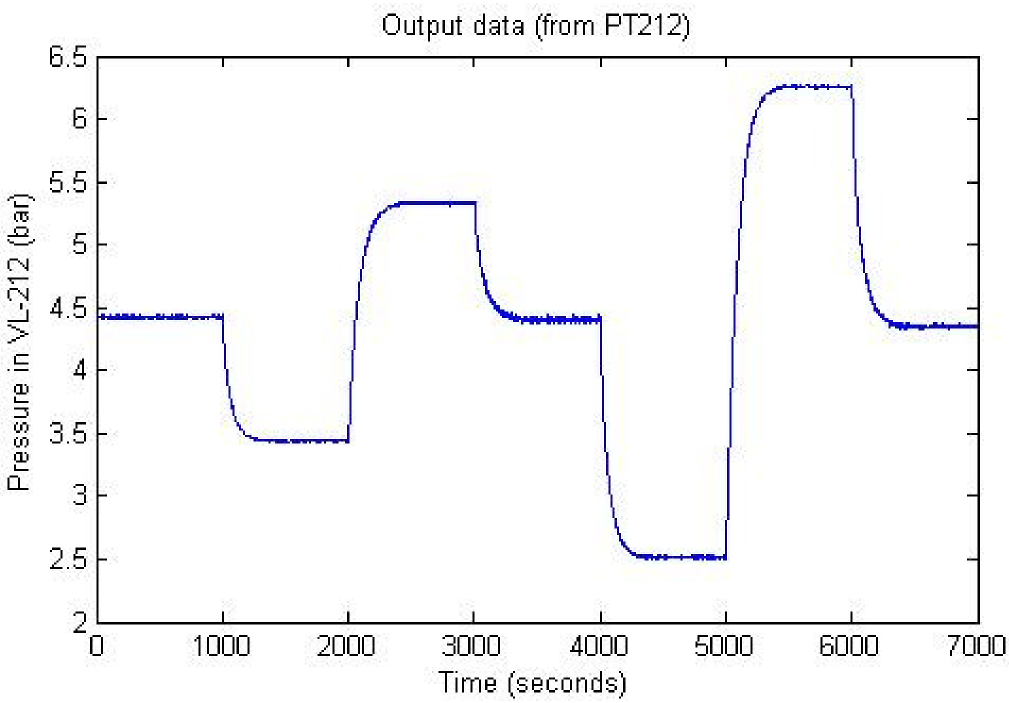



**Figure 10** Plot of output data (from PT212)

## 4.2 Model Estimation and Validation

A second-order transfer function (process model) with complex poles were estimated using the System Identification Toolbox, with a zero and an integrator (self-regulating) excluded from the model. The transfer function, poles and state space parameters of the estimated model are shown in **Table 3**.

**Table 3** Transfer function, poles and state space parameters of estimated model

| Model Parameters | Continuous-Time Model | Discrete-Time Model |
|---|---|---|
| Transfer function | $\dfrac{-0.00193}{s^2+1.701s+0.02039}$ | $\dfrac{-0.0005884z-0.0003376}{z^2-1.173z+0.1825}$ |
| System Poles | -1.6889, -0.0121 | 0.9880, 0.1847 |
| A (cont.) or F (disc.) | $\begin{bmatrix}-1.701 & -0.1631\\ 0.125 & 0\end{bmatrix}$ | $\begin{bmatrix}1.173 & -0.365\\ 0.5 & 0\end{bmatrix}$ |
| B (cont.) or G (disc.) | $\begin{bmatrix}0.125\\ 0\end{bmatrix}$ | $\begin{bmatrix}0.03125\\ 0\end{bmatrix}$ |
| C (cont.) or H (disc.) | $[0 \quad -0.1235]$ | $[-0.01883 \quad -0.0216]$ |
| D (cont.) or J (disc.) | [0] | [0] |

The simulated output is plotted with relative to the measured output, giving a best-fit value of 94.77% (determined from System Identification Toolbox). The simulated and measured output are shown in **Figure 11**.

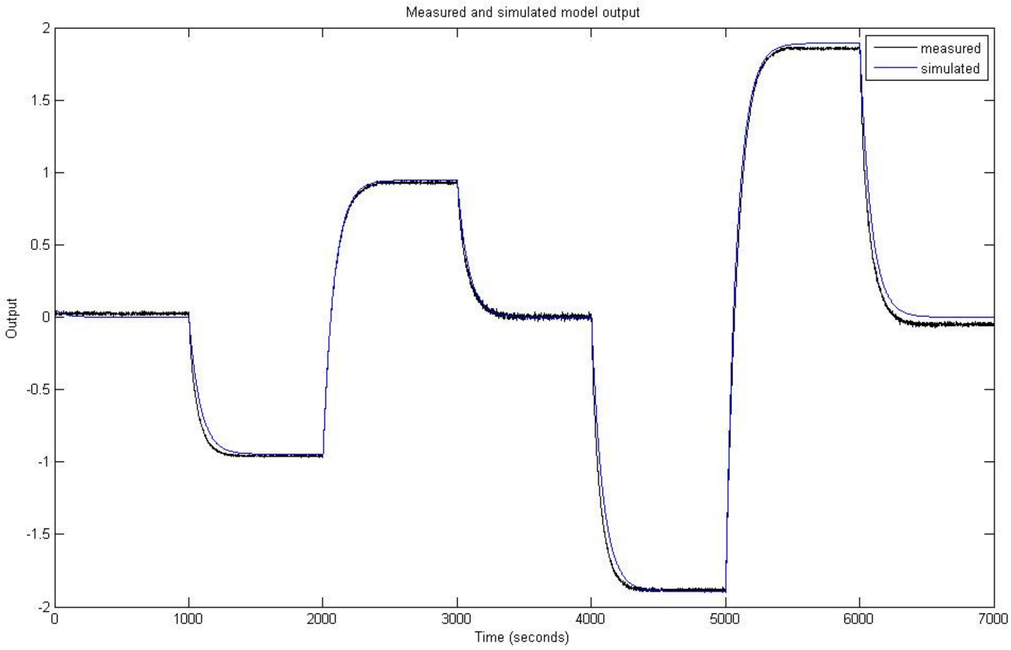



**Figure 11** Measured and simulated model output

## 4.3 Controller and Observers Design

In order to design controller and observers using pole placement method, the full-state feedback controller closed-loop poles are chosen to be further from 0 the system poles are (note: using continuous-time model). On the other hand, to design full-state and reduced-order observers with faster response, the observer closed-loop poles for both types are chosen to be further from zero than the controller's ones are.

Based on the data in **Table 3**, the following closed-loop poles are assumed and chosen for the controller and observers:

**Table 4** Assumed and chosen poles

| Type | Poles |
|---|---|
| Controller | -1.7, -0.1 |
| Full-state observer | -1.8, -0.2 |
| Reduced-order observer | -1.8 |

**Figure 12** illustrates the pole selections for the closed-loop systems. From the data in **Table 4**, the controller and observer gains are obtained as shown is **Table 5**. The steps taken to obtain the gains are shown in **Appendices G** and **H**.

**Table 5** Controller and observer gains

| Gain Type | Poles |
|---|---|
| Controller, K | [0.7920 9.5752] |
| Full-state observer, L1 | $\begin{bmatrix} 10.9465 \\ -2.4211 \end{bmatrix}$ |
| Reduced-order observer, L2 | -6.413 |

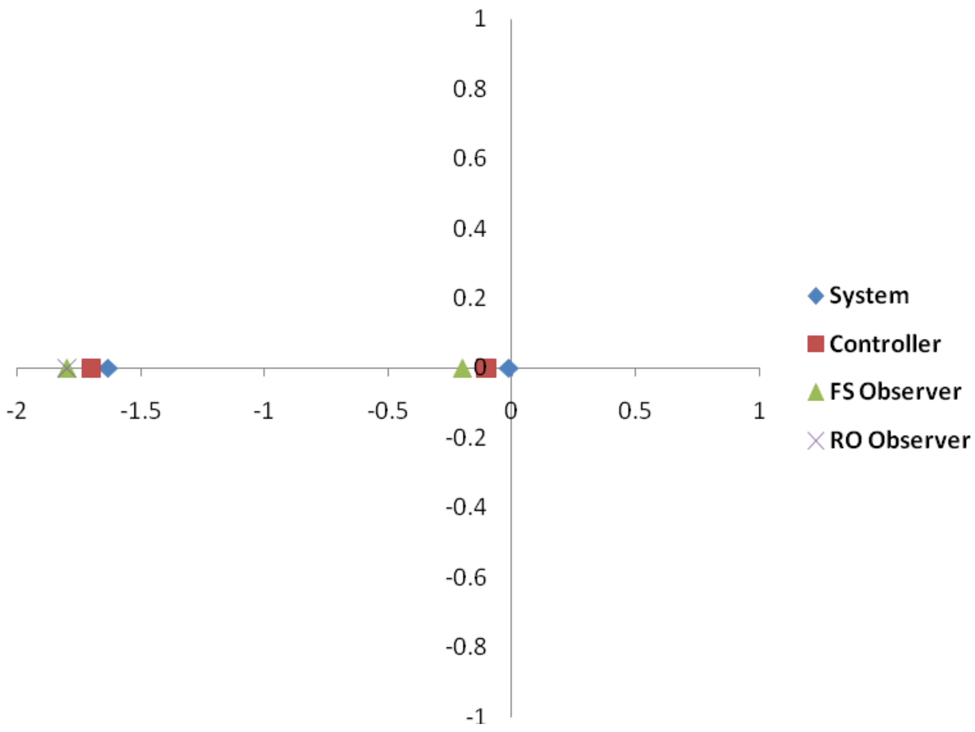



**Figure 12** Pole selections for closed-loop systems

## 4.4 Simulation of Controller and Observers

To simulate the controller and full-state observer on Simulink, the observer shown in **Figure 13** is constructed in a subsystem.

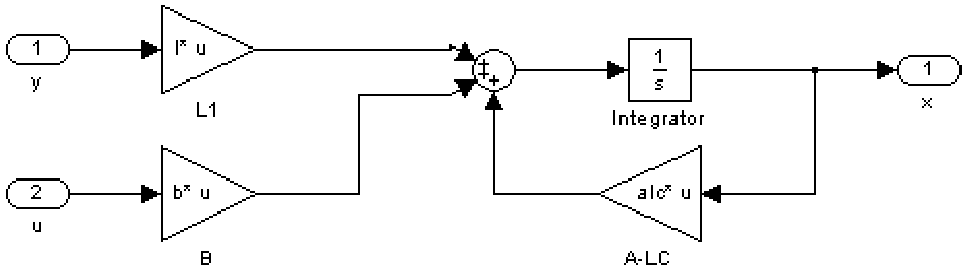



**Figure 13** Full-state observer (constructed in subsystem)

To simulate the controller and reduced-order observer on Simulink, the observer shown in **Figure 14** is constructed in the subsystem instead.

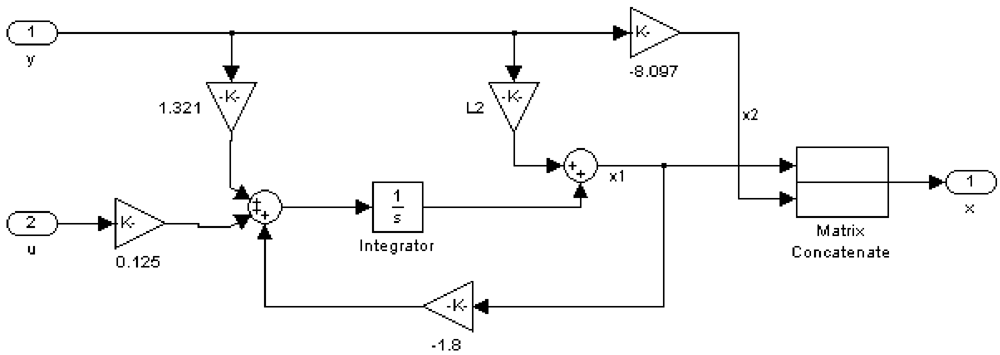



**Figure 14** Reduced-order observer (constructed in subsystem)

These subsystems are then used in all the block diagrams shown in the subsections to follow. The values of the model parameters and gains used in all Simulink block diagrams are taken from **Sections 4.2** and **4.3**.

The controller and observers are simulated for a period of 600 seconds for the following system types:

- Open-loop
- Open-loop with disturbance
- Closed-loop with disturbance
- Closed-loop with disturbance and set point
- Closed-loop with disturbance, set point and integrator

### *4.4.1 Open-Loop System*

As the midpoint value for PCV212, i.e. 50% is chosen to be the initial condition (with its corresponding output in PT212 of approximately 4.4 bar); any input into the system is first deducted by a value of 50 before proceeding into the transfer function. The output of the transfer function is then added with a value of 4.4 in order to obtain the value of PT212 from its corresponding PCV212 input. The block diagram in shown **Figure 15** is constructed to simulate the open-loop system.

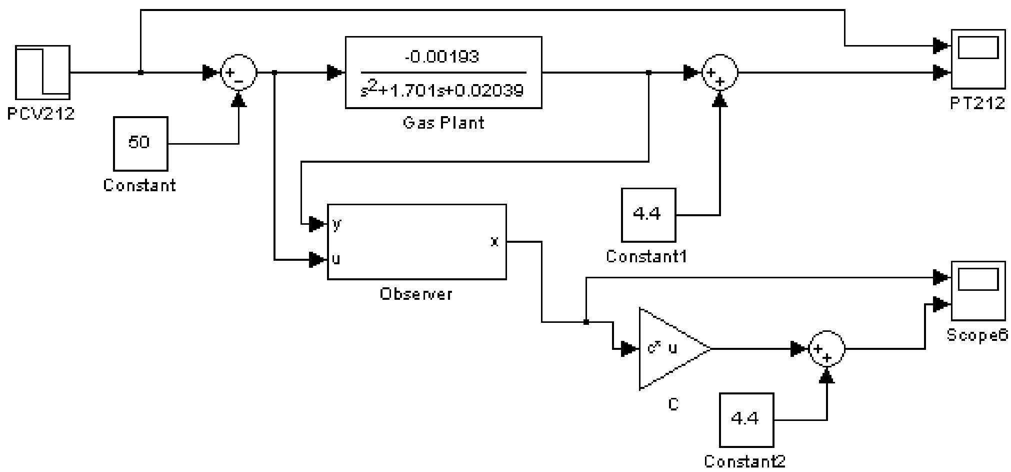



**Figure 15** Open-loop system

A step input with an initial value of 50% is introduced to the system before reducing to 30% after 10 seconds. The corresponding output of scopes *PT212* and *Scope6* are shown in **Figure 16** and **17**.

The output value of PT212 for an initial PCV212 input of 50% is approximately 4.4 bar. When the input changes to 30%, the output settles to a value of approximately 6.3 bar. These values are similar to the values shown in **Figures 9** and **10** earlier, thus proving that the obtained transfer function matches the behaviour of the plant and is thus feasible to be used for simulation. The observer's action is shown in **Figure 17**, estimating both the state variables and matching output Y.

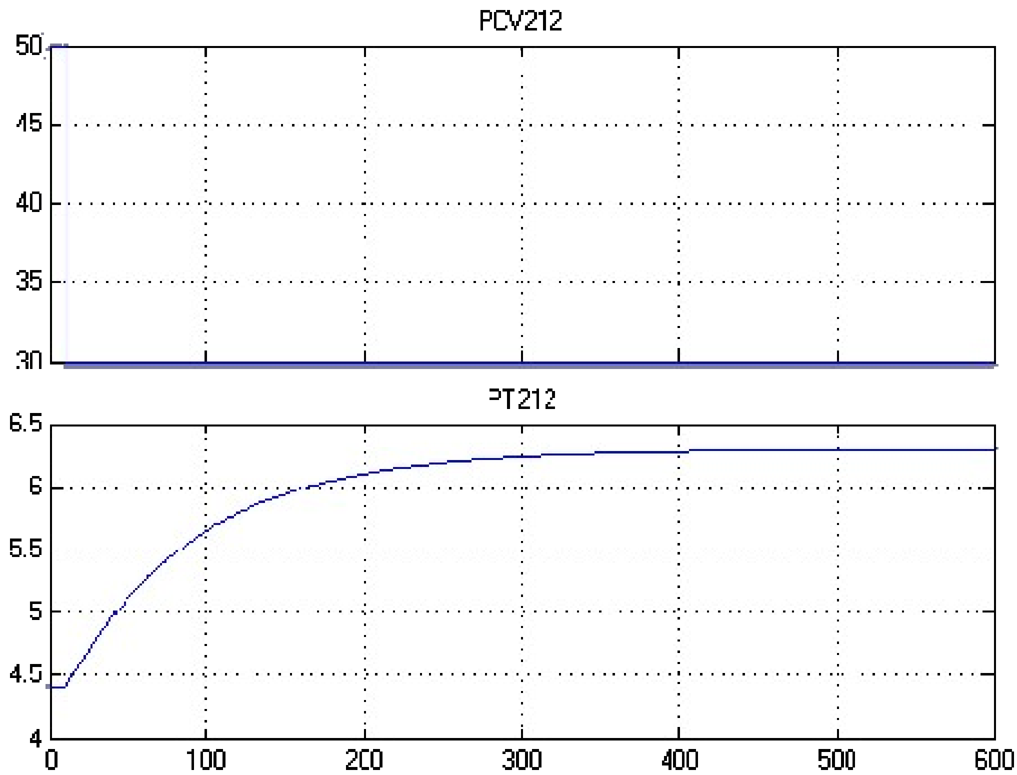



**Figure 16** Output of *PT212*

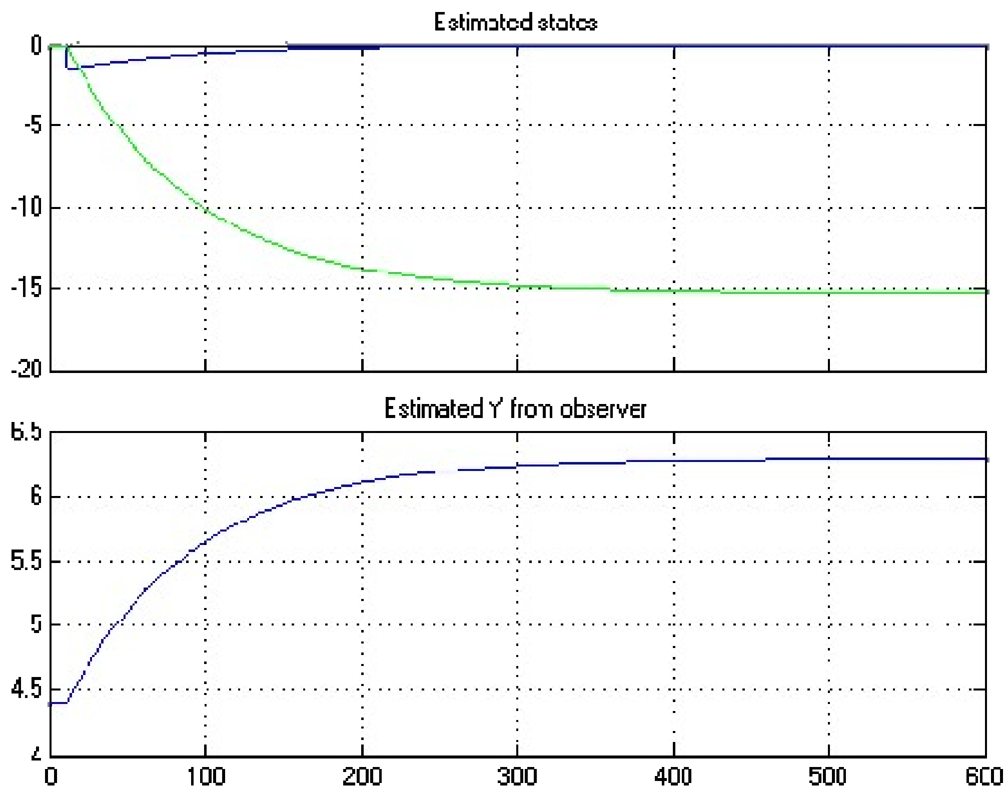



**Figure 17** Output of *Scope6*

### *4.4.2 Open-Loop System with Disturbance*

With zero input to the system, a pulse disturbance of 0.1 for a period of 100 seconds is inserted into the system after 20 seconds. The constructed block diagram for the open-loop system with disturbance is shown in **Figure 18**.

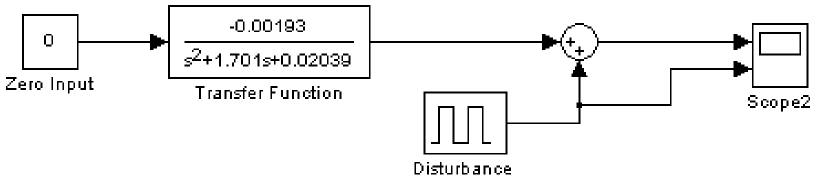



**Figure 18** Open-loop system with disturbance

The output of *Scope2* is shown in **Figure 19**. The output shows that the disturbance is added to the output of the transfer function from the 20$^{th}$ to the 120$^{th}$ second, producing an erroneous system output for that time period due to the absence of a controller and observer.

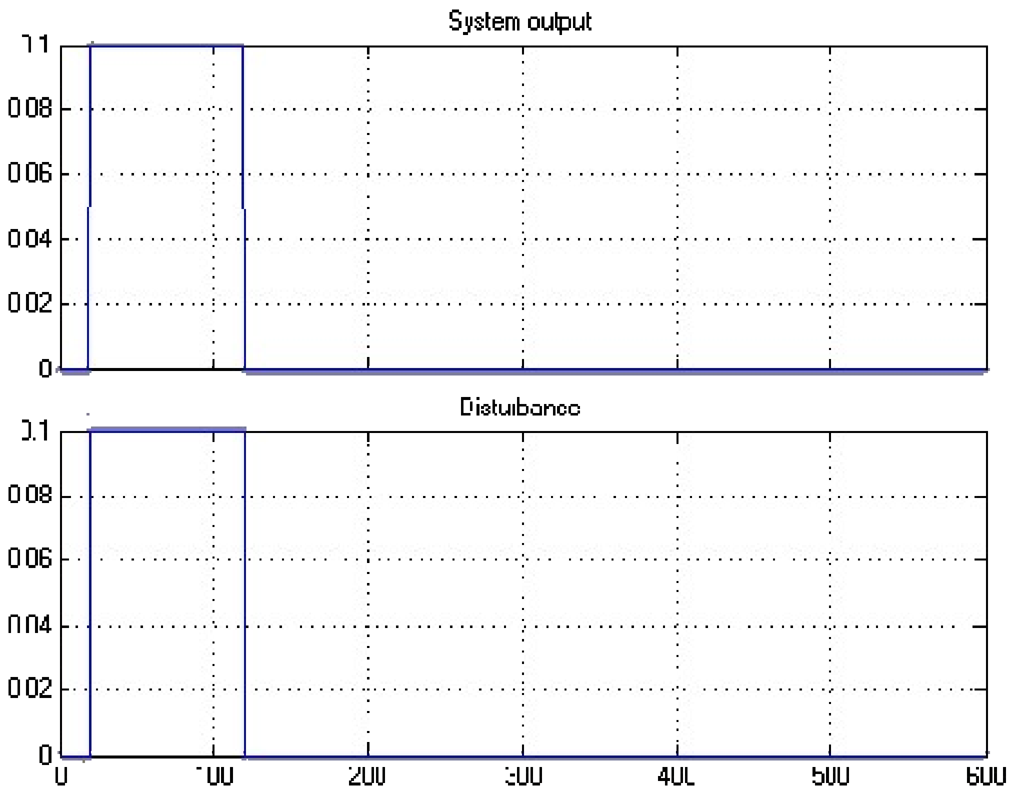



**Figure 19** Output of *Scope2*

### *4.4.3 Closed-Loop System with Disturbance*

A system similar to the system in **Subsection 4.4.2** is constructed, but with the exception that an observer and controller are included in the system. The block diagram in shown **Figure 20** is constructed to simulate the closed-loop system with disturbance.

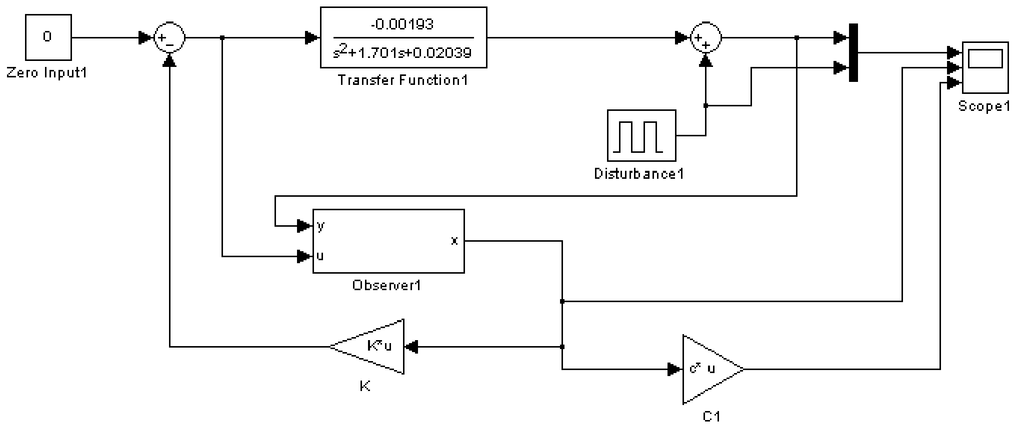



**Figure 20** Closed-loop system with disturbance

The output of *Scope1* is shown in **Figures 21** (when the full-state observer is used in the system) and **22** (when the reduced-order observer is used instead). The output shows the effects of the controller and observer on the system output during the presence of the disturbance. When the disturbance is added to the system from the 20$^{th}$ to the 120$^{th}$ second, the controller and observer attempt to reduce the output value back to 0, with the presence of a slight offset.

**Figure 23** shows the effects of a controller and observer when the disturbance is present on a system with (closed-loop) and without (open-loop) the pair.

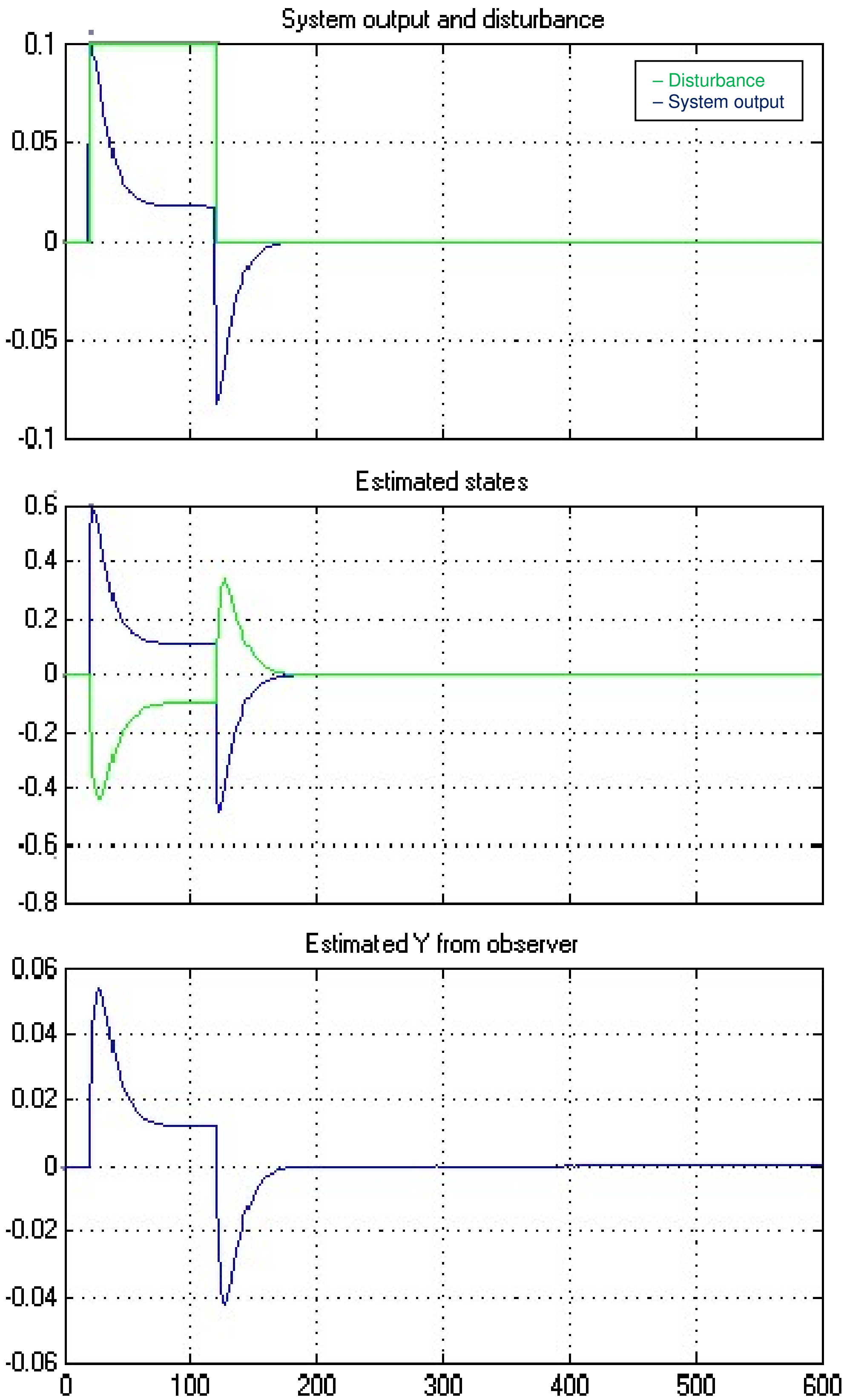



**Figure 21** Output of *Scope1* (full-state observer used)

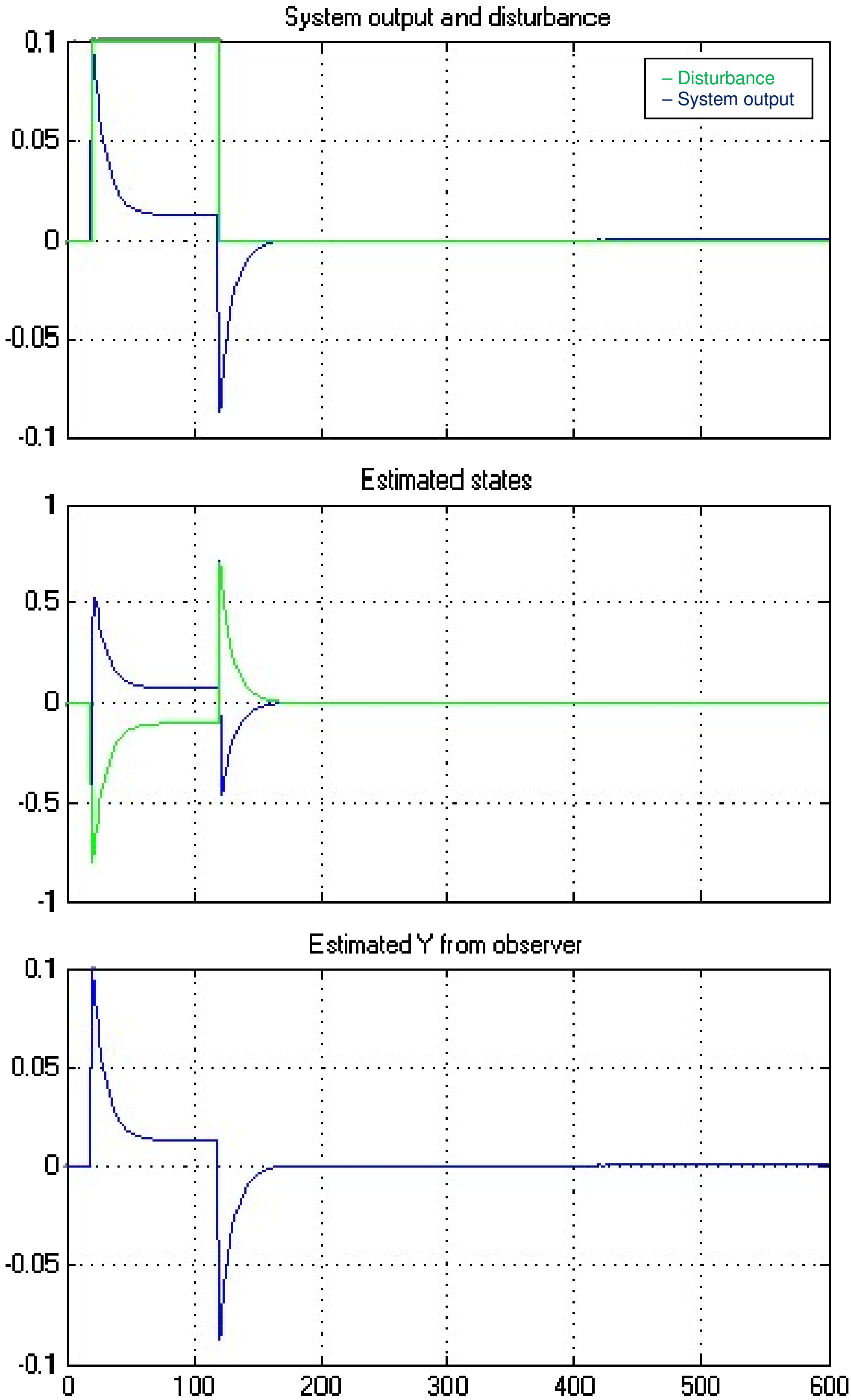



**Figure 22** Output of *Scope1* (reduced-order observer used)

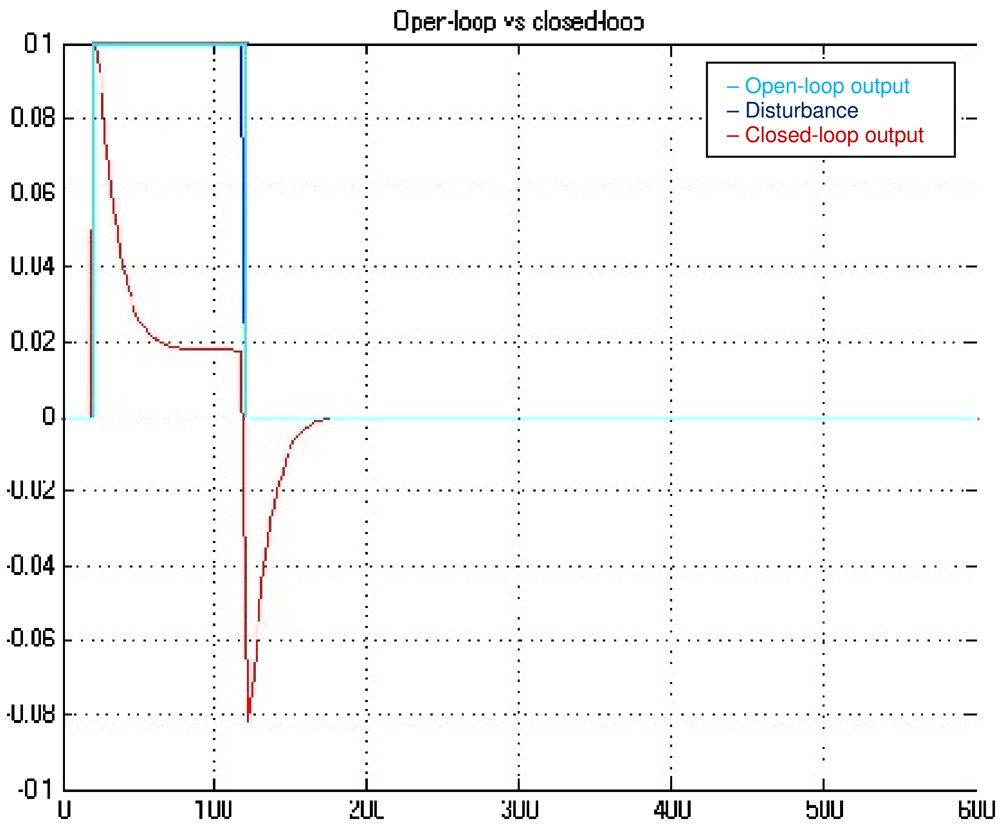



**Figure 23** Open-loop vs. closed-loop

### *4.4.4 Closed-Loop System with Disturbance and Set Point*

A system similar to the system in **Subsection 4.4.3** is constructed, but a set point value is given to the system and practical PCV212 values and PT212 values are utilized instead, thus simulating an almost actual scenario on Simulink. A disturbance of 0.5 is inserted into the system after 200 seconds for a period of 100 seconds.

The constructed block diagram for the closed-loop system with disturbance and set point is shown in **Figure 24**.

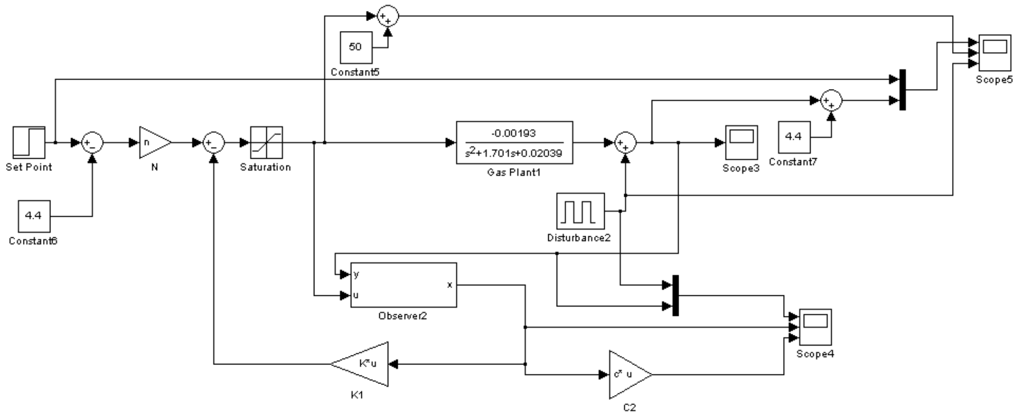



**Figure 24** Closed-loop system with disturbance and set point

The initial set point value is set at 4.4 bar, before increasing to 5.5 bar after 20 seconds. **Figures 25** and **27** show the outputs of *Scope5* and *Scope4* respectively when the full-state observer is used in the system. **Figures 26** and **28** show the outputs when the reduced-order observer is used instead.

The outputs of the scopes show that when the set point is changed from 4.4 bar to 5.5 bar, the values of PCV212 and PT212 change correspondingly. During the presence of the disturbance, the controller and observer are virtually successful in bringing the system output value back down to the desired set point, with the presence of a slight offset.

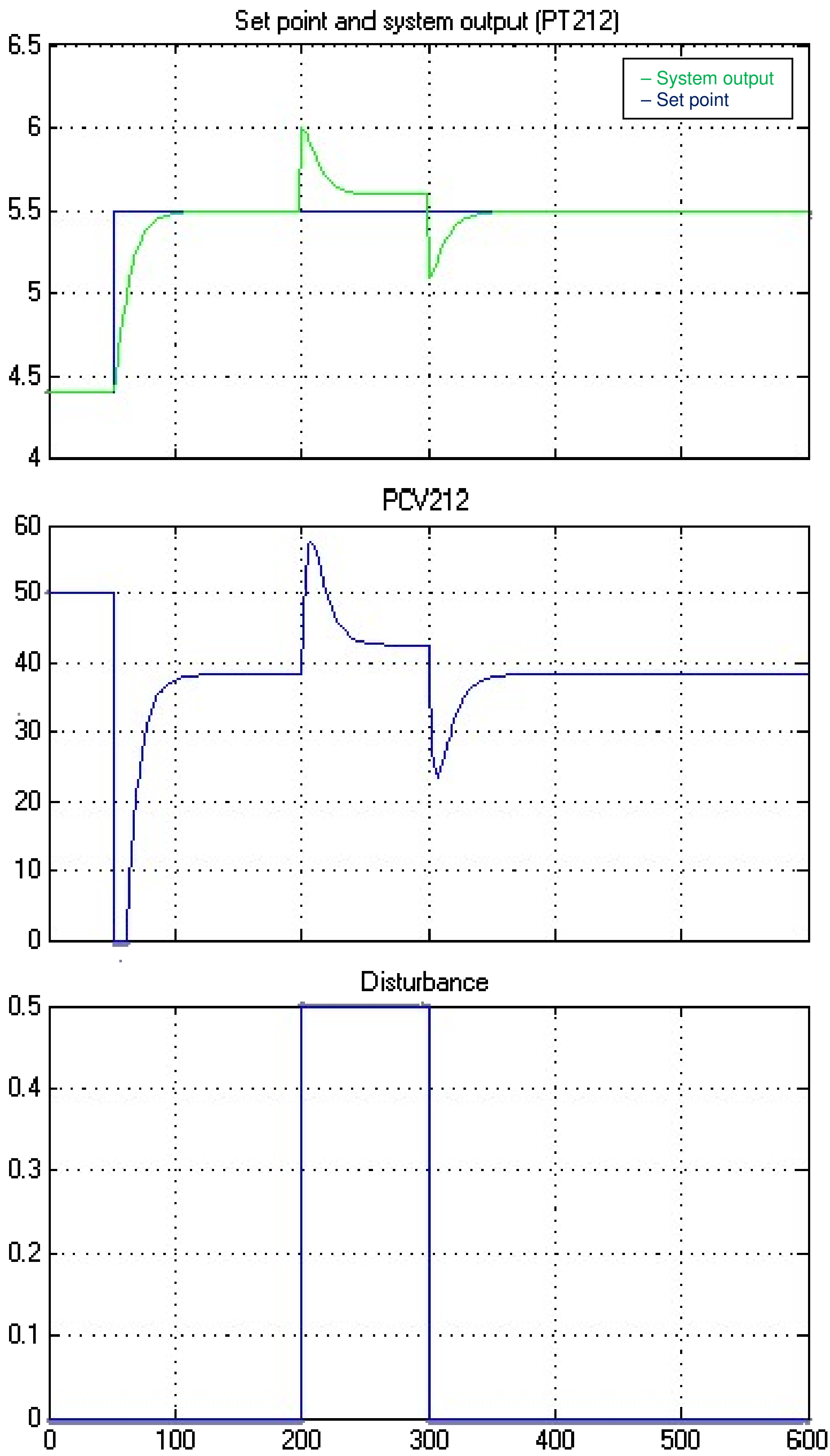



**Figure 25** Output of *Scope5* (full-state observer used)

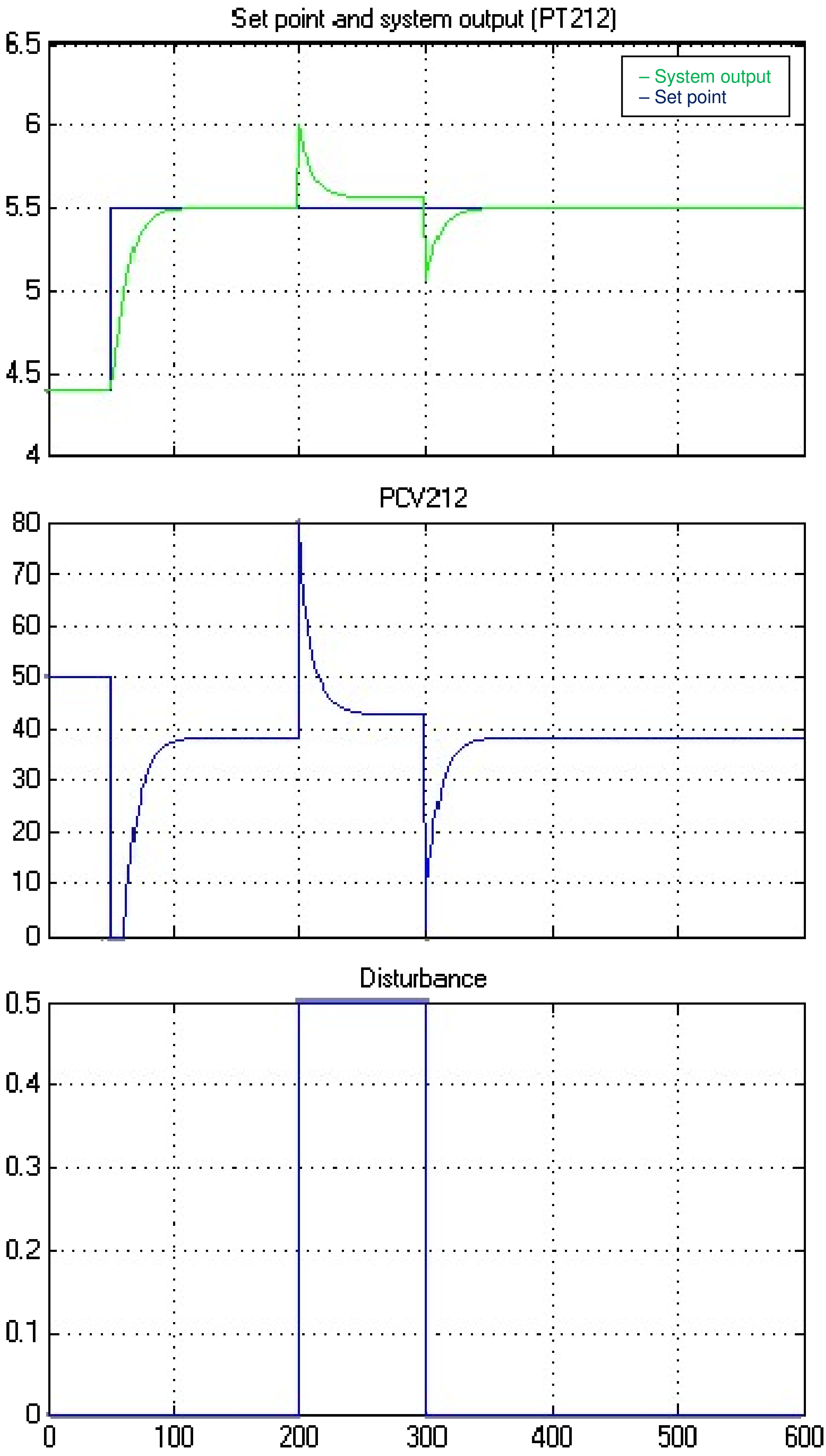



**Figure 26** Output of *Scope5* (reduced-order observer used)

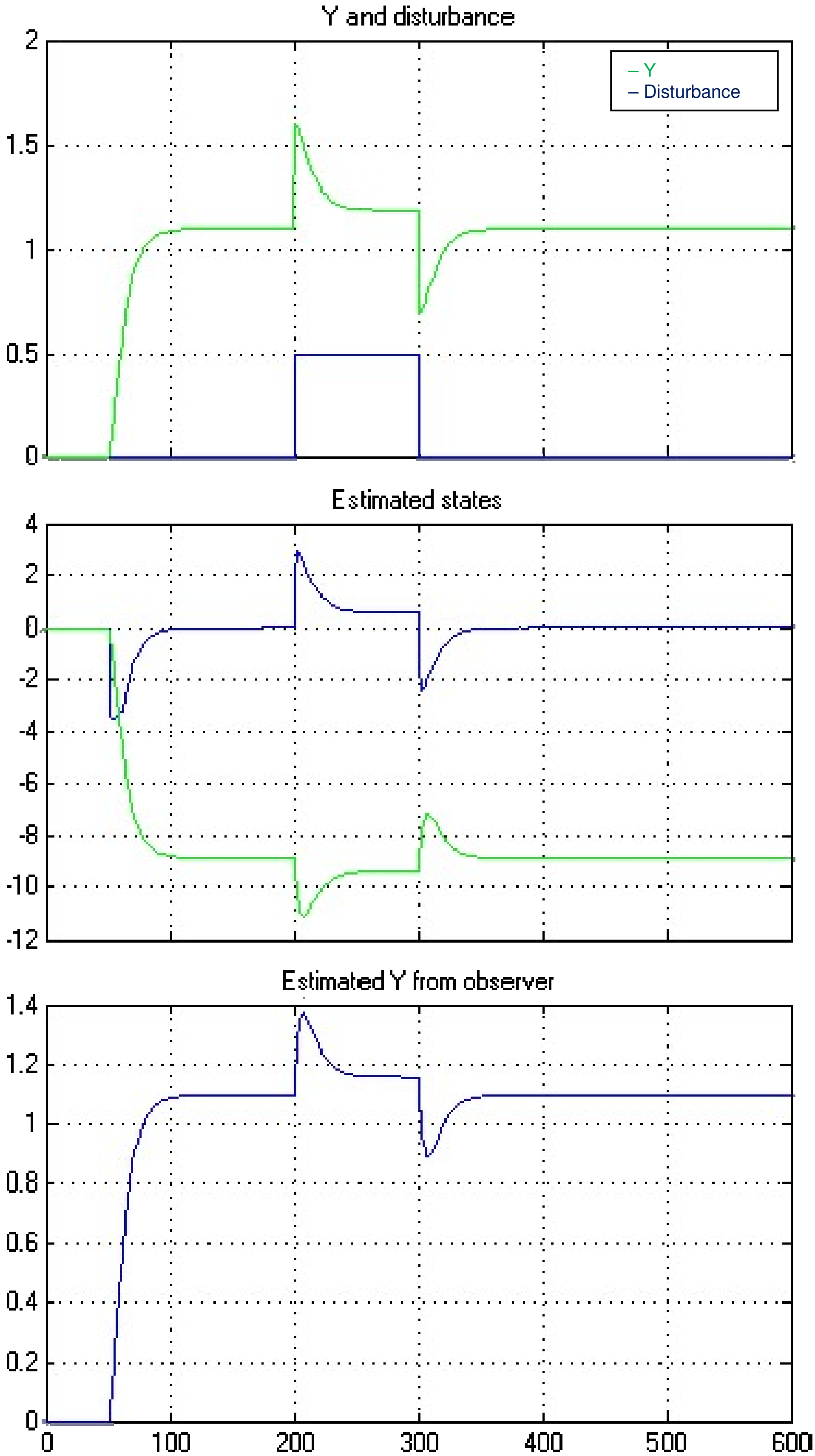



**Figure 27** Output of *Scope4* (full-state observer used)

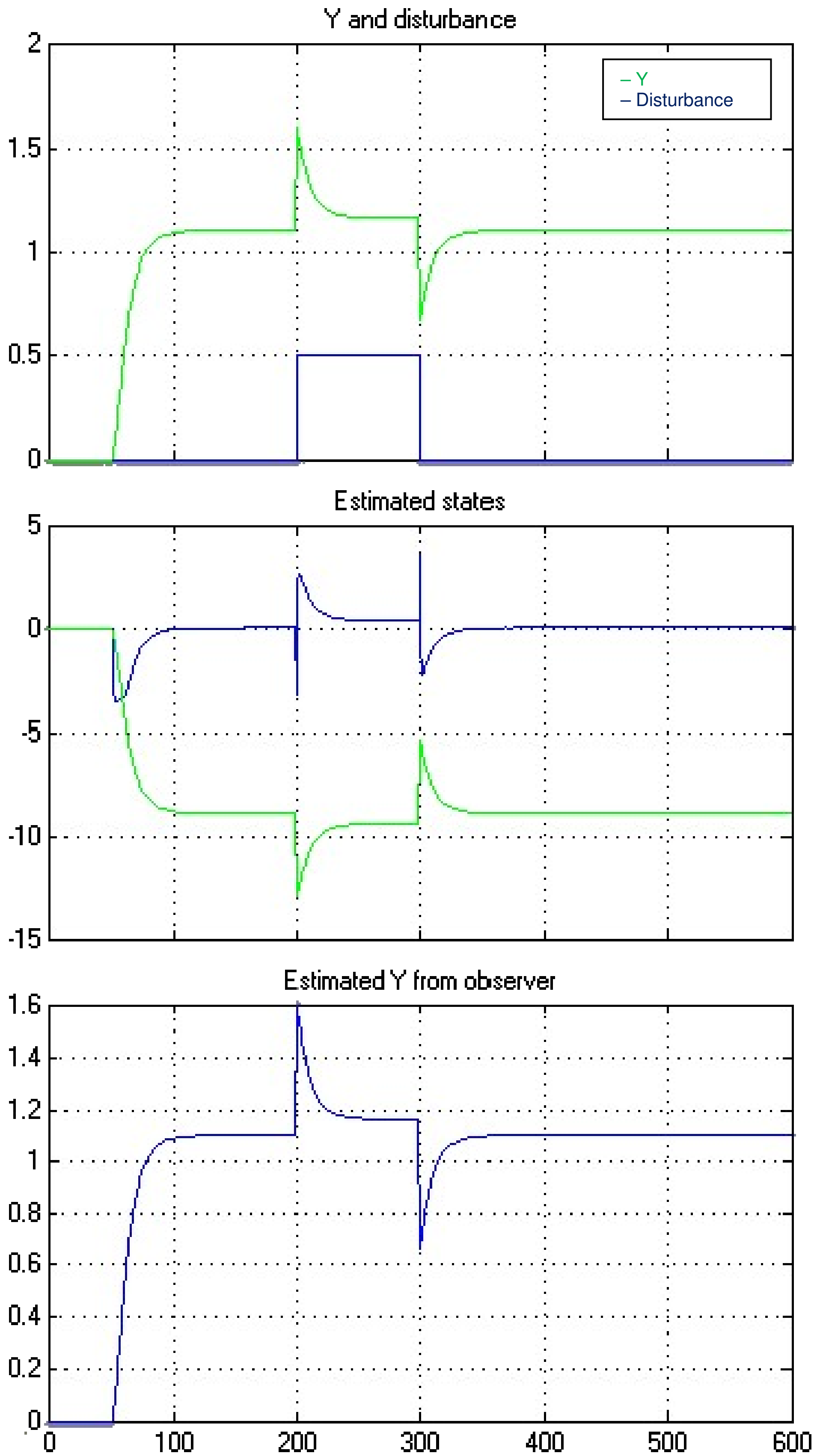



**Figure 28** Output of *Scope4* (reduced-order observer used)

### *4.4.5 Closed-Loop System with Disturbance, SP & Integrator*

As the system output in **Subsection 4.4.4** could not reach the desired set point during the presence of the disturbance (slight offset), an integrator is introduced into the system. A similarly constructed closed-loop system with disturbance, set point and integrator is shown in **Figure 29**.

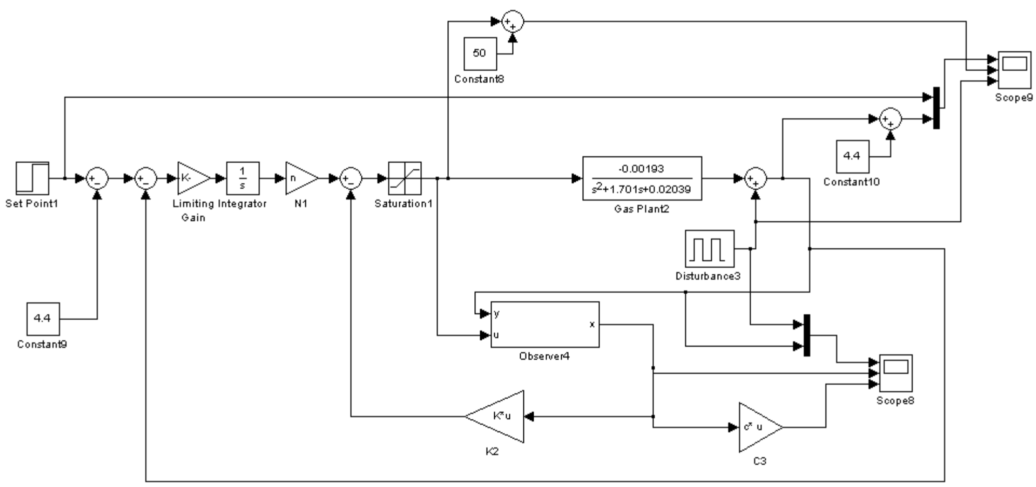



**Figure 29** Closed-loop system with disturbance, set point and integrator

The outputs of *Scope9* and *Scope8* are shown in **Figures 30** and **32** respectively when the full-state observer is used in the system. The effect of the integrator is clearly seen as the system output is returned to the desired set point of 4.4, without any offset. **Figures 31** and **33** show the outputs when the reduced-order observer is used instead.

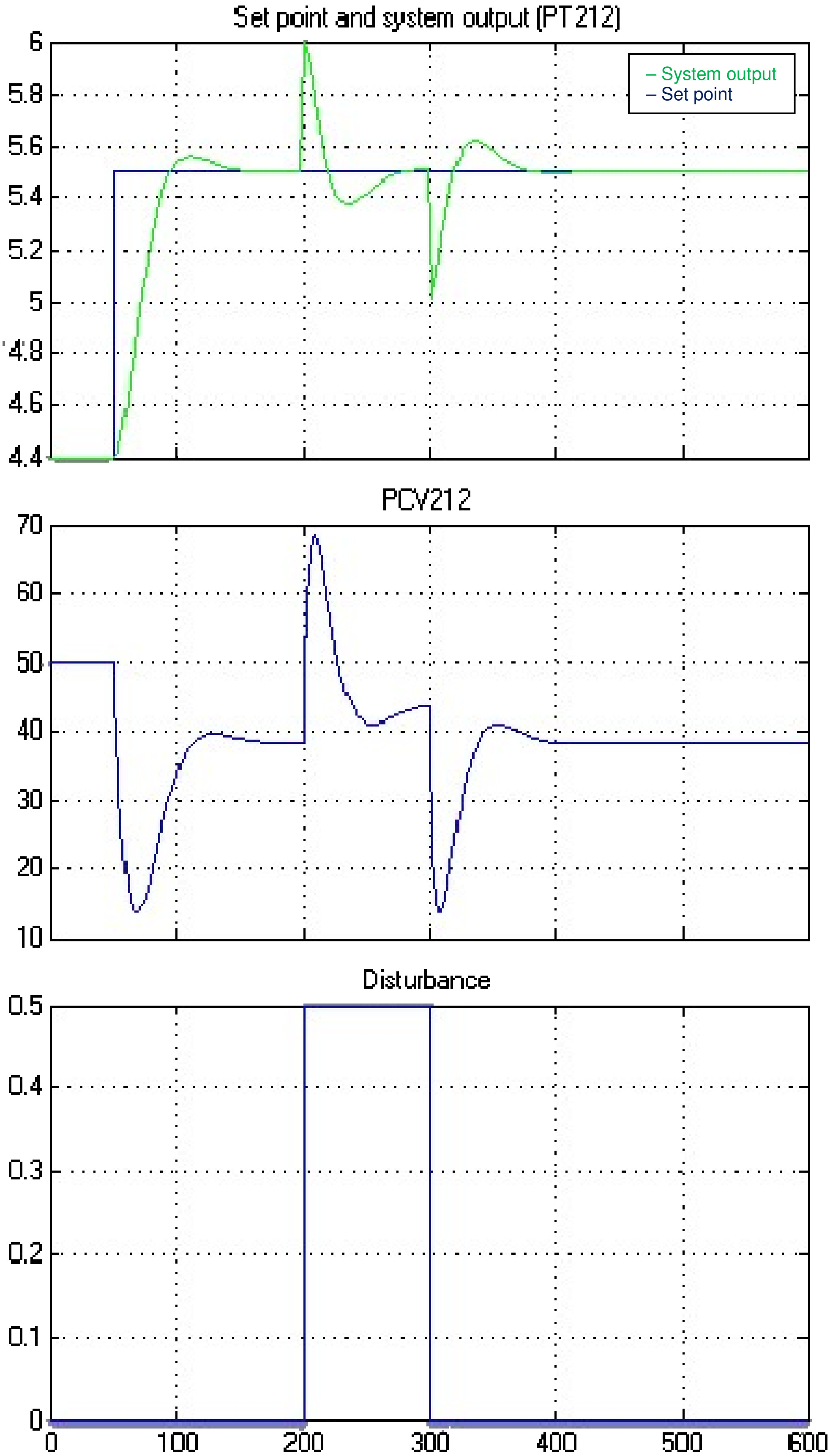



**Figure 30** Output of *Scope9* (full-state observer used)

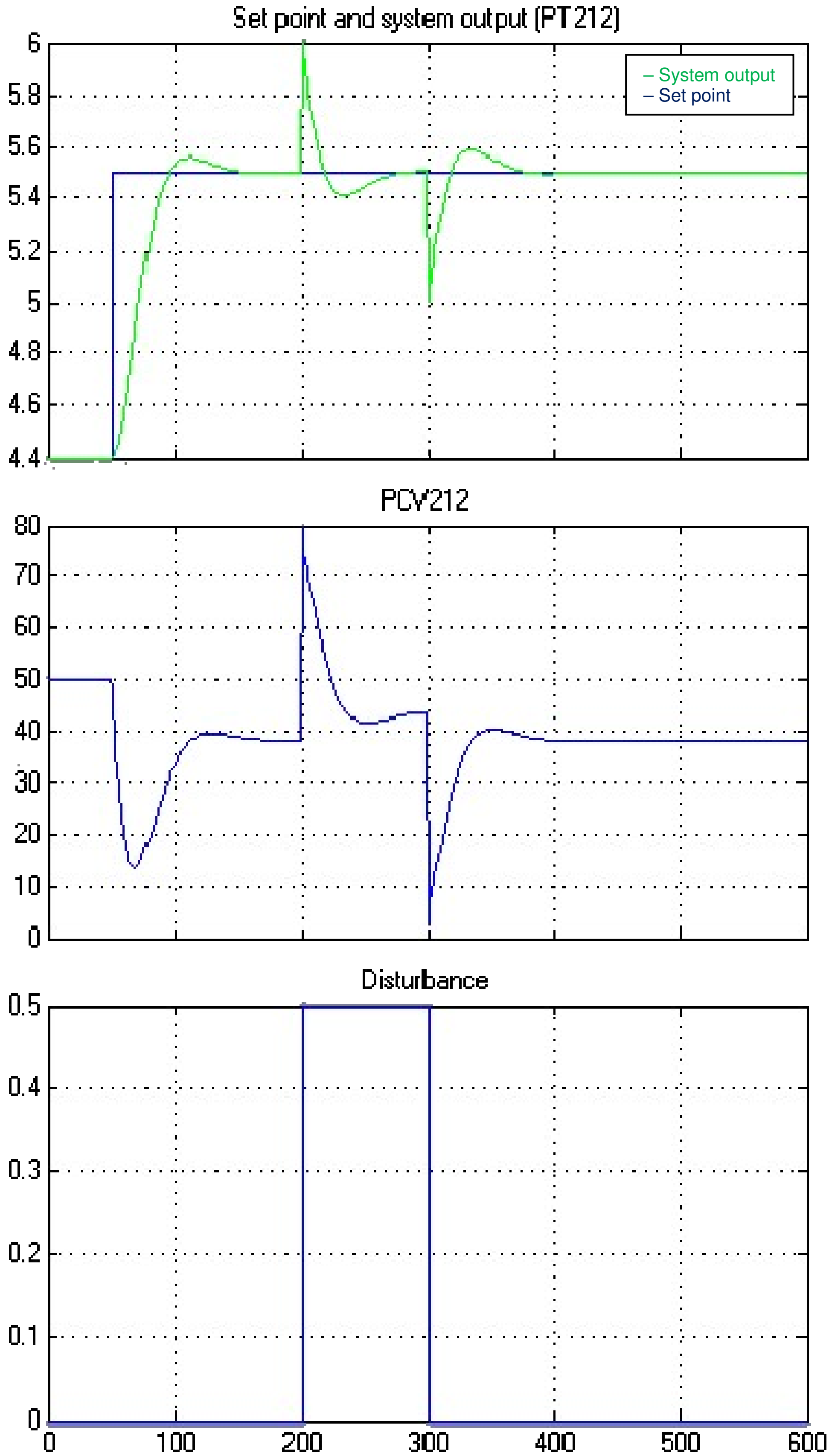



**Figure 31** Output of *Scope9* (reduced-order observer used)

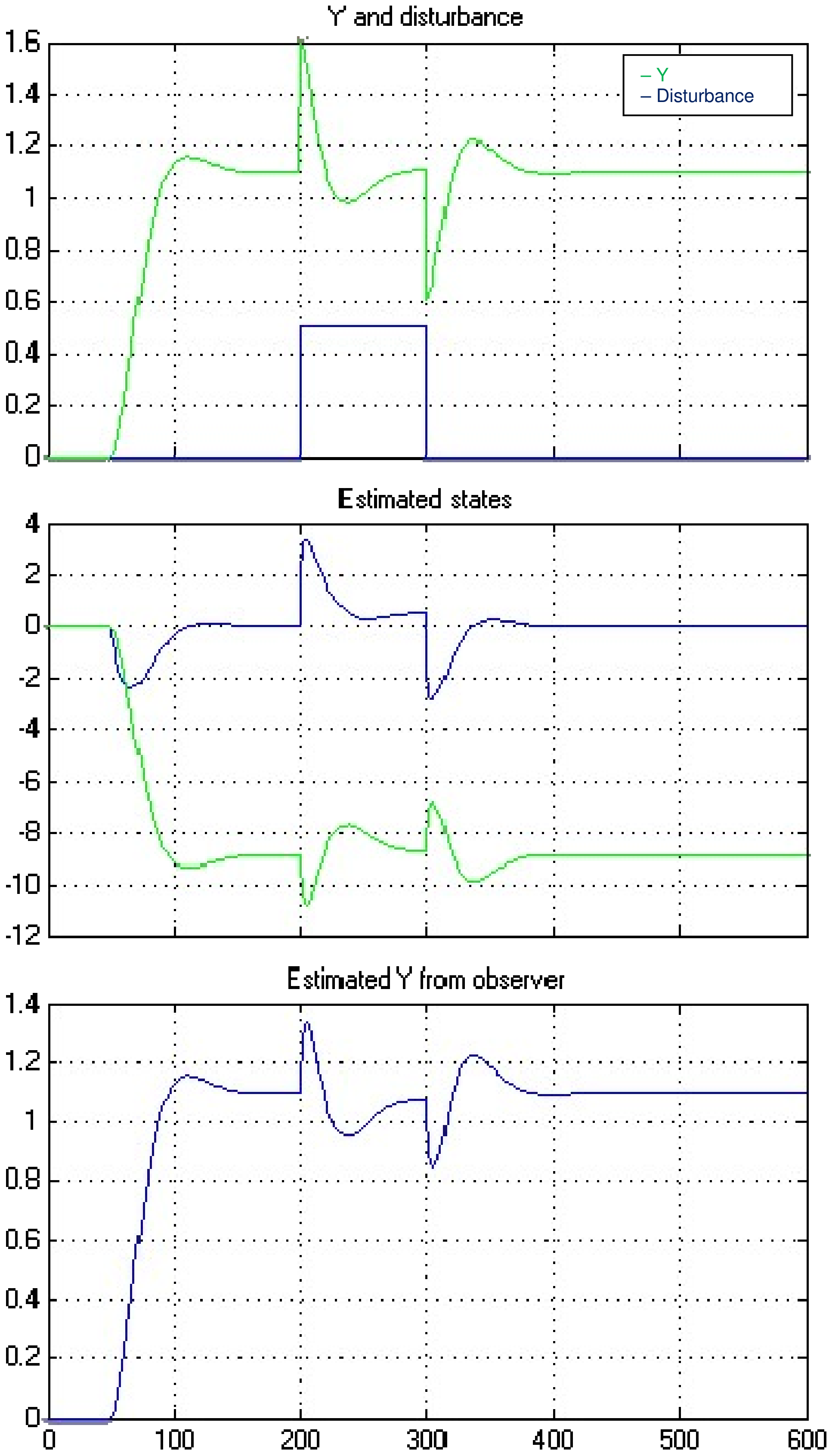



**Figure 32** Output of *Scope8* (full-state observer used)

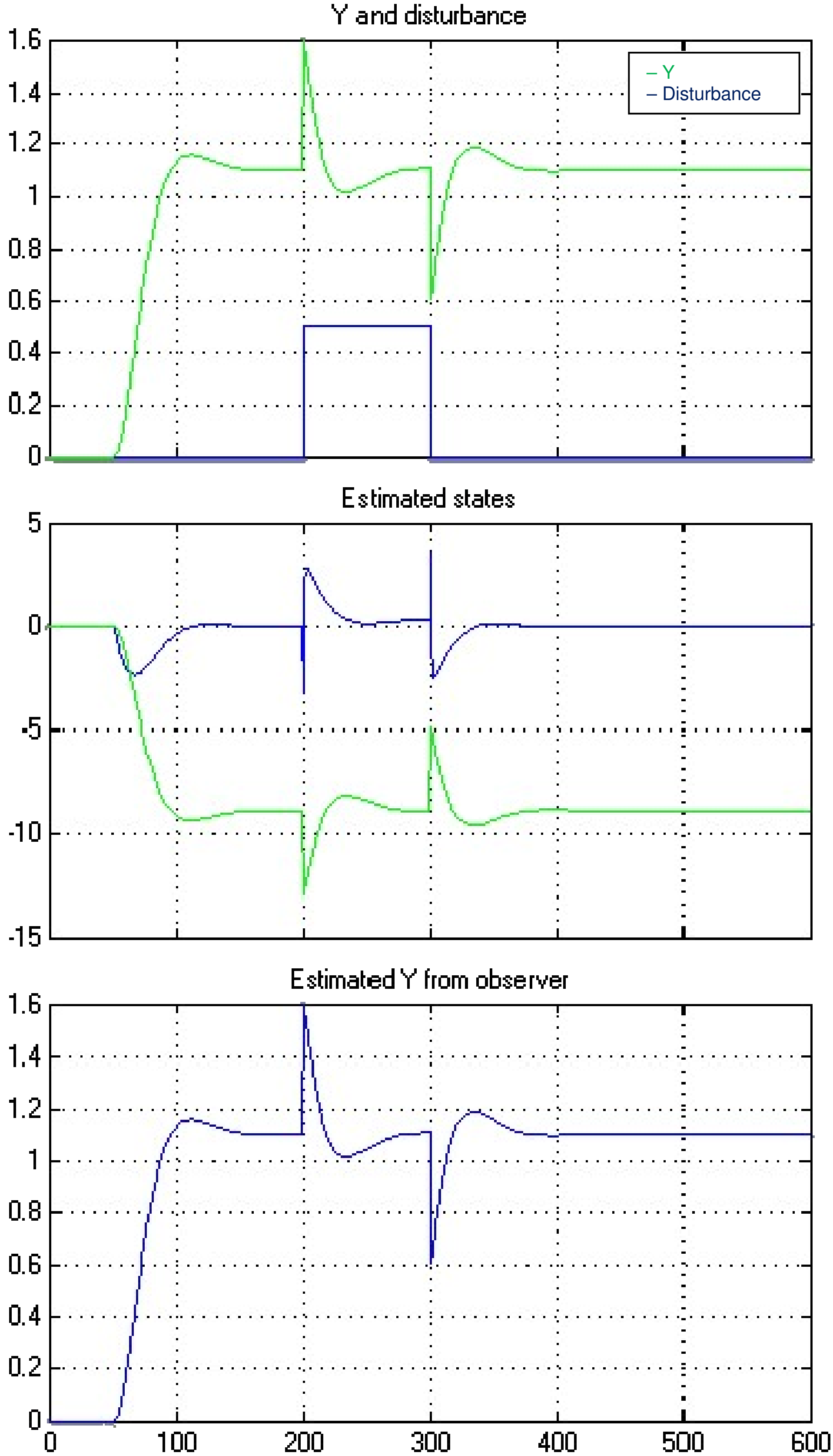



**Figure 33** Output of *Scope8* (reduced-order observer used)

The comparison of a closed-loop system output with and without the integrator is shown in **Figure 34**.

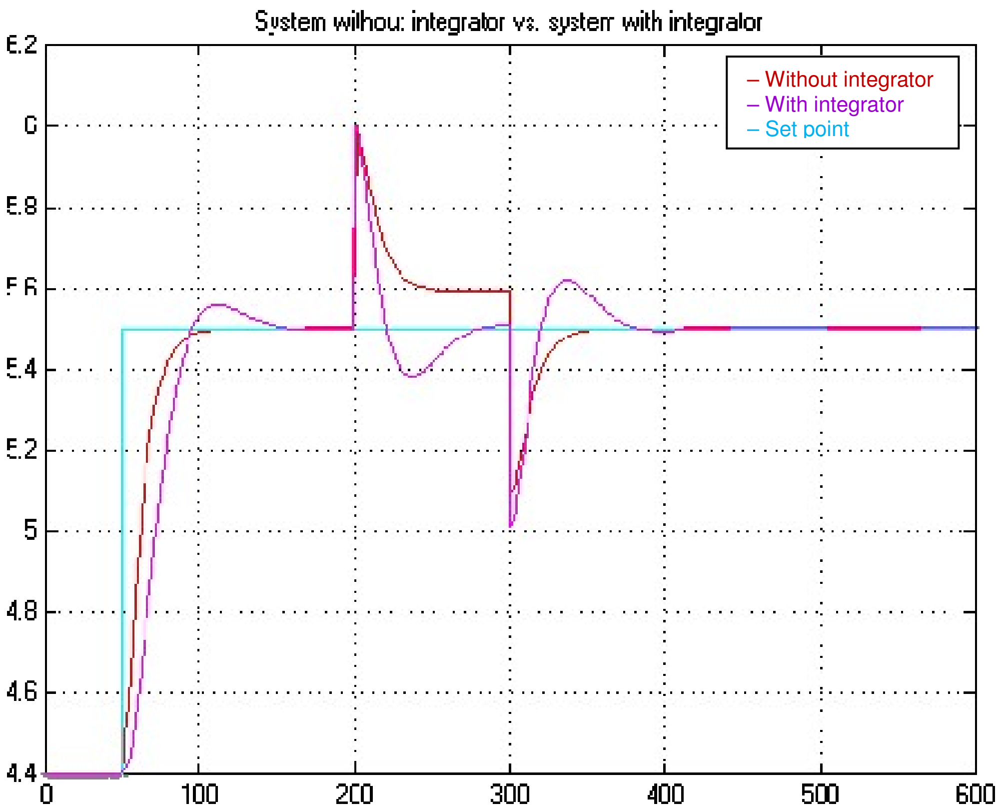



**Figure 34** System without integrator vs. system with integrator

# CHAPTER 5
# CONCLUSION AND RECOMMENDATIONS

This chapter concludes the entire project and proposes several recommendations, which could improve the outcomes of the project.

## 5.1 Recommendations

Despite the project achieving its objectives, there are several recommendations, which could be considered in order to improve its outcomes:

- **Determine the transfer function for the outlet valve**: A more complex but better control system could be designed if the transfer function for the valve PCV212 could be determined and included in the design.

- **Simulate using the transfer function of actual disturbances**: The disturbance in the plant was simulated using only a pulse generator, not an actual disturbance. Using the transfer function of actual disturbances will result in a more realistic simulation.

- **Compute the tuning gain for the integrator**: The tuning gain for the integrator was only assumed as 0.05. The proper method for computing a suitable gain should be studied and used in the design.

- **Implement controller and observers on actual plant**: The promising results from the simulations points only to one direction: implement the controller and observers on an actual plant to see its effectiveness as an alternative control strategy.

## 5.2 Conclusion

A gas plant was successfully modelled on Simulink from experimental data, producing simulated responses that characterize actual ones. Controller and observers were also then effectively designed using the pole placement method, producing promising results that indicate the practicality of modern control in plant process control systems. The success of this project signifies that an alternative to the current implementation of plant process control systems can be made possible with the design of a new controller and observer strategy that are robust, optimal and adaptive via modern control approach.

In conclusion, with the accomplishment of theoretically implementing the concepts of modern control engineering in plant process control systems, this project is a **success**.

# REFERENCES

## Published resources


[1] B. C. Kuo and F. Golnaraghi, *Automatic Control Systems*, 8th ed. New Jersey: Prentice Hall, 2003.

[2] B. Roffel and B. Betlem, *Process Dynamics and Control: Modeling for Control and Prediction*, West Sussex: John Wiley & Sons Ltd., 2006.

[3] B. W. Bequette, *Process Control: Modeling, Design and Simulation*, New Jersey: Prentice Hall, 2003.

[4] C. A. Smith and A. B. Corripio, *Principles and Practice of Automatic Process Control*, 3rd ed. Hoboken, NJ: Wiley, 2006.

[5] C. D. Johnson, *Process Control Instrumentation Technology*, 8th ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2006.

[6] G. F. Franklin, J. D. Powell and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, 5th ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2006.

[7] J. Dorsey, *Continuous and Discrete Control Systems: Modeling, Identification, Design and Implementation,* New York: McGraw Hill, 2002.

[8] K. Ogata, *Matlab for Control Engineers*, Upper Saddle River, NJ: Pearson Education: 2008.

[9] K. Ogata, *Modern Control Engineering*, 4th ed. New Jersey: Prentice-Hall, Inc., 2002.

[10] N. S. Nise, *Control Systems Engineering*, 5th ed. New York: John Wiley & Sons, 2008.

[11] R. C. Dorf and R. H. Bishop, *Modern Control Systems*, 11th ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2008.

[12] R. Pintelon and J. Schoukens, *System Identification: A Frequency Domain Approach*, New York: The Institute of Electrical and Electronics Engineers, Inc., 2001.

[13] T. E. Marlin, *Process Control: Designing Processes and Control Systems for Dynamic Performance*, 2nd ed. New York: McGraw Hill, 2000.

[14] W. Y. Svrcek, *A Real Time Approach to Process Control*, 2nd ed. Hoboken, NJ: Wiley, 2006.

**Online resources**

[15] ControlsWiki. "Chemical Process Dynamics and Controls", [Online]. Available: http://controls.engin.umich.edu/wiki/index.php/Main_Page. [Accessed: April 3, 2009].

[16] L. Ljung, "Perspectives on System Identification", April 20, 2008 [Online]. Available: http://www.control.isy.liu.se/~ljung/seoul2dvinew/plenary2.pdf [Accessed: April 12, 2009]

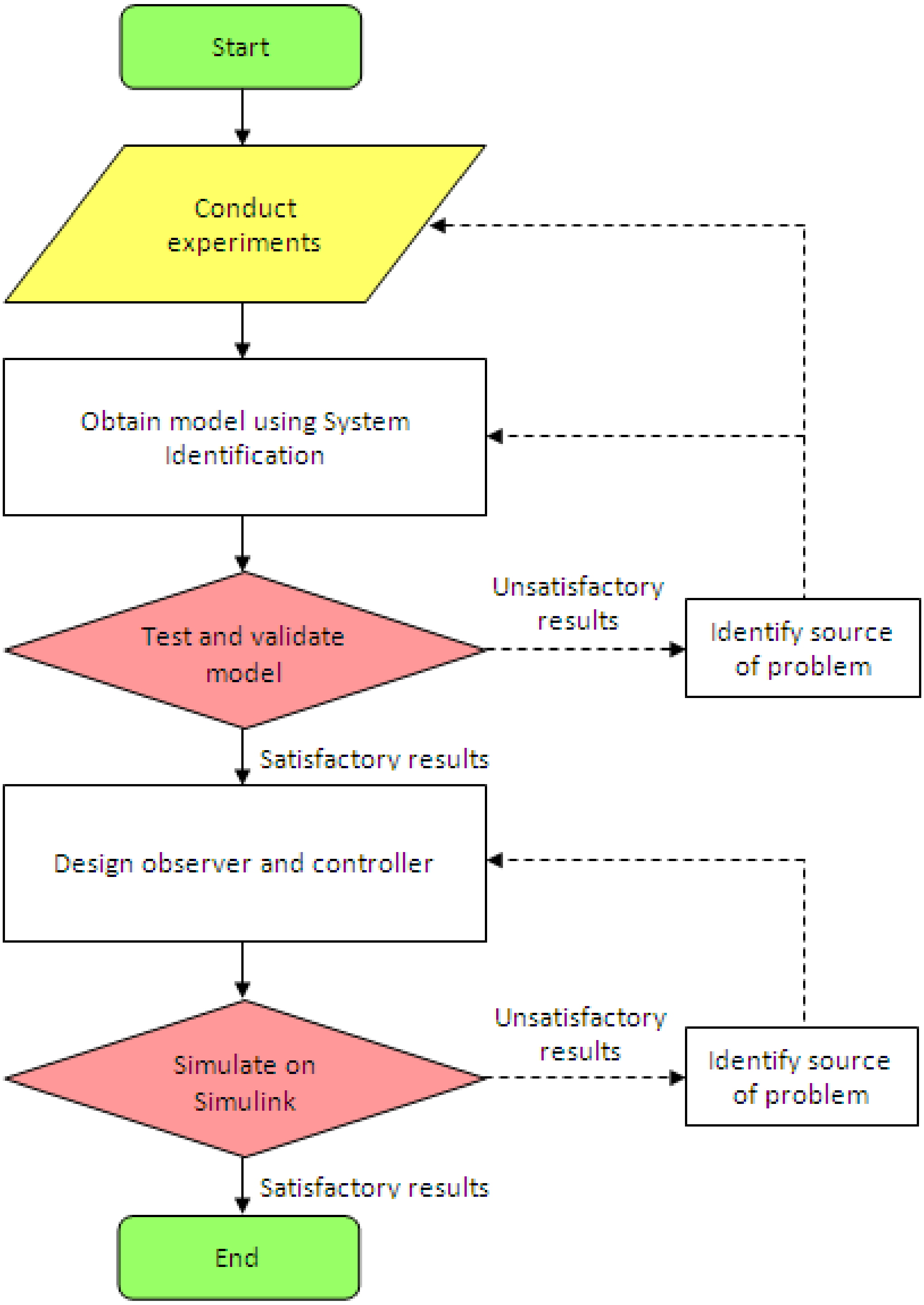
[17] MathWorks. "System Identification Toolbox 7.3", [Online]. Available: http://www.mathworks.com/products/sysid/ [Accessed: March 29, 2009]

[18] Wikipedia. "Process Control", [Online]. Available: http://en.wikipedia.org/wiki/Process_control [Accessed: January 30, 2009]

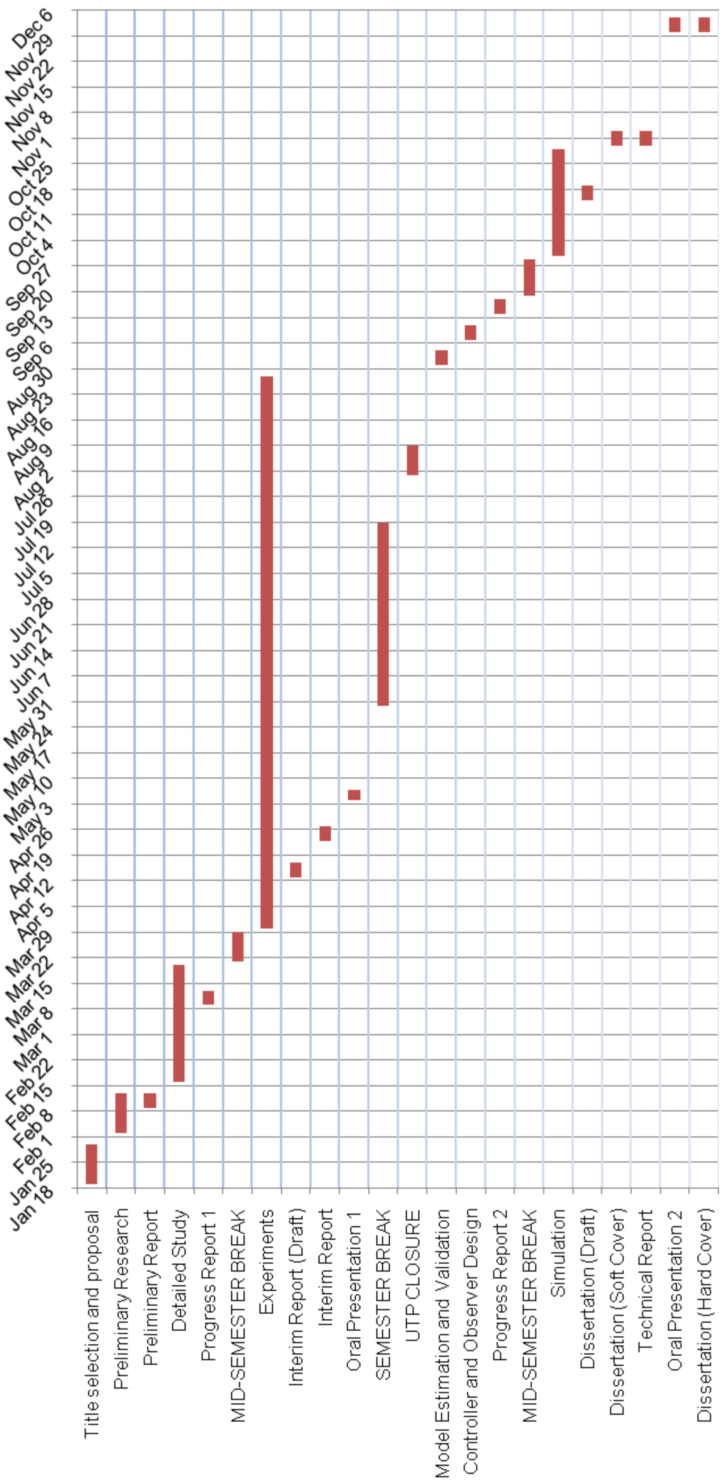
[19] Wikipedia. "System Identification", [Online]. Available: http://en.wikipedia.org/wiki/System_identification [Accessed: April 19, 2009]

# APPENDICES

# APPENDIX A
# PROJECT FLOWCHART

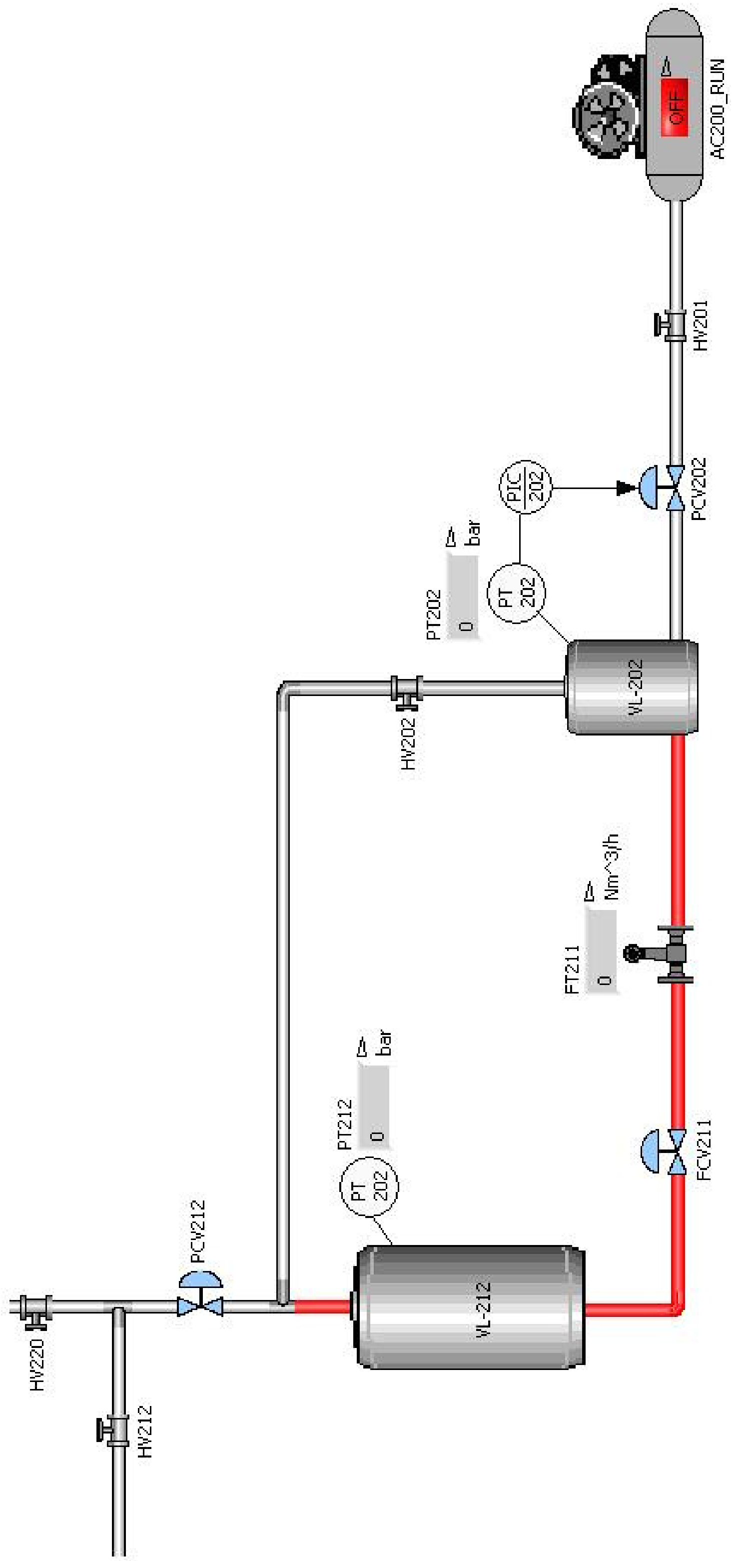


# APPENDIX B
# PROJECT GANTT CHART

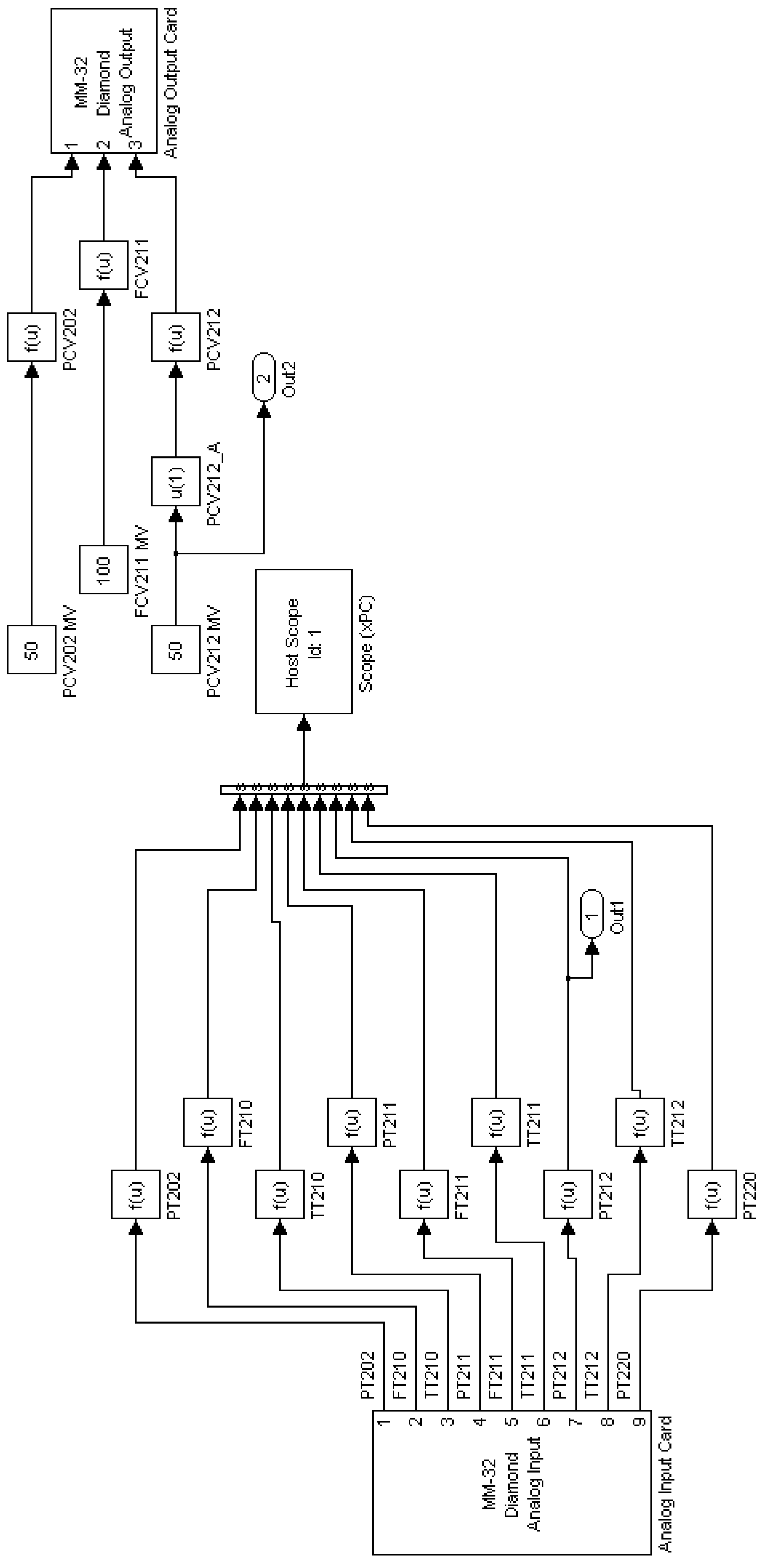


# APPENDIX C

# BLOCK DIAGRAM OF GAS PLANT



# APPENDIX D
# SIMULINK MODEL FOR EXPERIMENTS



# APPENDIX E

# M-FILE: DANIEL_DATALOG_1.M

```
D_PT212 = [];
D_PCV212 = [];

t=timer('ExecutionMode','fixedRate','Period',1,'TimerFc
n','daniel_datalog_2');
i=0;
start(t);
```

# APPENDIX F
# M-FILE: DANIEL_DATALOG_2.M

```
i=i+1;

if i>7000

    stop(t);

    xlswrite('daniel.xls',D_PT212','PT212');
    xlswrite('daniel.xls',D_PCV212','PCV212');

end

PCV212 = tg.getsignal(5);

PT212 = tg.getsignal(8);

D_PCV212 = [D_PCV212 PCV212];

D_PT212 = [D_PT212 PT212];
```

# APPENDIX G

# COMPUTING GAINS FOR CONTROLLER & FS OBSERVER

```
>> a=[-1.701 -0.1631;0.125 0];
>> b=[0.125;0];
>> c=[0 -0.1235];
>> d=0;
>> % Computing controller gain, K
>> jk=[-1.7 -0.1];
>> k=acker(a,b,jk)

k =

    0.7920    9.5752

>> % Computing full-state observer gain, L1
>> jl1=[-1.8 -0.2];
>> l1=acker(a',c', jl1)'

l1 =

   10.9465
   -2.4211
```

# APPENDIX H

# COMPUTING GAIN FOR REDUCED-ORDER OBSERVER

$$\dot{\hat{x}}_1 = A_{11}\hat{x}_1 + A_{12}x_2 + B_1 u + l(A_{21}x_1 - A_{21}\hat{x}_1)$$
$$\dot{\hat{x}}_1 = -1.701\hat{x}_1 - 0.1631x_2 + 0.125u + l(0.125x_1 - 0.125\hat{x}_1)$$

$$\dot{x}_1 = -1.701x_1 - 0.1631x_2 + 0.125u$$
$$\dot{x}_2 = 0.125x_1$$
$$y = -0.1235x_2$$

$$\dot{\hat{x}}_1 = -1.701\hat{x}_1 - 0.1631(-0.8.097y) + 0.125u + l(0.125\cdot -64.78\dot{y} - 0.125\hat{x}_1)$$
$$\dot{\hat{x}}_1 = -1.701\hat{x}_1 + 1.321y + 0.125u + l(-8.097\dot{y} - 0.125\hat{x}_1)$$
$$\dot{\hat{x}}_1 + 8.097l\dot{\hat{y}} = (-1.701 - 0.125l)\hat{x}_1 + 1.321y + 0.125u$$

$$-1.701 - 0.125l = -1.8$$
$$l = 0.792$$

$$\dot{\hat{x}}_1 + 6.413\dot{y} = -1.8\hat{x}_1 + 1.321y + 0.125u$$