

2003-09-01

## The Estimation and Control of a Laboratory Heating and Ventilation System

Robin P. Mooney

*Technological University Dublin, robin.mooney@mydit.ie*

Follow this and additional works at: <https://arrow.tudublin.ie/engscheledisae>



Part of the [Electrical and Electronics Commons](#)

---

### Recommended Citation

Mooney, R.:The Estimation and Control of a Laboratory Heating and Ventilation System. Masters Dissertation. Dublin, Technological University Dublin, 2003.

This Dissertation is brought to you for free and open access by the Dissertations at ARROW@TU Dublin. It has been accepted for inclusion in Masters in Advanced Engineering by an authorized administrator of ARROW@TU Dublin. For more information, please contact [yvonne.desmond@tudublin.ie](mailto:yvonne.desmond@tudublin.ie), [arrow.admin@tudublin.ie](mailto:arrow.admin@tudublin.ie), [brian.widdis@tudublin.ie](mailto:brian.widdis@tudublin.ie).



This work is licensed under a [Creative Commons Attribution-NonCommercial-Share Alike 3.0 License](#)

Faculty of Engineering

Dublin Institute of Technology

---

The Estimation and Control of a Laboratory

Heating and Ventilation System

---



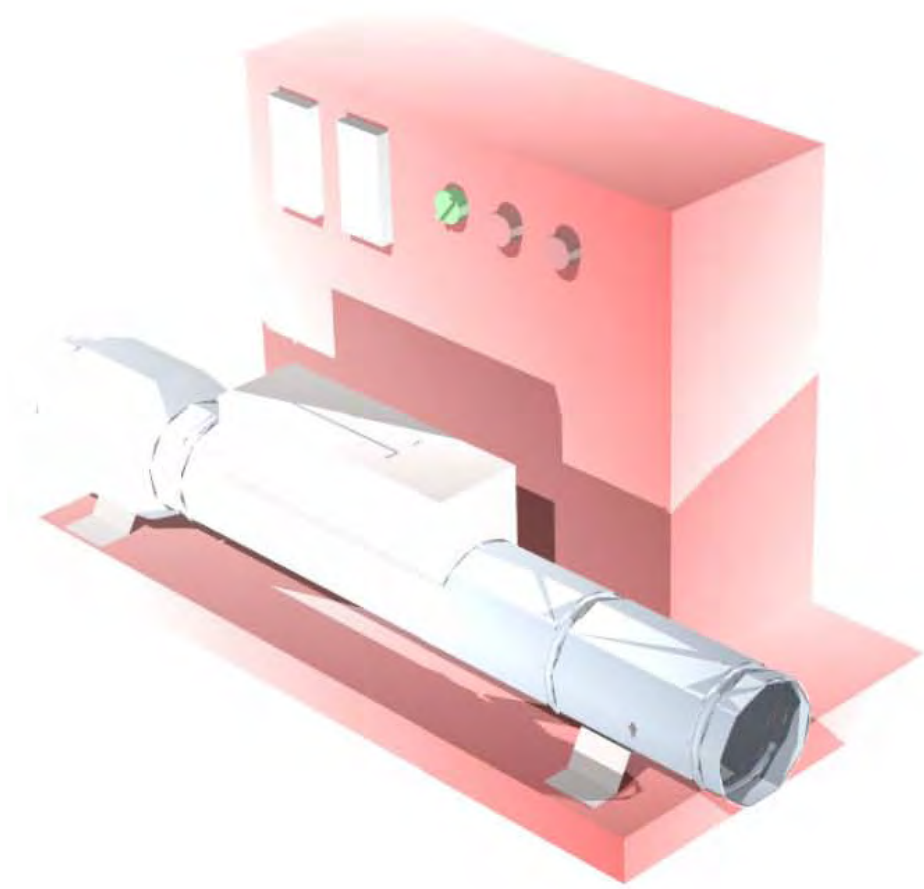
Robin Mooney Dip.Eng. B.E. MIEI

Supervisor: Dr. Aidan O'Dwyer

Masters in Advanced Engineering

September 2003

*“I often say that when you can measure what you are speaking about and express it in numbers you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind” —Lord Kelvin (1883).*



## Abstract

This dissertation is concerned with the estimation and control of a laboratory heating and ventilation system (Instrutek VVS-400). The system is a 2x2 multi-input multi-output process (MIMO). It has been shown that simple techniques such as the ultimate cycle method do not provide adequate control of the process. The system was interfaced with a PC using Matlab/Simulink via a data acquisition package (Humusoft). Continuous time process identification techniques were applied to the flow and temperature processes. The alternative tangent and point method was used to model the processes, and their interaction, using a first order lag plus delay model. Models were obtained for a range of operating conditions. The accuracy of the flow and temperature measurement transducers were investigated — some inaccuracies were determined. Tests revealed that both processes were continuously non-linear. This pointed toward adaptive control as appropriate. PI/PID controllers were used because both processes displayed a low time delay to time constant ratio. Tuning rules were selected on the basis of minimising the integral of absolute error. A strong interaction effect between the output temperature and input flow rate was reduced considerably using a static decoupler. A gain scheduler was designed, using look-up tables, to continuously interpolate for the most suitable controller settings and decoupler gain, as process operating conditions varied. The design was compared to an average model controller. Validation tests showed that the overall difference in performance was slim. It was concluded that discrete time identification methods would yield more appealing results for the gain scheduler, and that the design could be applied to other MIMO processes with relative ease.

## Acknowledgements

There are some people in particular that I would like to thank for their help with the completion of this project.

### **Dr. Aidan O'Dwyer** (project supervisor)

Aidan provided invaluable guidance throughout the project. His knowledge and experience was very much appreciated. Aidan could always point the best way forward, particularly in terms of relevant reading, and in this sense provided great motivation.

### **Mr. Tony Kealy**

Tony provided help in setting up the system to operate from a PC. His assistance at the beginning of the project saved valuable time. During the project Tony's know-how with Matlab/Simulink helped to diagnose some simulation problems.

Finally, I would like to thank my parents for their unending support and encouragement, for which I am extremely grateful. Thanks also to the rest of my family, my friends, and girlfriend for their help and advice.

## Declaration

I hereby declare that, except where otherwise indicated, this document is entirely my own work and has not been submitted in whole or in part to any other university.

Signed:.....Date:.....

# Table of Contents

|   |           |
|---|-----------|
| ABSTRACT.....   | I         |
| ACKNOWLEDGEMENTS.....   | II        |
| DECLARATION.....  | II        |
| TABLE OF CONTENTS.....  | III       |
| TABLE OF FIGURES & TABLES.....  | V         |
| <b>1. INTRODUCTION.....</b>   | <b>1</b>  |
| 1.1 THE BACKGROUND.....   | 1         |
| 1.2 THE EQUIPMENT.....  | 2         |
| 1.3 THE SOFTWARE.....   | 3         |
| 1.3.1 MATLAB®.....  | 3         |
| 1.3.2 SIMULINK®.....  | 3         |
| 1.3.3 HUMUSOFT®.....  | 3         |
| 1.4 AIMS AND OBJECTIVES.....  | 4         |
| 1.4.1 FAMILIARISATION WITH EQUIPMENT.....   | 4         |
| 1.4.2 FURTHER IDENTIFICATION AND INVESTIGATION INTO NON-LINEARITIES.....                          | 4         |
| 1.4.3 CONTROLLER DESIGN.....  | 5         |
| 1.4.4 FINAL DESIGN: IMPLEMENTATION AND VALIDATION TESTING.....                                    | 5         |
| 1.5 SUMMARY OF RESULTS.....   | 5         |
| <b>2. INVESTIGATION INTO PROCESS BEHAVIOUR - EXPERIMENTS.....</b>                                 | <b>7</b>  |
| 2.1 LOCAL FLOW CONTROLLER.....  | 7         |
| 2.1.1 PI CONTROLLER USING ZIEGLER AND NICHOLS ULTIMATE CYCLE METHOD.....                          | 7         |
| 2.1.2 PID CONTROLLER USING ZIEGLER AND NICHOLS ULTIMATE CYCLE METHOD.....                         | 8         |
| 2.1.3 MANUAL TUNING.....  | 10        |
| 2.1.4 AUTO-TUNING.....  | 11        |
| 2.2 PC FLOW CONTROLLER.....   | 12        |
| 2.2.1 PRO-REG PACKAGE.....  | 12        |
| 2.2.2 PI CONTROLLER USING ZIEGLER AND NICHOLS ULTIMATE CYCLE METHOD.....                          | 12        |
| 2.2.3 PID CONTROLLER USING ZIEGLER AND NICHOLS ULTIMATE CYCLE METHOD.....                         | 13        |
| 2.2.4 TESTS AT LOWER FLOWS.....   | 14        |
| 2.2.5 MANUAL TUNING.....  | 15        |
| 2.3 TEMPERATURE CONTROL.....  | 15        |
| <b>3. PROCESS IDENTIFICATION.....</b>   | <b>16</b> |
| 3.1 INTERFACING HARDWARE WITH MATLAB/HUMUSOFT.....  | 16        |
| 3.1.1 PROCEDURE FOR INTERFACING THE RIG WITH MATLAB/HUMUSOFT.....                                 | 16        |
| 3.2 FLOW PROCESS.....   | 18        |
| 3.2.1 FLOW STEP TESTS – LOAD VANE FULLY OPEN.....   | 18        |
| 3.2.2 FLOW FREQUENCY RESPONSE TEST.....   | 22        |
| 3.3 TEMPERATURE PROCESS.....  | 26        |
| 3.3.1 TEMPERATURE STEP TESTS.....   | 26        |
| <b>4. INVESTIGATION INTO PROCESS NON-LINEARITIES, MEASUREMENT ACCURACY, AND INTERACTIONS.....</b> | <b>32</b> |
| 4.1 FLOW PROCESS STATIC TESTS.....  | 32        |
| 4.1.1 FLOW MEASUREMENT ACCURACY OF THE RIG.....   | 34        |
| 4.2 TEMPERATURE PROCESS STATIC TESTS.....   | 39        |
| 4.2.1 TEMPERATURE MEASUREMENT ACCURACY OF THE RIG.....  | 40        |
| 4.2.2 TEMPERATURE PROCESS STATIC TESTS AT DIFFERENT FLOW RATES.....                               | 41        |
| 4.3 INVESTIGATION INTO PROCESS INTERACTIONS.....  | 42        |
| 4.3.1 INTERACTION STEP TESTS.....   | 44        |
| <b>5. CONTROLLER DESIGN.....</b>  | <b>46</b> |
| 5.1 CHOICE OF CONTROLLER.....   | 46        |
| 5.2 CHOICE OF ARCHITECTURE AND TUNING RULES.....  | 46        |
| 5.2.1 NOTE ON CONTROLLER IMPLEMENTATION.....  | 48        |
| 5.3 PRELIMINARY CONTROLLER TESTS.....   | 49        |
| 5.3.1 FLOW PROCESS CONTROLLER TESTS.....  | 49        |
| 5.3.1.1 PI CONTROLLER TESTS.....  | 49        |
| 5.3.1.2 PID CONTROLLER TESTS.....   | 51        |
| 5.3.2 TEMPERATURE PROCESS CONTROLLER TESTS.....   | 52        |
| 5.3.2.1 PI CONTROLLER TESTS.....  | 52        |
| 5.3.2.2 PID CONTROLLER TESTS.....   | 53        |
| 5.3.3 COMMENTS.....   | 53        |
| 5.4 PROCESS INTERACTIONS AND DECOUPLING.....  | 57        |

|           |  |            |
|-----------|--|------------|
| 5.4.1     | DECOUPLING CONTROL SYSTEMS.....  | 58         |
| 5.4.2     | CONTROLLER TESTS USING STATIC DECOUPLERS.....                              | 59         |
| 5.5       | INVERSE RESPONSE COMPENSATION METHODS.....                                 | 63         |
| 5.5       | GAIN SCHEDULING.....   | 65         |
| 5.5.1     | SIMPLE GAIN SCHEDULER DESIGN.....  | 65         |
| 5.5.2     | ADVANCED GAIN SCHEDULER DESIGN.....  | 69         |
| 5.5.2.1   | GAIN SCHEDULING USING LOOK-UP TABLES.....                                  | 69         |
| 5.5.2.2   | BUILDING PI/PID CONTROLLERS WITH UPDATEABLE SETTINGS IN SIMULINK.....      | 71         |
| 5.5.2.3   | OVERALL SYSTEM DESIGN.....   | 74         |
| 5.6       | FINAL DESIGN VALIDATION TESTS.....   | 75         |
| 5.6.1     | TESTS ON THE FLOW PROCESS.....   | 75         |
| 5.6.2     | TESTS ON THE TEMPERATURE PROCESS.....                                      | 78         |
| 5.6.3     | COMMENTS.....  | 80         |
| <b>6.</b> | <b>CONCLUSION.....</b>   | <b>83</b>  |
| 6.1       | SUMMARY AND DISCUSSION OF RESULTS.....                                     | 83         |
| 6.2       | FURTHER WORK AND RECOMMENDATIONS.....                                      | 88         |
| <b>7.</b> | <b>APPENDIX.....</b>   | <b>91</b>  |
|           | APPENDIX A – FLOW PROCESS OPEN LOOP STEP TEST PLOTS.....                   | 92         |
|           | APPENDIX B – FLOW PROCESS FREQUENCY RESPONSE PLOTS.....                    | 94         |
|           | APPENDIX C – FLOW PROCESS FREQUENCY BODE PLOT – POSSIBLE MODEL.....        | 97         |
|           | APPENDIX D – TEMPERATURE PROCESS OPEN LOOP STEP TEST PLOTS (30% FLOW)..... | 98         |
|           | APPENDIX E – TEMPERATURE PROCESS OPEN LOOP STEP TEST PLOTS (50% FLOW)..... | 102        |
|           | APPENDIX F – TEMPERATURE PROCESS OPEN LOOP STEP TEST PLOTS (70% FLOW)..... | 106        |
|           | APPENDIX G – PROCESS INTERACTION STEP TEST PLOTS.....                      | 110        |
|           | APPENDIX H – PROJECT TIMETABLE.....  | 111        |
|           | APPENDIX I – SCHEMATIC OF RIG.....   | 112        |
|           | APPENDIX J – MATLAB/SIMULINK CONTROLLER SETTINGS.....                      | 113        |
|           | APPENDIX K – RELATIVE GAIN ARRAY METHOD.....                               | 114        |
|           | APPENDIX L – PLOTS OF ADVANCED GAIN SCHEDULER CONTROLLER SETTINGS.....     | 118        |
|           | APPENDIX M – ALL ADVANCED GAIN SCHEDULER CONTROLLERS.....                  | 122        |
| <b>8.</b> | <b>BIBLIOGRAPHY.....</b>   | <b>126</b> |



## Table of Figures & Tables

|  |    |
|--|----|
| FIGURE 1-1 LLOYD'S BANK IN LONDON .....  | 1  |
| FIGURE 1-2 INSTRUTEK VVS-400 LABORATORY HEATING AND VENTILATION RIG .....                                      | 2  |
| FIGURE 2-1 SUSTAINED OSCILLATIONS IN OUTPUT DURING CONTINUOUS CYCLE TEST .....                                 | 7  |
| FIGURE 2-2 SERVO RESPONSE – 30%-50% STEP .....   | 7  |
| FIGURE 2-3 SERVO RESPONSE – 50%-30% STEP .....   | 7  |
| FIGURE 2-4 REGULATOR RESPONSE - CLOSING LOAD VANE BY 1/3.....  | 8  |
| FIGURE 2-5 REGULATOR RESPONSE OPENING LOAD VANE .....  | 8  |
| FIGURE 2-7 SERVO RESPONSE – 50%-30% STEP .....   | 9  |
| FIGURE 2-6 SERVO RESPONSE – 30%-50% STEP .....   | 9  |
| FIGURE 2-9 REGULATOR RESPONSE - OPENING LOAD VANE FULLY.....   | 9  |
| FIGURE 2-8 REGULATOR RESPONSE – CLOSING LOADS VANE BY 1/3 .....  | 9  |
| FIGURE 2-10 LARGE DISTURBANCE .....  | 9  |
| FIGURE 2-11 MANUAL TUNING STEP RESPONSES WITH GIVEN SETTINGS AND SETTLING TIMES .....                          | 10 |
| FIGURE 2-12 AUTO-TUNED PID CONTROLLER STEP RESPONSES FOR OPEN AND 1/3 CLOSED LOAD VANE.....                    | 11 |
| FIGURE 2-13 WIRING DIAGRAM.....  | 12 |
| FIGURE 2-14 Z&N PI CTR. (I) - SERVO (30-50% STEP) AND REGULATOR (LOAD VANE 1/3 CLOSED) RESPONSES .....         | 12 |
| FIGURE 2-15 Z&N PI CTR. (II) - SERVO (30-50% STEP) AND REGULATOR (LOAD VANE 1/3 CLOSED) RESPONSES .....        | 13 |
| FIGURE 2-16 Z&N PID CTR. - SERVO (30-50% STEP) AND REGULATOR (LOAD VANE 1/3 CLOSED) RESPONSES.....             | 13 |
| FIGURE 2-17 Z&N PI & PID CTR. - SERVO (15-30% STEP) AND REGULATOR (LOAD VANE 1/3 CLOSED) RESPONSES .....       | 14 |
| FIGURE 2-18 SELECTED SERVO AND REGULATOR RESPONSES FOR OPTIMAL PI CONTROL.....                                 | 15 |
| FIGURE 3-1 SIMULINK LIBRARY BROWSER.....   | 16 |
| FIGURE 3-2 REAL-TIME SINK PARAMETERS AND ADAPTER CARD SET-UP.....  | 17 |
| FIGURE 3-3 TYPICAL SIMULINK SET-UP .....   | 17 |
| FIGURE 3-4 ALTERNATIVE TANGENT AND POINT STEP RESPONSE IDENTIFICATION METHOD.....                              | 18 |
| FIGURE 3-5 SIMULINK MODEL FILE FOR FLOW OPEN LOOP STEP TEST .....  | 19 |
| FIGURE 3-6 EXAMPLE OF ALTERNATIVE TANGENT AND POINT METHOD APPLIED TO A FLOW STEP RESPONSE .....               | 19 |
| TABLE 3-1 SUMMARY OF RESULTS FROM FLOW OPEN LOOP STEP TESTS.....   | 20 |
| FIGURE 3-7 SUMMARY OF FLOW PROCESS MODEL PARAMETER VARIATION .....   | 21 |
| FIGURE 3-8 SIMULINK MODEL FILE USED FOR FLOW FREQUENCY RESPONSE TEST .....                                     | 22 |
| FIGURE 3-9 FREQUENCY RESPONSE PLOT WITH MEASUREMENTS SHOWN.....  | 22 |
| TABLE 3-2 BODE PLOT DATA .....   | 23 |
| FIGURE 3-10 BODE PLOT – FLOW PROCESS AT MID TO HIGH RANGE FLOWS .....  | 24 |
| FIGURE 3-11 SIMULINK MODEL FILE FOR TEMPERATURE OPEN LOOP STEP TEST.....                                       | 26 |
| FIGURE 3-12 EXAMPLE OF ALTERNATIVE TANGENT AND POINT METHOD APPLIED TO A FLOW STEP RESPONSE .....              | 27 |
| TABLE 3-3 SUMMARY OF RESULTS FROM TEMPERATURE OPEN LOOP STEP TESTS AT 30% FLOW .....                           | 28 |
| TABLE 3-4 SUMMARY OF RESULTS FROM TEMPERATURE OPEN LOOP STEP TESTS AT 50% FLOW .....                           | 29 |
| TABLE 3-5 SUMMARY OF RESULTS FROM TEMPERATURE OPEN LOOP STEP TESTS AT 70% FLOW .....                           | 30 |
| FIGURE 3-13 PLOTS OF PROCESS MODEL PARAMETER VARIANCE AT DIFFERENT FLOW RATES .....                            | 31 |
| FIGURE 4-1 SIMULINK MODEL FILE FOR STATIC FLOW TEST .....  | 32 |
| FIGURE 4-2 INCREASING AND DECREASING STATIC FLOW TEST CURVES – FULLY OPENED LOAD VANE .....                    | 33 |
| FIGURE 4-3 INPUT-OUTPUT CHARACTERISTIC CURVE – FLOW PROCESS .....  | 33 |
| FIGURE 4-4 FLOW CHARACTERISTIC CURVES – 1/3 CLOSED, 2/3 CLOSED, CLOSED LOAD VANE .....                         | 34 |
| FIGURE 4-5 ORIFICE PLATE FLOW MEASUREMENT DEVICE.....  | 34 |
| FIGURE 4-6 THE PITOT TUBE - VELOCITY MEASUREMENT DEVICE .....  | 35 |
| FIGURE 4-7 PHOTO OF THE RIG WITH FLOW MEASUREMENT EQUIPMENT SET-UP – AND CLOSE UP OF PITOT TUBE.....           | 36 |
| TABLE 4-1 RESULTS OF FLOW MEASUREMENT TEST.....  | 37 |
| TABLE 4-2 TEST SPECIFICATIONS .....  | 37 |
| FIGURE 4-8 DIFFERENCE BETWEEN ACTUAL FLOW RATE AND FLOW RATE MEASURED BY THE RIG.....                          | 38 |
| FIGURE 4-9 PITOT TUBE VELOCITY MEASUREMENT VS. ORIFICE PLATE VELOCITY MEASUREMENT .....                        | 38 |
| TABLE 4-3 STATIC TEMPERATURE TEST RESULTS.....   | 39 |
| FIGURE 4-10 TEMPERATURE CHARACTERISTIC CURVE AT 30% FLOW .....   | 40 |
| FIGURE 4-11 ACTUAL TEMPERATURE VS. TEMPERATURE OUTPUT CURVES – AT 30% FLOW .....                               | 40 |
| FIGURE 4-12 TEMPERATURE PROCESS STATIC STAIR TEST PLOT AT 50% FLOW.....  | 41 |
| FIGURE 4-13 TEMPERATURE PROCESS INPUT/OUTPUT CHARACTERISTIC CURVES .....                                       | 42 |
| FIGURE 4-15 BLOCK DIAGRAM DESCRIBING FLOW-TEMPERATURE INTERACTION .....  | 43 |
| FIGURE 4-16 INTERACTION STEP TEST PLOTS:–ALL INPUTS AND OUTPUTS PLOT, AND ZOOMED TEMPERATURE OUTPUT PLOT ..... | 44 |
| TABLE 4-4 INTERACTION STEP TEST RESULTS .....  | 44 |
| TABLE 5-1 SUMMARY OF TIME DELAY TO TIME CONSTANT RATIOS OBTAINED FROM PROCESS IDENTIFICATION .....             | 46 |
| TABLE 5-2 PI CONTROLLER TUNING RULES .....   | 47 |
| TABLE 5-3 PID CONTROLLER TUNING RULES .....  | 47 |
| TABLE 5-4 FLOW PROCESS – PI AND PID CONTROLLER SETTINGS .....  | 48 |

|   |     |
|---|-----|
| TABLE 5-5 TEMPERATURE PROCESS – PI AND PID CONTROLLER SETTINGS.....   | 48  |
| FIGURE 5-1 SIMULINK MODEL FILES FOR THE FLOE PROCESS REGULATOR AND SERVO PI CONTROLLER TESTS .....  | 50  |
| FIGURE 5-2 REGULATOR AND SERVO RESPONSE PLOTS OBTAINED FROM THE PI FLOW CONTROLLER TESTS .....  | 50  |
| FIGURE 5-3 SIMULINK MODEL FILES FOR THE FLOW PROCESS REGULATOR AND SERVO PID CONTROLLER TESTS.....  | 51  |
| FIGURE 5-4 REGULATOR AND SERVO RESPONSE PLOTS OBTAINED FROM THE PID FLOW CONTROLLER TESTS. ....   | 51  |
| FIGURE 5-5 SIMULINK MODEL FILES FOR THE TEMPERATURE PROCESS REGULATOR AND SERVO PI CONTROLLER TESTS.....  | 52  |
| FIGURE 5-6 REGULATOR AND SERVO RESPONSE PLOTS OBTAINED FROM THE PI TEMPERATURE CONTROLLER TESTS.....  | 52  |
| FIGURE 5-7 SIMULINK MODEL FILES FOR THE TEMPERATURE PROCESS REGULATOR AND SERVO PID CONTROLLER TESTS.....   | 53  |
| FIGURE 5-8 REGULATOR AND SERVO RESPONSE PLOTS OBTAINED FROM THE PID TEMPERATURE CONTROLLER TESTS.....   | 53  |
| FIGURE 5-9 SERVO RESPONSES FOR PI AND PID FLOW CONTROLLERS WITH CONTROLLER OUTPUT SHOWN .....   | 54  |
| FIGURE 5-10 INVERSE RESPONSE – PID TEMPERATURE CONTROLLER SERVO TEST .....  | 55  |
| FIGURE 5-11 TWO FIRST ORDER PROCESSES ACTING IN PARALLEL .....  | 55  |
| FIGURE 5-12 PID TEMPERATURE CONTROLLER TESTS WITH TEMPERATURE INPUT HELD CONSTANT .....   | 56  |
| FIGURE 5-13 MIMO BLOCK DIAGRAM.....   | 57  |
| FIGURE 5-14 TWO POSSIBLE 2X2 MIMO CONTROL CONFIGURATIONS.....   | 57  |
| FIGURE 5-15 A DECOUPLING CONTROL SYSTEM.....  | 58  |
| FIGURE 5-16 DECOUPLER CONTROL SYSTEM TEST.....  | 60  |
| FIGURE 5-17 SERVO RESPONSE PLOT OBTAINED FOR PI TEMPERATURE CONTROLLER WITH DECOUPLING .....  | 60  |
| FIGURE 5-18 CONSTANT INPUT AND SERVO RESPONSE PLOTS AT 30% FLOW IN THE LOW TEMPERATURE RANGE USING PI TEMPERATURE CONTROLLER WITH DECOUPLING.....         | 61  |
| FIGURE 5-19 CONSTANT INPUT AND SERVO RESPONSE PLOTS AT 30% FLOW IN THE LOW TEMPERATURE RANGE USING UPDATED PI TEMPERATURE CONTROLLER WITH DECOUPLING..... | 61  |
| FIGURE 5-20 TEMPERATURE PROCESS REGULATOR RESPONSE PLOT AT 30% FLOW IN THE LOW TEMPERATURE RANGE .....  | 62  |
| FIGURE 5-21 INPUT/OUTPUT TEMPERATURE PROCESS CHARACTERISTIC CURVE AT 50% FLOW, AND TEMPERATURE PROCESS SERVO RESPONSE PLOT AT 50% FLOW.....               | 62  |
| FIGURE 5-22 INVERSE RESPONSE MODELLING METHOD .....   | 63  |
| FIGURE 5-23 CONVENTIONAL FEEDBACK CONTROL OF AN INVERSE RESPONSE SYSTEM .....   | 64  |
| FIGURE 5-24 BLOCK DIAGRAM INCORPORATING INVERSE RESPONSE COMPENSATOR .....  | 64  |
| FIGURE 5-25 SIMULINK LOGIC BLOCK ARRANGEMENT TO MANIPULATE FLOW SIGNAL.....   | 66  |
| TABLE 5-6 TRUTH TABLE FOR CATEGORISING THE FLOW INPUT SIGNAL.....   | 66  |
| FIGURE 5-26 GAIN SCHEDULED PI FLOW CONTROL DESIGN.....  | 67  |
| FIGURE 5-27 GAIN SCHEDULED PI TEMPERATURE AND FLOW CONTROL DESIGN – IN REGULATOR MODE.....  | 68  |
| FIGURE 5-28 PLOT OF LINEAR INTERPOLATION CURVE FOR FLOW CONTROLLER GAIN (PI REGULATOR MODE).....  | 69  |
| TABLE 5-7 MATRIX OF INTEGRAL TIMES FOR A PID SERVO TEMPERATURE CONTROLLER .....   | 70  |
| FIGURE 5-29 3-DIMENSIONAL PLOT OF LINEAR INTERPOLATION OF INTEGRAL TIME FOR A PID SERVO TEMPERATURE CONTROLLER .....                                      | 70  |
| FIGURE 5-30 PI CONTROLLER USING LOOK-UP TABLES TO UPDATE CONTROLLER SETTINGS .....  | 71  |
| FIGURE 5-31 SIMULATION SET-UP AND SERVO RESPONSE OF IDEAL PI CONTROLLER TEST.....   | 71  |
| FIGURE 5-32 CLASSICAL PID CONTROLLER USING LOOK-UP TABLES TO UPDATE CONTROLLER SETTINGS.....  | 72  |
| FIGURE 5-33 SIMULATION SET-UP AND SERVO RESPONSE OF CLASSICAL PID CONTROLLER TEST.....  | 72  |
| FIGURE 5-34 SIMULATION SET-UP AND SERVO RESPONSE OF CLASSICAL PID CONTROLLER TEST WITH COMPROMISED DESIGN .....   | 73  |
| FIGURE 5-35 COMPARISON OF FLOW CONTROLLER (PID SERVO) DERIVATIVE AND INTEGRAL TIMES .....   | 73  |
| FIGURE 5-36 COMPARISON OF TEMPERATURE CONTROLLER (PID SERVO) DERIVATIVE AND INTEGRAL TIMES.....   | 73  |
| FIGURE 5-37 ADVANCED GAIN SCHEDULER – PI REGULATOR SET-UP .....   | 74  |
| FIGURE 5-38 FLOW PROCESS - ADVANCED GAIN SCHEDULER AND AVERAGE MODEL CONTROLLER PI REGULATOR RESPONSES .....  | 76  |
| TABLE 5-8 IAE RESULTS - FLOW REGULATOR CONTROLLERS.....   | 76  |
| FIGURE 5-39 FLOW PROCESS - ADVANCED GAIN SCHEDULER AND AVERAGE MODEL CONTROLLER PI SERVO RESPONSES .....  | 77  |
| TABLE 5-9 IAE RESULTS - FLOW SERVO CONTROLLERS.....   | 77  |
| TABLE 5-10 IAE RESULTS - TEMPERATURE REGULATOR CONTROLLERS .....  | 78  |
| FIGURE 5-40 TEMPERATURE PROCESS - ADVANCED GAIN SCHEDULER AND AVERAGE MODEL CONTROLLER PID REGULATOR RESPONSES.....                                       | 79  |
| FIGURE 5-41 TEMPERATURE PROCESS - ADVANCED GAIN SCHEDULER AND AVERAGE MODEL CONTROLLER PI SERVO RESPONSES.....  | 79  |
| TABLE 5-11 IAE RESULTS - TEMPERATURE SERVO CONTROLLERS .....  | 80  |
| FIGURE 5-42 IAE – FLOW REGULATOR CONTROLLERS.....   | 81  |
| FIGURE 5-43 IAE – FLOW SERVO CONTROLLERS.....   | 81  |
| FIGURE 5-44 IAE – TEMPERATURE REGULATOR CONTROLLERS .....   | 82  |
| FIGURE 5-45 IAE – TEMPERATURE SERVO CONTROLLERS .....   | 82  |
| TABLE 6-1 ALL FLOW PROCESS MODEL TRANSFER FUNCTIONS OBTAINED .....  | 83  |
| TABLE 6-2 ALL TEMPERATURE PROCESS MODELS OBTAINED .....   | 84  |
| TABLE 6-3 INTERACTION MODELS OBTAINED.....  | 86  |
| TABLE J-1 FLOW CONTROLLER SETTINGS FOR MATLAB/SIMULINK IMPLEMENTATION .....   | 113 |
| TABLE J-2 TEMPERATURE CONTROLLER SETTINGS FOR MATLAB/SIMULINK IMPLEMENTATION .....  | 113 |
| TABLE K STEADY STATE GAINS OBTAINED FROM PROCESS MODELLING.....   | 117 |
| FIGURE M-1 ADVANCED GAIN SCHEDULER – PI REGULATOR .....   | 122 |
| FIGURE M-2 ADVANCED GAIN SCHEDULER – PID REGULATOR.....   | 123 |
| FIGURE M-3 ADVANCED GAIN SCHEDULER – PI SERVO.....  | 124 |
| FIGURE M-4 ADVANCED GAIN SCHEDULER – PID SERVO.....   | 125 |

# 1. Introduction

## 1.1 The Background

Heating and ventilation systems play an important role in providing a comfortable, practical and healthy environment in our workplaces and homes. The provision of adequate heating and ventilation is becoming increasingly important in industry, particularly in office buildings with high employee densities. This is not only a key issue in providing work conditions that ensure employee satisfaction and hence increased work output; it is now a health and safety issue that has, in recent times, led to industrial relations problems related to air quality and smoking. Subsequently, fire, health and safety regulations have begun to put tighter limits on the acceptable number of air changes per hour in a building, and acceptable room temperature range. Figure 1-1 shows a picture of Lloyd's Bank in London, where the heating and ventilation system used was considered important from this point of view, as well as providing an unprecedented architectural design.



**Figure 1-1 Lloyd's Bank in London**

The accepted ISO standard<sup>1</sup> for 'comfort conditions' in the workplace describes an 'acceptable thermal environment' in the range 20 – 23.5°C. When the air temperature in an office goes above the 23.5°C limit people can suffer from headache, sweating, and nausea. If air temperatures drop below the 20°C limit shivering, colds, stiff joints and, in some extreme cases, frostbite can be suffered by

---

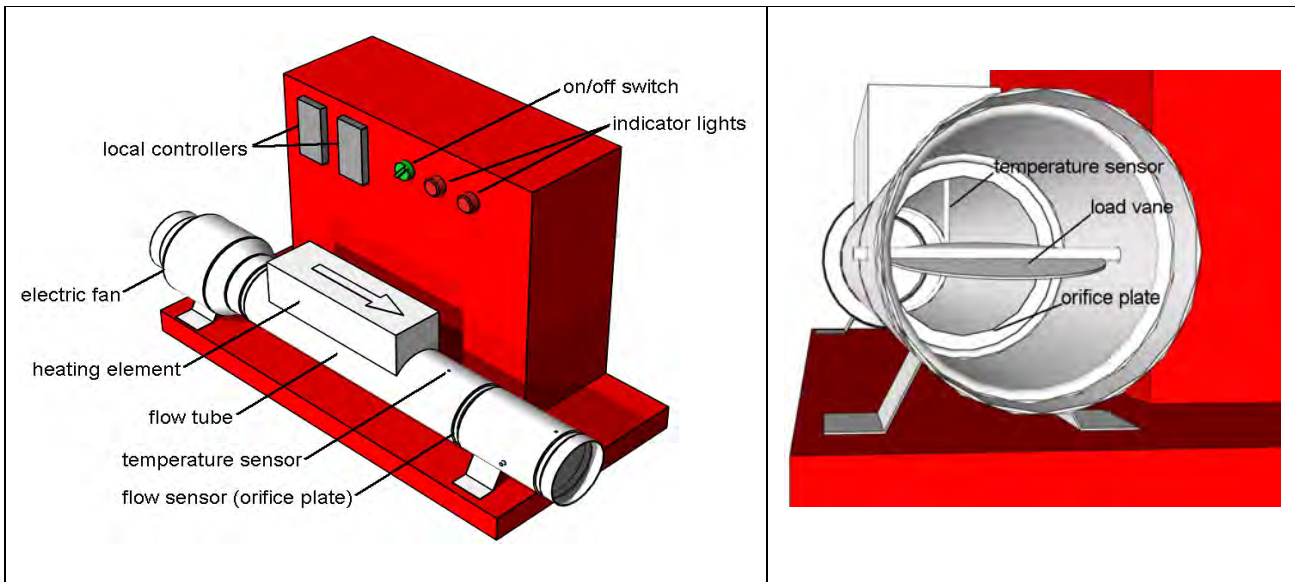
<sup>1</sup> ISO standard 7730 based on a seven point Likert scale and 10% dissatisfaction criterion (McQuiston *et. al.*, 2000).

office workers. Similar standards for ventilation rate (litres of air per second per person) are also available. These standards vary depending on several factors, such as maximum expected occupancy per 100m<sup>2</sup> and application type (i.e. public areas, educational institutions, hotels, garages, food and beverage services). An important problem to be considered in relation to these standards, for example, is the large amount of heat generated by telecommunications equipment. Thus highlighting the importance of heating and ventilation systems from a fire safety point of view.

It is important that a company maintains a work environment that is conducive to the well being of its employees. Satisfied employees are obviously more productive employees. “Leading-edge firms include well being factors such as health, safety, and ergonomics in their improvement activities” (Evans, 2002). So it is in a companies’ best interest to maintain a comfortable and safe work environment so that the company operates safely *and* profitably.

## 1.2 The Equipment

This project is concerned with the estimation and control of a *laboratory* heating and ventilation system. For obvious reasons it would be unfeasible to attempt this at full scale on an actual building with actual people in it. Nevertheless, the laboratory equipment provided functions in the same manner as a full-scale system. A 3-D model of the equipment used was drawn, and is shown in figure 1-2.



**Figure 1-2 Instrutek VVS-400 laboratory heating and ventilation rig**

An electric fan is located at the far end of the white tube and blows air over a heating element. The air exits to the surroundings at the near end of the tube. An orifice plate is situated just before the exit (see close up of inside the tube). The differential pressure across the orifice gives the flow rate. A platinum resistance temperature sensor is positioned inside the tube. A load vane provides a method of

restricting the airflow at the tube exit. The red box houses the power supply and other electrical components of the rig. Two independent local controllers for the flow and temperature processes, that have PI/PID and auto-tuning functions, are provided. It is possible to connect directly to the input and output of the fan and the heating element so that the processes may be controlled via a PC. Not shown are two flick switches that can be used to switch out the local controllers in favour of PC control.

### 1.3 The Software

#### 1.3.1 *Matlab*<sup>®</sup>

Matlab is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Matlab features a family of add-on application-specific solutions called toolboxes. Toolboxes allow the user to learn and apply specialized technology. Toolboxes are comprehensive collections of Matlab functions (M-files) that extend the Matlab environment to solve particular classes of problems, some of which include control systems and data acquisition. (Adapted from Matlab Help<sup>2</sup>).

#### 1.3.2 *Simulink*<sup>®</sup>

Simulink is a software package that operates in the Matlab environment. It enables the user to model, simulate, and analyse systems whose outputs change over time (dynamic systems). Simulink can be used to explore the behaviour of a range of real-world dynamic systems, including electrical circuits, shock absorbers, braking systems, and other electrical, mechanical, and thermodynamic systems. Simulating a dynamic system is a two-step process. First, a graphical model of the system to be simulated is created, using the Simulink model editor. The model depicts the time-dependent mathematical relationships among the system's inputs, states, and outputs. Then, the behaviour of the system is simulated over a specified time span. Simulink uses information that the user entered into the model to perform the simulation. (Adapted from Matlab Help<sup>3</sup>).

#### 1.3.3 *Humusoft*<sup>®</sup>

Humusoft provides an extended 'real time toolbox' that brings the power of MATLAB and Simulink to the real world. It allows the user to access external analog and digital signals via a data acquisition card. It provides a means of experimentation for control system design directly from the Simulink

---

<sup>2</sup> Matlab Help (version 6.5, release 13) – Directory tree: Matlab; Getting Started; Introduction.

<sup>3</sup> Matlab Help (version 6.5, release 13) – Directory tree: Simulink; Using Simulink; How Simulink Works.

environment. Real-time simulation is made easy – the user creates a simulation diagram, as normally done in Simulink, real-time input/output blocks are added from the real time toolbox library and the hardware is ready for simulation. The data acquisition board is represented by an ‘adapter block’. The difficult task of setting up a data acquisition board is simplified. A wide range of data acquisition boards are supported, including those manufactured by Advantech, Analog Devices, Axiom, Computer Boards, Data Translation, Humusoft, Keithley MetraByte, or National Instruments. (Adapted from Humusoft Website<sup>4</sup>).

#### **1.4 Aims and Objectives**

The main aim of this project is to estimate and control a laboratory heating and ventilation system. Initially, the methods of estimation (identification) or control to be used were undetermined. The outcome of each objective, on its completion, decided the nature of the next. The path chosen, that led to the final design, is justified throughout this report. The overall task was thus broken down into detailed objectives, as given below.

##### **1.4.1 *Familiarisation with Equipment***

This objective was carried out on a part-time basis prior to the May examinations. It consisted of the following activities:

- General investigation into process behaviour.
- Initial testing, i.e. carrying out previously existing laboratory experiments for the equipment.
- Controller design/testing using simple techniques.
- Interface the equipment with Matlab through Humusoft.
- Begin process identification using simple techniques.
- Investigate possibility of process interactions.

##### **1.4.2 *Further Identification and Investigation into Non-Linearities***

Subsequent to the completion of the previous objective the following activities were decided to be the most beneficial:

- Further and more complete temperature process identification.
- Static tests on the flow and temperature processes (to determine extent of non-linearity).
- Flow and temperature measurement accuracy tests.
- Process interaction tests.

---

<sup>4</sup> Humusoft Ltd., “Extended Real Time Toolbox – Product Description”, <http://www.humusoft.cz/rt/index.htm>, (6/8/2003).

### 1.4.3 *Controller Design*

When all identification and investigations into non-linearities were complete the following activities became clear:

- Choosing controller type to be used – implement and test.
- Investigation into inverse response compensation methods.
- Design, implement and test a static decoupling design.
- Design a simple gain scheduler.

### 1.4.4 *Final Design: Implementation and Validation Testing*

The results of the prior objectives led to a suitable final design stage where the activities were given as follows:

- Design advanced gain scheduler.
- Implement the design.
- Carry out validation testing.

## 1.5 Summary of Results

The initial tests and experiments carried out in chapter 2 gave a brief ‘picture’ of the processes’ behaviour. Simple control techniques were found to be inadequate. It was suspected that the processes were non-linear.

Chapter 3 details all identification procedures carried out. Step and frequency response methods were used. Three flow process models and nine temperature process models were determined. Each model obtained was categorised to specific operating regions of the processes. The flow process open loop characteristics were found to be faster at mid-to-high range flow rates. It was found that the temperature process characteristics depended on the conditions of the flow process. Hence models of the temperature process were determined at different flow rates. The temperature process gain was found to be higher at mid range input. It was found that a large variation in process model parameters existed in both processes.

In chapter 4 it was found that both processes were indeed non-linear in nature. Static input/output curves for each process were obtained. The flow process curve revealed that limits exist on its maximum and minimum operating region. The slope of the characteristic curve showed that the process has high gain at high inputs. The flow measurement accuracy of the rig was investigated. The orifice plate co-efficient of discharge was approximated to 0.55. It was therefore concluded that some flow measurement inaccuracies would be inherent in the closed loop feedback signal to the flow

process. The temperature process had different static characteristic curves at different flow rates. Three curves were found corresponding to three flow rates. At high the flow rates the maximum achievable temperature was decreased. The slope of the curve was greatest at mid-range temperatures. A lower temperature limit existed, for all curves, which was consistent with the ambient room temperature. A test measuring the accuracy of the temperature sensor revealed the rig consistently underestimates actual temperature by approximately 3°C. Finally, chapter 4 confirmed that an interaction exists between the output temperature and the input flow. Three models were obtained for the interaction corresponding to three temperature ranges. The model gain was negative and the interaction was more extreme at high temperatures. This established that a control strategy to eliminate or minimise process interactions would have to be found.

In Chapter 5, the Ideal PI and Classical PID controllers with suitable tuning rules, based on the step response identification results, were chosen to control the processes, with a view to minimising the integral of absolute error. Preliminary tests showed that the flow process controllers produced satisfactory performance, in regulator and servo mode. The temperature process was more difficult to control. An unexpected inverse like response was apparent, the cause for which was unknown. It was established that the behaviour could not be a true inverse response, but rather due to the way in which the simulation package operates, and/or due to the interacting effect of the flow process. An appropriately designed decoupler made a noticeable reduction in the so-called inverse response effect.

A simple gain scheduler design was developed that provided a method of skipping between controllers with fixed settings particular to given operating conditions. The method provided a suitable infrastructure for implementing a more complex design. The design incorporates look-up tables to continuously interpolate controller settings as process operating conditions vary. An alternative method of implementing an Ideal PI controller was found that utilises controller settings from look-up tables. Extending this design to a Classical PID controller was difficult because of derivative kick. A compromise of the simple gain scheduler design that interpolates for gain and integral time, but skips between specific 'D' controllers was found. A static decoupler was integrated into the design, the gain for which was determined through suitably designed look-up tables. The final design was named an 'advanced gain scheduler' (AGS).

Tests were carried out to compare the performance of the AGS to an average model controller (AMC) with both processes in regulator and servo mode, using Ideal PI and Classical PID control. The integral of absolute error (IAE) was the chosen performance criteria. The results show that in some cases the AGS outperformed the AMC. In other cases the AMC was clearly preferable to the AGS.



## 2. Investigation into Process Behaviour - Experiments

[Note: all experiments in this chapter were carried out at a room temperature of 23.5°C]

### 2.1 Local Flow Controller

#### 2.1.1 *PI Controller using Ziegler and Nichols Ultimate Cycle Method*

The output of the controlled process was connected to an oscilloscope. The load vane was opened fully and the set-point of the flow controller set to 30%. Integral and derivative settings were reset to zero and gradually the proportional band was decreased (equivalent to increasing the gain) until sustained oscillations were present in the output signal from the oscilloscope, see fig.2-1. The maximum measured proportional band and corresponding period of oscillation were;

$$P.B._U = 45\% \quad ; \quad T_U = 22 \text{ sec } s$$

Using the tuning rules given by Ziegler and Nichols (1942) PI controller settings were determined as follows;

$$P.B. = 2.2PB_U \quad ; \quad T_i = 0.83T_U$$

$$\therefore P.B. = 99\% \quad ; \quad T_i = 18.26 \text{ sec } s \approx 18 \text{ sec } s$$

These settings were applied to the local controller and simple step tests carried out. Plots of the servo response with a set value change of 30%-50% and then 50%-30% are shown in figures 2-2 and 2-3 below;

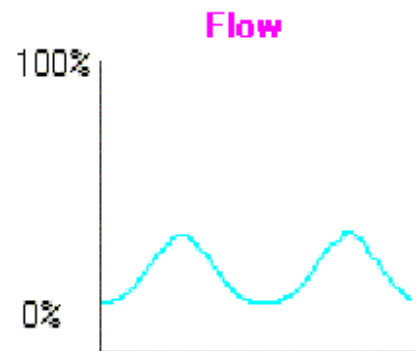


Figure 2-1 Sustained oscillations in output during continuous cycle test

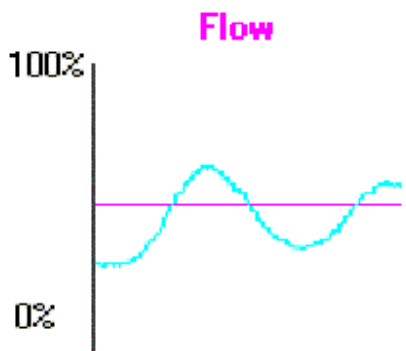


Figure 2-2 Servo response – 30%-50% step

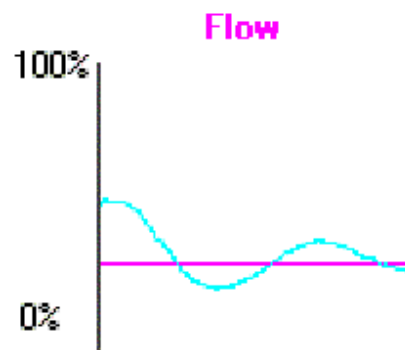


Figure 2-3 Servo response – 50%-30% step

As expected the responses very approximately exhibit a  $\frac{1}{4}$  decay ratio as per the design criteria. Although it should be noted that the response for an increasing step differed slightly to the response for a decreasing step. Regulator responses were also obtained. This was done by suddenly closing the load vane by approximately one third, allowing the signal to settle, and then suddenly opening the load vane again. The results of which are shown below in figures 2-4 and 2-5.

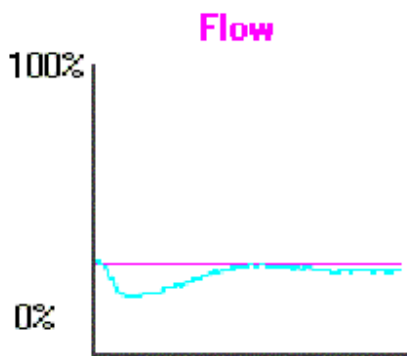


Figure 2-4 Regulator response - closing load vane by 1/3

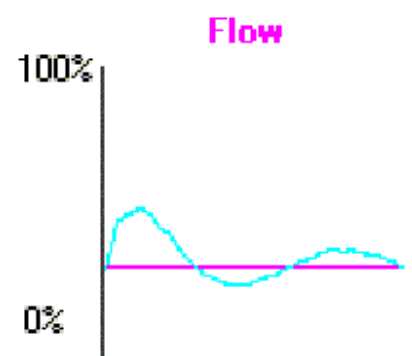


Figure 2-5 Regulator response opening load vane

The regulator response displayed a very visible difference in characteristic in these two plots. It seemed disturbances that required the system to increase flow (i.e. fig.2-4) were rejected faster than disturbances that require the system to decrease flow (i.e. fig.2-5). Both sets of responses suggest that the system is non-linear in nature perhaps with hysteresis present.

### 2.1.2 PID Controller using Ziegler and Nichols Ultimate Cycle Method

Using the PID tuning rules the appropriate controller settings were found using the same ultimate cycle information as for the PI controller design and went as follows;

$$P.B. = 1.7PB_U \quad ; \quad T_i = 0.5T_U \quad ; \quad T_d = 0.125T_U$$

$$\therefore P.B. = 76.5\% \quad ; \quad T_i = 11 \text{sec } s \quad ; \quad T_d = 2.75 \approx 3 \text{sec } s$$

The controller was set-up for these values and simple step tests were carried out. Plots of the servo response with a set value change of 30%-50% and then 50%-30% are shown in figures 2-6 and 2-7 below;

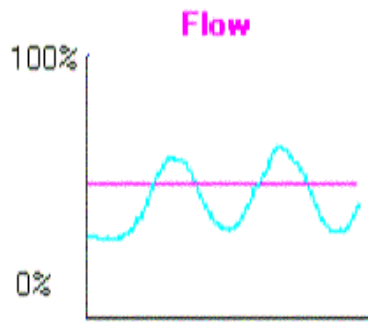


Figure 2-6 Servo response – 30%-50% step

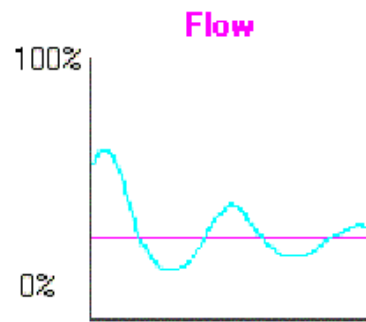


Figure 2-7 Servo response – 50%-30% step

The servo response for the PID controller again, as with the PI controller, seemed to behave differently depending on whether the step was increasing or decreasing. The regulator responses for the PID controller are shown in figures 2-8 and 2-9.

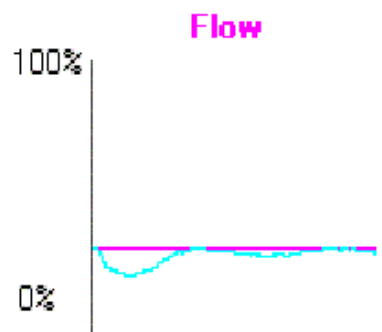


Figure 2-8 Regulator response – closing loads vane by 1/3

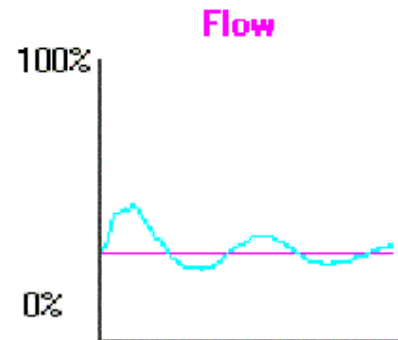


Figure 2-9 Regulator response - opening load vane fully

Again we see that the nature of the response is different depending on the direction of the disturbance. Therefore re-enforcing the point already mentioned that it is quite likely that non-linearities are inherent to the system. Another regulator response was carried out where a very large disturbance was introduced, see figure 2-10. The flow was unable to reject the disturbance and the fan motor was at full power, i.e. the system was saturated. Saturation is a non-linearity in itself and therefore this should be taken into consideration.

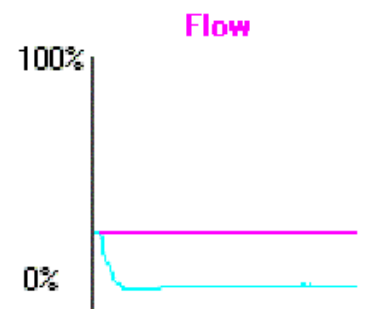


Figure 2-10 Large disturbance

### 2.1.3 Manual Tuning

After the observations made from the previous experiments, several sets of PI and PID controllers were set up manually and tested in terms of a step response from 30% to 50% of flow. The results and given settings are plotted in figure 2-11 below. It was possible to attain the speed of response, as it was noticed that the full length of the x-axis corresponded to 30 seconds.

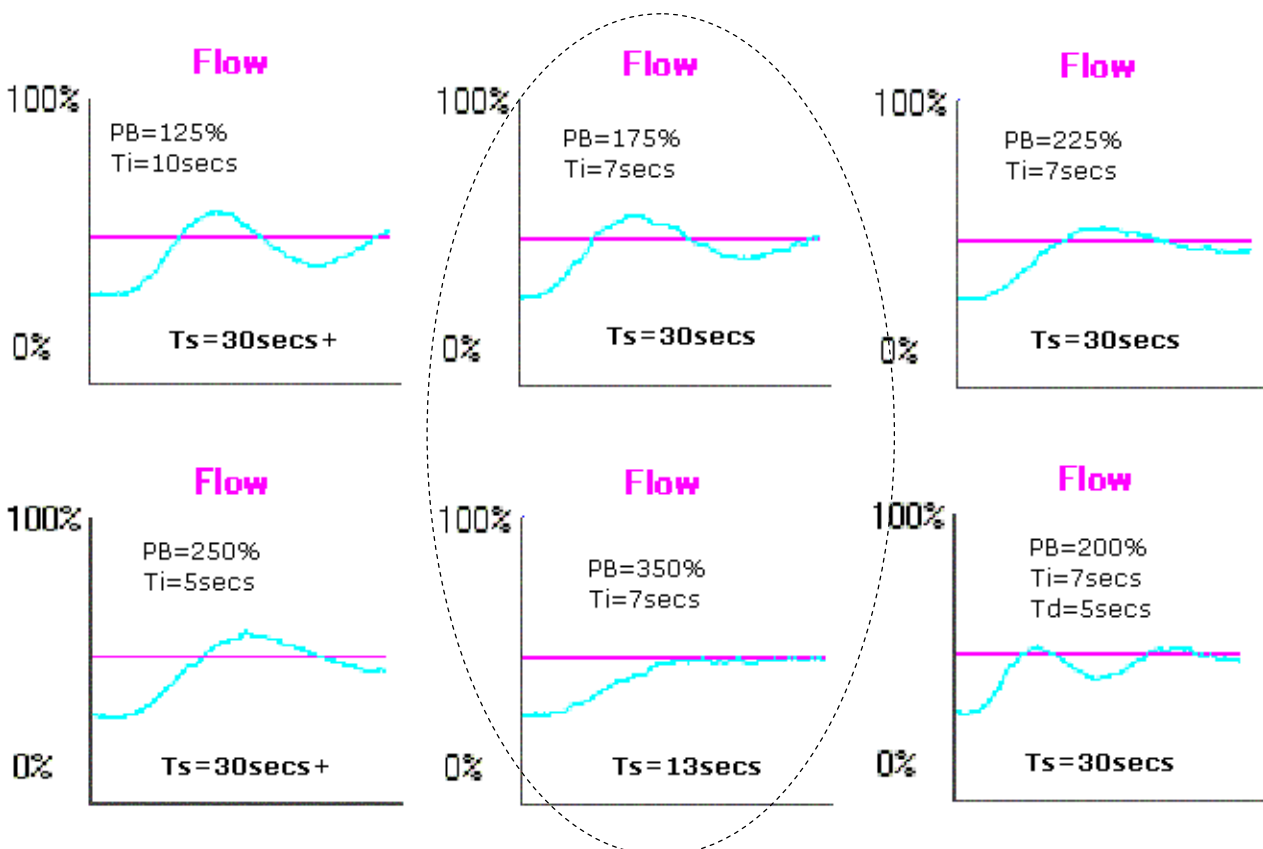


Figure 2-11 Manual tuning step responses with given settings and settling times

These results clearly display that at higher values of proportional band a lower settling time is achievable with less oscillations. This is what we would expect as a high proportional band corresponds to a low gain, and the higher the gain the more oscillatory the response. For example the plot for the PI controller, with  $PB=175\%$  and integral time  $T_i=7\text{secs}$ , displays an oscillatory response with settling time of 30secs, whereas if the proportional band is doubled to 350% the response becomes non-oscillatory with a settling time of 13secs. This type of controller tuning is difficult to do; it is simply a 'trial and error' exercise.

### 2.1.4 Auto-tuning

Built into the local controller is an auto-tuning function, where the controller issues a signal (or a series of signals) to the system and attempts to arrive at suitable PID controller settings for a given load. The method of auto-tuning is unknown but nevertheless it was worth examining how well it works.

Setting up the auto-tuner: (i) Press the 'parameter select key' to indicate A7  
(ii) Press the 'data' key and change the digit from 0 to 1  
(iii) Press the 'data entry key' to initiate auto-tuning

The decimal point in the right corner of the display flashes on and off while the controller auto-tunes. The controller was tuned in this way for a fully open load vane and for a one third closed load vane condition. The resulting PID controller settings and the corresponding step responses (30%-50% of flow) are shown in figure 2-12.

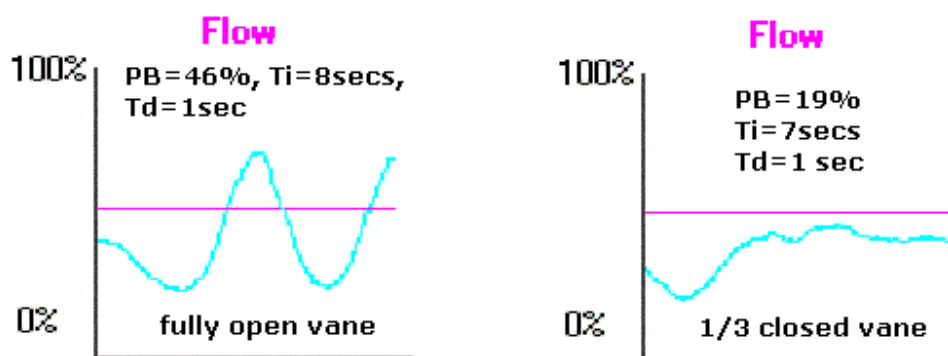


Figure 2-12 Auto-tuned PID controller step responses for open and 1/3 closed load vane

The quality of the tuning was clearly quite poor. In the first case above it was difficult to maintain the flow at 30% before the step was applied because the system was unstable to begin with. There was a similar problem in the second case above, although the response seemed to be somewhat better.

## 2.2 PC Flow Controller

### 2.2.1 PRO-REG Package

The rig was connected to a PC using the given input/output data acquisition card and PROCON – PROREG software. The connections from the rig to the input/output module (termination unit) were as shown below in figure 2-13.

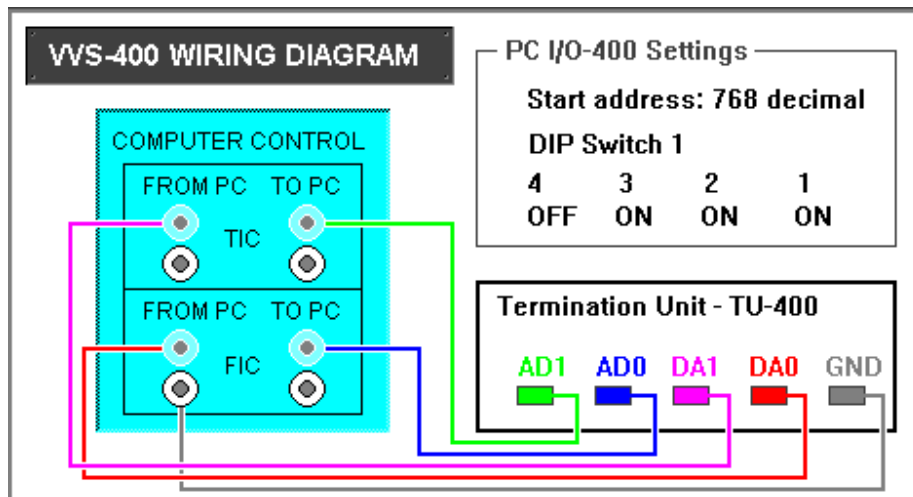


Figure 2-13 Wiring Diagram

### 2.2.2 PI Controller using Ziegler and Nichols Ultimate Cycle Method

The load vane was opened fully and the set-point of the flow controller set to 30%. Integral and derivative settings were reset to zero and gradually the proportional band was decreased (equivalent to increasing the gain) until sustained oscillations were present in the output signal from the oscilloscope. The maximum measured proportional band and corresponding period of oscillation were found to be;

$$P.B._v = 50\% ; T_v = 9.5 \text{ sec } s.$$

The Ziegler and Nichols PI tuning rules give settings of:  $P.B. = 110\% ; T_i = 8 \text{ sec } s$

The resulting servo and regulator responses for these settings are shown in fig. 2-14 below.

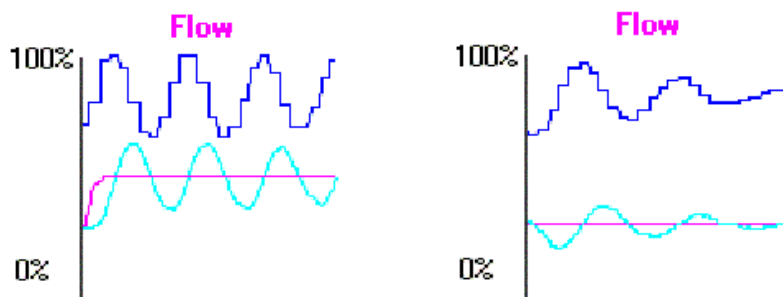


Figure 2-14 Z&N PI Ctr. (i) - Servo (30-50% step) and Regulator (load vane 1/3 closed) responses

The tuning parameters determined from the initial experiments using the local flow controller for the ultimate cycle method were also tested, the corresponding plots are shown below in fig. 2-15.

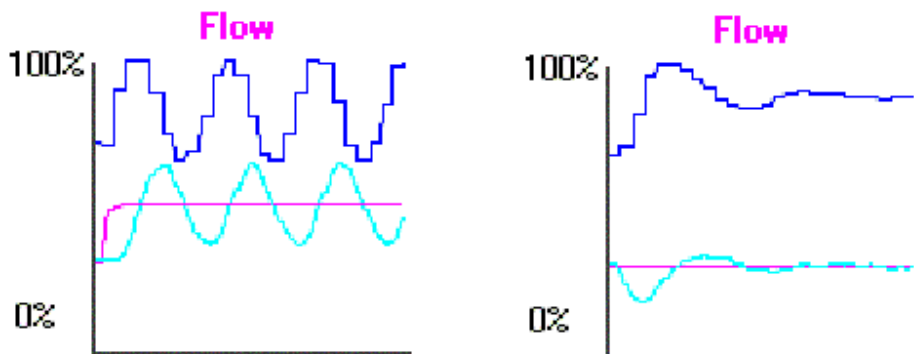


Figure 2-15 Z&N PI Ctr. (ii) - Servo (30-50% step) and Regulator (load vane1/3 closed) responses

In both cases the servo response is unsatisfactory while the regulator response is relatively good. The navy line of the plots represent the manipulated variable or the controller output. It should be considered that the local controller and the PROREG controller would have different sampling rates, hence explaining the difference in the tuning parameters ( $P.B.U$  and  $T_U$ ) obtained. This may be a possible explanation for the poor performance of this controller.

### 2.2.3 PID Controller using Ziegler and Nichols Ultimate Cycle Method

Using the PID tuning rules, the appropriate controller settings were found using the same ultimate cycle information as for the PI controller design.

$$P.B. = 1.7PB_U \quad ; \quad T_i = 0.5T_U \quad ; \quad T_d = 0.125T_U$$

$$\therefore P.B. = 76.5\% \quad ; \quad T_i = 11 \text{ sec } s \quad ; \quad T_d = 2.75 \approx 3 \text{ sec } s$$

The controller was set-up for these values. Plots of the servo and regulator responses are shown in fig 2-16 below.

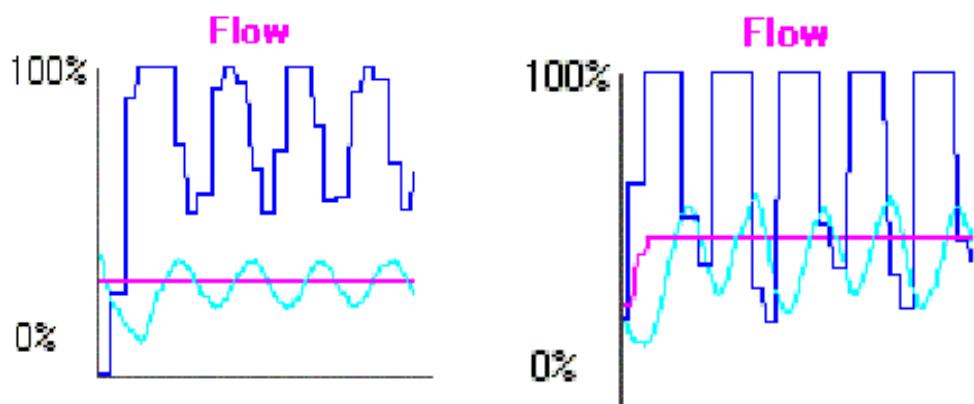


Figure 2-16 Z&N PID Ctr. - Servo (30-50% step) and Regulator (load vane1/3 closed) responses

It is quite clear that the PID controller tuning rules yield unacceptable results. This was most likely due to poor tuning.

#### 2.2.4 Tests at lower flows

The flow was set to 15% and  $P.B.U$  and  $T_U$  were found to be;  $P.B.U = 20\%$ ;  $T_U = 10\text{sec } s$

The resulting PI and PID settings were:

**PI Ctr.:**  $P.B. = 44\%$  ;  $T_i = 8.3\text{sec } s$

**PID Ctr.:**  $P.B. = 34\%$  ;  $T_i = 5\text{sec } s$  ;  $T_d = 1.25 \approx 1\text{sec } s$

The servo and regulator plots, for a step of 15-30% and disturbance caused by closing the load vane suddenly by 1/3, for both PI and PID controllers are shown in fig.2-17 below.

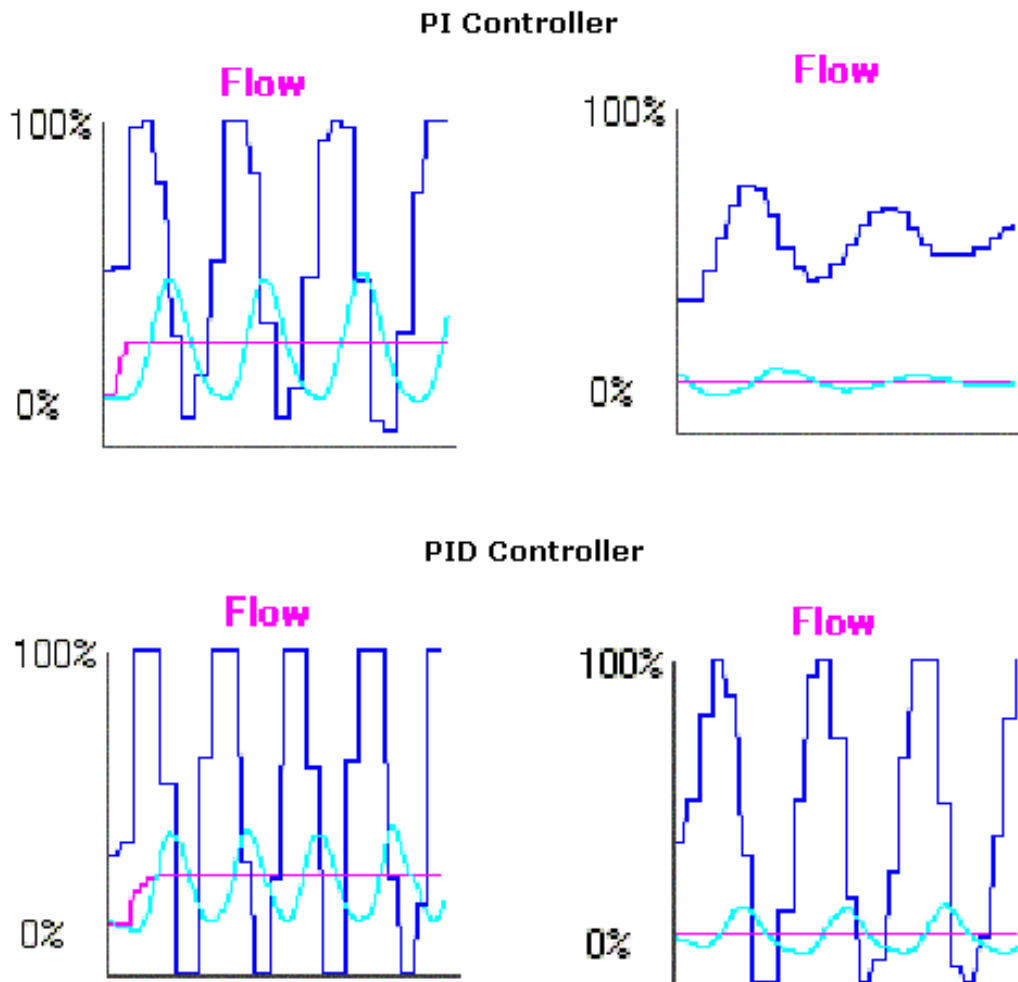


Figure 2-17 Z&N PI & PID Ctr. - Servo (15-30% step) and Regulator (load vane 1/3 closed) responses

The plots show that no real improvement of system performance was gained by reducing the operating point to 15% and re-tuning. Controller saturation still existed, in particular for the PID controller.



### 2.2.5 Manual Tuning

The PI controller was tuned manually by trial and error for optimal servo and regulator control. Some plots that were used to determine the best or optimum PI controller are shown below in fig.2-18.

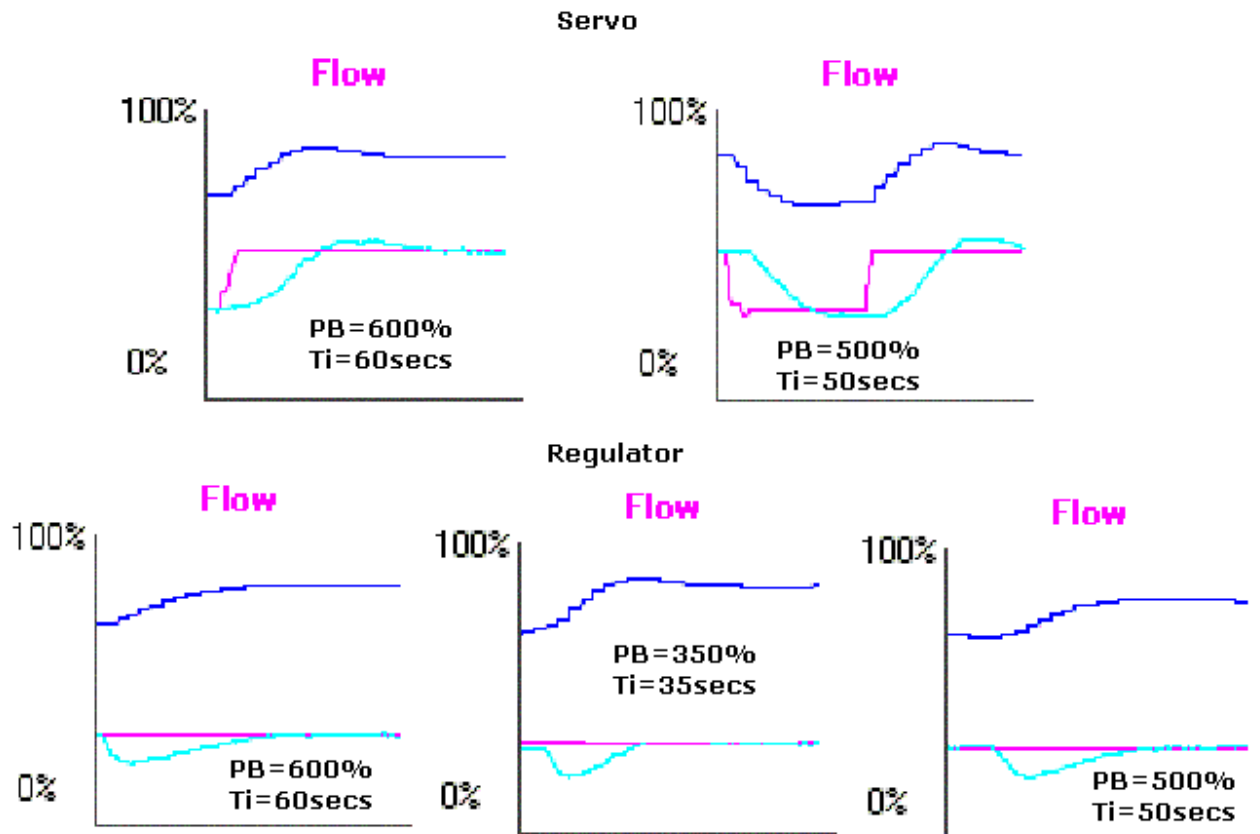


Figure 2-18 Selected servo and regulator responses for optimal PI control

From the plots it was clear that a PI controller with proportional band of 500% and integral time of 50secs produced quite good servo and regulator responses. These settings were consequently used during any of the tests regarding the temperature control to keep the flow at some constant level.

### 2.3 Temperature Control

The temperature process is very slow compared to the flow process, i.e. the combined time constant and time delay are bigger than that of the flow process. It was found that a simple step test on the temperature process took approximately 700secs. For this reason the temperature process was not as convenient to test as the flow process. No temperature tests were carried out using the PRO-REG software.

### 3. Process Identification

#### 3.1 Interfacing Hardware with Matlab/Humusoft

The Pro-Reg software used to provide the information detailed in chapter 2 only offers a visual representation of the processes behaviour. To carry out accurate process identification the rig had to be linked to more comprehensive software (Matlab), through an appropriate data acquisition package (Humusoft). This way the user can obtain raw data from the process as the process inputs are changed, for example the flow may be held constant as the input to the temperature process is changed; and then the subsequent change in output temperature can be measured, in terms of a voltage, in real time. This information can be then used to estimate the process parameters using various methods of identification.

##### 3.1.1 Procedure for interfacing the rig with Matlab/Humusoft

Open Matlab and open a new Simulink model file. The version of Matlab used must contain the ‘Real-time Toolbox’. This can be checked by looking in the Simulink Library browser, figure 3-1. Select the ‘Adapter’ from the toolbox and drag it into the model file window, this is the data acquisition card that should have been previously installed in the PC. To activate the ‘Adapter’ a driver must be specified. Matlab should prompt you to do this – select ‘dt2811h.rtd’. Select a real-time sink (‘RT Out’) and a real-time source (‘RT In’) from the appropriate sub-directories of the library tree and drag them into the model file. Each process will have one source and one sink. Each block is referenced in relation to the PC, so ‘RT In’ implies a signal coming into the PC from the rig (i.e. either the voltage from the temperature or flow measurement device) and ‘RT Out’ is the signal sent out from the PC to the process (i.e. the input to either process). The input and output blocks must be configured to the same sampling rate, and the appropriate adapter channels specified. The data acquisition card has two input/output channels — use one process per channel. Figure 3-2 shows the how a real-time sink can be set up and how the adapter card should be configured.

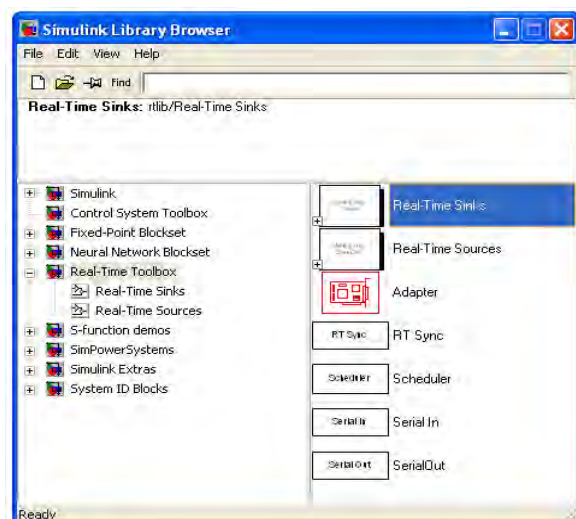


Figure 3-1 Simulink library browser

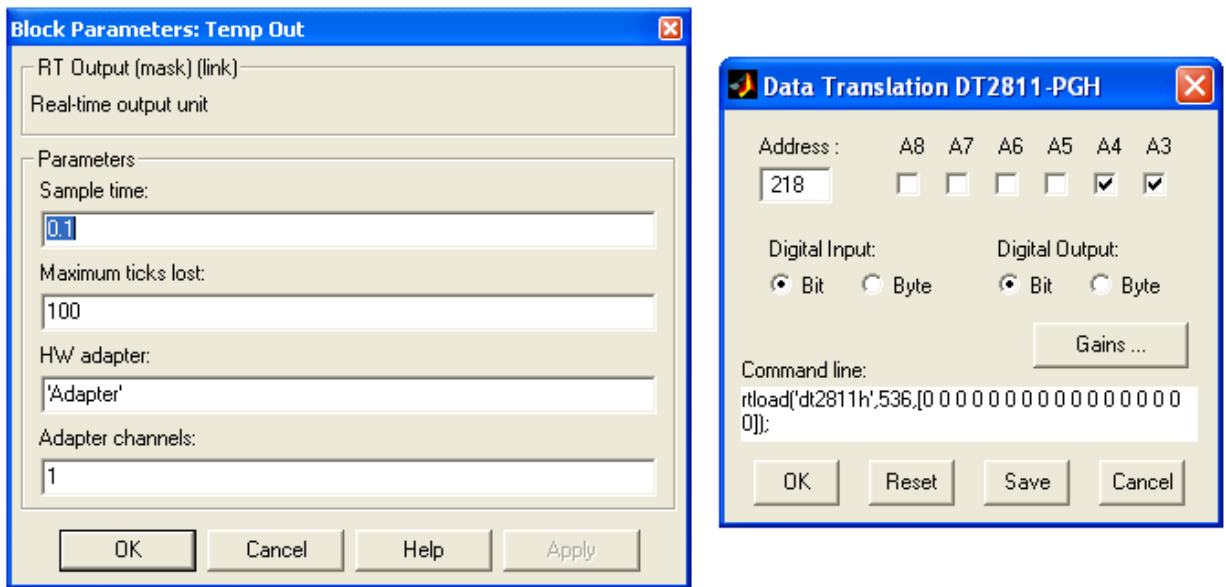


Figure 3-2 Real-time sink parameters and Adapter card set-up

Now, the software should be ready to interface with the hardware, i.e. wires from the process' inputs and outputs can be connected to the data acquisition card and the rig can be turned on. Inputs to the processes can be defined, i.e. step input or constant value, and the process' input and output can be viewed simultaneously by using a multiplexer connected to a scope, and subsequently plotted in Matlab using a workspace data sink. Figure 3-3 shows a typical set-up with the relevant blocks indicated.

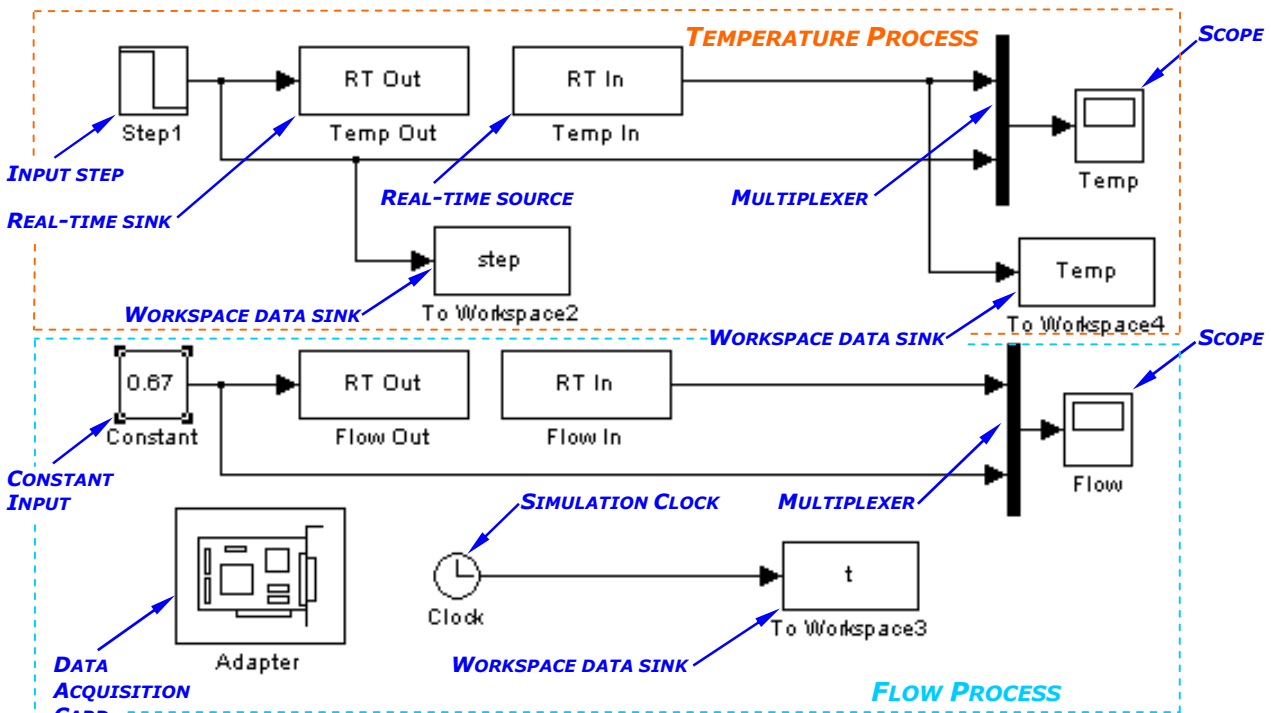


Figure 3-3 Typical Simulink set-up

Both processes take 0-5 Volts<sup>5</sup>. In Matlab/Humusoft this corresponds to an input or output of 0-1.

### 3.2 Flow Process

Two simple methods of identification were used to approximate the flow process. These were a range of open loop step tests and a frequency response experiment.

#### 3.2.1 Flow Step Tests – Load Vane Fully Open

Open loop step tests using small steps over the full range of flow were carried out. The alternative tangent and point method was used to approximate the process as a first order lag plus time delay (FOLPD) model. Figure 3-4 outlines this method of identification.

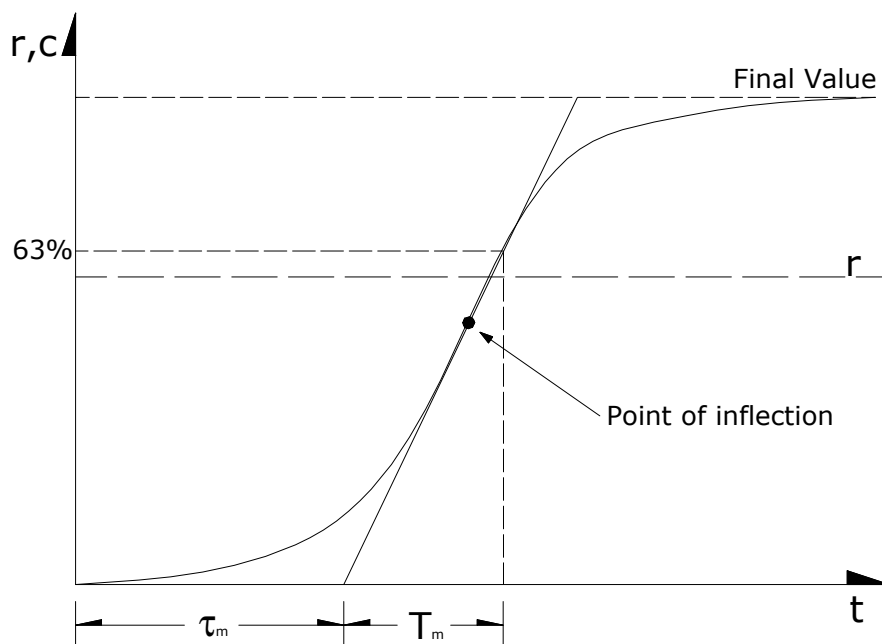


Figure 3-4 Alternative tangent and point step response identification method

The process model transfer function is then given by:

$$G_m(s) = \frac{K_m e^{-s\tau_m}}{1 + sT_m}$$

Given that:

$$K_m = \frac{\text{Final Value of } c}{r}$$

It was proposed that step tests should be carried out over a range of flows so that any non-linearity in the process could be determined. The static tests on the flow process showed that at any input less than 30%, no change in flow occurred. For this reason step tests began from 30% of flow. The step size chosen was 20%. Increasing and decreasing step responses were obtained experimentally and imported

<sup>5</sup> As specified in the Instrutek VVS-400 trainer manual.

into the Matlab/Simulink environment using HUMUSOFT. Figure 3-5 shows the Simulink set-up for the test. It should be noted that the temperature process was kept constant at room temperature (i.e. temperature process input of 0.24 corresponding to  $\approx 24^{\circ}\text{C}$ ).

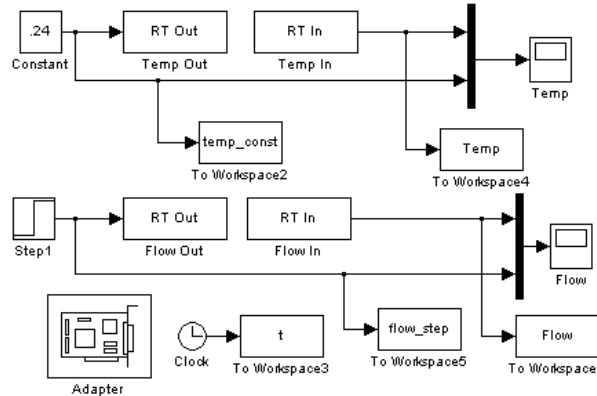


Figure 3-5 Simulink model file for flow open loop step test

The alternative tangent and point method was applied to each plot to determine the model parameters. All plots are given in appendix A. Figure 3-6 below shows an example of one such plot with the alternative tangent and point method applied.

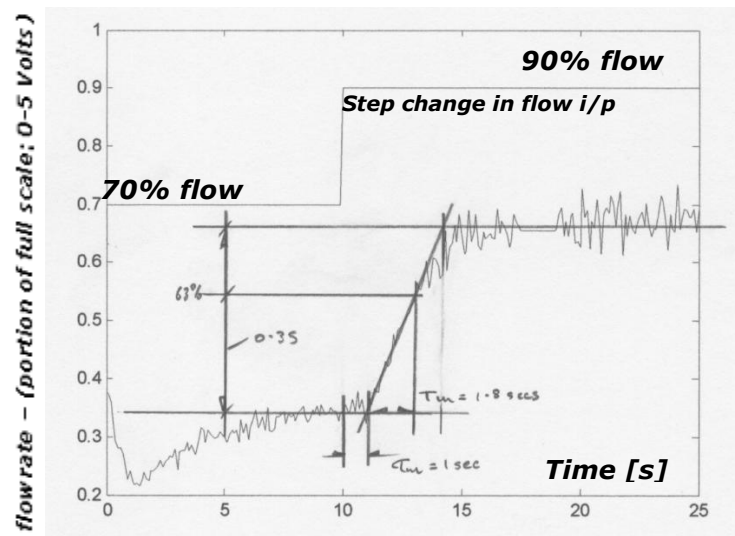


Figure 3-6 Example of alternative tangent and point method applied to a flow step response

It should be noted that during the first 5 to 7 seconds of the test the flow is adjusting to the initial value of the step (i.e. 0.7 = 70% flow). Also Simulink tends to momentarily cut the power to the plant when ‘run’ is clicked. This is evident in all the plots. When all of the plots were obtained, the change in output ( $\Delta o/p$ ), time constant ( $T_m$ ), and time delay ( $\tau_m$ ) were recorded. Each measurement was scaled by a ruler from the axes of the plot. All results are shown in Table3-1.

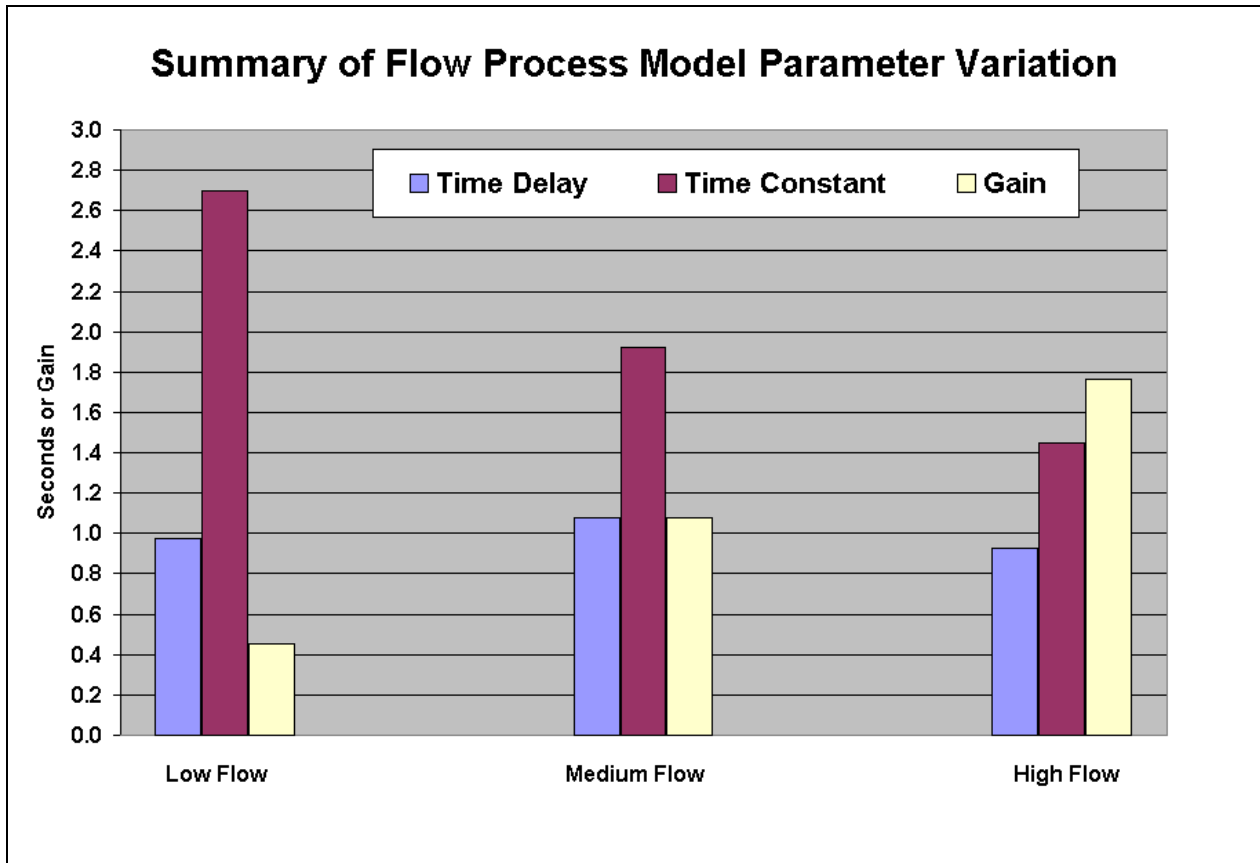
| <i>Flow Step Test Data @ 23.5 °C room temperature</i> |      |                |                |                |  |                |                |                    |                |      |
|---|------|----------------|----------------|----------------|--|----------------|----------------|--------------------|----------------|------|
| <i>Increasing Step</i>                                |      |                |                |                | <b>Average</b>                               |                |                |                    |                |      |
| step i/p  | Δo/p | τ <sub>m</sub> | T <sub>m</sub> | K <sub>m</sub> | τ <sub>m</sub>                               | T <sub>m</sub> | K <sub>m</sub> |                    |                |      |
| 0.3 - 0.5   | 0.07 | 0.7            | 4.3            | 0.35           | Low Flow                                     |                |                |                    |                |      |
| 0.4 - 0.6   | 0.11 | 1.0            | 3.0            | 0.55           | 0.85   | 3.65           | 0.45           |                    |                |      |
| 0.5 - 0.7   | 0.17 | 1.2            | 2.5            | 0.85           | Medium Flow                                  |                |                |                    |                |      |
| 0.6 - 0.8   | 0.28 | 1.1            | 2.0            | 1.40           | 1.15   | 2.25           | 1.13           | <b>Average</b>     |                |      |
| 0.7 - 0.9   | 0.35 | 1.0            | 1.8            | 1.75           | High Flow                                    |                |                | τ <sub>m</sub>     | T <sub>m</sub> |      |
| 0.8 - 1.0   | 0.36 | 0.9            | 1.5            | 1.80           | 0.95   | 1.65           | 1.78           | <b>Low Flow</b>    |                |      |
| <b>'load vane fully open'</b>                         |      |                |                |                | <i>Units of input and output: 1 = 5Volts</i> |                |                | 0.98               | 2.70           | 0.45 |
| <i>Decreasing Step</i>                                |      |                |                |                | <b>Average</b>                               |                |                | <b>Medium Flow</b> |                |      |
| step i/p  | Δo/p | τ <sub>m</sub> | T <sub>m</sub> | K <sub>m</sub> | τ <sub>m</sub>                               | T <sub>m</sub> | K <sub>m</sub> | 1.08               | 1.93           | 1.08 |
| 1.0 - 0.8   | 0.33 | 1.0            | 1.0            | 1.65           | High Flow                                    |                |                | <b>High Flow</b>   |                |      |
| 0.9 - 0.7   | 0.37 | 0.8            | 1.5            | 1.85           | 0.90   | 1.25           | 1.75           | 0.93               | 1.45           | 1.76 |
| 0.8 - 0.6   | 0.25 | 1.0            | 1.5            | 1.25           | Medium Flow                                  |                |                |                    |                |      |
| 0.7 - 0.5   | 0.16 | 1.0            | 1.7            | 0.80           | 1.00   | 1.60           | 1.03           |                    |                |      |
| 0.6 - 0.4   | 0.10 | 1.0            | 1.5            | 0.50           | Low Flow                                     |                |                |                    |                |      |
| 0.5 - 0.3   | 0.08 | 1.2            | 2.0            | 0.40           | 1.10   | 1.75           | 0.45           |                    |                |      |
|   |      |                |                |                | flows < 55% are 'low'                        |                |                |                    |                |      |
|   |      |                |                |                | flows >55% and < 75% are 'medium'            |                |                |                    |                |      |
|   |      |                |                |                | flows > 75% are 'high'                       |                |                |                    |                |      |
| <b>Overall Average</b>                                |      | 0.99           | 2.03           | 1.10           |  |                |                |                    |                |      |

Table 3-1 Summary of results from flow open loop step tests

The table shows data obtained for increasing and decreasing steps. In each case the average value of time delay, time constant and process gain was obtained for low (<55%), medium (55%-75%), and high (>75%) rates of flow. Then average values of these two sets of figures were found and are shown in the coloured boxes above. This information gives three process models, as follows;

$$G_{m_{LOW-FLOW}}(s) = \frac{0.45e^{-0.98s}}{1 + 2.70s} \quad G_{m_{MED-FLOW}}(s) = \frac{1.08e^{-1.08s}}{1 + 1.93s} \quad G_{m_{HIGH-FLOW}}(s) = \frac{1.76e^{-0.93s}}{1 + 1.45s}$$

The results show quite a large variance in the process gain and time constant while the time delay remains fairly constant, as would be expected, because it is related to the transport time of the air in the tube. This shows that the process is indeed non-linear. Figure 3-7 graphically illustrates how the process model parameters vary over the range of flows.



**Figure 3-7 Summary of flow process model parameter variation**

The process has a high time constant and low gain at low flows. At high flows the time constant is low and the gain is high. The time delay remains relatively constant over the range of flows. From this it is clear that the process performs faster at flows greater than 75% of its full range, i.e. 'high flows'.

### 3.2.2 Flow Frequency Response Test

The flow process was also modelled in the frequency domain. A range of sine waves over different frequencies were input to the flow process while the temperature process was held constant at room temperature, with the load vane fully open. The Simulink set-up used is shown in figure 3-8 below.

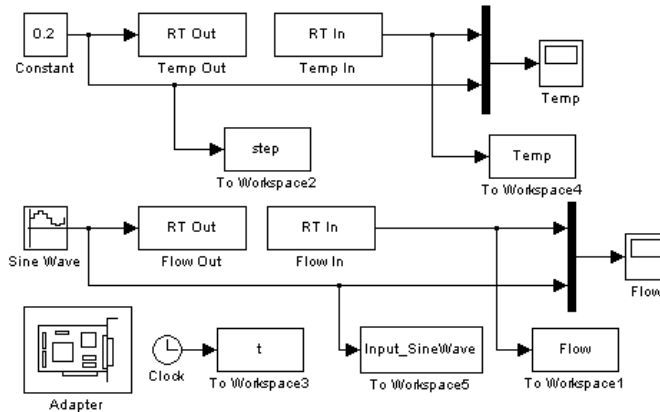


Figure 3-8 Simulink model file used for flow frequency response test

The peak-to-peak amplitude and phase shift of the output was recorded for each frequency. An example of one such plot is shown in figure 3-9. Note: All plots are given in appendix B. The figure

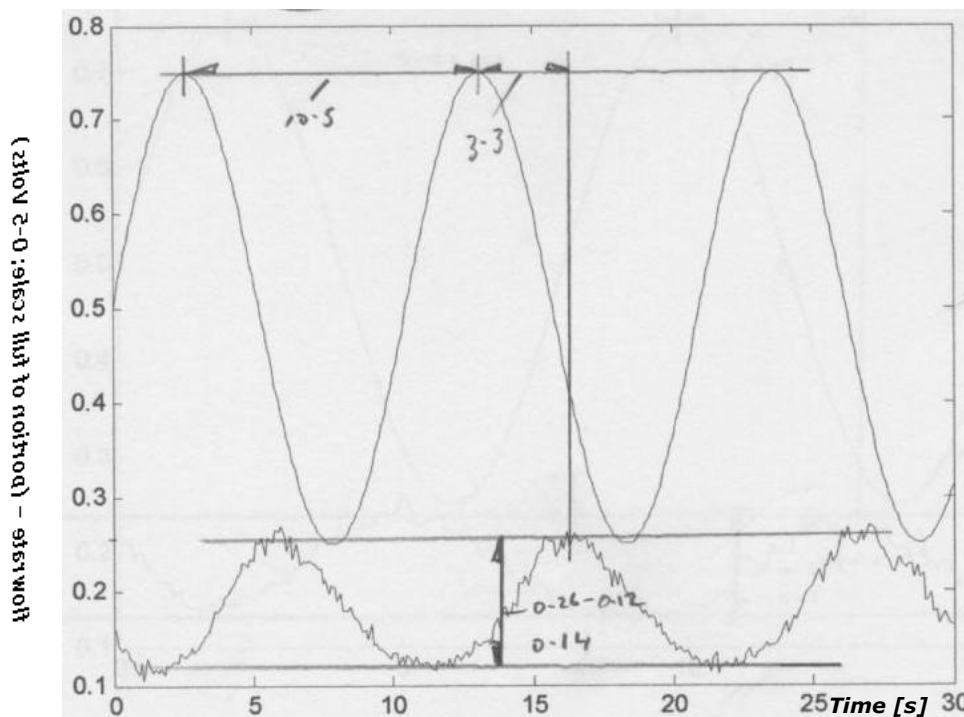


Figure 3-9 Frequency response plot with measurements shown

shows the measurements for the period of the input sine wave, the phase shift of the response, and the peak-to-peak amplitude of the response. This information from each plot over a range of frequencies



was gathered together, from which it was possible to draw a Bode plot for the process. Table 3-2 shows the data obtained for the Bode plot.

***Sine wave Input - peak to peak amplitude 0.5  
from 25% to 75% flow - room temperature 24°C***

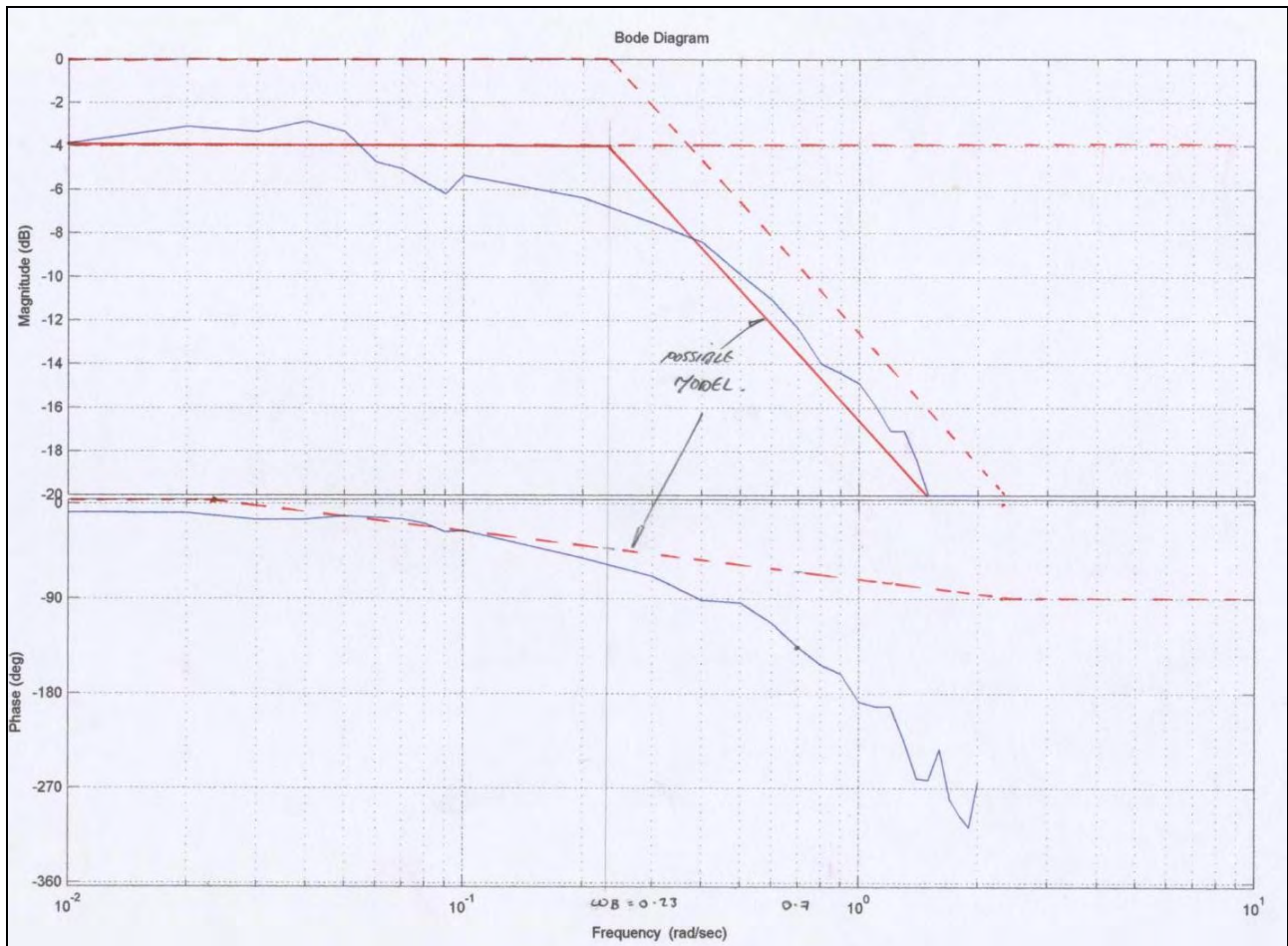
| Input     |        |        | Output        |           | Bode Plot Data |      |
|-----------|--------|--------|---------------|-----------|----------------|------|
| Frequency | Period | Shift  | peak-peak amp | Magnitude | Phase          |      |
| [rads/s]  | [Hz]   | [s]    | [s]           | [%]       | [dB]           | [°]  |
| 0.01      | 0.002  | 628.32 | 14.28         | 0.32      | -3.88          | -8   |
| 0.02      | 0.003  | 314.16 | 8.00          | 0.35      | -3.10          | -9   |
| 0.03      | 0.005  | 209.44 | 9.03          | 0.34      | -3.35          | -16  |
| 0.04      | 0.006  | 157.08 | 6.59          | 0.36      | -2.85          | -15  |
| 0.05      | 0.008  | 125.66 | 4.15          | 0.34      | -3.35          | -12  |
| 0.06      | 0.010  | 104.72 | 3.90          | 0.29      | -4.73          | -13  |
| 0.07      | 0.011  | 89.76  | 3.63          | 0.28      | -5.04          | -15  |
| 0.08      | 0.013  | 78.54  | 4.15          | 0.26      | -5.68          | -19  |
| 0.09      | 0.014  | 69.81  | 5.20          | 0.25      | -6.20          | -27  |
| 0.10      | 0.016  | 62.83  | 4.50          | 0.27      | -5.35          | -26  |
| 0.20      | 0.032  | 31.42  | 4.50          | 0.24      | -6.38          | -52  |
| 0.30      | 0.048  | 20.94  | 4.00          | 0.21      | -7.54          | -69  |
| 0.40      | 0.064  | 15.71  | 4.00          | 0.19      | -8.40          | -92  |
| 0.50      | 0.080  | 12.57  | 3.28          | 0.16      | -9.90          | -94  |
| 0.60      | 0.095  | 10.47  | 3.30          | 0.14      | -11.06         | -113 |
| 0.70      | 0.111  | 8.98   | 3.40          | 0.12      | -12.40         | -136 |
| 0.80      | 0.127  | 7.85   | 3.33          | 0.10      | -13.98         | -153 |
| 0.90      | 0.143  | 6.98   | 3.13          | 0.10      | -14.42         | -161 |
| 1.00      | 0.159  | 6.28   | 3.28          | 0.09      | -14.89         | -188 |
| 1.10      | 0.175  | 5.71   | 3.05          | 0.08      | -15.92         | -192 |
| 1.20      | 0.191  | 5.24   | 2.80          | 0.07      | -17.08         | -193 |
| 1.30      | 0.207  | 4.83   | 3.02          | 0.07      | -17.08         | -225 |
| 1.40      | 0.223  | 4.49   | 3.25          | 0.06      | -18.42         | -261 |
| 1.50      | 0.239  | 4.19   | 3.05          | 0.05      | -20.00         | -262 |
| 1.60      | 0.255  | 3.93   | 2.54          | 0.05      | -20.00         | -233 |
| 1.70      | 0.271  | 3.70   | 2.88          | 0.05      | -20.00         | -281 |
| 1.80      | 0.286  | 3.49   | 2.87          | 0.05      | -20.00         | -296 |
| 1.90      | 0.302  | 3.31   | 2.82          | 0.05      | -20.00         | -307 |
| 2.00      | 0.318  | 3.14   | 2.30          | 0.05      | -20.00         | -264 |

Table 3-2 Bode plot Data

A Bode plot could then be drawn. It was possible to do this in Matlab by importing the data from table 2-2 as vectors ('frequency', 'magnitude', and 'phase') and by then using the 'bode' function. The code used is as follows;

```
response=mag.*exp(i*phase)    %convert from polar to Cartesian form
sys=frd(response,freq)        %continuous-time frequency response data model
bode(sys); grid                %draw bode plot
```

The resulting Bode plot is shown in figure 3-10. Note: a larger version is available in appendix C.



**Figure 3-10 Bode Plot – flow process at mid to high range flows**

A possible FOLPD model was fitted to the magnitude plot from which a value for the model gain could be estimated;

$K_m = -4dB = 10^{\frac{-4}{20}} = 0.63$  An approximate break frequency ( $\omega_B$ ) was apparent at 0.23 [rads/s] at which the magnitude begins to decay at a rate of -20dB per decade. Therefore the first order term is

given by;  $\frac{1}{1+Ts}$  where;  $T = \frac{1}{\omega_b} = \frac{1}{0.23} = 4.35 \text{ sec}$

By fitting  $\frac{1}{1+4.35s}$  on the phase data it is revealed that a time delay is unaccounted for. This may be estimated by looking at a particular frequency, measuring the first order model phase, and measuring the actual phase. The difference between them is the extra phase attributed to the time delay, i.e.;

Taking;  $\omega_1 = 0.7 \text{ rad/s}$

Actual phase =  $-135^\circ$       model phase =  $-65^\circ$

$\therefore$  phase lag due to delay =  $-70^\circ = -1.22 \text{ rads}$

$\therefore -\omega_1 \tau_m = -1.22$        $\therefore \tau_m = \frac{-1.22}{-0.7} = 1.75 \text{ sec}$

Also try;  $\omega_2 = 0.2 \text{ rad/s}$

Actual phase =  $-53^\circ$       model phase =  $-35^\circ$

$\therefore$  phase lag due to delay =  $-18^\circ = -0.314 \text{ rads}$

$\therefore -\omega_2 \tau_m = -0.314$        $\therefore \tau_m = \frac{-0.314}{-0.2} = 1.57 \text{ sec}$

Therefore ; Average time delay,  $\tau_m = 1.66 \text{ secs}$

The resulting possible process model for low-to-medium range flows is given by;

$$G_{m_{\text{LOW-MED-FLOW}}}(s) = \frac{0.63e^{-1.66s}}{1+4.35s} \dots \text{from frequency response identification}$$

The input sine waves used to generate the Bode plot data had amplitude of 0.25 about 0.5. So the input to the fan varied from 1.25V to 3.75V (i.e. 25% to 75% of 5Volts). This is the low to medium flow range, as defined by the previous step test identification. It was possible to make a comparison to the results from the step tests by taking an average of the low and medium flow process models;

$$G_{m_{\text{LOW-MED-FLOW}}}(s) = \frac{0.76e^{-1.03s}}{1+2.31s} \dots \text{from step test identification}$$

Although the time constants are quite different, the system time delay and gain are quite similar. The difference in models highlights the problem of modelling in the frequency domain. That is, the model time constant can vary a great deal, depending on where the  $-20\text{dB/decade}$  line is cuts the zero decibel line. The Bode plot is estimated from a number of specific points, and the resulting plot may not suit the selected model as well as expected. This leads to inescapable room for variation in the model obtained. For this reason the model parameters used for controller tuning in chapter 5 were based on step response models only.

### 3.3 Temperature Process

The temperature process is very slow compared to the flow process. It was found that the combined system time constant and time delay could be as high as five minutes. For this reason it was not practical to carry out frequency response tests. Nevertheless simple step tests were carried out.

#### 3.3.1 Temperature Step Tests

Open loop step tests using small steps over the full range of temperature were carried out. The alternative tangent and point method was used to approximate the process as a first order lag plus time delay (FOLPD) model as was explained in section 3.1.1. It was proposed that step tests should be carried out over a range of temperatures and at different flow rates so that any non-linearity in the process could be accounted for. The ambient room temperature determined the minimum of the temperature range to be tested. The step size chosen was 10%. Increasing and decreasing step responses at three flow rates (30%, 50%, and 70%) were obtained experimentally. Figure 3-11 shows the set-up to record the data from the test. It should be noted that the flow process was kept constant (at 30% for this test, i.e. corresponding to a constant input to the flow process of 0.67).

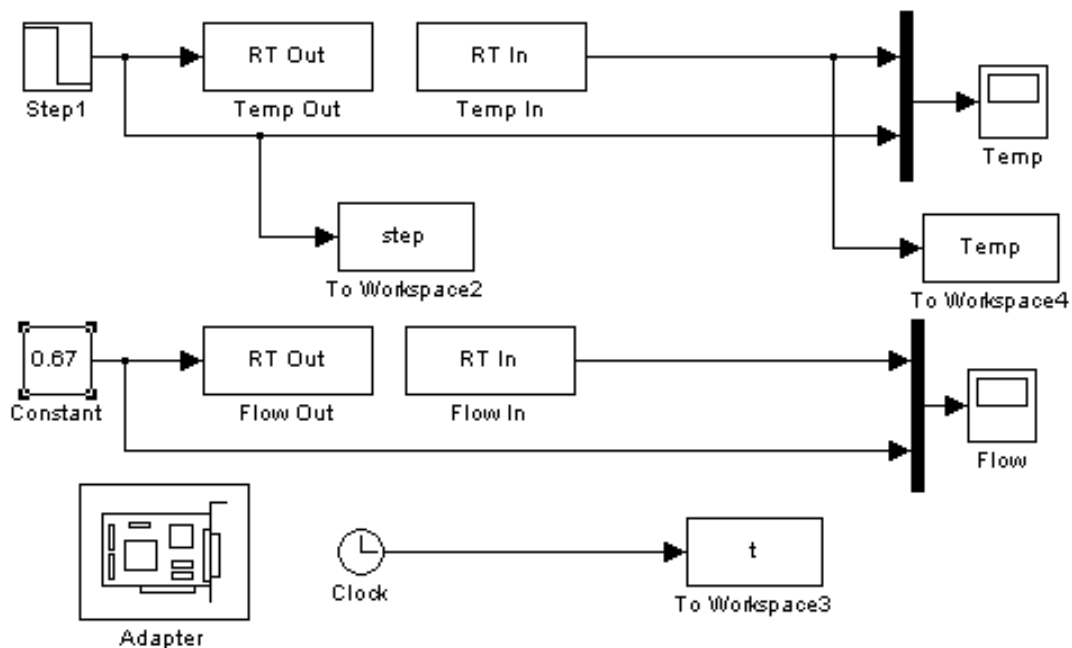


Figure 3-11 Simulink model file for temperature open loop step test

The alternative tangent and point method was used on each plot to determine the model parameters. All plots are given in appendices D, E and F. Figure 3-12 shows an example of one such plot with the alternative tangent and point method applied.

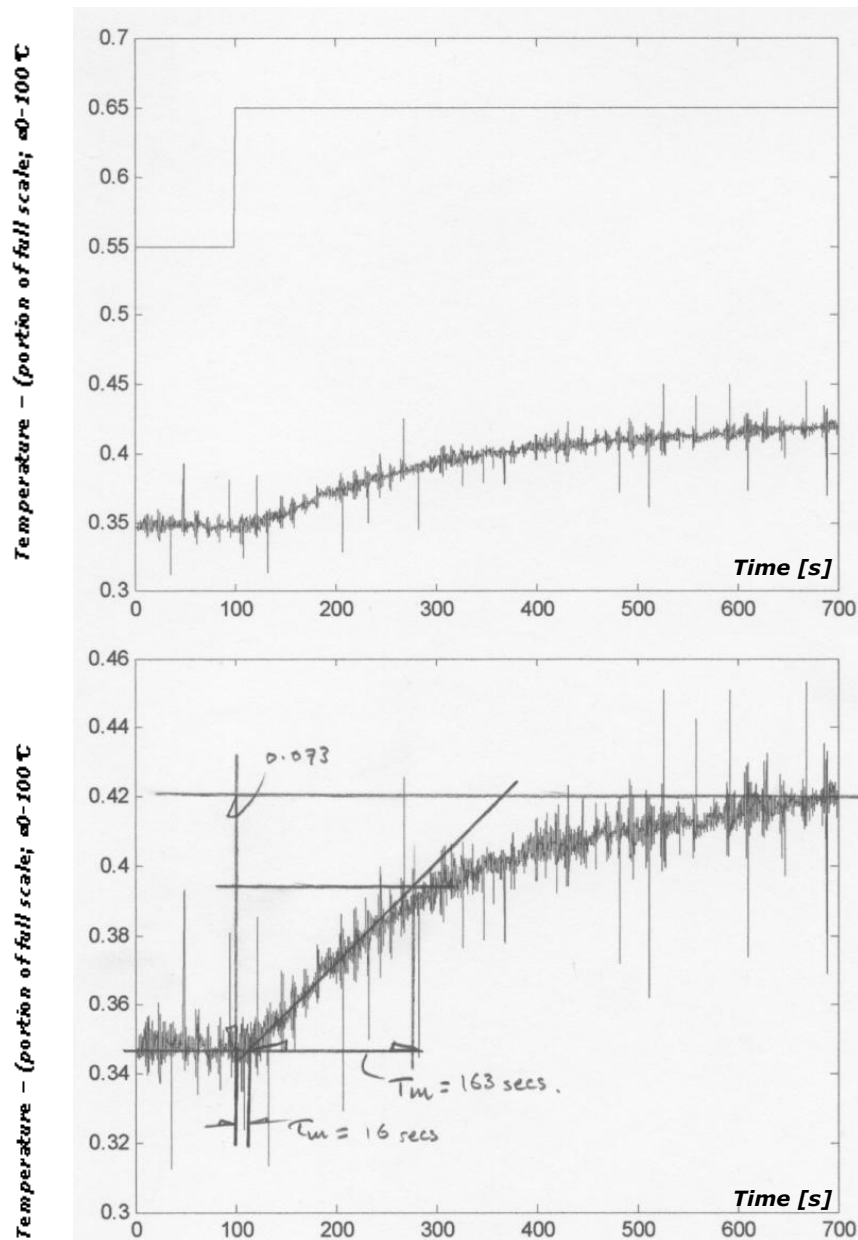


Figure 3-12 Example of alternative tangent and point method applied to a flow step response

The top plot shows both the input step and the output response. The bottom plot is a zoomed in version of the response. This was done to make it easier to record measurements. Most of the responses displayed noise. This makes applying the tangent and point method difficult, though possible with careful work. All measurements were scaled from the axes of the plot using a ruler. The step was applied at 100 seconds and most tests took about 700 seconds. When all the plots were obtained the change in output ( $\Delta o/p$ ), time constant ( $T_m$ ), and time delay ( $\tau_m$ ) were recorded. All results are shown in Tables 3-3, 3-4, and 3-5, corresponding to flows of 30, 50, and 70% respectively.

| <b>Temperature Step Test Data @ 30% flow</b> |              |          |       |       |   |       |       |                     |            |             |
|--|--------------|----------|-------|-------|---|-------|-------|---------------------|------------|-------------|
| <b>Increasing Step</b>                       |              |          |       |       | (units of input and output: 1 = 5Volts)       |       |       |                     |            |             |
| step i/p                                     | $\Delta o/p$ | $\tau_m$ | $T_m$ | $K_m$ | <b>Average</b>                                |       |       |                     |            |             |
| 0.15 - 0.25                                  | 0.013        | 55       | 145   | 0.13  | $\tau_m$                                      | $T_m$ | $K_m$ |                     |            |             |
| 0.25 - 0.35                                  | 0.035        | 28       | 100   | 0.35  | Low Temp.                                     |       |       |                     |            |             |
| 0.35 - 0.45                                  | 0.044        | 24       | 133   | 0.44  | 42  | 123   | 0.24  |                     |            |             |
| 0.45 - 0.55                                  | 0.042        | 25       | 127   | 0.42  | Medium Temp.                                  |       |       |                     |            |             |
| 0.55 - 0.65                                  | 0.073        | 16       | 163   | 0.73  | 21  | 145   | 0.58  |                     |            |             |
| 0.65 - 0.75                                  | 0.050        | 22       | 145   | 0.50  | High Temp.                                    |       |       |                     |            |             |
| 0.75 - 0.85                                  | 0.033        | 27       | 189   | 0.33  | 25  | 167   | 0.42  | <b>Average</b>      |            |             |
| 0.85 - 0.95                                  | nui          | nui      | nui   | nui   |   |       |       | $\tau_m$            | $T_m$      | $K_m$       |
|  |              |          |       |       |   |       |       | <b>Low Temp.</b>    |            |             |
|  |              |          |       |       |   |       |       | <b>32</b>           | <b>124</b> | <b>0.32</b> |
|  |              |          |       |       |   |       |       | <b>Medium Temp.</b> |            |             |
|  |              |          |       |       |   |       |       | <b>22</b>           | <b>153</b> | <b>0.61</b> |
|  |              |          |       |       |   |       |       | <b>High Temp.</b>   |            |             |
|  |              |          |       |       |   |       |       | <b>24</b>           | <b>150</b> | <b>0.45</b> |
| <b>Decreasing Step</b>                       |              |          |       |       | <b>Average</b>                                |       |       |                     |            |             |
| step i/p                                     | $\Delta o/p$ | $\tau_m$ | $T_m$ | $K_m$ | $\tau_m$                                      | $T_m$ | $K_m$ |                     |            |             |
| 0.95 - 0.85                                  | nui          | nui      | nui   | nui   | High Temp.                                    |       |       |                     |            |             |
| 0.85 - 0.75                                  | 0.035        | 26       | 136   | 0.35  | 23  | 134   | 0.48  |                     |            |             |
| 0.75 - 0.65                                  | 0.060        | 19       | 131   | 0.60  | Medium Temp.                                  |       |       |                     |            |             |
| 0.65 - 0.55                                  | 0.070        | 19       | 191   | 0.70  | 24  | 161   | 0.64  |                     |            |             |
| 0.55 - 0.45                                  | 0.058        | 29       | 131   | 0.58  | Low Temp.                                     |       |       |                     |            |             |
| 0.45 - 0.35                                  | 0.048        | 30       | 114   | 0.48  | 23  | 126   | 0.39  |                     |            |             |
| 0.35 - 0.25                                  | 0.030        | 15       | 137   | 0.30  |   |       |       |                     |            |             |
| 0.25 - 0.15                                  | nui          | nui      | nui   | nui   |   |       |       |                     |            |             |
| <i>[Note: nui = no useful information]</i>   |              |          |       |       | for input <45% the temp. is 'low'             |       |       |                     |            |             |
|  |              |          |       |       | for input <65% and >45% the temp. is 'medium' |       |       |                     |            |             |
|  |              |          |       |       | for input >65% the temp. is 'high'            |       |       |                     |            |             |
| <b>Overall Average</b>                       |              | $\tau_m$ | $T_m$ | $K_m$ |   |       |       |                     |            |             |
|  |              | 26       | 142   | 0.45  |   |       |       |                     |            |             |

**Table 3-3 Summary of results from temperature open loop step tests at 30% flow**

The table shows data obtained for increasing and decreasing steps. In each case the average value of time delay, time constant and process gain was obtained for low, medium, and high temperatures. An average value of these two sets of figures was found and are as shown in the coloured boxes above. This information gives three process models, for this flow rate (i.e. 30%), and are as follows:

$$G_{m_{LOW-TEMP}}(s) = \frac{0.32e^{-32s}}{1+124s} \quad G_{m_{MED-TEMP}}(s) = \frac{0.61e^{-22s}}{1+153s} \quad G_{m_{HIGH-TEMP}}(s) = \frac{0.45e^{-24s}}{1+150s}$$

The results of this test when carried out at a flow rate of 50% are shown in table 3-4 below;

| <i>Temperature Step Test Data @ 50% flow</i> |              |          |       |       |   |       |       |              |       |       |
|--|--------------|----------|-------|-------|---|-------|-------|--------------|-------|-------|
| <i>Increasing Step</i>                       |              |          |       |       | (Units of input and output: 1 = 5Volts)       |       |       |              |       |       |
| step i/p                                     | $\Delta o/p$ | $\tau_m$ | $T_m$ | $K_m$ | Average                                       |       |       |              |       |       |
| 0.15 - 0.25                                  | 0.01         | 18       | 97    | 0.12  | $\tau_m$                                      | $T_m$ | $K_m$ |              |       |       |
| 0.25 - 0.35                                  | 0.02         | 27       | 112   | 0.20  | Low Temp.                                     |       |       |              |       |       |
| 0.35 - 0.45                                  | 0.03         | 12       | 110   | 0.30  | 19  | 106   | 0.21  |              |       |       |
| 0.45 - 0.55                                  | 0.03         | 12       | 76    | 0.30  | Medium Temp.                                  |       |       |              |       |       |
| 0.55 - 0.65                                  | 0.05         | 15       | 142   | 0.50  | 14  | 109   | 0.40  |              |       |       |
| 0.65 - 0.75                                  | nui          | nui      | nui   | nui   | High Temp.                                    |       |       |              |       |       |
| 0.75 - 0.85                                  | 0.03         | 18.0     | 127   | 0.30  | 18  | 127   | 0.3   | Average      |       |       |
| 0.85 - 0.95                                  | nui          | nui      | nui   | nui   |   |       |       | $\tau_m$     | $T_m$ | $K_m$ |
|  |              |          |       |       |   |       |       | Low Temp.    |       |       |
|  |              |          |       |       |   |       |       | 16           | 123   | 0.32  |
|  |              |          |       |       |   |       |       | Medium Temp. |       |       |
|  |              |          |       |       |   |       |       | 16           | 109   | 0.43  |
|  |              |          |       |       |   |       |       | High Temp.   |       |       |
|  |              |          |       |       |   |       |       | 17           | 113   | 0.30  |
| <i>Decreasing Step</i>                       |              |          |       |       | Average                                       |       |       |              |       |       |
| step i/p                                     | $\Delta o/p$ | $\tau_m$ | $T_m$ | $K_m$ | $\tau_m$                                      | $T_m$ | $K_m$ |              |       |       |
| 0.95 - 0.85                                  | 0.03         | 18       | 91    | 0.30  | High Temp.                                    |       |       |              |       |       |
| 0.85 - 0.75                                  | 0.03         | 12       | 106   | 0.30  | 15  | 99    | 0.30  |              |       |       |
| 0.75 - 0.65                                  | 0.04         | 12       | 115   | 0.40  | Medium Temp.                                  |       |       |              |       |       |
| 0.65 - 0.55                                  | 0.05         | 24       | 103   | 0.50  | 18  | 109   | 0.45  |              |       |       |
| 0.55 - 0.45                                  | 0.05         | 15       | 151   | 0.50  | Low Temp.                                     |       |       |              |       |       |
| 0.45 - 0.35                                  | 0.04         | 12       | 140   | 0.44  | 12  | 140   | 0.44  |              |       |       |
| 0.35 - 0.25                                  | nui          | nui      | nui   | nui   |   |       |       |              |       |       |
| 0.25 - 0.15                                  | nui          | nui      | nui   | nui   |   |       |       |              |       |       |
| [Note: nui = no useful information]          |              |          |       |       | for input <45% the temp. is 'low'             |       |       |              |       |       |
|  |              |          |       |       | for input <65% and >45% the temp. is 'medium' |       |       |              |       |       |
|  |              |          |       |       | for input >65% the temp. is 'high'            |       |       |              |       |       |
| Overall Average                              |              | $\tau_m$ | $T_m$ | $K_m$ |   |       |       |              |       |       |
|  |              | 16       | 114   | 0.35  |   |       |       |              |       |       |

**Table 3-4 Summary of results from temperature open loop step tests at 50% flow**

The table shows data as before. Some of the plots were not clear enough to yield good information and are denoted by ‘nui’ above. The range of input over which averages for low, medium and high temperature were measured are shown. The three process models obtained, for this flow rate (i.e. 50%), are as follows:

$$G_{m_{LOW-TEMP}}(s) = \frac{0.32e^{-16s}}{1 + 123s} \quad G_{m_{MED-TEMP}}(s) = \frac{0.43e^{-16s}}{1 + 109s} \quad G_{m_{HIGH-TEMP}}(s) = \frac{0.30e^{-17s}}{1 + 113s}$$

The results of this test when carried out at a flow rate of 70% are shown in table 3-5 below;

| <i>Temperature Step Test Data @ 70% flow</i> |              |          |       |       |   |       |       |   |  |  |
|--|--------------|----------|-------|-------|---|-------|-------|---|--|--|
| <i>Increasing Step</i>                       |              |          |       |       | (units of input and output: 1 = 5Volts) |       |       |   |  |  |
| step i/p                                     | $\Delta o/p$ | $\tau_m$ | $T_m$ | $K_m$ | Average                                 |       |       |   |  |  |
| 0.15 - 0.25                                  | nui          | nui      | nui   | nui   | $\tau_m$                                | $T_m$ | $K_m$ |   |  |  |
| 0.25 - 0.35                                  | 0.016        | 28.0     | 80.0  | 0.16  | Low Temp.                               |       |       |   |  |  |
| 0.35 - 0.45                                  | 0.034        | 19.8     | 98.7  | 0.34  | 24                                      | 89    | 0.25  |   |  |  |
| 0.45 - 0.55                                  | 0.036        | 19.6     | 118.0 | 0.36  | Medium Temp.                            |       |       |   |  |  |
| 0.55 - 0.65                                  | 0.042        | 28.0     | 98.0  | 0.42  | 24                                      | 108   | 0.39  |   |  |  |
| 0.65 - 0.75                                  | 0.041        | 22.00    | 116.0 | 0.41  | High Temp.                              |       |       |   |  |  |
| 0.75 - 0.85                                  | 0.027        | 19.3     | 124.0 | 0.27  | 21                                      | 120   | 0.34  |   |  |  |
| 0.85 - 0.95                                  | nui          | nui      | nui   | nui   |   |       |       |   |  |  |
| <i>Decreasing Step</i>                       |              |          |       |       | Average                                 |       |       |   |  |  |
| 0.95 - 0.85                                  | nui          | nui      | nui   | nui   | $\tau_m$                                | $T_m$ | $K_m$ |   |  |  |
| 0.85 - 0.75                                  | 0.025        | 11.0     | 127   | 0.25  | High Temp.                              |       |       |   |  |  |
| 0.75 - 0.65                                  | 0.038        | 23.0     | 108   | 0.38  | 17                                      | 118   | 0.32  |   |  |  |
| 0.65 - 0.55                                  | 0.046        | 21.3     | 79    | 0.46  | Medium Temp.                            |       |       |   |  |  |
| 0.55 - 0.45                                  | 0.039        | 17.2     | 106   | 0.39  | 22                                      | 94    | 0.42  |   |  |  |
| 0.45 - 0.35                                  | 0.031        | 22.1     | 110   | 0.31  | Low Temp.                               |       |       |   |  |  |
| 0.35 - 0.25                                  | 0.017        | 18.40    | 107   | 0.17  | 20                                      | 108   | 0.35  |   |  |  |
| 0.25 - 0.15                                  | nui          | nui      | nui   | nui   |   |       |       |   |  |  |
| [Note: nui = no useful information]          |              |          |       |       |   |       |       |   |  |  |
|  |              | $\tau_m$ | $T_m$ | $K_m$ |   |       |       | for input <45% the temp. is 'low'             |  |  |
| <b>Overall Average</b>                       |              | 21       | 106   | 0.33  |   |       |       | for input <65% and >45% the temp. is 'medium' |  |  |
|  |              |          |       |       |   |       |       | for input >65% the temp. is 'high'            |  |  |

Table 3-5 Summary of results from temperature open loop step tests at 70% flow

The table shows data as before. The three process models obtained, for this flow rate (i.e. 70%), are as follows:

$$G_{m_{LOW-TEMP}}(s) = \frac{0.30e^{-22s}}{1 + 99s} \quad G_{m_{MED-TEMP}}(s) = \frac{0.41e^{-23s}}{1 + 101s} \quad G_{m_{HIGH-TEMP}}(s) = \frac{0.33e^{-19s}}{1 + 119s}$$

The difference between the process model parameters at various flow rates are shown in table 3-6.



| Time Delay |     |     | Time Constant |     |     | Gain |      |      | Parameter  |
|------------|-----|-----|---------------|-----|-----|------|------|------|------------|
| 32         | 16  | 22  | 124           | 123 | 99  | 0.32 | 0.32 | 0.30 | Low Temp.  |
| 22         | 16  | 23  | 153           | 109 | 101 | 0.61 | 0.43 | 0.41 | Med. Temp. |
| 24         | 17  | 19  | 150           | 113 | 119 | 0.45 | 0.30 | 0.33 | High Temp. |
| 30%        | 50% | 70% | 30%           | 50% | 70% | 30%  | 50%  | 70%  | Flow Rate  |

Table 3-6 Summary of process model parameters at various flow rates

Plotting the data, as shown in figure 3-13, can show the variance of each parameter more easily.

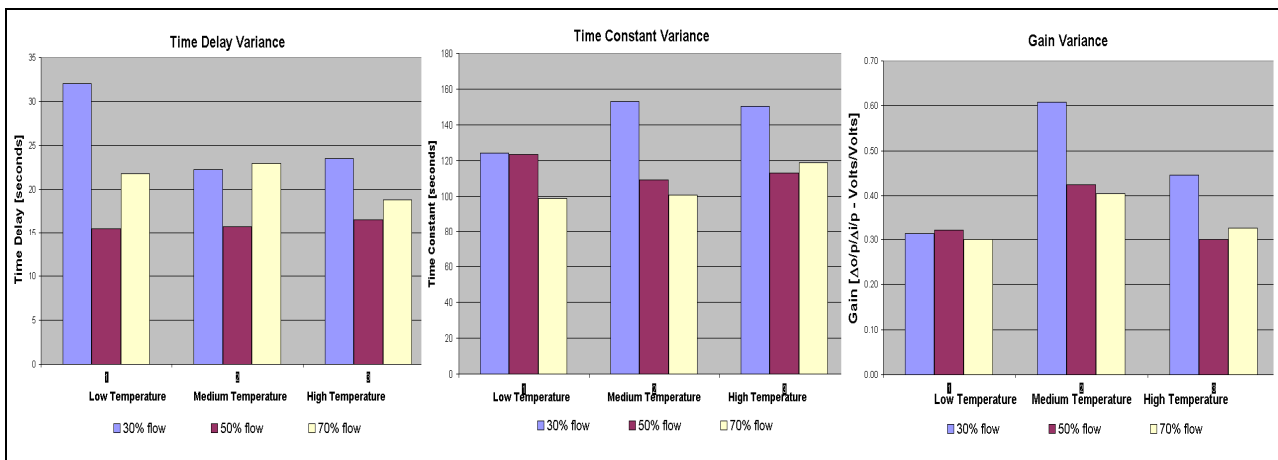


Figure 3-13 Plots of process model parameter variance at different flow rates

At first it seemed that no particular trend existed in these results but on closer inspection some interesting relationships became apparent. The time delay seemed to be relatively constant over the range of temperatures for a particular flow rate (i.e. at 50% the time delay remains around 16 or 17 seconds over the temperature range). The time constant displays similar behaviour; i.e. it remains relatively constant over the range of temperatures and no other particular trend is apparent. The gain, on the other hand, was very clearly higher in the medium temperature range (i.e. between 45% and 65 %) compared to the high and low temperature ranges. The effect of having a higher gain at medium temperatures means that the temperature process will behave better in this range. From an intuitive point of view this makes sense. The heating element has a maximum power output, so at high temperatures we would expect it to saturate; i.e. a limit is put on the processes operating range. Similarly, the ambient temperature of the room or surroundings limits the minimum temperature at which the process can operate. This can be seen in more detail in the next chapter.

## 4. Investigation into Process Non-Linearities, Measurement Accuracy, and Interactions.

It was quite obvious from the process identification that both flow and temperature processes were non-linear. This was shown by the large variations in process model parameters determined. For this reason it was interesting to attempt to attain a static characteristic curve for both processes. The curve describes the steady state (or static) output from each process over its full range of input. Both processes take input voltages of 0-5Volts representing 0-100% of full scale. The output voltage has the same range as the input. By measuring this with a voltmeter, a percentage value of output could be calculated.

### 4.1 Flow Process Static Tests

It was required to import data from an Excel spreadsheet and use the 'from Workspace' option in Simulink to set up the input to the flow process. This is shown in fig. 4-1 as 'stair\_data'. This was simply a staircase function. The step increment was 5% of full-range and occurred every 20secs. This was considered long enough for the process to settle between steps as, in general, from the previous tests on the flow, the combined time constant and time delay did not exceed 4 seconds. During this test the temperature was held constant at the ambient room temperature (24°C). The experimental set-up used for this test is shown in figure 4-1.

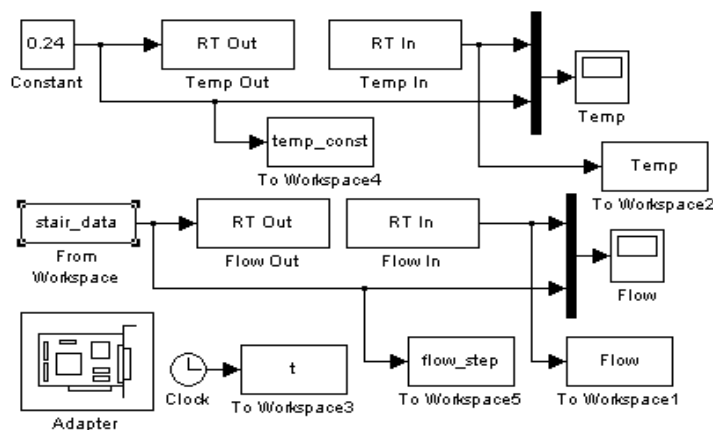
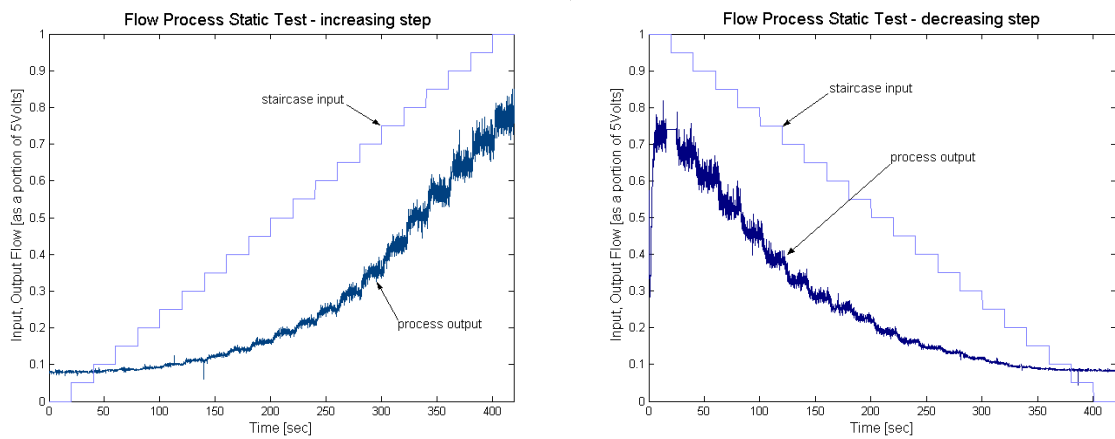


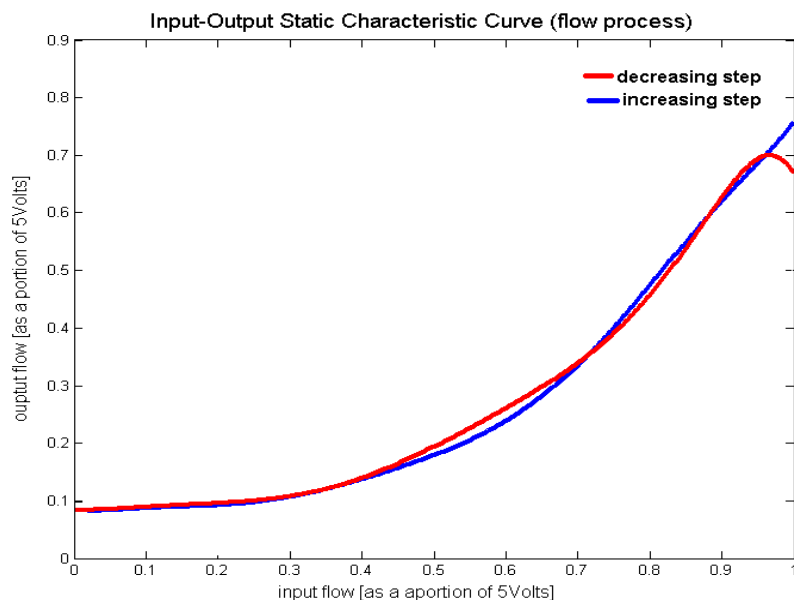
Figure 4-1 Simulink model file for static flow test

All previous tests on the flow process were carried out with the load vane fully open. Because the static test was more practical to carry out, it was possible to plot static responses for various load vane conditions. The resulting plots of the increasing and decreasing input staircase function and the output responses for a fully open load vane are shown in figure 4-2.



**Figure 4-2 Increasing and decreasing static flow test curves – fully opened load vane**

The plots show that the flow process doesn't seem to respond to a change in input of less than 15%. It also shows that at higher flows the level of noise is significantly increased. This would be expected as the flow may become turbulent at high flows. It should also be noted that the flow never actually reaches 100% of full scale even at 100% input with the load vane fully opened. Figure 4-2 also shows that at zero input the process has an output of nearly 10%. This may be some minimum operating point for the flow process. It was possible to generate an input-output characteristic curve by plotting the staircase input against the output response, as shown in figure 4-3 below.



**Figure 4-3 Input-output characteristic curve – flow process**

The blue line represents the process for increasing step responses, and the red line represents the process for decreasing step responses. They are not the same but very similar. The slight falloff at the end of the red line is due to the zero flow condition at the beginning of the decreasing step test (see the

right hand plot of figure 4-2). Figure 4-3 clearly shows that the flow process is non-linear. It also justifies the results from the step tests. The slope of the curve is greater at high inputs, i.e. the process gain is bigger for a large input. This had already been established through the process model transfer functions for the flow process.

The effect of closing the load vane by 1/3 and 2/3, and eventually closing it altogether was also investigated, the plots for which are shown below in figure 4-4.

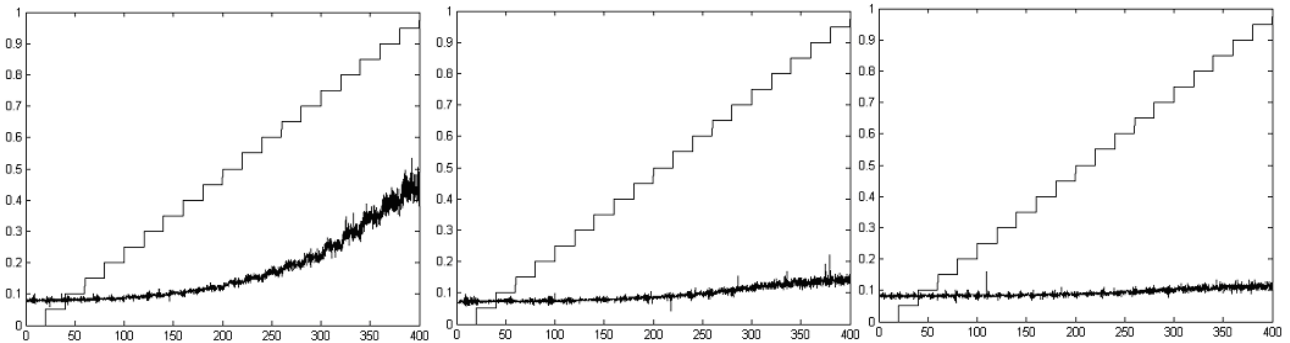


Figure 4-4 Flow characteristic curves – 1/3 closed, 2/3 closed, closed load vane

Again we can see that at low flow rates the process tends not to respond to a change in input. Also these results highlight that the process is far less able to react to a change in input when the load vane is more than 1/3 closed.

#### 4.1.1 Flow Measurement Accuracy of the Rig

The rig uses an orifice plate to measure the flow rate. The accuracy of most orifice plates are usually not very good and a co-efficient of discharge ( $C_d$ ) has to be calculated specifically for a particular orifice, to compensate for frictional head loss and the formation of a *vena contracta* or necking. Figure 4-5 shows the set-up of an orifice plate measuring device.

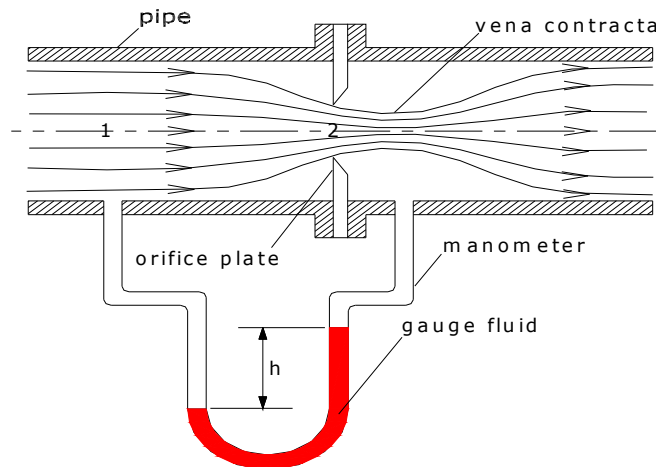


Figure 4-5 Orifice plate flow measurement device

The co-efficient of discharge ( $C_d$ ) relates the actual flow rate to the theoretical flow rate as calculated from Bernoulli's equation, and can be calculated as follows;

*Nomenclature;*  $P_1 =$  pressure at section 1 [ $N/m^2$ ];  $P_2 =$  pressure at section 2 [ $N/m^2$ ]  
 $V_1 =$  velocity at section 1 [ $N/m^2$ ];  $V_2 =$  velocity at section 2 [ $N/m^2$ ]  
 $z_1 =$  elevation at section 1 [ $N/m^2$ ];  $z_2 =$  elevation at section 2 [ $N/m^2$ ]  
 $A_1 =$  cross sectional area at section 1 [ $m^2$ ];  $A_2 =$  cross sectional area at section 2 [ $m^2$ ]  
 $\rho =$  flowing fluid density [ $kg/m^3$ ];  $g =$  acceleration due to gravity [ $m/s^2$ ]  
 $Q_T =$  theoretical flow rate [ $m^3/s$ ]

$$C_d = \frac{\text{ActualFlowrate}}{\text{TheoreticalFlowrate}} \dots(1) \text{ co-efficient of discharge}$$

$$\frac{P_1}{\rho g} + \frac{V_1^2}{2g} + z_1 = \frac{P_2}{\rho g} + \frac{V_2^2}{2g} + z_2 \dots (2) \text{ Bernoulli's equation}$$

Assuming incompressible flow then;  $Q_T = A_1V_1 = A_2V_2 \dots(3) \text{ Continuity equation}$

Substituting  $V_2$  in terms of  $V_1$  from (3) into (2) we can find  $V_1$  and hence  $Q_T$ ;

$$Q_T = A_1V_1 = A_1 \frac{1}{\sqrt{\left(\frac{A_1}{A_2}\right)^2 - 1}} \sqrt{\frac{P_1 - P_2}{\rho}}$$

Now, the actual flow rate must be measured so that the co-efficient of discharge may be calculated. A Pitot tube can be used to do this. A Pitot tube measures velocity at a point in a flowing fluid and is a very accurate measuring device. Figure 4-6 shows how a Pitot tube is set up.

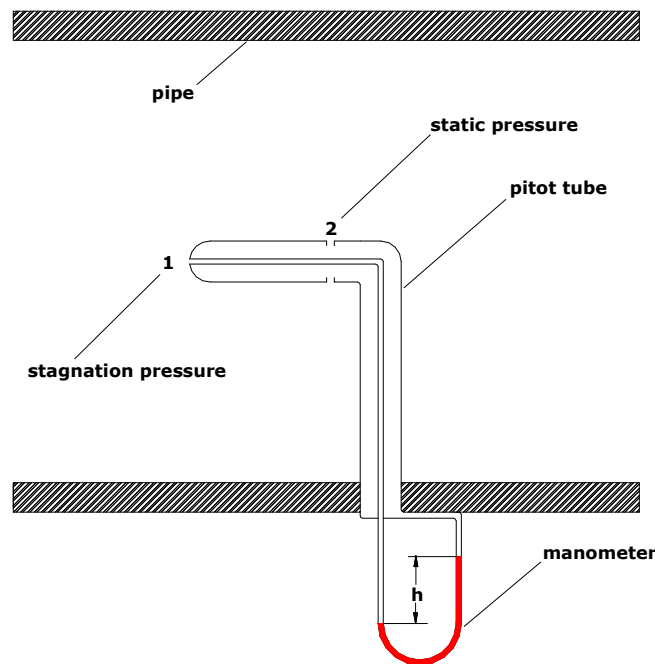


Figure 4-6 The Pitot tube - velocity measurement device

The reason for the Pitot tube’s accuracy is because they can be made from a single slender annular tube and thus the effect of friction loss is minimized. Point 1 is stagnant (i.e. the velocity has been brought to zero) and measures stagnation pressure. Point 2 is static and sees the true velocity of the fluid but measures static pressure. The difference between stagnation pressure and static pressure is given by the head of manometer fluid  $h$ . Applying Bernoulli’s equation between points 1 and 2 we get;

$$\frac{P_1}{\rho g} + \frac{V_1^2}{2g} + z_1 = \frac{P_2}{\rho g} + \frac{V_2^2}{2g} + z_2 \dots \text{but } V_1 = 0 \text{ and } z_1, z_2 = 0$$

$$\therefore V_2 = \sqrt{\left(\frac{P_1 - P_2}{\rho g}\right) 2g} = \sqrt{2gh} = \text{fluid\_velocity}$$

Taking  $V_2$  as actual velocity;  $Q_{ACTUAL} = AV_2 \dots$ where  $A$  is the cross sectional area of the pipe

So now the co-efficient of discharge may be calculated. A small hole was drilled in the top of the tube of the rig so that the Pitot tube could be inserted. The Pitot tube must face directly into the flow for accurate results. Figure 4-7 shows the rig with a manometer connected to the orifice plate pressure tapings and another manometer connected to the Pitot tube, and a close up of the Pitot tube.

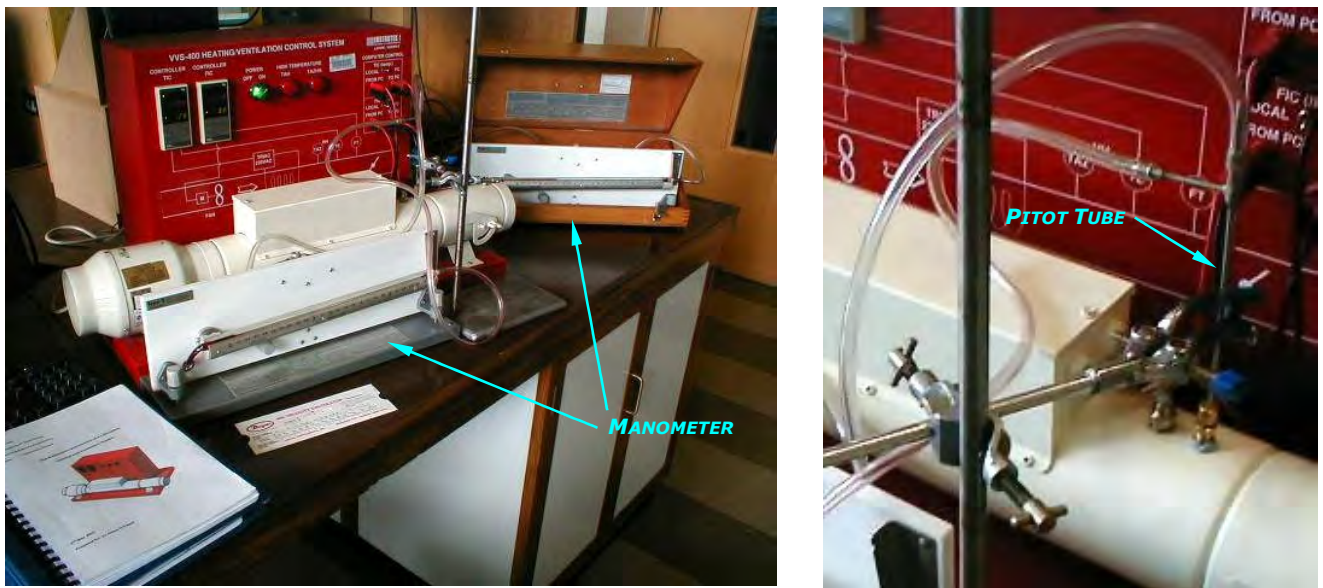


Figure 4-7 Photo of the rig with flow measurement equipment set-up – and close up of Pitot tube

The measurements of differential pressure head were taken from both manometers over the full range of input to the fan from 0-100% in increments of 5%. The manometers were inclined for greater accuracy. The viscosity and density of air were taken as standard for atmospheric pressure (White, 1999). The flow velocity, flow rate, and Reynolds number were calculated and tabulated as shown in Table 4-1. The co-efficient of discharge was then calculated by dividing the Pitot tube flow rate (taken as actual flow rate) by the orifice plate theoretical flow rate.

| Flow<br>% input | Pitot Tube  |        |         |                       | $R_e$ | Orifice Plate |        |         |                       | $R_e$                           | $C_d$       |
|-----------------|-------------|--------|---------|-----------------------|-------|---------------|--------|---------|-----------------------|---------------------------------|-------------|
|                 | scale units | h [mm] | V [m/s] | Q [m <sup>3</sup> /s] |       | scale units   | h [mm] | V [m/s] | Q [m <sup>3</sup> /s] |                                 |             |
| 5               | 0.001       | 0.006  | 0.011   | 0.00011               | 77    | 0.007         | 0.045  | 0.023   | 0.00023               | 155                             | 0.49        |
| 10              | 0.002       | 0.013  | 0.016   | 0.00016               | 108   | 0.010         | 0.065  | 0.027   | 0.00027               | 186                             | 0.58        |
| 15              | 0.004       | 0.026  | 0.022   | 0.00022               | 153   | 0.020         | 0.129  | 0.039   | 0.00039               | 263                             | 0.58        |
| 20              | 0.007       | 0.045  | 0.030   | 0.00030               | 203   | 0.028         | 0.181  | 0.046   | 0.00046               | 311                             | 0.65        |
| 25              | 0.010       | 0.065  | 0.036   | 0.00036               | 242   | 0.030         | 0.194  | 0.047   | 0.00047               | 322                             | 0.75        |
| 30              | 0.011       | 0.071  | 0.037   | 0.00037               | 254   | 0.040         | 0.258  | 0.055   | 0.00055               | 371                             | 0.68        |
| 35              | 0.013       | 0.084  | 0.041   | 0.00041               | 276   | 0.050         | 0.323  | 0.061   | 0.00061               | 415                             | 0.67        |
| 40              | 0.015       | 0.097  | 0.044   | 0.00044               | 297   | 0.061         | 0.393  | 0.067   | 0.00067               | 459                             | 0.65        |
| 45              | 0.016       | 0.103  | 0.045   | 0.00045               | 306   | 0.075         | 0.484  | 0.075   | 0.00075               | 508                             | 0.60        |
| 50              | 0.017       | 0.110  | 0.046   | 0.00046               | 316   | 0.085         | 0.548  | 0.080   | 0.00080               | 541                             | 0.58        |
| 55              | 0.019       | 0.123  | 0.049   | 0.00049               | 334   | 0.130         | 0.839  | 0.098   | 0.00098               | 669                             | 0.50        |
| 60              | 0.020       | 0.129  | 0.050   | 0.00050               | 343   | 0.161         | 1.038  | 0.109   | 0.00109               | 745                             | 0.46        |
| 65              | 0.025       | 0.161  | 0.056   | 0.00056               | 383   | 0.197         | 1.271  | 0.121   | 0.00121               | 824                             | 0.46        |
| 70              | 0.030       | 0.194  | 0.062   | 0.00062               | 419   | 0.225         | 1.451  | 0.129   | 0.00129               | 881                             | 0.48        |
| 75              | 0.040       | 0.258  | 0.071   | 0.00071               | 484   | 0.297         | 1.916  | 0.149   | 0.00149               | 1012                            | 0.48        |
| 80              | 0.049       | 0.316  | 0.079   | 0.00079               | 536   | 0.359         | 2.316  | 0.163   | 0.00163               | 1112                            | 0.48        |
| 85              | 0.057       | 0.368  | 0.085   | 0.00085               | 578   | 0.430         | 2.774  | 0.179   | 0.00179               | 1218                            | 0.47        |
| 90              | 0.065       | 0.419  | 0.091   | 0.00091               | 617   | 0.500         | 3.225  | 0.193   | 0.00193               | 1313                            | 0.47        |
| 95              | 0.072       | 0.464  | 0.095   | 0.00095               | 650   | 0.565         | 3.644  | 0.205   | 0.00205               | 1396                            | 0.47        |
| 100             | 0.080       | 0.516  | 0.101   | 0.00101               | 685   | 0.610         | 3.935  | 0.213   | 0.00213               | 1450                            | 0.47        |
|                 |             |        |         |                       |       |               |        |         |                       | <b>Average <math>C_d</math></b> | <b>0.55</b> |

**Table 4-1 Results of flow measurement test**

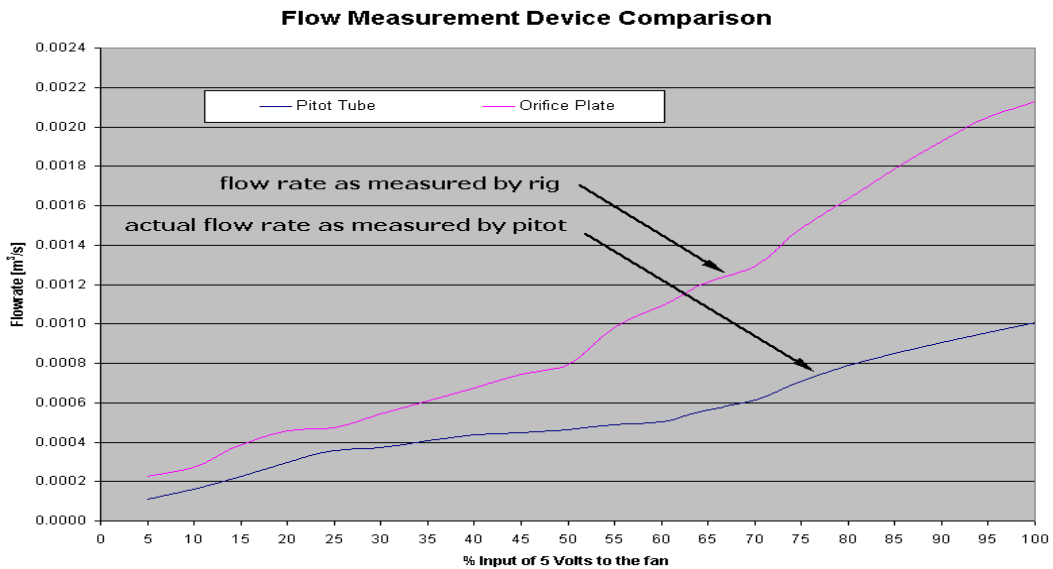
The test yielded some interesting results. The flow rate was never high enough to be sure that it was turbulent, i.e. the Reynolds number ( $R_e$ ) never exceeded 2000 over the full range of flow. The coefficient of discharge  $C_d$  was found to be about 0.55, which is reasonable because a sharp edged orifice would usually be expected to have  $C_d$  equal to 0.62<sup>6</sup>. Table 4-2 shows the test specifications used to calculate table 4-1.

|  |          |
|--|----------|
| Tube Diameter [mm]                           | 100      |
| Tube C.S. Area [m <sup>2</sup> ]             | 0.01     |
| Orifice Diameter [mm]                        | 78       |
| Orifice Area [m <sup>2</sup> ]               | 0.006084 |
| Manometer Scale Multiplier                   | 0.05     |
| Manometer Fluid Density [kg/m <sup>3</sup> ] | 782.4    |
| Air Density [kg/m <sup>3</sup> ]             | 1.2255   |
| Air Viscosity [Ns/m <sup>2</sup> ]           | 0.000018 |

**Table 4-2 Test specifications**

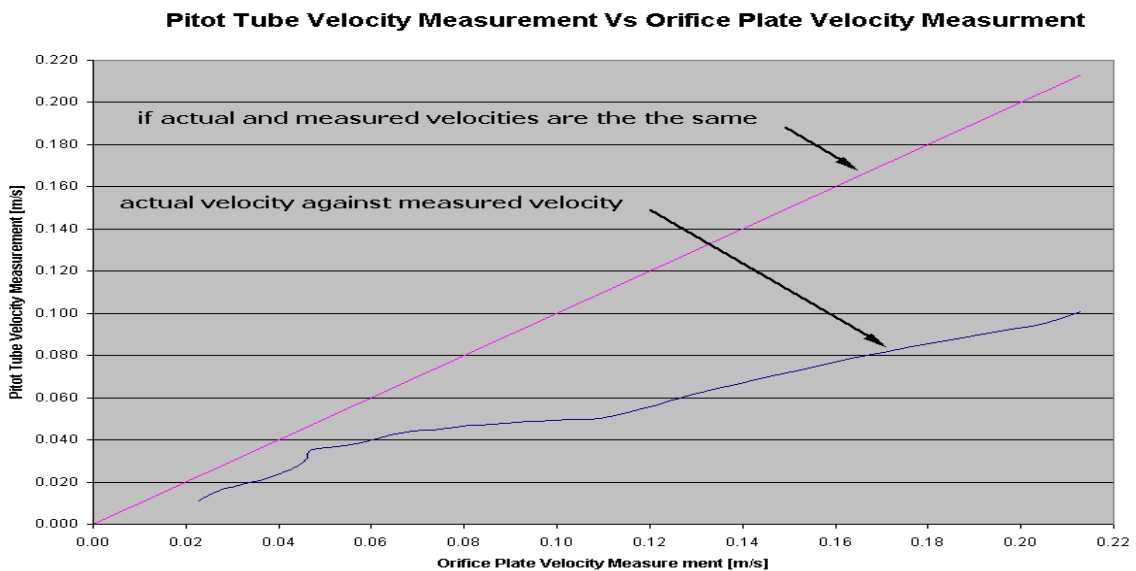
<sup>6</sup> As given by J.F. Douglas *et. al.* (1996) for a sharp edged orifice irrespective of pipe diameter (section 6.3).

The difference between the actual flow rate and the flow rate as measured by the orifice plate is shown in Figure 4-8.



**Figure 4-8 Difference between actual flow rate and flow rate measured by the rig**

The orifice plate continuously overestimates the actual flow rate by a factor of approximately 2 (i.e.  $1/C_d$ ). Clearly, the flow rate information available to the closed loop flow control system contains significant inaccuracies.



**Figure 4-9 Pitot tube velocity measurement vs. orifice plate velocity measurement**

Figure 4-9 show the actual velocity plotted against the velocity as measured by the rig (navy line). The pink line indicates where the situation where measured and actual velocities would be the same. At higher velocities these lines tend to diverge.



#### 4.2 Temperature Process Static Tests

It was more difficult to obtain the information for this test than it was for the flow process, because of the long time required between readings. The input to the temperature process was connected to a power supply and the voltage was increased slowly, in steps of 0.25Volts (i.e. 5% change in input), from 0 to 5Volts. The output voltage from the temperature sensor was allowed 3mins to settle between steps before a reading was taken. Also to investigate the calibration of the temperature sensor, a mercury thermometer was used to measure the actual temperature at each time interval. Results for increasing and decreasing step changes were gathered and are as shown in table 4-3. The test was carried out at 30% flow output.

|           |     | Increasing |    |                   | Decreasing |    |                   |
|-----------|-----|------------|----|-------------------|------------|----|-------------------|
| I/p volts | %   | o/p volts  | %  | actual temp. [°C] | o/p volts  | %  | actual temp. [°C] |
| 0         | 0   | 0.98       | 20 | 22.5              | 0.99       | 20 | 23                |
| 0.25      | 5   | 0.99       | 20 | 22.5              | 0.99       | 20 | 23                |
| 0.5       | 10  | 1          | 20 | 23                | 1          | 20 | 23                |
| 0.75      | 15  | 1.01       | 20 | 23                | 1          | 20 | 23                |
| 1         | 20  | 1.03       | 21 | 23                | 1.02       | 20 | 23.5              |
| 1.25      | 25  | 1.08       | 22 | 23                | 1.05       | 21 | 24.5              |
| 1.5       | 30  | 1.13       | 23 | 25                | 1.09       | 22 | 25                |
| 1.75      | 35  | 1.22       | 24 | 27                | 1.19       | 24 | 27                |
| 2         | 40  | 1.34       | 27 | 30                | 1.24       | 25 | 28                |
| 2.25      | 45  | 1.44       | 29 | 32                | 1.33       | 27 | 30                |
| 2.5       | 50  | 1.61       | 32 | 34                | 1.45       | 29 | 32                |
| 2.75      | 55  | 1.83       | 37 | 37                | 1.54       | 31 | 34                |
| 3         | 60  | 1.92       | 38 | 41                | 1.67       | 33 | 35.5              |
| 3.25      | 65  | 2.03       | 41 | 43                | 1.83       | 37 | 39.5              |
| 3.5       | 70  | 2.27       | 45 | 51.5              | 1.99       | 40 | 43                |
| 3.75      | 75  | 2.35       | 47 | 53                | 2.18       | 44 | 45                |
| 4         | 80  | 2.41       | 48 | 54                | 2.37       | 47 | 48                |
| 4.25      | 85  | 2.47       | 49 | 56                | 2.45       | 49 | 50                |
| 4.5       | 90  | 2.49       | 50 | 57                | 2.54       | 51 | 54                |
| 4.75      | 95  | 2.5        | 50 | 58                | 2.58       | 52 | 56                |
| 5         | 100 | 2.56       | 51 | 60                | 2.58       | 52 | 58                |

Table 4-3 Static temperature test results

A plot of percentage voltage input versus percentage voltage output is shown in figure 4-10 below. It should be borne in mind that this plot represents steady state information only about the temperature process.

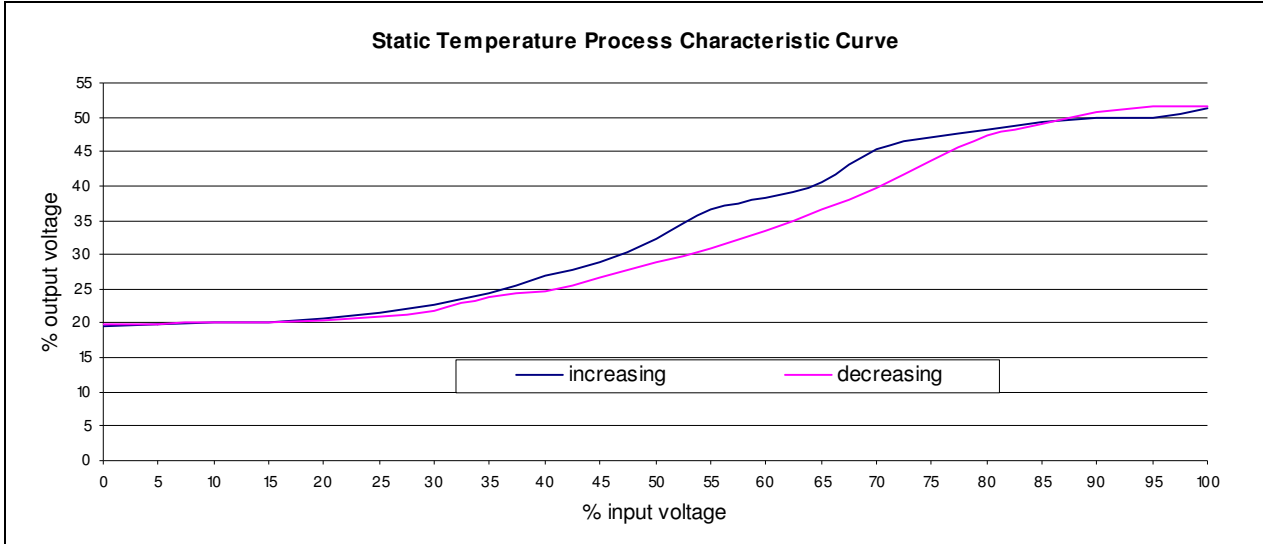


Figure 4-10 Temperature characteristic curve at 30% flow

This graph quite clearly shows that the temperature process is non-linear. It also displays a hysteresis effect. The author is of the opinion that this hysteresis is due to the nature of the temperature process. When the heating element is supplied with power the temperature rises; when the power is switched off, and the air flowing cools the element, the temperature falls. So the heating of the air is a different process to the cooling of the air.

4.2.1 Temperature Measurement Accuracy of the Rig

The actual temperature and the measured temperature were also plotted, as shown in figure 4-11.

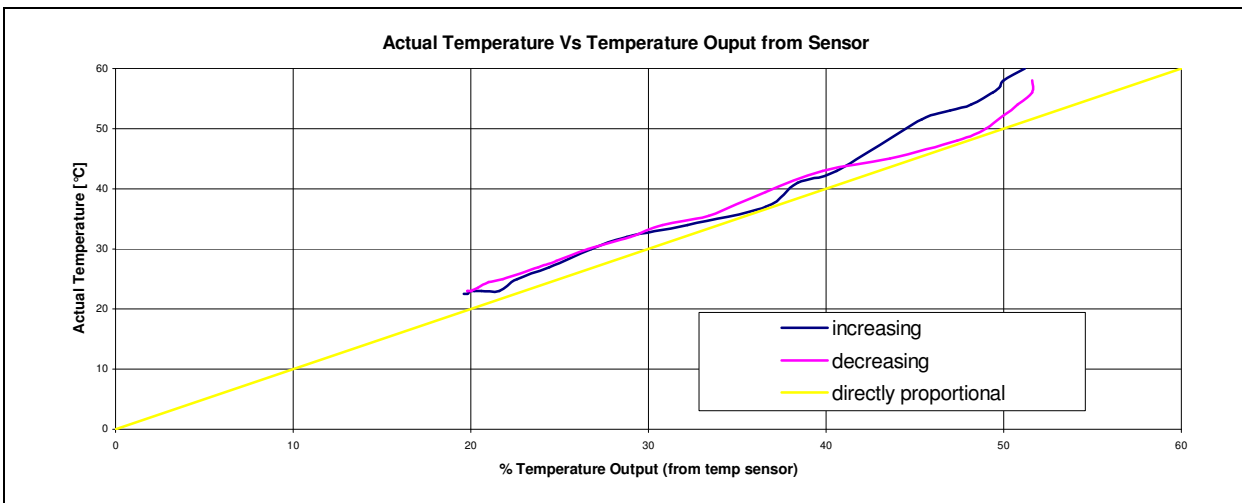


Figure 4-11 Actual temperature vs. temperature output curves – at 30% flow

Figure 4-11 tells us that if we assume that 0-100% temperature output correlates directly with a 0-100°C temperature scale, then the temperature sensor tends to underestimate the actual temperature. For example, at 30% temperature output, from the temperature sensor, the actual temperature was measured to be approximately 34°C. This underestimation was continuous across the range of temperatures.

#### 4.2.2 Temperature process static tests at different flow rates

As in section 4.1, it was possible to generate a *staircase* function using an Excel spreadsheet, and import it into Matlab to be used as the input to the temperature process. The point of doing this was to obtain an input-output characteristic curve of the process. This test has the advantage of being fully automated. Although the test times are long, the rig needs no attention during tests. The reason for the long test time is because of the large time constant of the temperature process.

The average temperature process time constant was found to be 121 seconds, and average time delay of 21 seconds. For a FOLPD model the time to reach 99.33% of the final value is given by:  $5T_m + \tau_m$ , i.e.  $5(121) + 21 = 626\text{sec}$ . So the time between steps was chosen to be some value larger than this, i.e. 700 seconds, to allow for the output value to reach its final value between steps. Eight steps of 0.125 every 700 seconds made up the input staircase function. Figure 4-12 shows a plot of the input staircase function and the resulting output temperature curve at a flow rate of 50%.

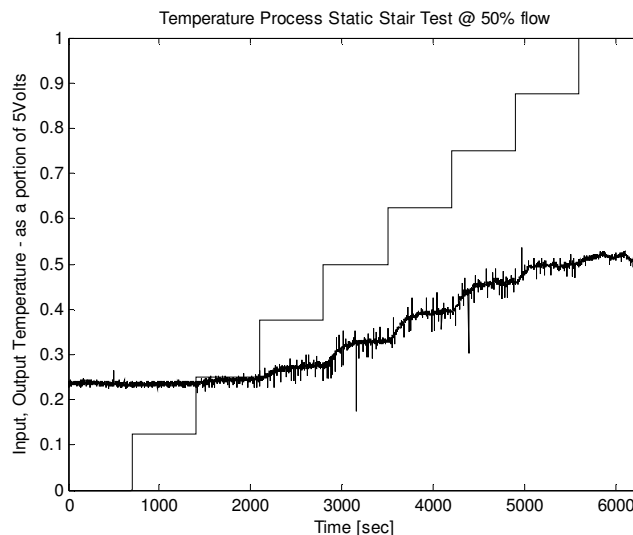
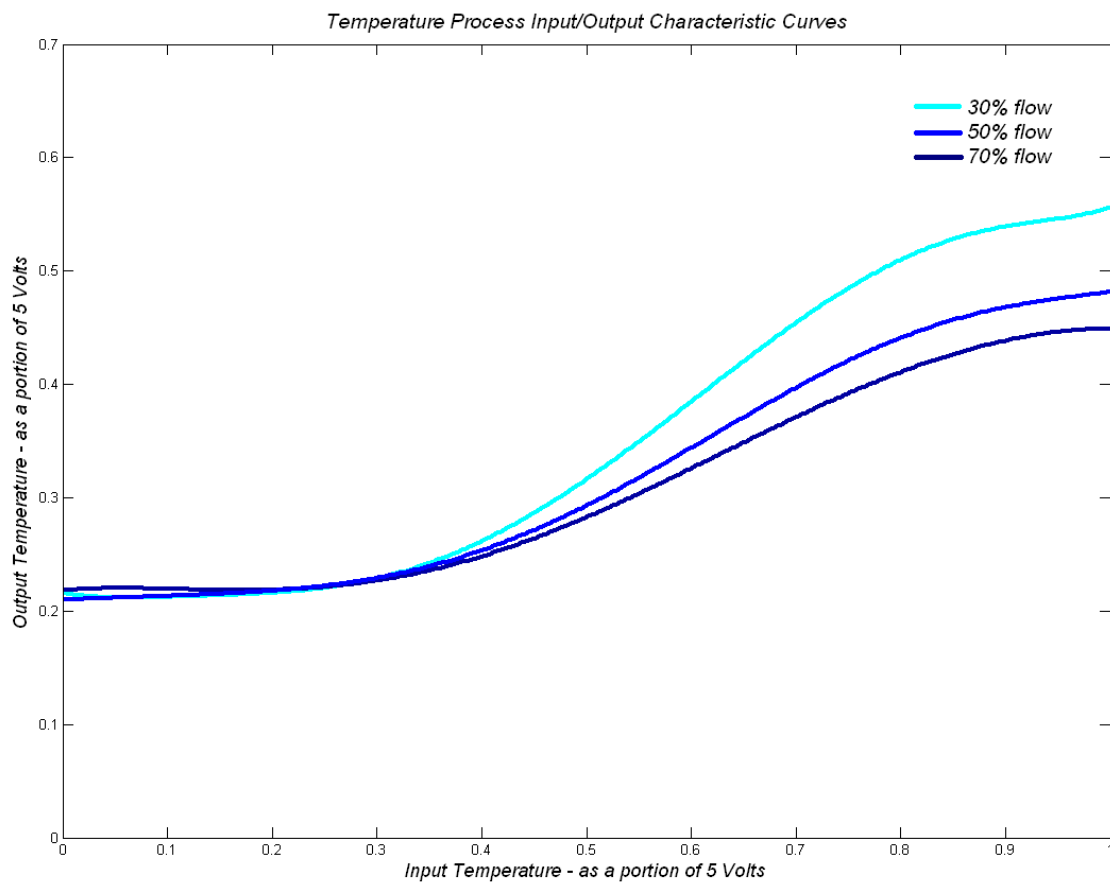


Figure 4-12 Temperature process static stair test plot at 50% flow

This test was carried out at three different rates of flow, i.e. 30, 50 and 70 per cent. The input stair can be plotted against the resulting output and an input-output characteristic curve can be obtained. Figure 4-13 shows all curves obtained.

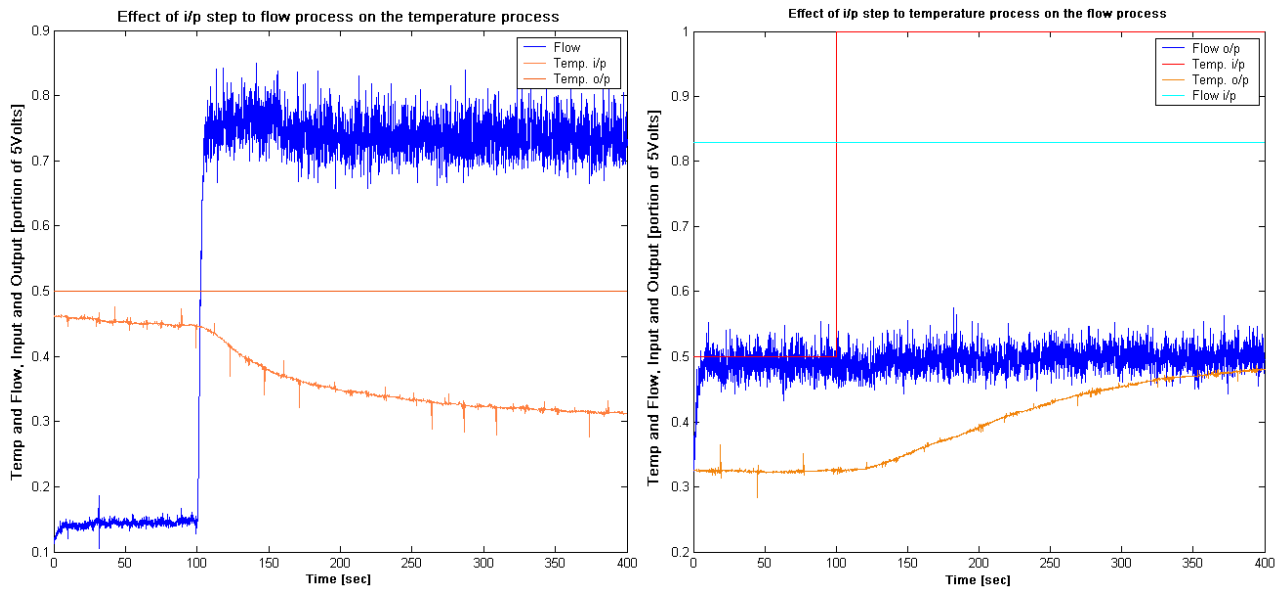


**Figure 4-13 Temperature process input/output characteristic curves**

The temperature process is clearly non-linear. All the curves are limited (in output) to above 0.23 at low input. This is because of the limit imposed on the process due to the surrounding conditions, i.e. an ambient room temperature of 23°C. The upper region of the each curve tends to level off at high input. The lower the flow rate, the higher the maximum possible temperature. It was found that at zero flow the temperature could reach as high as 100°C. The figure tells us that the realistic operating region for the temperature process should be 20-45°C.

### **4.3 Investigation into Process Interactions**

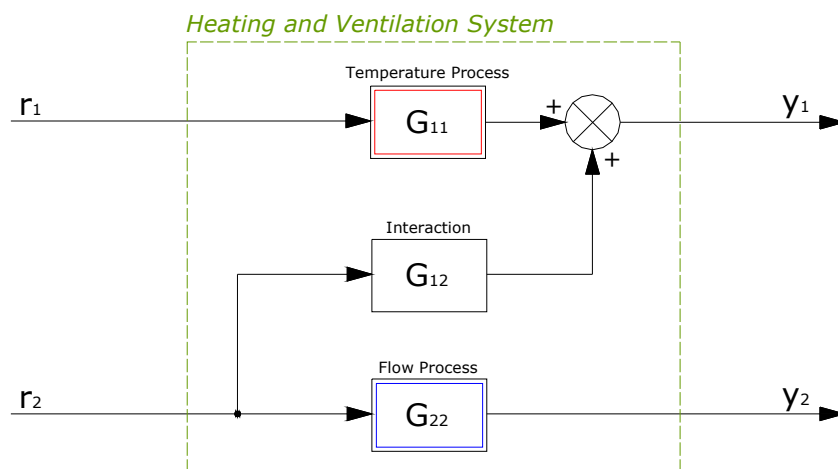
Any heating and ventilation system will contain two processes, i.e. one process that heats air and another to move or ventilate the air. If the dynamics of one of these processes affect the dynamics of the other, then a process interaction exists. This can lead to difficulties when designing effective controllers for each process. To examine the possibility of process interactions between the temperature and flow processes, two simple tests were carried out. In each case the input to both processes were held constant and allowed to settle. Then one of the process inputs would undergo a step change. The output of the other process was observed to see if this change had any effect on it.



**Figure 4-14 Interaction between the flow process on the temperature process and vice-versa**

Figure 4-14 shows the results from both tests. The left hand plot shows when the temperature process output is held at some constant (i.e. 0.45 or  $\approx 45^{\circ}\text{C}$ ) and the flow process undergoes a step change so that the output temperature reduces considerably (to 0.31 or  $\approx 31^{\circ}\text{C}$ ). This is a very large change and it is likely that it would be noticed in the environment being heated. It should be noted that the change in flow was very large (from 15% to 75% of full range) and therefore represents the worst-case scenario. The right hand plot shows that when the flow process output is held constant and the temperature process undergoes a step the flow process remains undisturbed.

The results show that the temperature process dynamics depend on the operating conditions of the flow process and the flow process dynamics are unaffected by the operating conditions of the temperature process. Figure 4-15 summarises this relationship in block diagram form.



**Figure 4-15 Block diagram describing flow-temperature interaction**

4.3.1 Interaction Step Tests

This section attempts to estimate the interaction transfer function  $G_{12}$ . A large step was applied to the flow process (at  $r_2$ ) and the output of the temperature process ( $y_1$ ) was observed over a range of temperature inputs (at  $r_1$ ). The alternative tangent and point method was then applied to the temperature output, as before in section 3.3.1. (i.e. we are assuming a FOLPD model for the interaction process). Figure 4-16 shows the results of one of the tests.

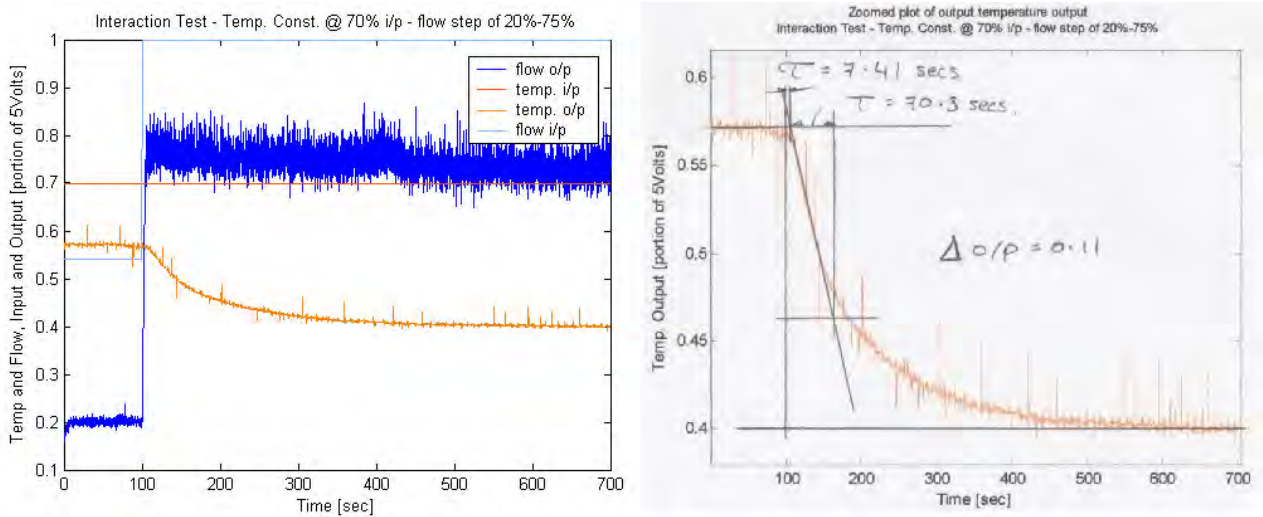


Figure 4-16 Interaction step test plots:--all inputs and outputs plot, and zoomed temperature output plot

The left-hand plot shows a test carried out with a constant temperature input of 70%. The flow process receives a step input at 100seconds. The output flow change is large (20-75%). This results in a change in the temperature output. The right-hand plot displays a zoomed plot of the temperature output with the alternative tangent and point method applied. This test was carried out at three different constant temperature inputs (30, 50, and 70 per cent). All plots can be found in appendix G. Table 4-4 shows all test results.

(units of input and output: 1 = 5Volts)

| const. temp. i/p ( $r_1$ ) | $\Delta o/p$ ( $y_1$ ) | flow step i/p ( $r_2$ ) | flow step o/p ( $y_2$ ) | $\tau_m$                   | $T_m$ | $K_m$ |
|----------------------------|------------------------|-------------------------|-------------------------|----------------------------|-------|-------|
| 0.3                        | -0.011                 | 0.54 - 1                | 0.2 - 0.75              | 16                         | 63    | -0.02 |
| 0.5                        | -0.085                 | 0.54 - 1                | 0.2 - 0.75              | 8                          | 85    | -0.18 |
| 0.7                        | -0.110                 | 0.54 - 1                | 0.2 - 0.75              | 7                          | 70    | -0.24 |
| Temperature process        |                        | Flow process            |                         | Interaction model $G_{12}$ |       |       |

Table 4-4 Interaction step test results

The interaction model gain is more than ten times larger at 70% constant temperature input than at 30%. The time taken for the temperature output to change (i.e. the time delay) becomes shorter at

higher temperatures. The temperature output decreased as the flow input increased therefore the change in temperature output was negative. So an inverse relationship exists between the flow input and temperature output, as expected. These results show that the effect of the interaction is more noticeable at higher temperatures. Controlling the temperature process at high temperatures may become difficult because of this. The following are the three interaction models obtained.

$$G_{12-LowTemp}(s) = \frac{-0.02e^{-16s}}{1+63s} \quad G_{12-Med.Temp}(s) = \frac{-0.18e^{-8s}}{1+85s} \quad G_{12-HighTemp}(s) = \frac{-0.24e^{-7s}}{1+70s}$$

## 5. Controller Design

### 5.1 Choice of controller

The rationale for deciding what type of controller to use was based on the process model time delay to time constant ratio obtained from the system identification chapter. Table 5-1 shows a summary of these ratios, over the range of operating conditions tested, for each process.

| Operating Conditions | 30% flow                            |      |      | 50% flow |      |      | 70% flow |      |      | (any temperature)     |      |      |
|----------------------|-------------------------------------|------|------|----------|------|------|----------|------|------|-----------------------|------|------|
|                      | L                                   | M    | H    | L        | M    | H    | L        | M    | H    | L                     | M    | H    |
| $\tau_m/T_m$         | 0.26                                | 0.15 | 0.16 | 0.13     | 0.14 | 0.15 | 0.22     | 0.23 | 0.16 | 0.36                  | 0.56 | 0.64 |
|                      | T e m p e r a t u r e P r o c e s s |      |      |          |      |      |          |      |      | F l o w P r o c e s s |      |      |

Table 5-1 Summary of time delay to time constant ratios obtained from process identification

It has been suggested that the PID implementation is recommended for the control of processes of low order and with small time delays (O'Dwyer *et. al.*, 1999). The process models used are first order and the time delay to time constant ratio ( $\tau_m/T_m$ ), as shown above, is small. Therefore the processes are non-dominant delay processes. For this reason it was decided that proportional integral (PI) and proportional integral derivative (PID) controllers would be used. If the process identification yielded larger time delays then perhaps alternative controllers could have been considered, such as the Smith predictor.

### 5.2 Choice of architecture and tuning rules

The Ideal PI and Classical PID controller structures (equations 5-1 and 5-2, respectively) were chosen mainly because of their wide use in industry, and also for their relatively simple implementation. The latter controller is used in a range of Honeywell, Toshiba, and Foxboro products (O'Dwyer, 2003a).

$$G_C(s) = K_C \left( 1 + \frac{1}{T_i s} \right) \quad \dots \text{Ideal PI Controller architecture}$$

$$G_C(s) = K_C \left( 1 + \frac{1}{T_i s} \right) \frac{1 + T_d s}{1 + \frac{T_d}{N} s} \quad \dots \text{Classical PID Controller architecture}$$

The amount of filtering in the derivative term of the PID controller (i.e. the  $N$  value) was chosen to be as ten<sup>7</sup>. The tuning rules selected were specific to the model and modelling method used, i.e. FOLPD model and the alternative tangent and point modelling method, as discussed in chapter 3. All tuning

<sup>7</sup> as per the Honeywell TDC3000 Process Manager product – Type A, interactive mode with  $N=10$  (O'Dwyer, 2003a)



rules are designed to minimise of the integral of the absolute error (minimum IAE criteria). Servo and regulator settings are given for both controller architectures as shown in tables 5-2 and 5-3. Where possible, the same authors work was used for servo and regulator tuning rules for consistency.

| Rule  | $K_c$   | $T_i$   | Comment                                |
|---|---|---|--|
| <b>Regulator Tuning</b>                               | <b>Performance index minimisation</b>                         |   |  |
| Minimum IAE<br>Murrill (1967)<br><i>Pages 358-363</i> | $\frac{0.984}{K_m} \left( \frac{T_m}{\tau_m} \right)^{0.986}$ | $\frac{T_m}{0.608} \left( \frac{\tau_m}{T_m} \right)^{0.707}$ | $0.1 \leq \frac{\tau_m}{T_m} \leq 1.0$ |
| <b>Servo Tuning</b>                                   | <b>Performance index minimisation</b>                         |   |  |
| Minimum IAE<br>Rovira <i>et. al.</i> (1969)           | $\frac{0.758}{K_m} \left( \frac{T_m}{\tau_m} \right)^{0.861}$ | $\frac{T_m}{1.020 - 0.323 \frac{\tau_m}{T_m}}$                | $0.1 \leq \frac{\tau_m}{T_m} \leq 1.0$ |

Table 5-2 PI Controller Tuning Rules

| Rule                                     | $K_c$  | $T_i$   | $T_d$   | Comment                         |
|--|--|---|---|---------------------------------|
| <b>Regulator Tuning</b>                  | <b>Performance index minimisation</b>                              |   |   |                                 |
| Minimum IAE<br>Kaya and Scheib<br>(1988) | $\frac{0.98089}{K_m} \left( \frac{T_m}{\tau_m} \right)^{0.716167}$ | $\frac{T_m}{0.91032} \left( \frac{\tau_m}{T_m} \right)^{1.05211}$ | $0.59974 T_m \left( \frac{\tau_m}{T_m} \right)^{0.89819}$ | $0 < \frac{\tau_m}{T_m} \leq 1$ |
| <b>Servo Tuning</b>                      | <b>Performance index minimisation</b>                              |   |   |                                 |
| Minimum IAE<br>Kaya and Scheib<br>(1988) | $\frac{0.65}{K_m} \left( \frac{T_m}{\tau_m} \right)^{1.04432}$     | $\frac{T_m}{0.9895 + 0.09539 \frac{\tau_m}{T_m}}$                 | $0.50814 T_m \left( \frac{\tau_m}{T_m} \right)^{1.08433}$ | $0 < \frac{\tau_m}{T_m} \leq 1$ |

Table 5-3 PID Controller Tuning Rules

The comment for each set of rules specifies the acceptable process time delay-to-time constant ratio. Table 5-1 shows that both processes satisfied this criterion. These rules allow values of the controller settings to be calculated, i.e.  $K_c$ ,  $T_i$  and  $T_d$ , from the process model parameters obtained from identification, i.e.  $K_m$ ,  $T_m$  and  $\tau_m$ . These rules were applied to all process models obtained for both processes. For each process PI and PID controller settings were determined for servo and regulator tuning. Therefore 48 different controllers were subsequently determined, i.e. 12 flow process controllers and 36 temperature controllers. The flow process does not depend on the operating conditions of the temperature process. Therefore three flow controllers were determined corresponding to three operating conditions (high, medium and low flows), for each type of controller i.e. PI

regulator, PI servo, PID regulator, and PID servo. Whereas, in the case of the temperature process, nine of each type had to be determined. This was because of the temperature process' dependency on the flow process, i.e. three temperature process operating conditions times three flow process operating conditions, for each type of controller. Tables 5-4 and 5-5 show the resulting controllers for the flow and temperature processes respectively. Average model controller settings were also determined and are highlighted in green.

| Operating Condition | PI Controllers           |                |                                |                | PID Controllers                  |                |                |                                  |                |                |
|---------------------|--------------------------|----------------|--------------------------------|----------------|----------------------------------|----------------|----------------|----------------------------------|----------------|----------------|
|                     | Min. IAE (Murrill, 1967) |                | Min. IAE(Rovira et. al., 1969) |                | Min. IAE (Kaya and Scheib, 1988) |                |                | Min. IAE (Kaya and Scheib, 1988) |                |                |
| Flow                | K <sub>c</sub>           | T <sub>i</sub> | K <sub>c</sub>                 | T <sub>i</sub> | K <sub>c</sub>                   | T <sub>i</sub> | T <sub>d</sub> | K <sub>c</sub>                   | T <sub>i</sub> | T <sub>d</sub> |
| L                   | 5.97                     | 2.16           | 4.05                           | 2.99           | 4.74                             | 1.02           | 0.65           | 4.18                             | 2.64           | 0.45           |
| M                   | 1.63                     | 2.10           | 1.16                           | 2.29           | 1.42                             | 1.15           | 0.68           | 1.11                             | 1.85           | 0.52           |
| H                   | 0.87                     | 1.74           | 0.63                           | 1.78           | 0.78                             | 0.99           | 0.58           | 0.59                             | 1.38           | 0.45           |
| Av.                 | 1.82                     | 2.01           | 1.28                           | 2.35           | 1.54                             | 1.05           | 0.64           | 1.25                             | 1.95           | 0.47           |
|                     | Regulator Tuning         |                | Servo Tuning                   |                | Regulator Tuning                 |                |                | Servo Tuning                     |                |                |

Table 5-4 Flow Process – PI and PID controller settings

| Operating Conditions | PI Controllers           |                |                                |                | PID Controllers                  |                |                |                                  |                |                |       |
|----------------------|--------------------------|----------------|--------------------------------|----------------|----------------------------------|----------------|----------------|----------------------------------|----------------|----------------|-------|
|                      | Min. IAE (Murrill, 1967) |                | Min. IAE(Rovira et. al., 1969) |                | Min. IAE (Kaya and Scheib, 1988) |                |                | Min. IAE (Kaya and Scheib, 1988) |                |                |       |
| Flow Temp.           | K <sub>c</sub>           | T <sub>i</sub> | K <sub>c</sub>                 | T <sub>i</sub> | K <sub>c</sub>                   | T <sub>i</sub> | T <sub>d</sub> | K <sub>c</sub>                   | T <sub>i</sub> | T <sub>d</sub> |       |
| 30 % flow            | L                        | 11.88          | 78.27                          | 7.72           | 132.39                           | 8.74           | 32.76          | 22.03                            | 8.49           | 122.27         | 14.51 |
|                      | M                        | 10.84          | 64.38                          | 6.56           | 157.24                           | 7.01           | 22.11          | 16.24                            | 8.01           | 152.49         | 9.61  |
|                      | H                        | 13.78          | 66.57                          | 8.42           | 154.98                           | 9.06           | 23.44          | 17.02                            | 10.14          | 149.59         | 10.21 |
| 50 % flow            | L                        | 23.49          | 46.79                          | 13.97          | 125.76                           | 14.71          | 15.28          | 11.48                            | 17.51          | 122.98         | 6.61  |
|                      | M                        | 15.60          | 45.66                          | 9.43           | 111.99                           | 10.07          | 15.64          | 11.50                            | 11.53          | 108.64         | 6.80  |
|                      | H                        | 21.82          | 47.66                          | 13.22          | 115.91                           | 14.13          | 16.40          | 12.03                            | 16.12          | 112.36         | 7.13  |
| 70 % flow            | L                        | 14.56          | 55.74                          | 9.29           | 104.00                           | 10.34          | 22.10          | 15.22                            | 10.50          | 97.65          | 9.73  |
|                      | M                        | 10.44          | 58.27                          | 6.68           | 106.46                           | 7.47           | 23.37          | 16.02                            | 7.51           | 99.63          | 10.31 |
|                      | H                        | 18.47          | 53.11                          | 11.30          | 122.57                           | 12.18          | 18.79          | 13.62                            | 13.58          | 118.20         | 8.19  |
| Av.                  | 11.62                    | 69.84          | 7.23                           | 147.40         | 7.90                             | 25.90          | 18.38          | 8.48                             | 140.73         | 11.34          |       |
|                      | Regulator Tuning         |                | Servo Tuning                   |                | Regulator Tuning                 |                |                | Servo Tuning                     |                |                |       |

Table 5-5 Temperature Process – PI and PID controller settings

5.2.1 Note on controller implementation

PI and PID controllers will be implemented in the Matlab/Simulink (digital) environment. All available process information is digital. Therefore the controllers used are in fact digital controllers,

and should be treated as such when determining controller settings. This is a very important point when dealing with processes that have particularly fast dynamics. The sampling time of the data acquisition system used, in effect, adds an extra time delay in the signal to the final control element. This can narrow the margin of stability somewhat. This effect is negligible when the sampling time is relatively small in comparison to the process time constant. Continuous time and discrete time PI and PID controllers will have essentially the same behaviour as long as  $\Delta t/T \leq 0.1$  (Seborg *et. al.*, 1989), where  $\Delta t$  is the sampling time and  $T$  is the processes time constant. Hence controller settings obtained through continuous (analogue) controller design can be used in the discrete (digital) time domain. The dynamic behaviour of a sampler plus zero order hold used in a data acquisition system can be approximated by a time delay equal to one-half the sampling period. Thus it is common practice to include this delay when tuning PID controllers.  $\Delta t/2$  should be added to the process time delay before computing controller settings. The sampling time chosen was 0.1 seconds. The effect of halving this and adding to the smallest process model time constant was not enough to make any significant change to the calculated settings.

### 5.3 Preliminary Controller Tests

The controller settings in tables 5-4 and 5-5 had to be tested and validated before implementation. The large number of controllers (48 in total) meant that it was not feasible to test all of them. This would take a considerable amount of time, especially in the case of the temperature process where the system dynamics were considerably slower than the flow process. Instead, all flow controllers designed for medium flow and all temperature controllers designed for medium temperature at 50% flow were tested, i.e. all settings highlighted in tables 5-4 and 5-5.

#### 5.3.1 Flow process controller tests

During all of these tests the temperature was held constant at the ambient room temperature of 24°C.

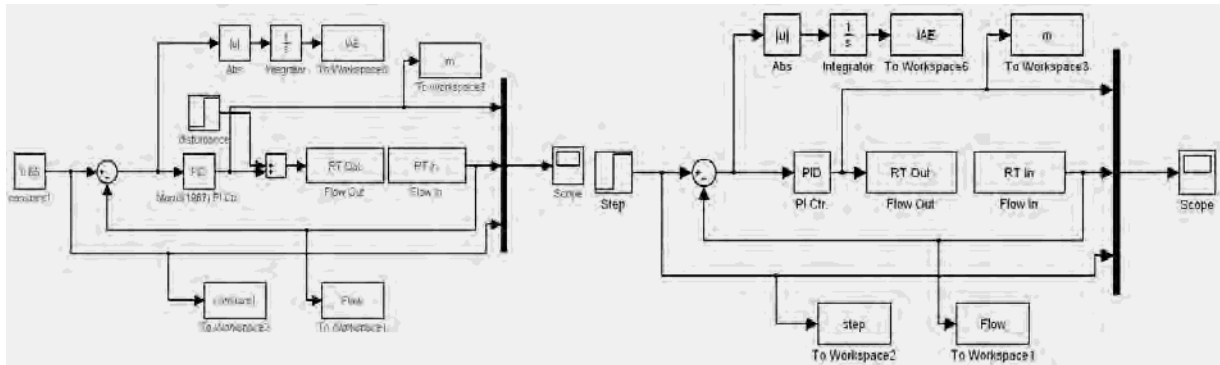
##### 5.3.1.1 PI controller tests

The standard Simulink PID controller block was used with settings given table 5-4. It should be noted that the default set-up of this block is slightly different to the PI controller architecture chosen, i.e. equation 5-1. Equation 5-3 shows the Simulink set-up of the PID controller block.

$$G_C(s) = P + \frac{I}{s} + Ds \quad \text{Equation 5-3 Simulink PID controller set-up}$$

Comparing this to equation 5-1, the 'D' portion should be set to zero,  $P = K_c$ , and  $I = K_c/T_i$ . A full set of values used for the Simulink implementation can be found in appendix J. The PI controllers were

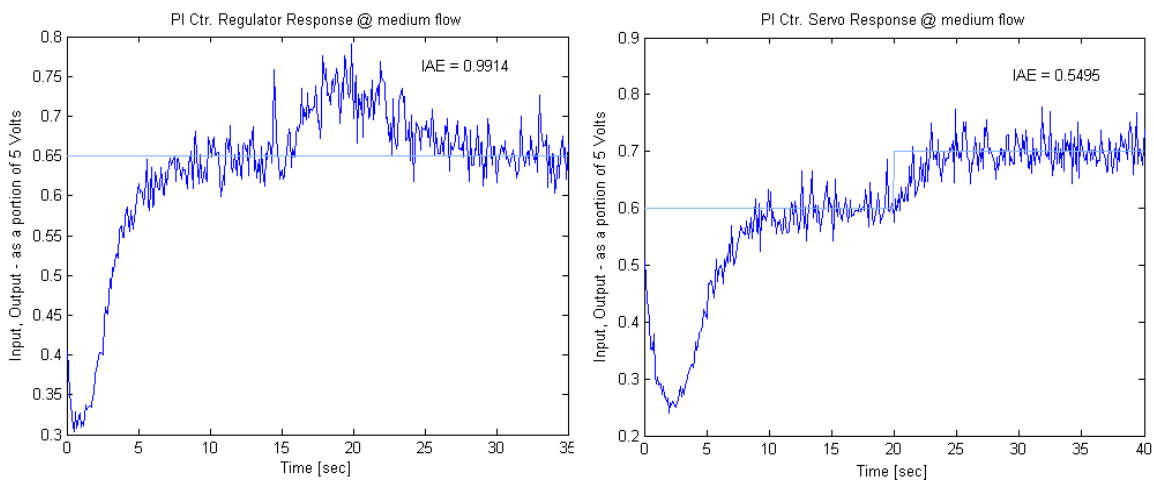
set up in closed loop regulator and servo mode as shown in figure 5-1. The controller had different settings for regulator and servo mode, and in each case the integral of absolute error (IAE) was recorded . The closed loop input (constant1 or step), output (Flow), and the controller output (m) were recorded to the workspace. Each signal was viewed through a multiplexer to a scope.



**Figure 5-1 Simulink model files for the flow process regulator and servo PI controller tests**

In the regulator response test the flow was held constant at an input of 65% flow, corresponding to medium flow (i.e. halfway between the upper and lower flow limits of 75% and 55%). The disturbance size was a step change of 0 to 0.5, corresponding to 0 to 2.5 Volts, and occurred 15 seconds after the beginning of the simulation. The simulation was 35 seconds in length.

In the servo response test the input step was small so that it would remain in the limits of medium flow operating conditions (i.e. between 55% and 75% input). The step was from 60% to 70% and occurred 20 seconds after the beginning of the simulation, this was to allow for the output to initially settle. In both regulator and servo tests the IAE value was recorded from 5 seconds before the input disturbance and step respectively, so that the initial settling error was not recorded. Figure 5-2 shows the regulator and servo response plots obtained from the PI flow controller tests.



**Figure 5-2 Regulator and servo response plots obtained from the PI flow controller tests**

5.3.1.2 PID controller tests

The PID controller used was made up from a standard Simulink PID controller block, with the derivative portion set to zero as before. The derivative part of the controller was given by a separate transfer function block, marked 'D' in figure 5-3. As before the regulator and servo controllers had separate settings as given in table 5-4. The PID controllers were set up in closed loop regulator and servo mode, as shown in figure 5-3. Again the integral of absolute error was recorded (IAE), closed loop input (constant1 or step), output (Flow), and the controller output (m) were recorded to the workspace.

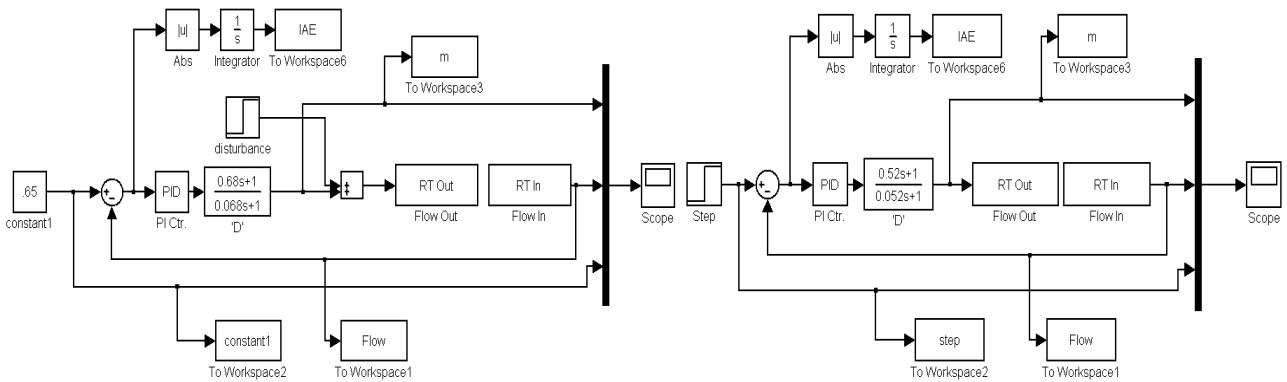


Figure 5-3 Simulink model files for the flow process regulator and servo PID controller tests

All test conditions (i.e. disturbance and step size and time) were as before for the PI test. Figure 5-4 shows the regulator and servo response plots obtained from the PID flow controller tests.

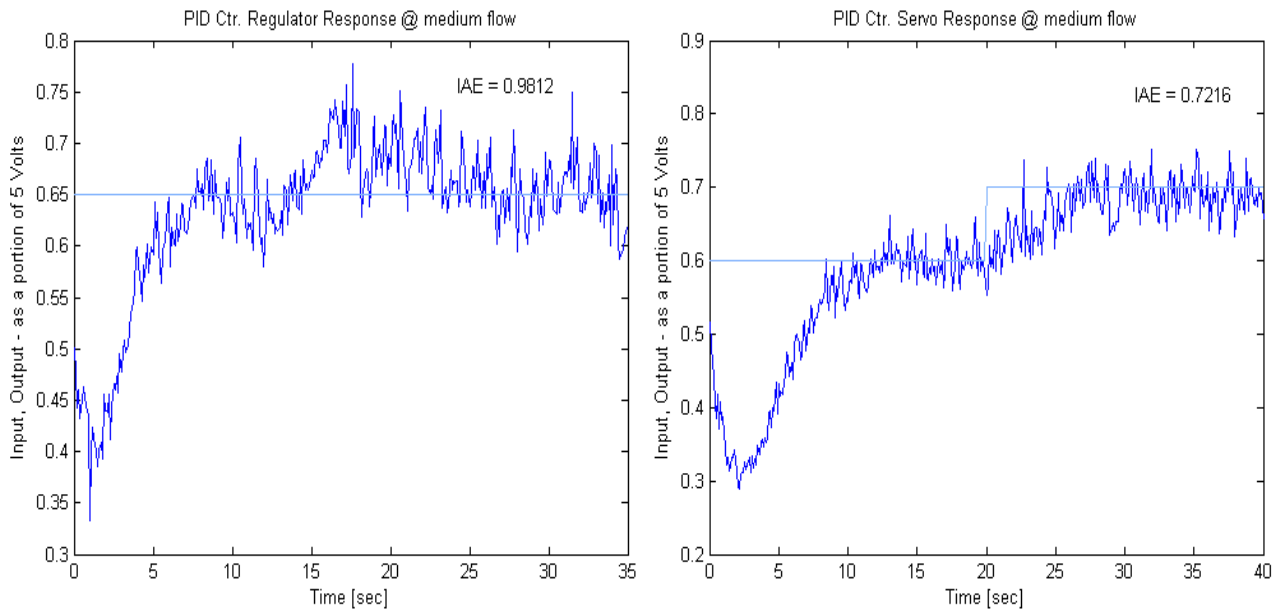


Figure 5-4 Regulator and servo response plots obtained from the PID flow controller tests.

5.3.2 Temperature process controller tests

During the following tests the flow output was held constant at 50% in open loop, i.e. input of 83%.

5.3.2.1 PI controller tests

The PI controllers for the temperature process were set up as before for the flow process in section 5.3.1.1, and as shown in figure 5-5. The controller had different settings for regulator and servo mode, as given in table 5-4. The closed loop input (constant1 or step), output (Temp), and the controller output (m) were recorded to the workspace.

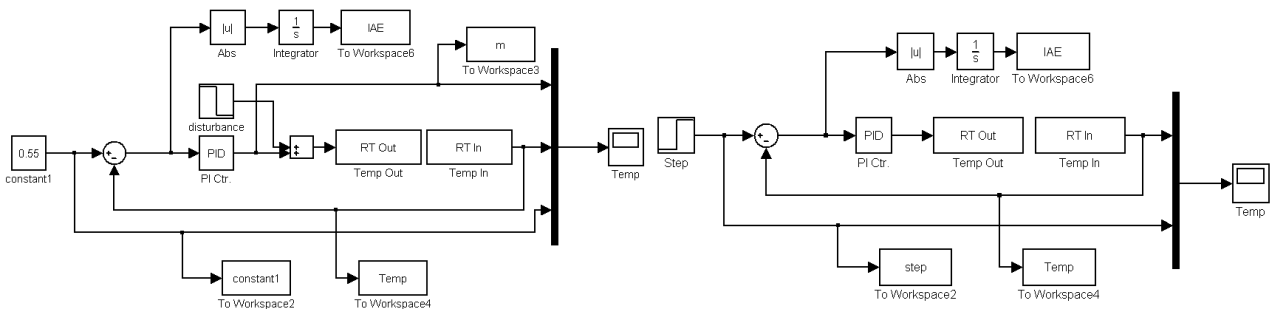


Figure 5-5 Simulink model files for the temperature process regulator and servo PI controller tests

In the regulator response test, the temperature was held constant at an input of 55% (or  $\approx 55^{\circ}\text{C}$ ), corresponding to medium temperature (i.e. halfway between the upper and lower temperature limits of 65% and 45%). The disturbance size was a step change of 0 to 1, corresponding to 0 to 5 Volts, and occurred 100 seconds after the beginning of the simulation. The simulation was 700 seconds in length.

In the servo response test the input step was small so that it would remain in the limits of medium temperature operating conditions (i.e. between 45% and 65% input). The step was from 50% to 60% and occurred 100 seconds after the beginning of the simulation, this was to allow for the output to initially settle. Figure 5-6 shows the regulator and servo response plots obtained from the PI temperature controller tests.

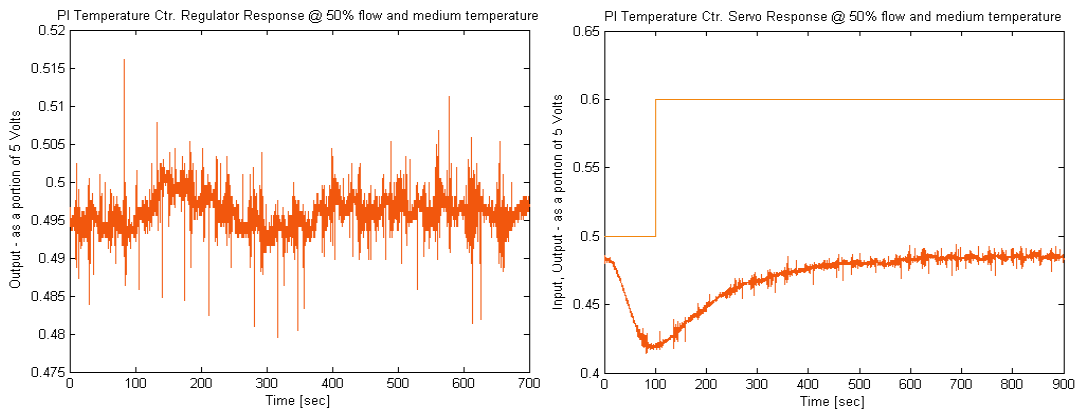


Figure 5-6 Regulator and servo response plots obtained from the PI temperature controller tests.

5.3.2.2 PID controller tests

The PID controllers for the temperature process were set up as before for the flow process in section 5.3.1.2, as in figure 5-7. As before the regulator and servo controllers had separate settings as given in table 5-4. All test conditions (i.e. disturbance and step size and time) were as before for the PI test. Figure 5-8 shows the regulator and servo response plots obtained from the PID temperature controller.

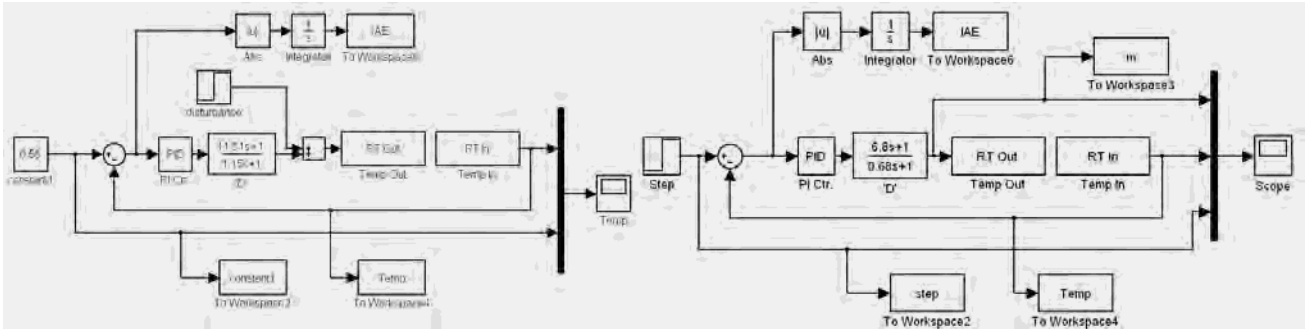


Figure 5-7 Simulink model files for the temperature process regulator and servo PID controller tests

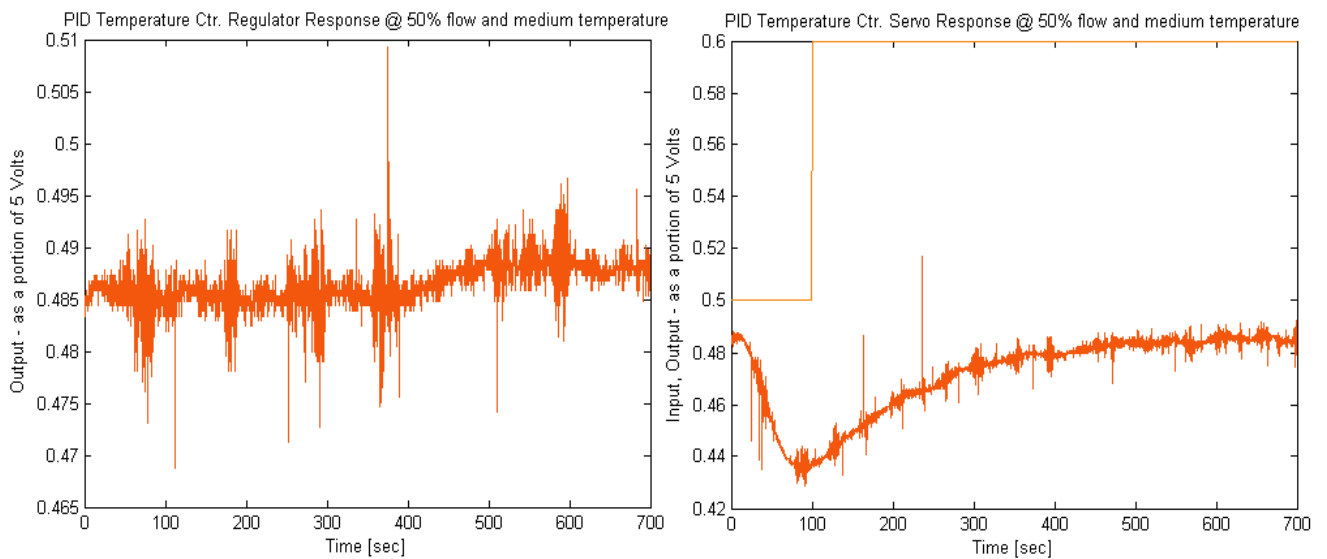
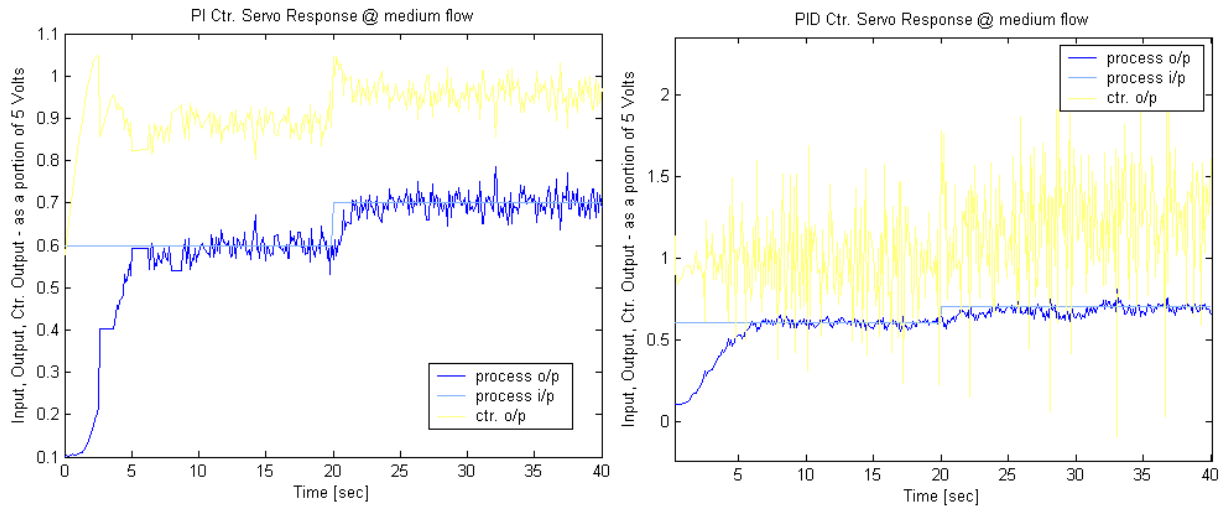


Figure 5-8 Regulator and servo response plots obtained from the PID temperature controller tests.

5.3.3 Comments

It was found from the preliminary tests on the flow process that both PI and PID controllers performed well. The operating conditions of the test corresponded to the controller settings used, so good performance was to be expected here. The regulator and servo responses (figures 5-2 and 5-3), for both controllers, were satisfactory, i.e. a large disturbance to the process was successfully rejected and the process output could adequately follow a change in input. Interestingly, the minimum IAE values showed that, overall, the PI controller performed better than the PID controller. The regulator IAE values were only slightly different (0.9812 for PID and 0.9914 for PI). The servo IAE value for the PI

controller was substantially less than that of the PID controller (0.5495 for PI and for 0.7216 PID). However, it should be borne in mind that the level of noise (particularly for the PID controller) will have a detrimental effect on the IAE value. This can be seen from looking at the controller output during these tests. Figure 5-9 shows the controller output for a servo test on the flow process for PI (left-hand plot) and PID (right-hand plot) controllers.



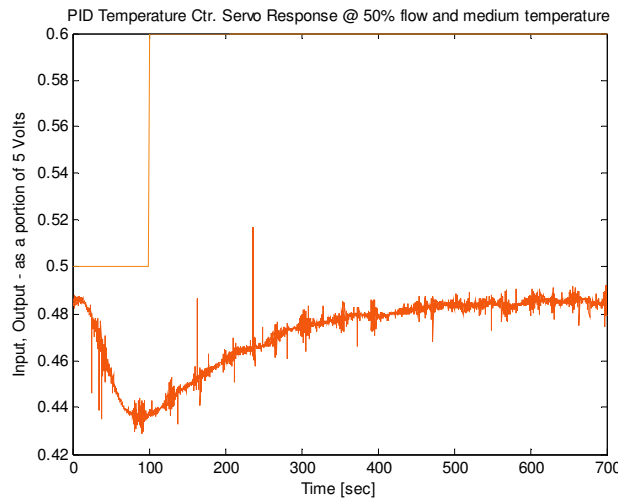
**Figure 5-9 Servo responses for PI and PID flow controllers with controller output shown**

The PID controller tends to suffer from the effect of noise in the feedback signal. The ‘D’ portion of the controller exaggerates the noise and the result is a controller output that is more erratic and varied than that of the PI controller. The process itself may not be able to keep up with the rate of change of the control signal and hence leading to slightly deteriorated performance. Similar behaviour can be seen when comparing the regulator responses of the PI and PID flow controller outputs. Nevertheless the flow controller settings were validated for implementation.

The temperature controller tests yielded some other interesting information, see figures 5-6 and 5-8. The regulator responses seemed relatively unaffected by the disturbance with only a very small change in system output. This could be for two reasons. It is possible that the regulator controller settings were suitable and the system performance was exceptionally good. It is also possible that the disturbance to the temperature process could not have had any effect on the output temperature anyhow because of the relatively high flow rate (50%). Perhaps similar tests at lower flow rates could establish the answer to this. So it may be the case that regulator performance of the temperature process is not an issue one way or another.

Conversely, the servo responses did not perform as they should. An unexpected *inverse response* was observed in the temperature output, i.e. the temperature output initially decreased steadily for a time and then recovered, as shown figure 5-10.



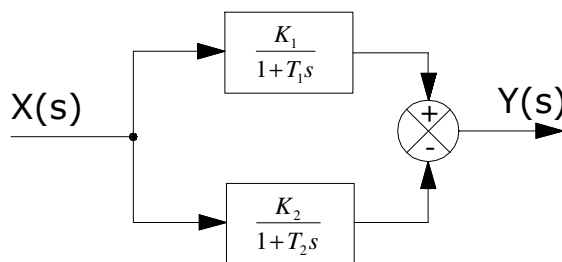


**Figure 5-10 Inverse response – PID temperature controller servo test**

An inverse response usually occurs when two physical effects act on the process output variable in opposite ways and with different time scales (Seborg et. al., 1989). It may be the case that the *fast* effect is due to the interaction associated with the flow rate (as discussed in section 4.3) and that the *slow* effect is the temperature process. This argument can be validated by comparing the average temperature process model ( $G_{11}$ ) with the average interaction model ( $G_{12}$ ), as shown below.

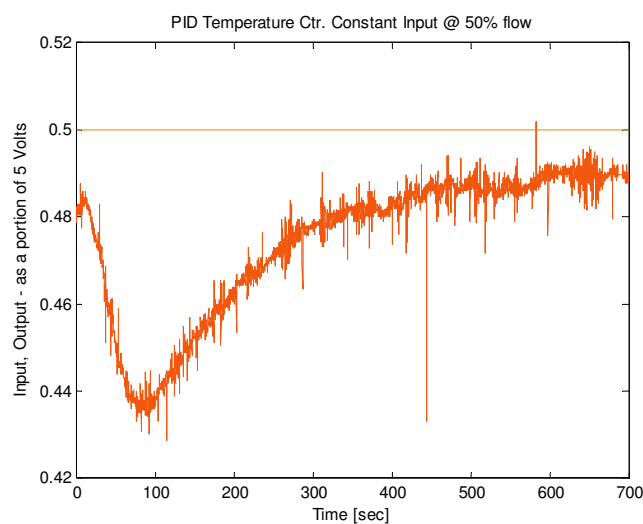
$$G_{11-Av.Temp}(s) = \frac{0.38e^{-21s}}{1 + 121s} \quad G_{21-Av.Temp}(s) = \frac{-0.15e^{-10s}}{1 + 73s}$$

It is correct to say that the two effects act on the output variable in opposite ways because the interaction effect tends to reduce the temperature as flow increases, hence the negative gain. It is also true that the interaction model is faster than the temperature model. The combined time constant and time delay of the temperature process is 142 seconds compared to 83 seconds for the interaction model. So it seems that the inverse response may be due to the interaction. Strictly speaking this cannot be entirely true. Usually this type of response will occur when two physical effects have the same input and one combined output. Figure 5-11 shows an example of this where two first order processes are acting in parallel.



**Figure 5-11 Two first order processes acting in parallel**

In our case the two processes (the temperature process and the interaction process) have the same output but two different inputs, as shown before in figure 4-13. Nevertheless, an inverse response could occur if both processes experience a step input at the same time. In the tests carried out, the flow was held constant and the temperature process was allowed to settle before it received a step input. This again breaches the conditions stated that are required for a true inverse response. It was noticed that at the beginning of simulations the power is initially cut to both processes (Matlab/Simulink default simulation condition) so by simply starting the test both processes receive an instantaneous step input. This makes sense because the temperature output begins to fall before the step to the temperature process occurs. To prove that this was the case another test was carried out where the temperature and flow inputs were simply left constant with the PID temperature controller in place. The resulting response is shown in figure 5-12.



**Figure 5-12 PID Temperature controller tests with temperature input held constant**

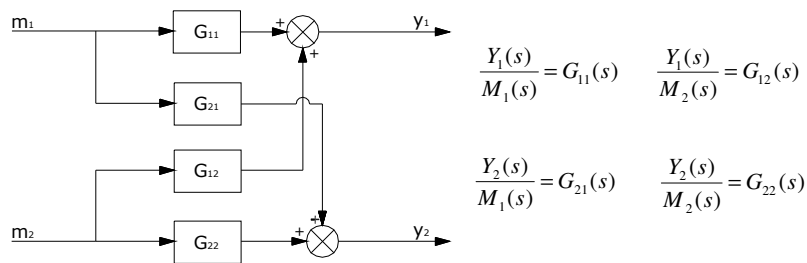
The effect of this inverse response is to add an extra delay to the temperature process that had not been accounted for before. So from this point of view the controller settings developed for the temperature controllers were not adequate.

It is the opinion of the author that the inverse response observed was due the interacting effect of changes in flow conditions on the temperature process, as discussed in section 4.3. This problem may be resolved in two ways; an appropriately designed *decoupler* may eliminate some or all of the interaction effect and hence remove the inverse response; alternatively tests could be carried out that estimate the extra delay due to the inverse response, and hence the relevant controller settings can be updated, and re-testing could occur. It is also possible to design a specific inverse response compensator instead of using a PID controller with adjusted settings. Both methods of dealing with

this problem will be explored. The design and implementation of a decoupling system is relatively easier than the inverse response methods mentioned. For this reason it was proposed that a decoupler be designed and tested before inverse response compensation methods were considered.

**5.4 Process Interactions and Decoupling**

Until now each process has been treated as single-input single-output (SISO) process and any attempt to control each process has been based on this assumption. However, it has been established that an interaction exists between the two processes and that its effect is strong enough not to be ignored. So now the overall system, as a multi-input multi-output (MIMO) process, should be treated as such in terms of controller design. Figure 5-13 shows a 2x2 MIMO block diagram and its process transfer functions.



**Figure 5-13 MIMO block diagram**

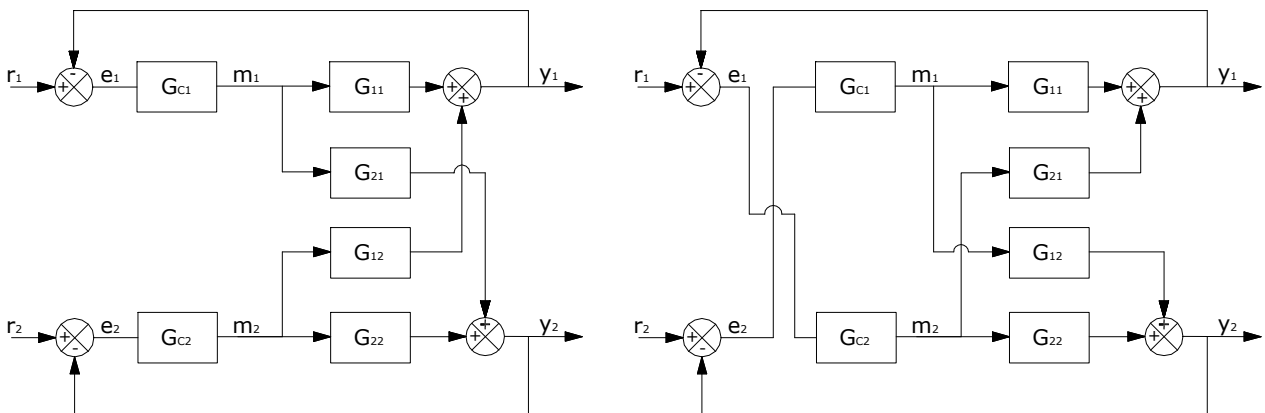
Using the principle of superposition the output variables  $y_1$  and  $y_2$  are given as;

$$y_1 = G_{11}.m_1 + G_{12}.m_2 \quad ; \quad y_2 = G_{21}.m_1 + G_{22}.m_2$$

Or in Matrix form;

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{pmatrix} \begin{pmatrix} m_1 \\ m_2 \end{pmatrix}$$

When a conventional multi-loop control scheme consisting of two feedback controllers is used two possible configurations are possible, and are shown in figure 5-14.



**Figure 5-14 Two possible 2x2 MIMO control configurations**

In the left hand block diagram  $y_1$  is coupled with  $m_1$  and  $y_2$  is coupled with  $m_2$ . In the right hand block diagram  $y_1$  is coupled with  $m_2$  and  $y_2$  is coupled with  $m_1$ . It can be difficult to establish which configuration to use, especially when dealing with systems with more than two input and output variables. It has been shown using the *relative gain array* method<sup>8</sup>, proposed by Bristol (1966), that the most suitable configuration is as shown in the left-hand block diagram of figure 5-14.

Section 4.3 found that an interaction existed between  $m_2$  (flow i/p) and  $y_1$  (temperature o/p), and that no interaction existed between  $m_1$  (temperature i/p) and  $y_2$  (flow o/p). Three models were developed for  $G_{12}$  corresponding to three temperature operating conditions. As no interaction was observed between  $m_1$  and  $y_2$  then  $G_{21} = 0$ . Process models for  $G_{11}$  (temperature process) and  $G_{22}$  (flow process) have been established in detail in chapter 3. With this information gathered it is possible to design a decoupler to reduce the process interaction  $G_{12}$ .

### 5.4.1 Decoupling Control Systems

A decoupling controller is an additional controller added the conventional multi-loop configuration, as shown in figure 5-15. Consider the 2x2 process shown in figure 5-14 (i.e.  $y_1$  is coupled with  $m_1$  and  $y_2$  is coupled with  $m_2$ ). Assume both outputs are initially at their desired values, and a disturbance causes the controller of loop 2 to vary the value of  $m_2$ . This will then cause an unwanted disturbance in loop 1 and hence cause  $y_1$  to vary from its desired value. Given that;  $y_1 = G_{11}.m_1 + G_{12}.m_2$ , then to keep  $y_1$  constant (i.e.  $Y_1(s) = 0$ ),  $m_1$  must be adjusted by  $-(G_{12}/G_{11}).m_2$ . So by introducing a *decoupler* of transfer function  $-(G_{12}/G_{11})$ , the interacting effect of loop 2 on loop 1 is eliminated. The same argument can be applied to the effect of loop 1 on loop 2, and hence yield a decoupler with transfer function  $-(G_{21}/G_{22})$ . Figure 5-15 shows such a decoupling control system.

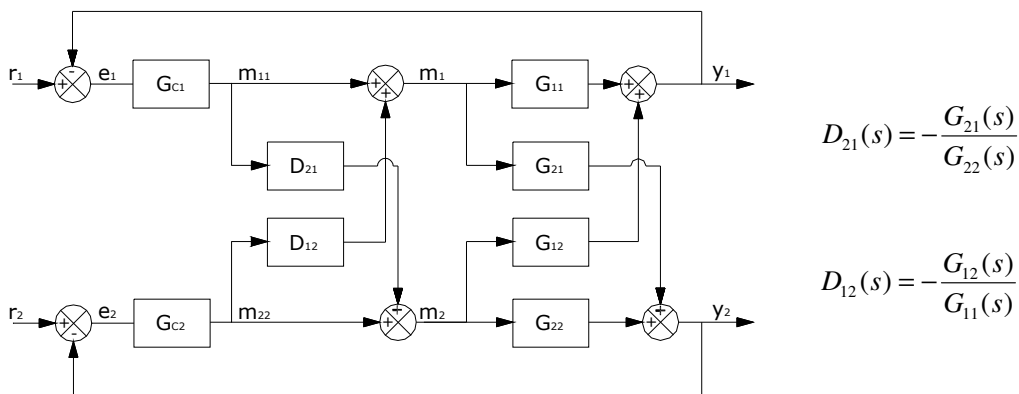


Figure 5-15 A decoupling control system

<sup>8</sup> See Appendix K for Bristol's RGA method and proof of chosen controlled to manipulated variables.

The decouplers  $D_{12}$  and  $D_{21}$  may not always be physically realizable especially when dealing with models with different time delays. For example, sometimes it may occur that the ideal decoupler has a time advance term (i.e.  $e^{+2s}$ ), which is obviously impossible to implement. A less ambitious approach to full decoupling but still very effective is *static decoupling*. This is where the decouplers are designed based on the steady state process interactions only. The design equations for the decouplers in figure 5-15 can be adjusted by setting  $s = 0$ , i.e. the process transfer functions are simply replaced by their corresponding steady state gains, so that;  $D_{12} = -K_{12}/K_{11}$  and  $D_{21} = -K_{21}/K_{22}$ . Since static decouplers are merely constants they are always physically realizable and easily implemented.

#### 5.4.2 Controller tests using static decouplers

From section 4.3 the following transfer functions describe the process interactions observed at medium temperature (between 45 and 65 per cent temperature input) and over the low-to-medium flow range approximately (tests were carried out using a very large flow step of 20 to 75 percent flow output);

$$G_{12-Med.Temp}(s) = \frac{-0.18e^{-8s}}{1+85s} \quad G_{21}(s) = 0$$

The appropriate process models corresponding to these conditions were given by;

$$G_{11_{MED-TEMP}}(s) = \frac{0.43e^{-16s}}{1+109s} \quad G_{22_{LOW-MED-FLOW}}(s) = \frac{0.76e^{-1.03s}}{1+2.31s}$$

Applying the static decoupling design equations yields;

$$D_{12} = 0.18/0.43 = 0.419 \quad \text{and} \quad D_{21} = 0/0.76 = 0$$

Since the decoupler for the flow process ( $D_{21}$ ) is zero it may be omitted completely from the block diagram. Also this means that the flow process controllers do not need to be re-tested. The temperature process controllers were re-tested with the above decoupler ( $D_{12}$ ) in place. All operating conditions and controller settings were as before in section 5.3.2. Figure 5-16 shows the Simulink model file for the test.

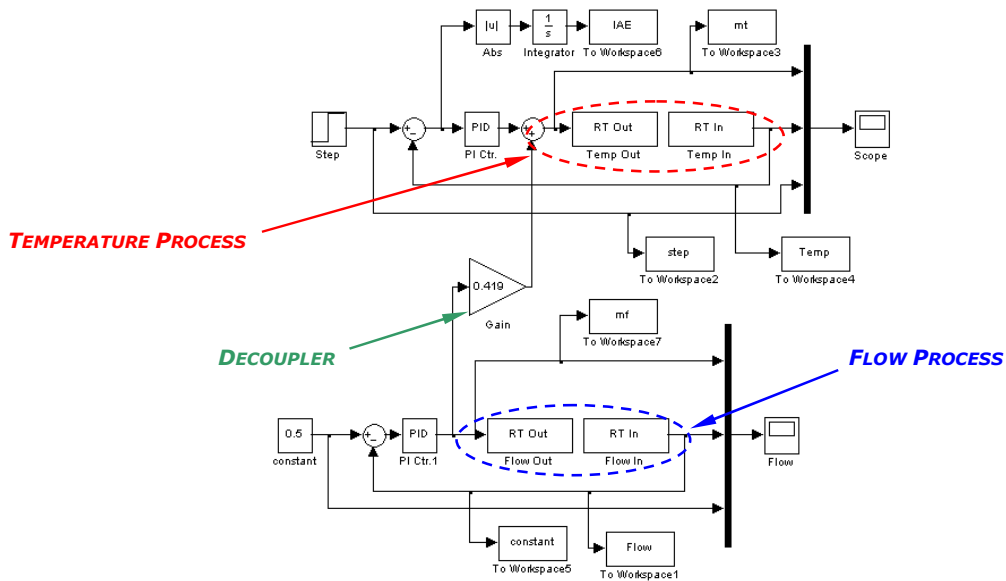


Figure 5-16 Decoupler control system test

Figure 5-17 shows the servo response for a PI temperature controller using the decoupling control system shown above. The controller settings used were specific to the operating conditions, as given in table 5-5.

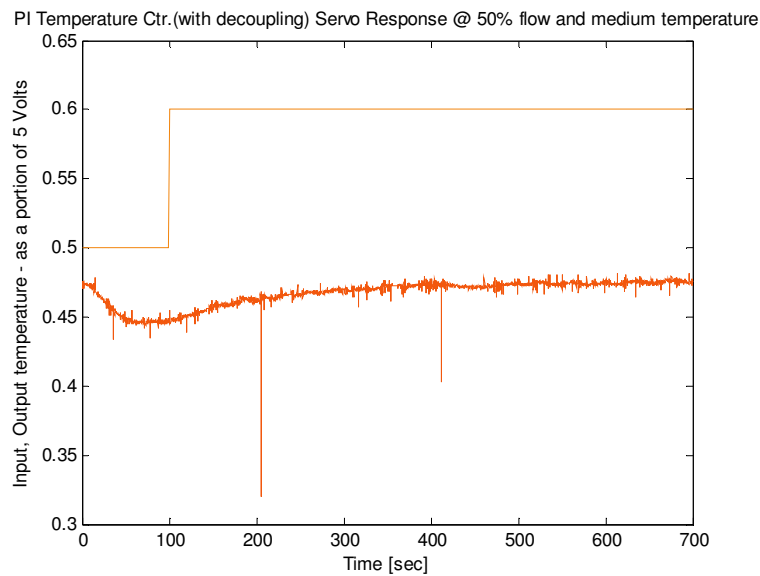
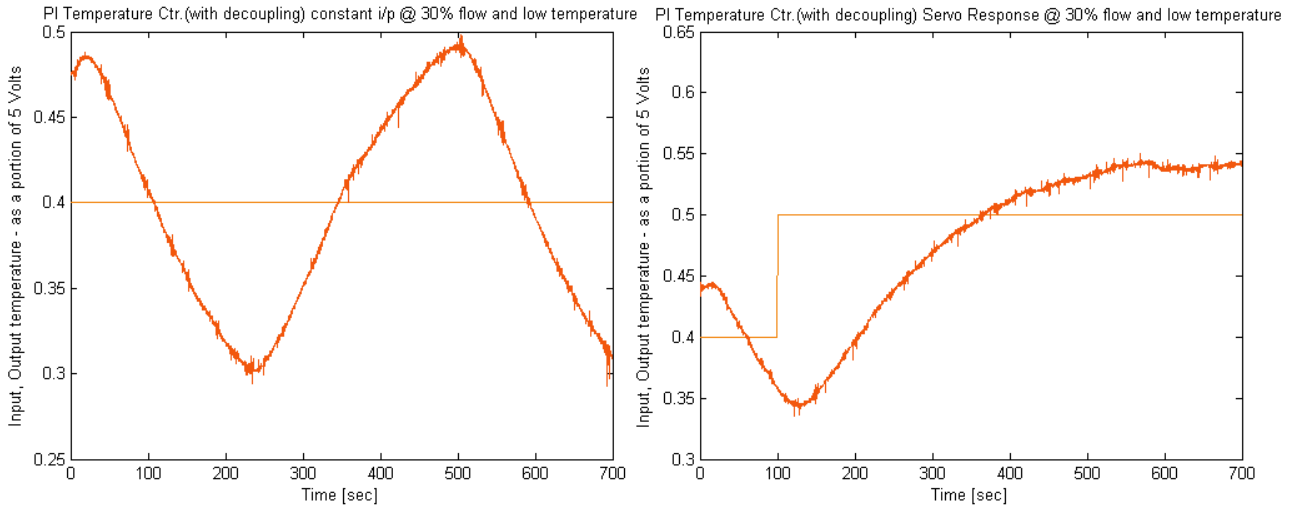


Figure 5-17 Servo response plot obtained for PI temperature controller with decoupling

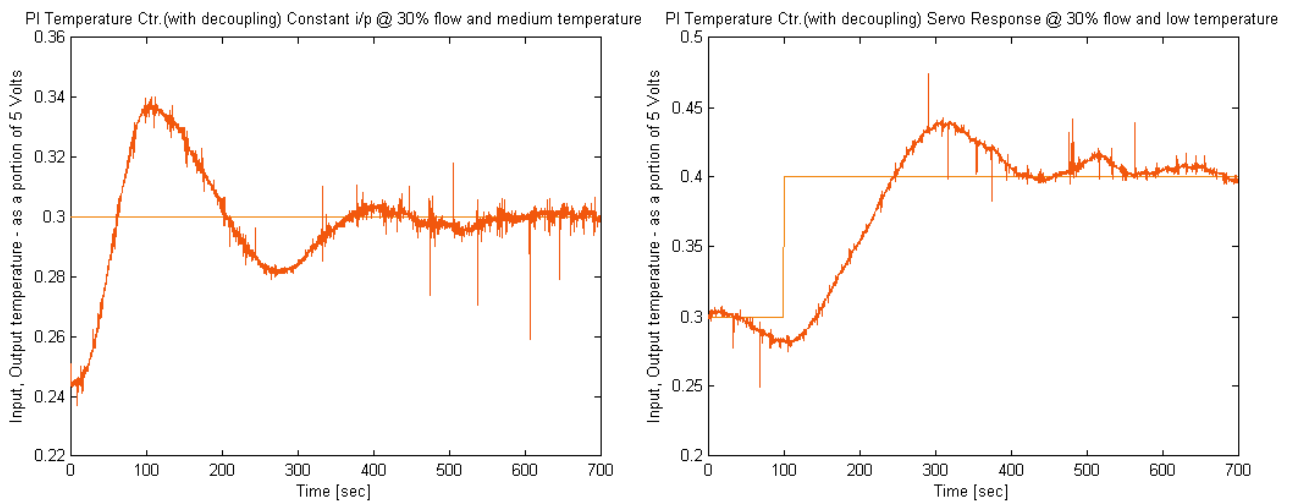
The plot is very similar to that of the previous tests carried out without the decoupler in place. What is noticeably better is that the size of the inverse response is reduced. The previous controller saw the temperature output drop to 0.4 before recovering. In the interest of curiosity the same test was carried out using a lower input step and at a lower flow rate. The stimulus for doing this was a suspicion that the inverse response observed could have been due to the temperature element being physically unable

to produce a temperature as high as what was sing demanded of it at such a high rate of flow (i.e. 50%). Figure 5-18 show plots for a constant input and servo response tests carried out at 30% flow and in the low temperature range.



**Figure 5-18 Constant input and servo response plots at 30% flow in the low temperature range using PI temperature controller with decoupling.**

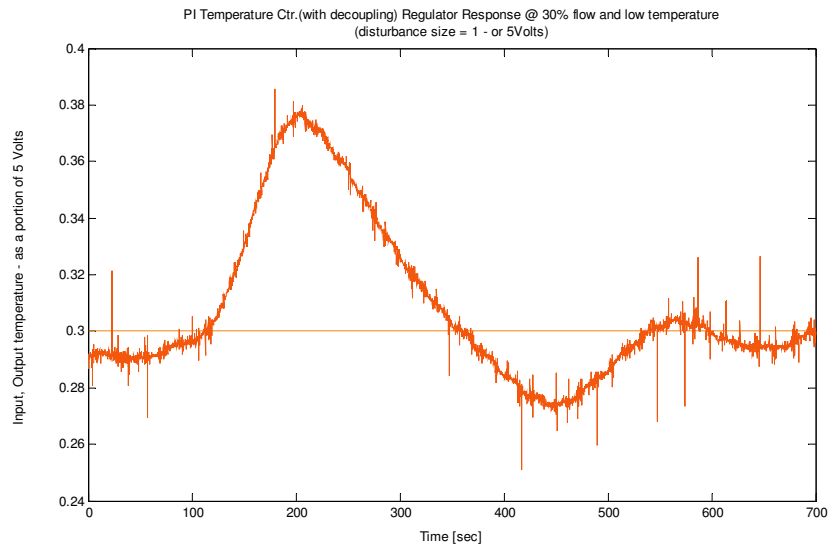
Initially it seemed that the decoupler was making the performance worse. It was then found on further inspection that the PI settings had not been re-adjusted for the new operating conditions, i.e. 30% flow and low temperature range. Figure 5-19 shows the plots from a similar test carried out with appropriately adjusted PI controller settings.



**Figure 5-19 Constant input and servo response plots at 30% flow in the low temperature range using updated PI temperature controller with decoupling.**

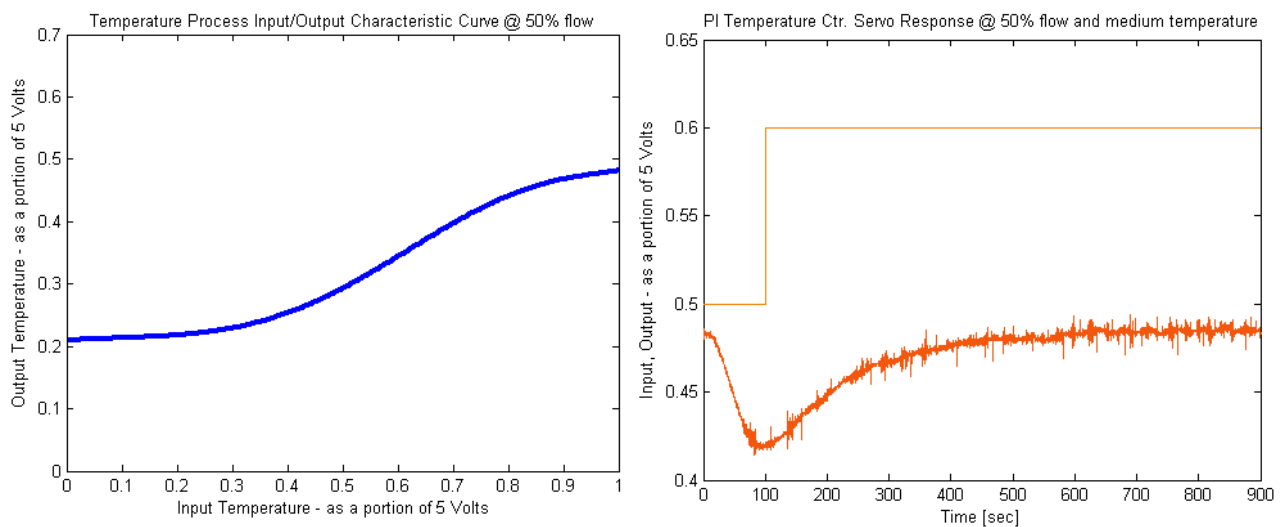
It is clear that the response is noticeably better with adjusted controller settings. Also the plots in figure 5-18 highlight the importance of updating the controller settings. The temperature process stability will depend on accurate and appropriate settings. The test was also carried out in regulator

mode. Figure 5-20 shows a plot of the response obtained. The disturbance step size was from 0-1, i.e. 0-5Volts, and was applied at time = 100 seconds.



**Figure 5-20 Temperature process regulator response plot at 30% flow in the low temperature range**

These results for both servo and regulator control were satisfactory. The problem involving the inverse response seen earlier was resolved. The initial test conditions used in section 5.3.2 were such that the temperature process was being asked to reach values that it was physically unable to reach. This can be confirmed by looking at the input-output characteristic curve for the temperature process developed in section 4.2.2. At 50% flow the output temperature cannot go beyond about 48°C (0.48 output). Figure 5-21 shows the temperature process characteristic curve for 50% and the servo test at 50% flow that yielded the inverse response.



**Figure 5-21 Input/output temperature process characteristic curve at 50% flow, and temperature process servo response plot at 50% flow**



From the characteristic curve it is clear that the temperature process is unable to produce temperatures greater than around 48°C when the flow rate is 50%. The process is therefore unable to reach the set point that it is being asked to reach in the servo test plot shown. Reasons for the presence of the inverse response have been discussed in section 5.3.3. It has also been noted that it is not a true inverse response. Referring to figures 5-19 and 5-20, the decoupling controller design has noticeably reduced the size of the initial drop in temperature (due to the so-called inverse response), the process output managed to follow the set point adequately, and the process successfully rejected a large disturbance. For these reasons inverse response compensation methods will not be implemented, although they will be briefly discussed.

### 5.5 Inverse Response Compensation Methods

The major problem created by inverse response control systems is the confusing scenario it presents to the controller. Having taken the appropriate action to give the desired result the controller is led to believe that it in fact took the wrong action, as the initial response does the opposite of what it should. The controller is then likely to compound the problem further in reacting to this illogical response and hence the system's stability may be put in jeopardy. Usually inverse response systems are controlled using conventional PID control or using an inverse response compensator. The derivative mode of the PID controller allows it to somewhat anticipate the "wrong way behaviour" of an inverse response system and therefore a PID controller can be suitable. In tuning the controller, the process model time delay should be approximated by including the time for the inverse response effect, as in figure 5-22.

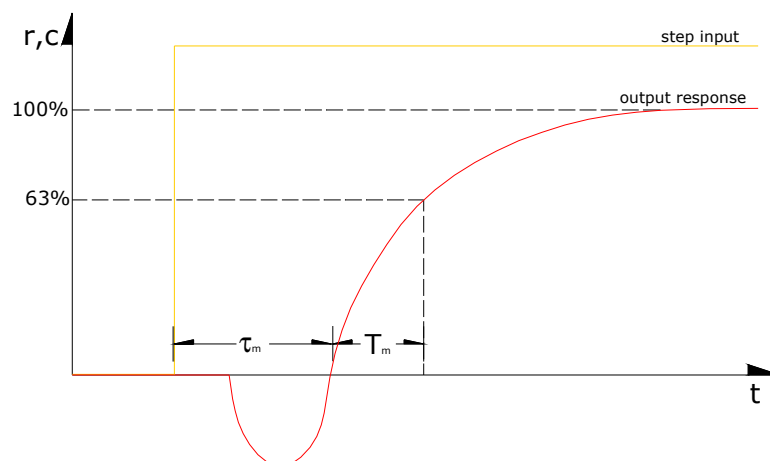


Figure 5-22 Inverse response modelling method

A PID controller that is tuned from such a model should be able to cope with an inverse response process. It can be shown that of all the conventional feedback controllers only the PID type can be used to control the inverse response system with any degree of success (Ogunnaike, 1994).

Alternatively an inverse response compensator can be used. The design first developed by Iinoya and Altpeter (1962), uses an inner feedback loop for the controller that is based on the process model. The design is similar to that of the Smith predictor compensator. A true inverse response process will have a right-hand-side zero present in the process transfer function, i.e.;

$$G(s) = \frac{1 - ns}{(1 + 2s)(1 + 5s)} \dots \text{typical inverse response process transfer function}$$

The  $1 - ns$  term is what causes an inverse response. Figure 5-23 shows the conventional feedback control of an inverse response system.

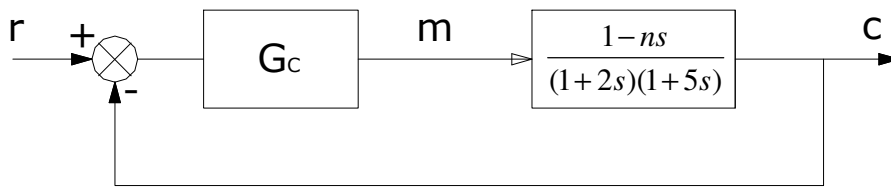


Figure 5-23 Conventional feedback control of an inverse response system

Now consider the situation where a minor feedback loop is introduced to the controller with the feedback transfer function given by the two first order lag terms of the process times a variable ( $\lambda$ ) of  $s$ , as shown in figure 5-24.

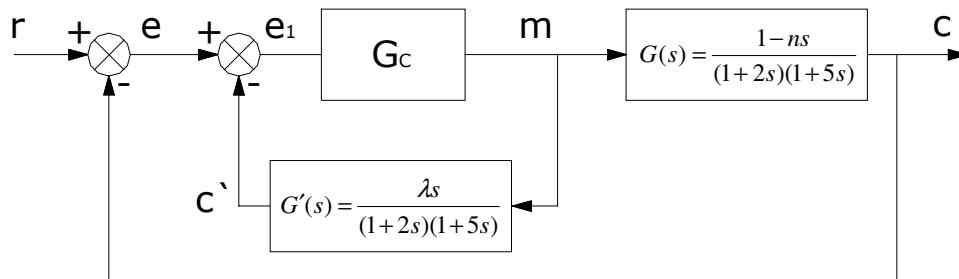


Figure 5-24 Block diagram incorporating inverse response compensator

The objective here is to make the controller think that it is receiving a signal from a “normal” system (as opposed to an inverse response system) by suitably choosing a value for  $\lambda$ . Now,  $c' = G'(s)m$  is the signal generated by the inner feedback loop, and therefore the signal reaching the controller is given by;

$$e_1 = r - c - c' \quad \text{or;} \quad e_1 = r - [G(s) + G'(s)].m$$

Now, let:  $G^*(s) = G(s) + G'(s)$  and;  $c^* = G^*(s).m$

So that;  $e_1 = r - c^*$ ; so now the controller only sees the apparent process  $G^*(s)$  given by;

$$G^*(s) = \frac{(1 - ns)}{(1 + 2s)(1 + 5s)} + \frac{\lambda s}{(1 + 2s)(1 + 5s)} = \frac{1}{(1 + 2s)(1 + 5s)} [1 + (\lambda - n)s]$$

So now, by choosing  $\lambda \geq n$  the transfer function of the apparent process  $G^*(s)$  no longer contains a right-hand side zero. Thus the minor loop provides a corrective signal that essentially eliminates the inverse response from the feedback loop. Owing to its similarities to the Smith predictor this design also suffers from sensitivity to model inaccuracies. Choosing  $\lambda > n$  (as opposed to  $\lambda = n$ ) has some advantages in case of model mismatch. It has been found that for optimal performance  $\lambda$  should be chosen such that:  $\lambda = 2n$ , providing minimum mean square deviation of the output from the set-point.

## 5.5 Gain Scheduling

Generally controllers are designed by obtaining a model for the process, designing a controller based on the process model, testing it, possibly by simulation prior to implementation, and then tuning it after installation. This approach works well provided that the plant parameters do not change with time or plant load (O'Dwyer, 2003b). It was found from the process identification that the dynamics of both processes depended on the given operating conditions; for example, the flow process displayed a low gain at low flows and a high gain at high flows. Similar variances were found to exist in the temperature process characteristics. It was also found that the flow process interacted with the temperature process. Designing one controller for each process that accounts for all operating conditions and interactions would be very difficult. An adaptive control approach offers a solution to this problem. An *adaptive control system* is one in which the controller parameters are adjusted automatically to compensate for changing process conditions (Seborg *et. al.*, 1989). One method of adaptive control is gain scheduling.

### 5.5.1 Simple gain scheduler design

The aim of the design is to adjust the controller settings appropriately as process operating conditions vary. Tables 5-4 and 5-5 show settings for the flow and temperature processes that are particular to the processes operating conditions. Once the settings were available, a method of skipping between these settings had to be determined. The gain scheduler was to be implemented in the Matlab/Simulink environment. The initial problem was determining how the processes input and output signals could be manipulated to distinguish between the various levels of operating conditions, i.e. high, medium, or low flow and temperature. It was found that this could be done using functions found in the *Math Operations* menu in Simulink. Simple *logic operators* such as *and* (AND), *or* (OR), *exclusive or* (XOR), and *negative or* (NOR) gates were used with *relational operators* (i.e.  $>$  - greater than) to manipulate the signal. Figure 5-25 shows how the flow signal was categorised as high, medium, or low.

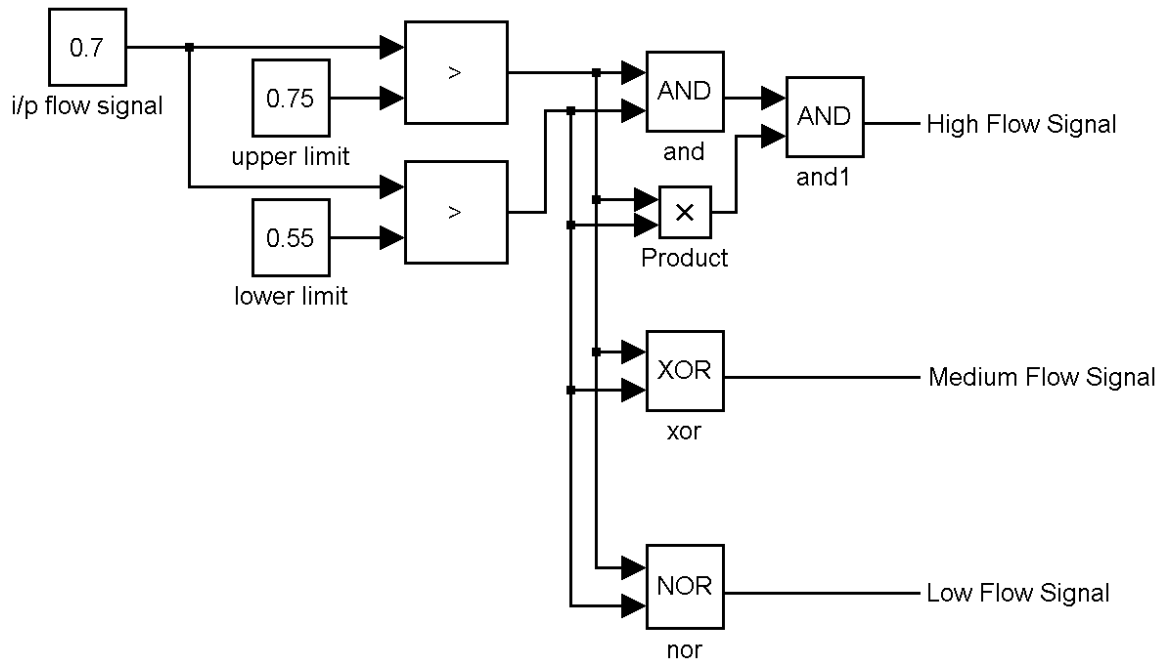


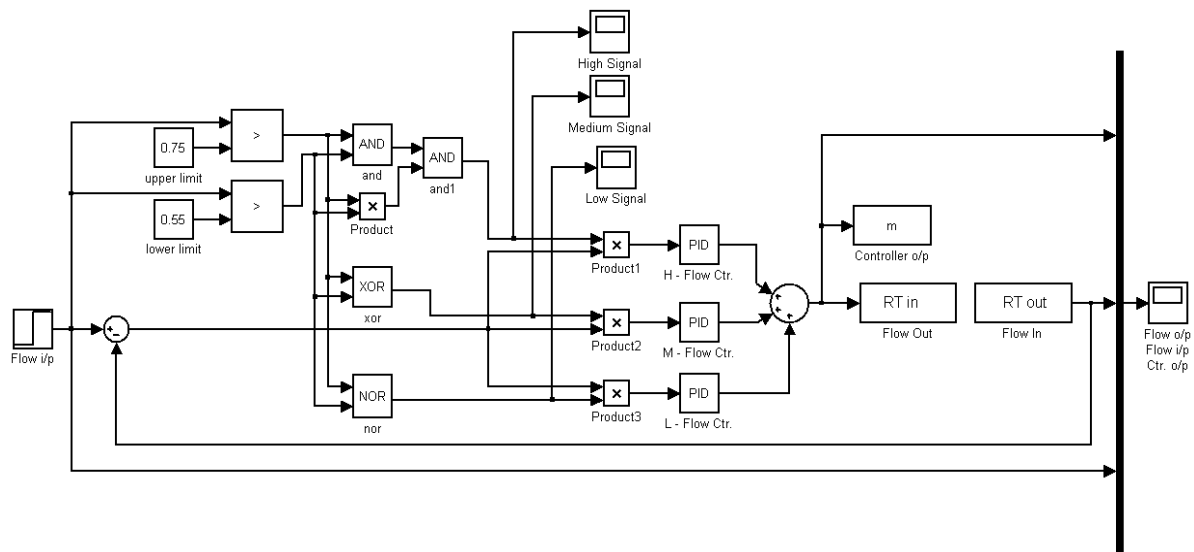
Figure 5-25 Simulink logic block arrangement to manipulate flow signal

The large blocks marked '>' output either 1 or 0. The output is '1' if the upper input is greater than the lower input, and '0' if the upper input is **not** greater than the lower input. So in figure 5-25 the upper block outputs '0' and the lower block outputs '1', i.e. the flow is between 55 and 75 per cent. The range of flow was defined earlier as being *high* if the flow input is greater than 75%, *medium* if the flow input is between 55% and 75%, and *low* when the flow input is less than 55%. The two signals from the relational operator blocks are routed through logic operator blocks to categorize the flow input signal. Table 5-6 shows how low, medium, and high signals were determined.

| Flow i/p<br>(Operating<br>Condition) | relational operator |      | logic operator |     |     |
|--------------------------------------|---------------------|------|----------------|-----|-----|
|                                      | >55%                | >75% | AND            | XOR | NOR |
| L                                    | 0                   | 0    | 0              | 0   | 1   |
| M                                    | 1                   | 0    | 0              | 1   | 0   |
| H                                    | 1                   | 1    | 1              | 0   | 0   |

Table 5-6 Truth table for categorising the flow input signal

This means that for any input signal only one of the three logic operator blocks will have an output of 1. This signal was then used to activate a controller specific to that operating condition. Each signal is multiplied by the error signal, the output of which was fed to PI controllers specific to the flow input category. The output of all three PI controllers is summed and subsequently fed to the process input, see figure 5-26.



**Figure 5-26** Gain scheduled PI flow control design

The figure shows a *scope* joined to each signal from the logic gates. These were used to ensure that at any one time only one controller was operating on the process. This design approach was also used for the temperature process. The temperature process had nine controllers to switch between, so this made the design a little more involved. Nevertheless the format of the design was the same. The temperature process had three controllers (for high, medium, and low temperature inputs) at three different flow rates; 30, 50, and 70 per cent (flow output). The same method of manipulating the process signal (to categorise it) was used, only this time the logic block arrangement shown in figure 5-25 was used twice. One set of logic blocks was used to determine whether the temperature input was in the high, medium or low range, and another set was used to determine the flow output conditions. Temperature inputs less than 45% are low, between 45% and 65% are medium, and above 65% are high. Controller settings were determined for each of these (see table 5-5) at three different output flow rates (i.e. 30, 50, 70 per cent). The cut-off limits to determine which settings to use were given by the median between the flow rates, i.e. 40% and 60% flow output. Figure 5-27 shows the entire design (including the flow process gain scheduler). In the figure both processes are set up in regulator mode. So the set-up should correspond to when the heating and ventilation system is up and running at the desired temperature and flow rate. It was decided that this design would not be tested. The gain scheduler design simply switches between fixed controller settings. Any tests carried out on this design would be equivalent to the tests already investigated in section 5.3. It should be noted also that the decoupling design discussed earlier is not built into this gain scheduler. Nevertheless, the design in figure 5-26 provides the infrastructure for a more advanced gain scheduling design.

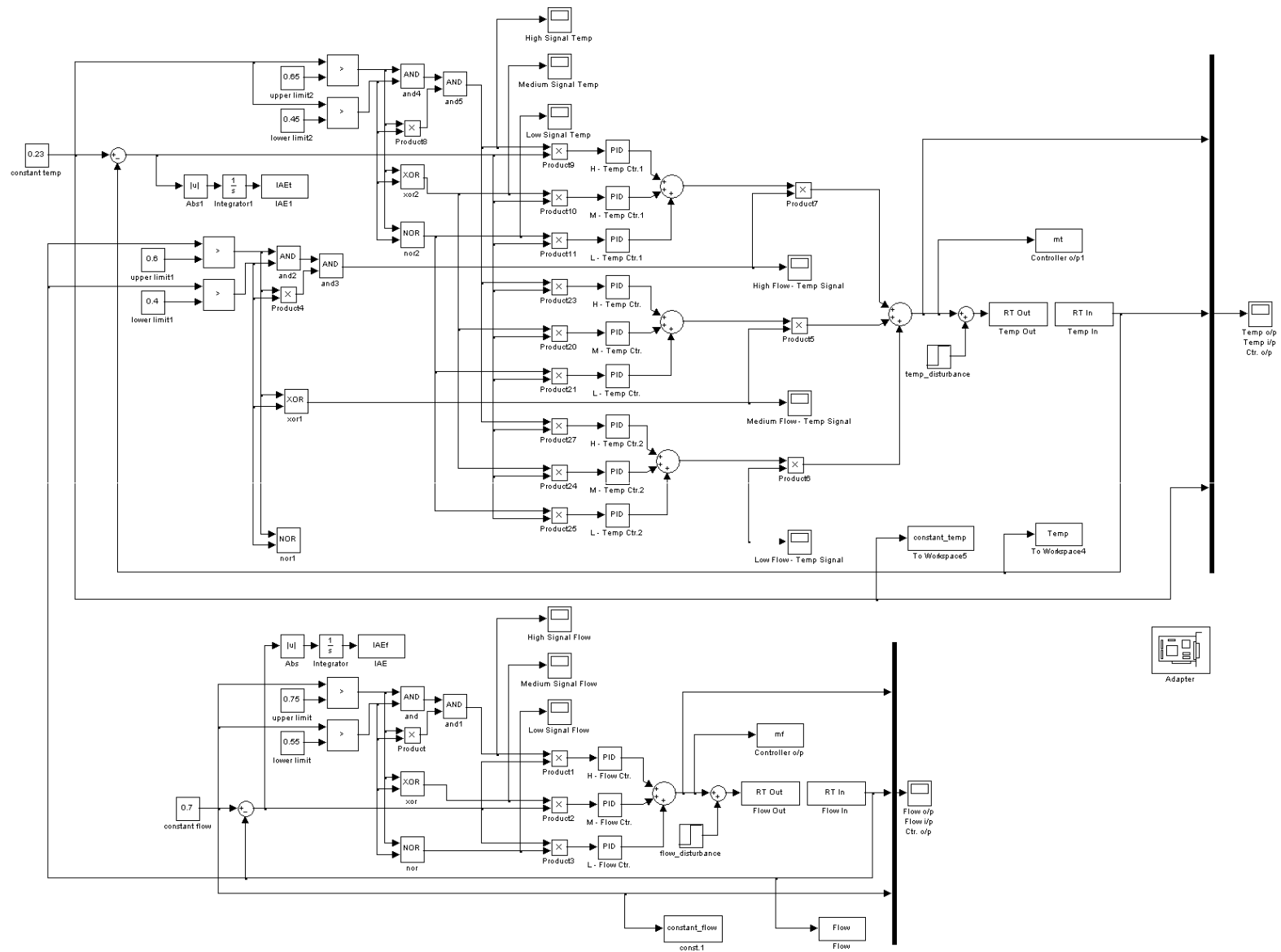
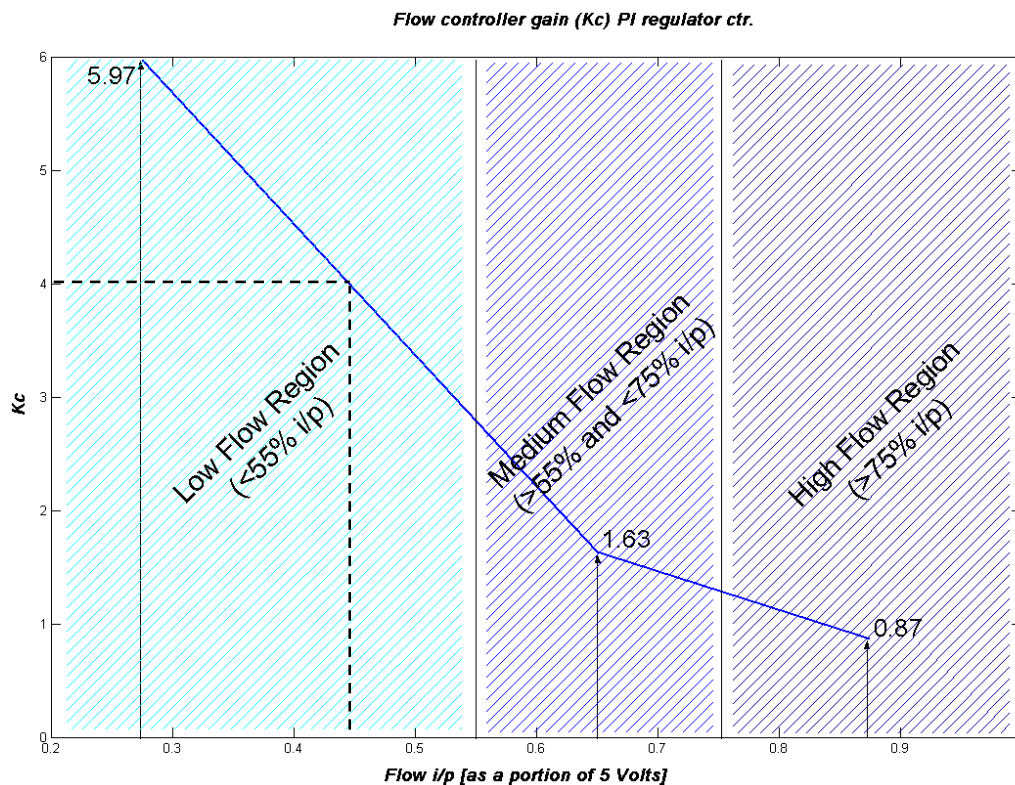


Figure 5-27 Gain scheduled PI temperature and flow control design – in regulator mode

## 5.5.2 Advanced gain scheduler design

### 5.5.2.1 Gain scheduling using look-up tables

It would be desirable to somehow interpolate between the controller settings for specific operating conditions as opposed to crudely ‘skipping’ between particular fixed settings. Figure 5-28 illustrates what this means.



**Figure 5-28** Plot of linear interpolation curve for flow controller gain (PI regulator mode)

The values; 5.97, 1.63, and 0.87 were taken from table 5-4 for the flow controller gain in PI regulator mode. They correspond to the median of their respective flow regions, i.e. midway of the low flow region is given by 27.5% flow input and the appropriate gain for this flow is given as 5.97 (from table 5-4). The blue line joins the three gain values determined for each flow region. Simulink has a *look-up table* function that easily allows one to pick values of gain off the plot for a given flow input, i.e. at 45% flow input the gain should be 4. In the previous gain scheduler design this flow condition would give a gain of 5.97 because it would only recognize that the flow is in the ‘low region’. This offers a more ‘smooth’ approach to adapting the controller settings as the process operating conditions vary.

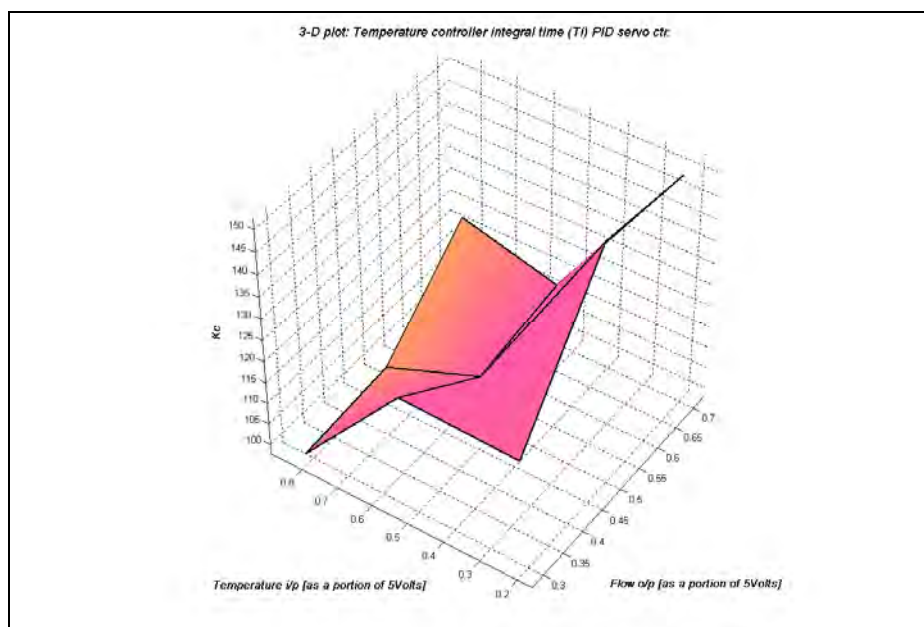
So a look up table would be specified for each controller setting, i.e.  $K_c$ ,  $T_i$ , and  $T_d$ . In the case of the flow process each look up table is 1-dimensional, i.e. it requires a single vector of input values (operating conditions) and their corresponding vector of outputs (gain, integral time, or derivative

time). This is because the flow controller settings only vary as the flow input varies. The temperature controller settings vary with the temperature input and the flow output. So the problem becomes a little more involved. The solution is to use a 2-dimensional look up table, i.e. two vectors of input values, or operating conditions (temperature input and flow output), and their corresponding matrix of output values (gain, integral time, or derivative time) are specified. Table 5-7 shows the matrix values for the temperature controller integral time over a range of input temperatures and output flows for a PID controller in servo mode (taken from table 5-5).

| $T_i$       |      | Temperature input |        |        |
|-------------|------|-------------------|--------|--------|
|             |      | 0.225             | 0.550  | 0.875  |
| Flow output | 0.30 | 122.27            | 152.49 | 149.59 |
|             | 0.50 | 122.98            | 108.64 | 112.36 |
|             | 0.70 | 97.65             | 99.63  | 118.20 |

**Table 5-7 Matrix of integral times for a PID servo temperature controller**

The two input vectors are given by the values of temperature input (0.225, 0.550, 0.875) and flow output (0.3, 0.5, 0.7). Using a 2 dimensional lookup table the corresponding values of integral time can be used to interpolate an integral time for any operating conditions in the range 0-1 of input temperature and output flow. Figure 5-29 shows a plot<sup>9</sup> of the matrix of integral times given in table 5-7. A look up table can pick values off this plot as the process operating conditions change.



**Figure 5-29 3-dimensional plot of linear interpolation of integral time for a PID servo temperature controller**

<sup>9</sup> Plots of all flow and temperature controller parameters can be found in Appendix L.



So look-up tables can be used to continuously give an accurate estimation of the most suitable controller settings for given operating conditions. They can also be incorporated into the de-coupler design discussed earlier. The output of the look-up table is a single value for controller gain, integral time or derivative time. This presents another problem, as the standard *PID* function block in Simulink does not allow for its settings to be updated in this way. Nevertheless, it was possible to design controllers in Simulink, corresponding to the given controller architectures, that can incorporate look-up tables.

5.5.2.2 Building PI/PID controllers with updateable settings in Simulink

Figure 5-30 shows a set-up that corresponds to the standard PI controller block in Simulink. The difference is that the controller settings  $K_c$  and  $T_i$  can be updated as the process operating conditions change using look-up tables.

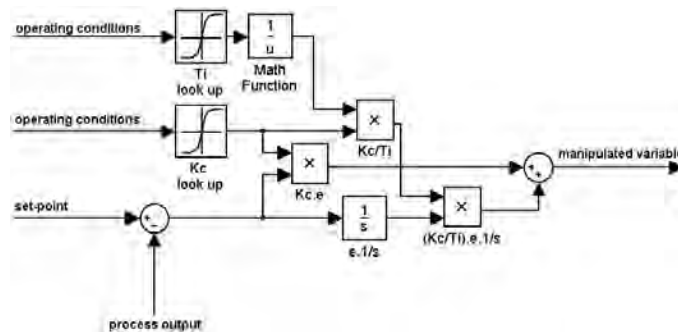


Figure 5-30 PI controller using look-up tables to update controller settings

The design is consistent with the Ideal PI controller architecture, i.e.  $G_c(s) = K_c \left( 1 + \frac{1}{T_i s} \right)$ .

The set-up was vindicated by comparing its performance to that of a PI controller (with  $K_c$  and  $T_i$  set to 1) in a simulation. The process was given as a simple first order lag (time constant of 1 second and gain of 1). Figure 5-31 shows the set-up and results of the simulation.

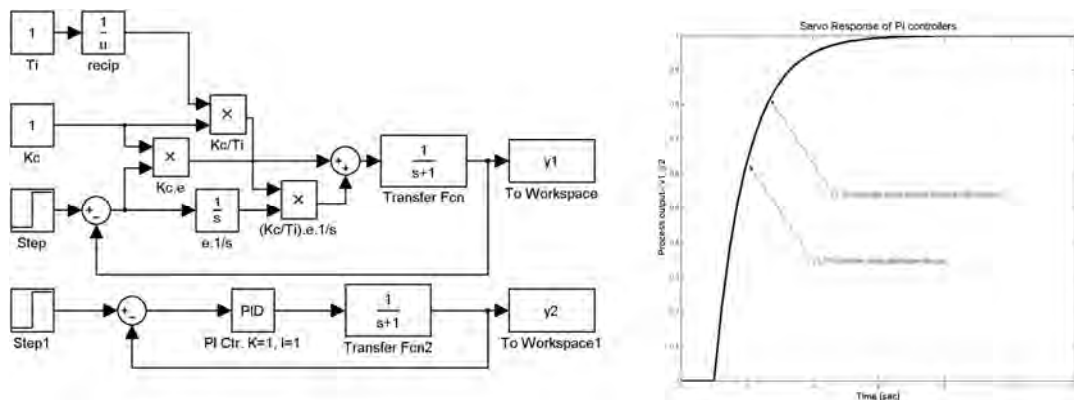


Figure 5-31 Simulation set-up and servo response of Ideal PI controller test

The design was extended to the Classical PID controller architecture, i.e.  $G_c(s) = K_c \left( 1 + \frac{1}{T_i s} \right) \frac{1 + T_d s}{1 + \frac{T_d}{N} s}$ .

Figure 5-32 shows the Simulink set-up. The design was tested in the same way as the Ideal PI controller, where the process was given by a simple first order lag with controller settings;  $K_c$ ,  $T_i$ , and  $T_d = 1$ . The simulation set-up and servo response are shown in figure 5-33. The value for  $N$  in the above equation was 10, as already discussed in section 5.2

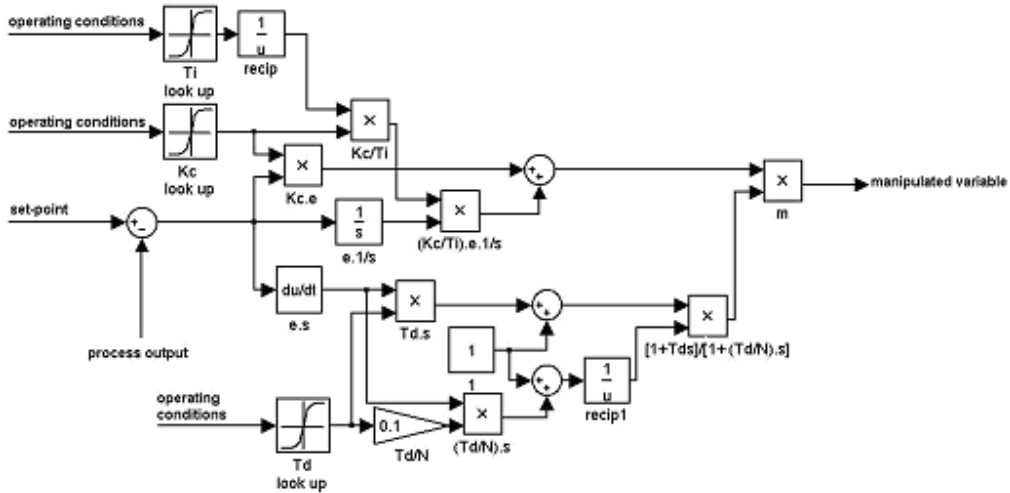


Figure 5-32 Classical PID controller using look-up tables to update controller settings

Unlike the Ideal PI controller response,  $y_1$  and  $y_2$  did not correspond exactly. The responses were quite different from each other. This is a problem associated with the derivative portion of the controller called *derivative kick*. This is when a sudden change in set-point (and hence  $e$ ) will cause the derivative term to become very large and thus provides a large initial kick to the final control element.

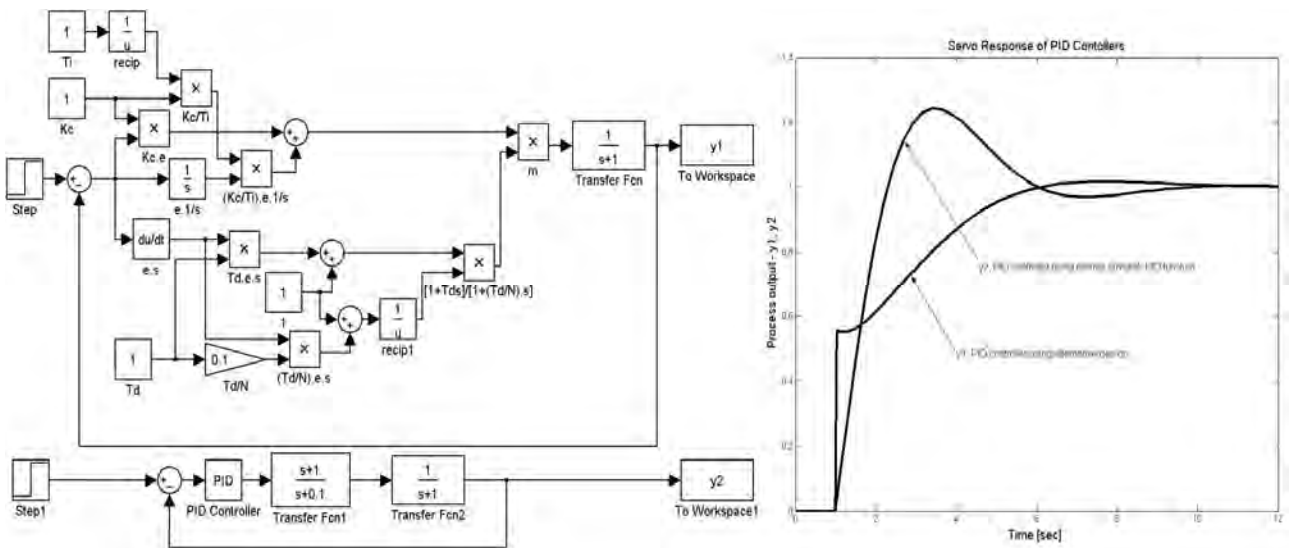


Figure 5-33 Simulation set-up and servo response of Classical PID controller test

This sudden jolt is undesirable and it was found that a compromise of both designs removed it, and produced an identical response compared to the normal Simulink PID set-up, see figure 5-34.

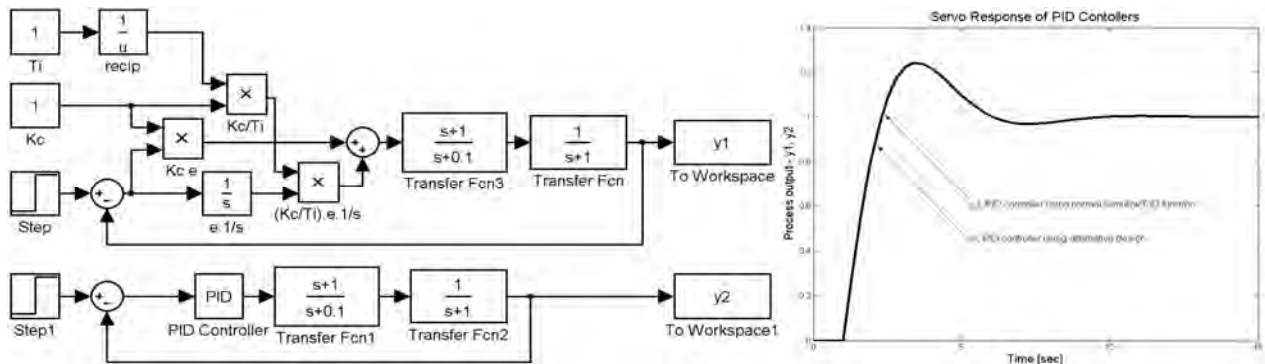


Figure 5-34 Simulation set-up and servo response of Classical PID controller test with compromised design

This design approach meant that it would not be possible to use look-up tables to estimate the derivative time of the controller. Nevertheless, the framework developed earlier for the simple gain scheduled controller can be adapted to produce the ‘D’ portion of the control signal. Further advocating this approach, it was found that the variance of the derivative time compared to that of the other controller parameters was quite small. For example, figures 5-35 and 5-36 clearly show that the variance of the derivative time is small in comparison to that of the integral time, for both flow and temperature controllers, respectively. Thus, using a  $T_d$  look-up table is less important.

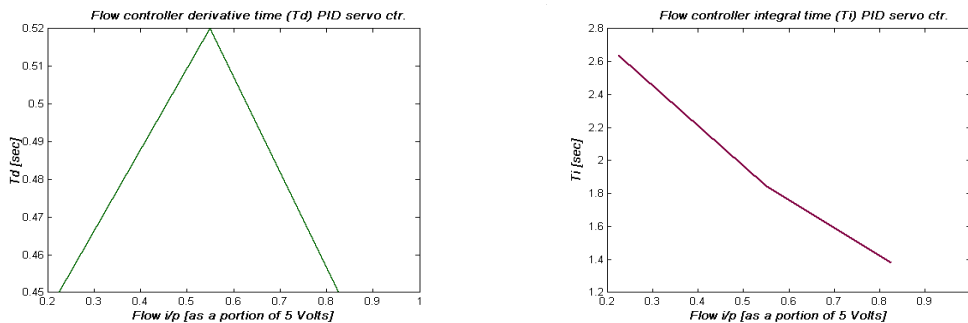


Figure 5-35 Comparison of flow controller (PID servo) derivative and integral times

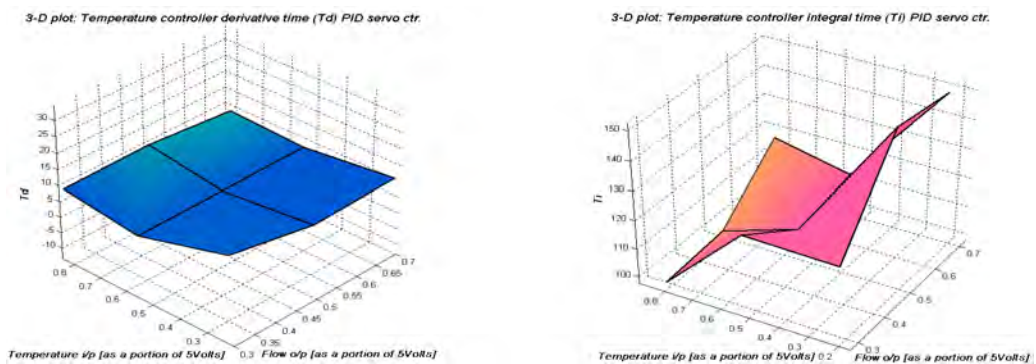


Figure 5-36 Comparison of temperature controller (PID servo) derivative and integral times

5.5.2.3 Overall system design

This section takes the ideas outlined in sections 5.4, 5.5.2.1, and 5.5.2.2 and merges them together to obtain the overall system design. Four Simulink files were developed that correspond to four different controller set-ups; PI regulator, PI servo, PID regulator, and PID servo controllers. The difference between the regulator and servo designs is slight. The set-point is constant in regulator mode as opposed to a step input in servo mode, and a load disturbance (step) exists at the process input in regulator mode, and no disturbance exists in servo mode. Look-up tables are used to linearly interpolate the controller gain ( $K_c$ ) and integral time ( $T_i$ ) in all cases. The values given in the look-up tables will of course vary depending on the type of controller and control mode. In the case of the PID controllers the value for derivative time ( $T_d$ ) is not linearly interpolated, but rather has a fixed value for a particular operating condition. Figure 5-37 shows the PI regulator set-up.

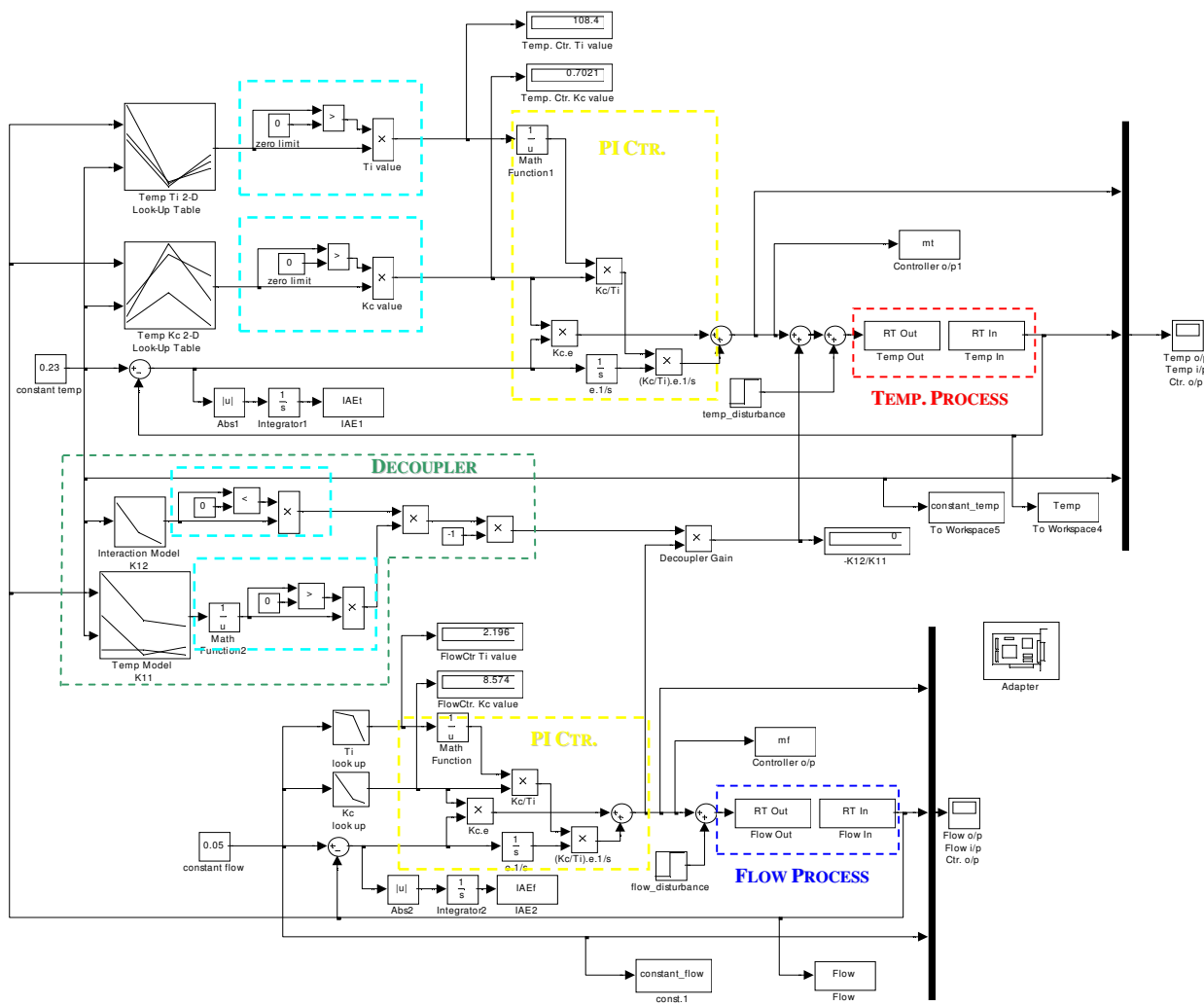


Figure 5-37 Advanced gain scheduler – PI regulator set-up

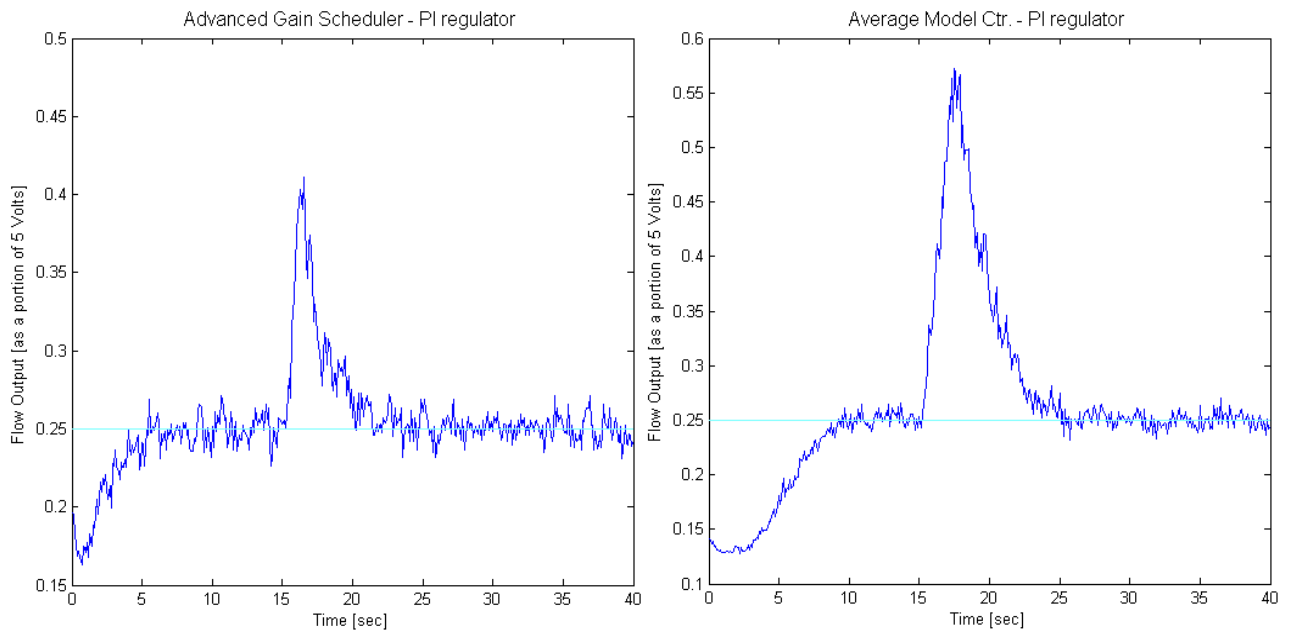
The temperature and flow processes are shown in red and blue dashed boxes respectively. The yellow boxes show the PI controllers for both processes. The green box shows the decoupler design. A 2-d look-up table delivers the temperature process model gain ( $K_{11}$ ) and a 1-d look-up table gives the process interaction gain ( $K_{12}$ ). The values used in these look-up tables were taken from appendix K. The light blue boxes indicate where the signal from a look-up table is limited above or below zero. The PID version of this design uses the same set-up with the 'D' portion of the signal put in series with the 'PI' signal, using the simple gain scheduler design seen already in section 5.5.1. Larger diagrams showing all four controllers are given in appendix M.

## 5.6 Final Design Validation Tests

Tests were carried out separately on both processes in regulator and servo mode. Before testing the appropriate 'test region' for each process had to be defined. The static tests detailed in chapter 4 help to give realistic test regions. For example the flow process was found to have a minimum possible flow rate of about 10% and a maximum possible flow rate of about 75%, see figure 4-3. So it would be sensible to only carry out tests in this region. The temperature process tests region had different limits depending on the flow rate. For example, the maximum possible temperature at a high flow rate of 70% output was 45°C, see figure 4-13. On the other hand, the ambient temperature of the surroundings, i.e. the laboratory room temperature, determined the minimum temperature possible. Testing could begin after the appropriate test regions were determined. The final controller design (advanced gain scheduler or AGS) was compared to an Ideal PI and Classical PID controller with fixed settings based on an average model (average model controller or AMC). The basis for performance comparison was the integral of absolute error (IAE). The large number of tests carried out did not allow for their inclusion in the Appendix. However, an accompanying CD contains all plots and simulation files.

### 5.6.1 Tests on the flow process

The flow process test region was given by flows between 10 and 75 per cent. In regulator tests, six values within these limits were chosen for the constant input. PI and PID controllers were tested using the AGS design and the AMC design. The test time was 40 seconds. The step disturbance was from 0 to 1 (i.e. 0 - 5 Volts) at a time of 15 seconds. The value for IAE was recorded at 15 and 40 seconds. The difference in these values being the cumulative error due to the disturbance only. A total of 24 regulator tests were carried out on the flow process. Figure 5-38 compares the results of two tests; the AGS and AMC PI regulator responses. It is clear that the AGS performance was better. All regulator test results are shown in table 5-8



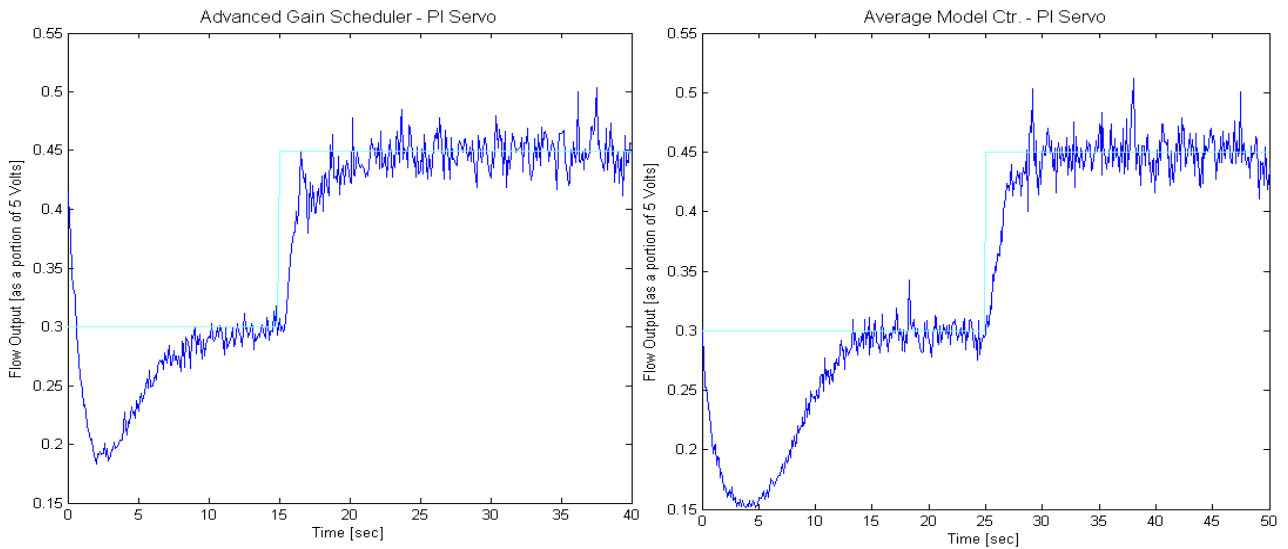
**Figure 5-38 Flow process - Advanced gain scheduler and average model controller PI regulator responses**

The values shown in the columns are the IAE calculated from the beginning of the process disturbance, i.e. from 15 seconds onwards. The mean, standard deviation, maximum, and minimum values for each controller are given in italics. It can be seen that overall the AGS out performs the AMC. The ‘percentage performance increase’ is calculated for the PI and PID controllers. This is calculated by expressing the decrease in mean IAE, when using the AGS, as a percentage of mean IAE, when using the AMC.

| <b>IAE</b>   |      | <b>Advanced gain scheduler</b> |              | <b>Average model controller</b> |              |
|--|------|--------------------------------|--------------|---------------------------------|--------------|
| <b>(Flow control - Regulator)</b>                            |      | <b>PI</b>                      | <b>PID</b>   | <b>PI</b>                       | <b>PID</b>   |
| <b>Constant flow input<br/>[as a portion of 5<br/>Volts]</b> | 0.15 | 0.441                          | 0.244        | 0.777                           | 1.172        |
|  | 0.25 | 0.342                          | 0.365        | 0.863                           | 1.201        |
|  | 0.35 | 0.613                          | 0.513        | 0.877                           | 1.233        |
|  | 0.45 | 0.781                          | 0.674        | 0.959                           | 1.277        |
|  | 0.55 | 1.016                          | 0.885        | 0.957                           | 1.330        |
|  | 0.65 | 1.437                          | 1.069        | 0.915                           | 1.265        |
| <i>Arithmetic mean</i>                                       |      | <i>0.772</i>                   | <i>0.625</i> | <i>0.891</i>                    | <i>1.246</i> |
| <i>Standard deviation</i>                                    |      | <i>0.405</i>                   | <i>0.314</i> | <i>0.069</i>                    | <i>0.057</i> |
| <i>Minimum</i>   |      | <i>0.342</i>                   | <i>0.244</i> | <i>0.777</i>                    | <i>1.172</i> |
| <i>Maximum</i>   |      | <i>1.437</i>                   | <i>1.069</i> | <i>0.959</i>                    | <i>1.330</i> |
| <b>% Performance increase</b>                                |      | <b>PI</b>                      | <b>13</b>    | <b>PID</b>                      | <b>50</b>    |

**Table 5-8 IAE results - flow regulator controllers**

The servo tests were carried out using six different step inputs within the given test region. Servo responses were obtained for increasing and decreasing step inputs. 48 servo tests were carried out in total. The step was applied 15 seconds after the beginning of the simulation. The simulation time was 40 seconds, as before with the regulator tests. Figure 5-39 compares the results of two tests; the AGS and AMC PI servo responses.



**Figure 5-39 Flow process - Advanced gain scheduler and average model controller PI servo responses**

All servo test results are shown in table 5-9. The ‘percentage performance increase’ indicates that the AMC performed slightly better than the AGS in servo mode. It should be noted, however, that the minimum IAE values for the AGS are mostly lower than the AMC.

| <b>IAE<br/>(Servo)</b>                           |            | <b>Advanced gain scheduler</b> |             |             |             | <b>Average model controller</b> |             |             |             |
|--|------------|--------------------------------|-------------|-------------|-------------|---------------------------------|-------------|-------------|-------------|
|  |            | <b>PI</b>                      |             | <b>PID</b>  |             | <b>PI</b>                       |             | <b>PID</b>  |             |
|  |            | <b>inc.</b>                    | <b>dec.</b> | <b>inc.</b> | <b>dec.</b> | <b>Inc.</b>                     | <b>dec.</b> | <b>inc.</b> | <b>dec.</b> |
| <b>Flow step input [as a portion of 5 Volts]</b> | 0.1 – 0.25 | 0.543                          | 0.470       | 0.651       | 0.458       | 1.270                           | 0.909       | 1.342       | 0.855       |
|  | 0.2 – 0.35 | 0.393                          | 0.299       | 0.581       | 0.416       | 0.597                           | 0.438       | 0.533       | 0.424       |
|  | 0.3 - 0.45 | 0.503                          | 0.400       | 0.650       | 0.495       | 0.491                           | 0.401       | 0.582       | 0.414       |
|  | 0.4 - 0.55 | 0.709                          | 0.635       | 0.745       | 0.725       | 0.505                           | 0.421       | 0.639       | 0.483       |
|  | 0.5 - 0.65 | 1.188                          | 1.148       | 0.264       | 1.198       | 0.579                           | 0.510       | 0.908       | 0.152       |
|  | 0.6 - 0.75 | 1.434                          | 1.254       | 1.536       | 1.222       | 0.875                           | 0.680       | 1.011       | 1.116       |
| <b>Arithmetic mean</b>                           |            | 0.795                          | 0.701       | 0.738       | 0.752       | 0.719                           | 0.560       | 0.836       | 0.574       |
| <b>Standard deviation</b>                        |            | 0.420                          | 0.404       | 0.425       | 0.370       | 0.303                           | 0.199       | 0.312       | 0.348       |
| <b>Minimum</b>                                   |            | 0.393                          | 0.299       | 0.264       | 0.416       | 0.491                           | 0.401       | 0.533       | 0.152       |
| <b>Maximum</b>                                   |            | 1.434                          | 1.254       | 1.536       | 1.222       | 1.270                           | 0.909       | 1.342       | 1.116       |
| <b>Percentage Performance Increase</b>           |            | <b>PI</b>                      |             | <b>-14</b>  |             | <b>PID</b>                      |             | <b>-5</b>   |             |

**Table 5-9 IAE results - flow servo controllers**

**5.6.2 Tests on the temperature process**

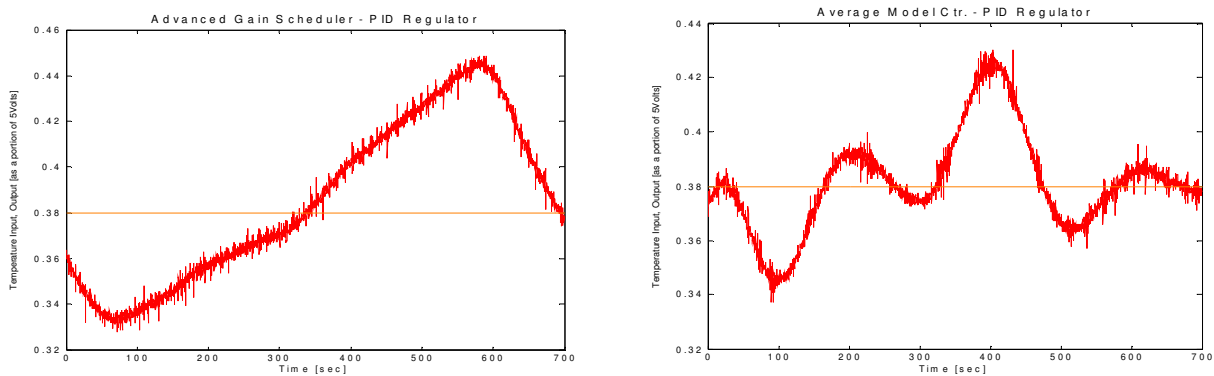
The temperature process test region was given by temperatures between 20°C and 45°C. In regulator tests, three values within these limits (26°C, 32°C, and 38°C) were chosen for the constant input, and at three different flow rates (20, 40, and 60%). These flow rates were deliberately chosen contrary to the flow rates at which the temperature controller look-up tables were calibrated (i.e. 30, 50, and 70%). PI and PID controllers were tested using the AGS design and the AMC design. The test time was 700 seconds. The step disturbance was from 0 to 1 (i.e. 0 - 5 Volts) at a time of 300 seconds. The value for IAE was recorded at 300 and 700 seconds so that only the IAE due to the disturbance was accounted for. A total of 36 regulator tests were carried out. All regulator test results are shown in table 5-10.

| <b>IAE</b><br><b>Temp - regulator</b>                 |      | <b>Advanced gain scheduler</b>  |           |           |             |            |              | <b>Flow o/p %</b> |
|---|------|---------------------------------|-----------|-----------|-------------|------------|--------------|-------------------|
|   |      | <b>PI</b>                       |           |           | <b>PID</b>  |            |              |                   |
| <b>Temp. constant input [as a portion of 5 Volts]</b> |      | <b>60</b>                       | <b>40</b> | <b>20</b> | <b>60</b>   | <b>40</b>  | <b>20</b>    |                   |
|   | 0.38 | 0.38                            | 4.350     | 4.722     | 11.622      | 7.885      | 9.975        |                   |
| 0.32  | 0.32 | 4.517                           | 5.129     | 13.577    | 8.879       | 6.362      | 13.118       |                   |
| 0.26  | 0.26 | 6.172                           | 7.244     | 16.822    | 4.023       | 2.420      | 3.971        |                   |
| <b>Mean</b>   |      | 5.013                           | 5.698     | 14.007    | 6.929       | 6.252      | 10.012       |                   |
| <b>Overall mean</b>                                   |      | 8.239                           |           |           | 7.731       |            |              |                   |
| <b>Standard deviation</b>                             |      | 1.007                           | 1.354     | 2.627     | 2.565       | 3.778      | 5.232        |                   |
| <b>Minimum</b>  |      | 4.350                           | 4.722     | 11.622    | 4.023       | 2.420      | 3.971        |                   |
| <b>Maximum</b>  |      | 6.172                           | 7.244     | 16.822    | 8.879       | 9.975      | 13.118       |                   |
| <b>IAE</b><br><b>Temp - regulator</b>                 |      | <b>Average model controller</b> |           |           |             |            |              | <b>Flow o/p %</b> |
|   |      | <b>PI</b>                       |           |           | <b>PID</b>  |            |              |                   |
| <b>Temp. constant input [as a portion of 5 Volts]</b> |      | <b>60</b>                       | <b>40</b> | <b>20</b> | <b>60</b>   | <b>40</b>  | <b>20</b>    |                   |
|   | 0.38 | 6.850                           | 6.281     | 6.638     | 4.405       | 4.835      | 0.436        |                   |
| 0.32  | 0.32 | 7.880                           | 6.705     | 7.947     | 6.082       | 6.391      | 9.231        |                   |
| 0.26  | 0.26 | 10.585                          | 9.914     | 9.308     | 3.316       | 8.504      | 8.415        |                   |
| <b>Mean</b>   |      | 8.438                           | 7.633     | 7.964     | 4.601       | 6.576      | 6.027        |                   |
| <b>Overall mean</b>                                   |      | 8.012                           |           |           | 5.735       |            |              |                   |
| <b>Standard deviation</b>                             |      | 1.929                           | 1.987     | 1.335     | 1.394       | 1.842      | 4.859        |                   |
| <b>Minimum</b>  |      | 6.850                           | 6.281     | 6.638     | 3.316       | 4.835      | 0.436        |                   |
| <b>Maximum</b>  |      | 10.585                          | 9.914     | 9.308     | 6.082       | 8.504      | 9.231        |                   |
| <b>Percentage performance increase</b>                |      |                                 |           | <b>PI</b> | <b>-2.8</b> | <b>PID</b> | <b>-25.8</b> |                   |

**Table 5-10 IAE results - temperature regulator controllers**



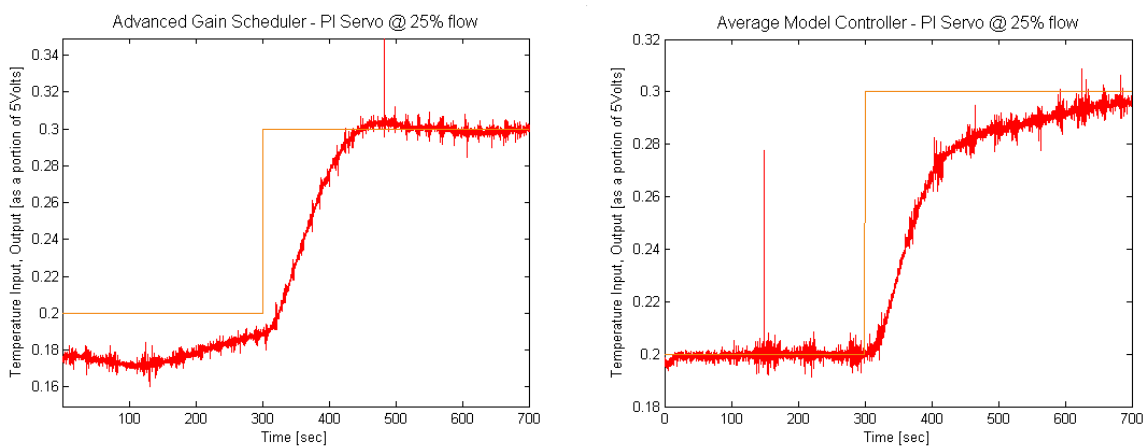
Figure 5-40 compares the results of two tests, the AGS and AMC PID temperature regulator responses.



**Figure 5-40 Temperature process - Advanced gain scheduler and average model controller PID regulator responses**

The results show that the AMC performs better in PID regulator mode than the AGS (percentage performance index of  $-25.8\%$ ). In PI regulator mode the difference in performance was very small (percentage performance index of  $-2.8\%$ ). Figure 5-40 shows that the maximum temperature reached in response due to the disturbance was larger for the AGS ( $\approx 0.445$ ) than the AMC ( $\approx 0.435$ ). It should also be noted that the AGS response did not settle as quickly as the AMC response before the disturbance (at 300 seconds).

In servo tests, three different step inputs (20-30%, 25-35%, and 35-45%) within the given test region were used. Increasing and decreasing step responses were obtained at two different flow rates (25% and 55%). PI and PID controllers were tested using the AGS design and the AMC design. The test time was 700 seconds. The step input occurred at a time of 300 seconds. The value for IAE was recorded at 300 and 700 seconds so that only the IAE due to the response only was accounted for. A total of 48 servo tests were carried out. All temperature servo test results are shown in table 5-11. Figure 5-41 compares the results of two tests; the AGS and the AMC PI servo responses.



**Figure 5-41 Temperature process - Advanced gain scheduler and average model controller PI servo responses**

| <b>IAE</b>                                |             | <b>Advanced gain scheduler</b>  |               |                      |               |                      |               |                      |               |
|---|-------------|---------------------------------|---------------|----------------------|---------------|----------------------|---------------|----------------------|---------------|
|   |             | <b>PI</b>                       |               |                      |               | <b>PID</b>           |               |                      |               |
|   |             | <b>25 % o/p flow</b>            |               | <b>55 % o/p flow</b> |               | <b>25 % o/p flow</b> |               | <b>55 % o/p flow</b> |               |
|   |             | <b>Inc.</b>                     | <b>dec.</b>   | <b>inc.</b>          | <b>dec.</b>   | <b>inc.</b>          | <b>dec.</b>   | <b>inc.</b>          | <b>dec.</b>   |
| Flow step input [as a portion of 5 Volts] | 0.2 - 0.3   | 8.0689                          | 12.196        | 8.56                 | 13.5763       | 15.107               | 14.506        | 9.12862              | 16.2343       |
|   | 0.25 - 0.35 | 10.2482                         | 9.788         | 12.4569              | 12.4149       | 24.3261              | 5.635         | 15.991               | 12.9473       |
|   | 0.35 - 0.45 | 19.277                          | 5.729         | 13.9867              | 6.032         | 16.885               | 2.837         | 22.0152              | 9.065         |
| <b>Mean</b>                               |             | <i>12.531</i>                   | <i>9.238</i>  | <i>11.668</i>        | <i>10.674</i> | <i>18.773</i>        | <i>7.659</i>  | <i>15.712</i>        | <i>12.749</i> |
| <b>Overall mean</b>                       |             | 11.028                          |               |                      |               | 13.723               |               |                      |               |
| <b>Std. deviation</b>                     |             | <i>5.943</i>                    | <i>3.268</i>  | <i>2.798</i>         | <i>4.062</i>  | <i>4.891</i>         | <i>6.092</i>  | <i>6.448</i>         | <i>3.589</i>  |
| <b>Minimum</b>                            |             | <i>8.069</i>                    | <i>5.729</i>  | <i>8.560</i>         | <i>6.032</i>  | <i>15.107</i>        | <i>2.837</i>  | <i>9.129</i>         | <i>9.065</i>  |
| <b>Maximum</b>                            |             | <i>19.277</i>                   | <i>12.196</i> | <i>13.987</i>        | <i>13.576</i> | <i>24.326</i>        | <i>14.506</i> | <i>22.015</i>        | <i>16.234</i> |
| <b>IAE</b>                                |             | <b>Average model controller</b> |               |                      |               |                      |               |                      |               |
|   |             | <b>PI</b>                       |               |                      |               | <b>PID</b>           |               |                      |               |
|   |             | <b>25 % o/p flow</b>            |               | <b>55 % o/p flow</b> |               | <b>25 % o/p flow</b> |               | <b>55 % o/p flow</b> |               |
|   |             | <b>Inc.</b>                     | <b>dec.</b>   | <b>inc.</b>          | <b>dec.</b>   | <b>inc.</b>          | <b>dec.</b>   | <b>inc.</b>          | <b>dec.</b>   |
| Flow step input [as a portion of 5 Volts] | 0.2 - 0.3   | 10.62203                        | 15.5972       | 12.1244              | 13.878        | 12.44                | 17.7068       | 10.85                | 16.7433       |
|   | 0.25 - 0.35 | 8.4886                          | 14.791        | 12.8317              | 9.904         | 8.5361               | 11.0811       | 12.6996              | 10.7317       |
|   | 0.35 - 0.45 | 14.784                          | 7.736         | 22.27                | 4.562         | 14.7692              | 8.297         | 18.292               | 5.267         |
| <b>Mean</b>                               |             | <i>11.298</i>                   | <i>12.708</i> | <i>15.742</i>        | <i>9.448</i>  | <i>11.915</i>        | <i>12.362</i> | <i>13.947</i>        | <i>10.914</i> |
| <b>Overall mean</b>                       |             | 12.299                          |               |                      |               | 12.284               |               |                      |               |
| <b>Std. deviation</b>                     |             | <i>3.202</i>                    | <i>4.325</i>  | <i>5.664</i>         | <i>4.675</i>  | <i>3.150</i>         | <i>4.834</i>  | <i>3.875</i>         | <i>5.740</i>  |
| <b>Minimum</b>                            |             | <i>8.489</i>                    | <i>7.736</i>  | <i>12.124</i>        | <i>4.562</i>  | <i>8.536</i>         | <i>8.297</i>  | <i>10.850</i>        | <i>5.267</i>  |
| <b>Maximum</b>                            |             | <i>14.784</i>                   | <i>15.597</i> | <i>22.270</i>        | <i>13.878</i> | <i>14.769</i>        | <i>17.707</i> | <i>18.292</i>        | <i>16.743</i> |
| <b>Percentage performance increase</b>    |             | <b>PI</b>                       |               | <b>10.3</b>          |               | <b>PID</b>           |               | <b>-11.7</b>         |               |

Table 5-11 IAE results - temperature servo controllers

### 5.6.3 Comments

Overall it is difficult to say whether the AGS is superior to the AMC. In some cases the AGS showed noticeably large improvements in performance over the AMC, in others both displayed similar performance, and occasionally the AMC proved to be preferable to the AGS. There are various reasons why this was the case. However, in all cases the processes were controlled successfully.

The results show, for regulator PI control of the flow process, that the AGS offers a 13% reduction in IAE over the AMC, and 50% reduction in IAE for PID control. However, the standard

deviation of IAE is noticeably lower for the AMC. The reason for this is that, while the overall IAE mean is lower for the AGS, the IAE tends to increase at higher rates of flow. For both PI and PID controllers the minimum IAE for the AGS is lower than that of the AMC, similarly the maximum IAE is higher. It is most likely that this is related to how efficient the gain scheduler is at varying the controller parameters as the process operating conditions vary. This in turn relates back to the quality of process identification. The AMC controller settings are fixed and are based on an overall process model, therefore modelling inaccuracies should be somewhat ‘ironed out’. On the other hand the non-linearity of the process should hinder the AMC’s performance. Figure 5-42 displays the IAE values obtained for the AGS and AMC in PI and PID regulator control.

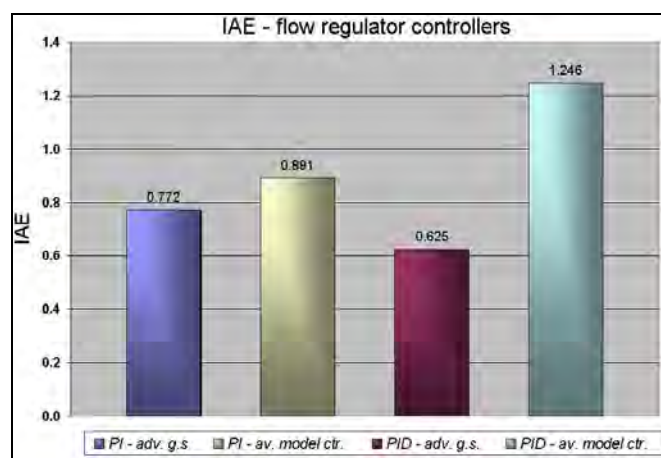


Figure 5-42 IAE – flow regulator controllers

For servo flow control, the AMC proved to be marginally better than the AGS, in both PI and PID mode. The IAE values are very similar in standard deviation and mean. The improvement in IAE or percentage performance increase, using the AMC, was small, i.e. -14% for PI and -5% for PID. Figure 5-43 displays the IAE values obtained for AGS and AMC in PI and PID servo control.

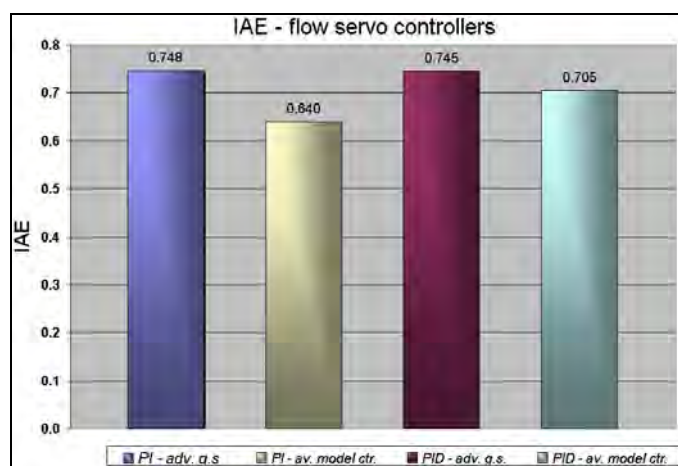


Figure 5-43 IAE – flow servo controllers

The tests on the temperature process produced somewhat inconsistent results. The AMC produced mean IAE values that were lower than that of the AGS, for PID control, in both regulator and servo modes. Conversely, the AGS gave better performance than the AMC, for PI control, in regulator mode and worse performance in servo mode. Figure 5-44 (IAE – temperature regulator controllers) and figure 5-45 (IAE – temperature servo controllers) illustrate this point. However, it should be noted that both controller designs managed to control the more troublesome temperature process.

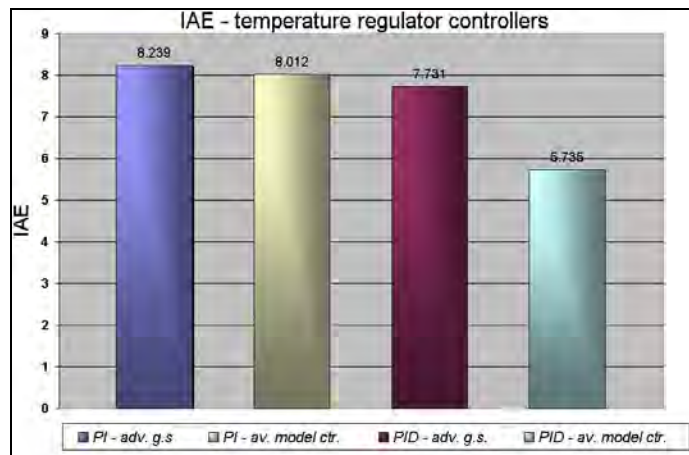


Figure 5-44 IAE – temperature regulator controllers

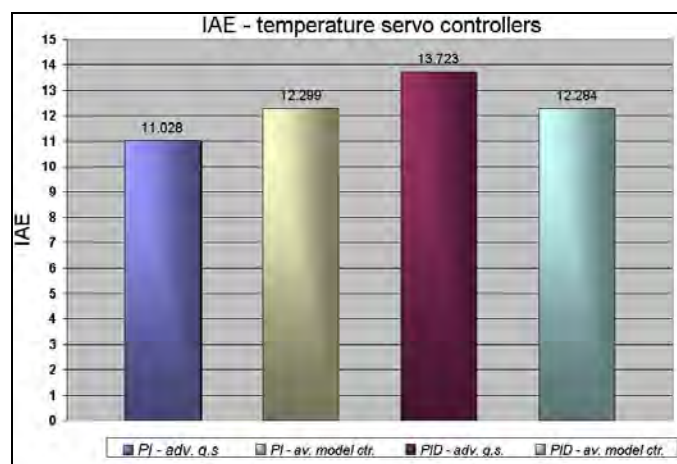


Figure 5-45 IAE – temperature servo controllers

It is possible that a wider range of tests would give a more comprehensive representation of the controllers' performance. The quality of the results could have been improved by extending the simulation time. For example, in the left hand plot of figure 5-40 it is clear that the regulator response of the AGS had not settled before the disturbance at 300 seconds, nor had it settled by the end of the simulation. Perhaps the results would have been more coherent if the simulation time was 1400 seconds and the step input or disturbance applied at 700 seconds. Unfortunately time restrictions did not allow for this to be possible.

## 6. Conclusion

### 6.1 Summary and Discussion of Results

The initial tests and experiments discussed in chapter 2 characterise, in outline, the temperature and flow processes. They also highlight the inadequacies of simple control techniques when used on these processes. These experiments pointed towards the possibility that the processes were non-linear. Chapters 3 and 4 characterise fully, the processes' behaviour through process identification and investigations into non-linearities, measurement accuracy, and interactions. Chapter 5 details the entire controller design procedure, problems encountered, implementation issues, and final design validation testing.

In Chapter 3 process models were determined over a range of operating conditions for both processes using a FOLPD model. Three flow process models were obtained corresponding to low, medium, and high flow ranges using step response identification. A frequency response model was determined for low-to-medium flows. Table 6-1 shows a summary of all flow process models obtained. All models describe the flow process for a fully opened load vane.

|  |  |  |   |
|--|--|--|---|
| <i>Step Response Models</i>                              | $G_{m_{LOW-FLOW}}(s) = \frac{0.45e^{-0.98s}}{1+2.70s}$     | $G_{m_{MED-FLOW}}(s) = \frac{1.08e^{-1.08s}}{1+1.93s}$ | $G_{m_{HIGH-FLOW}}(s) = \frac{1.76e^{-0.93s}}{1+1.45s}$ |
| <i>Average (low-to medium flow) Step Response Model:</i> | $G_{m_{LOW-MED-FLOW}}(s) = \frac{0.76e^{-1.03s}}{1+2.31s}$ |  |   |
| <i>Frequency Response Model:</i>                         | $G_{m_{LOW-MED-FLOW}}(s) = \frac{0.63e^{-1.66s}}{1+4.35s}$ |  |   |

**Table 6-1 All flow process model transfer functions obtained**

It can be seen that the process gain varies quite substantially at different flow rates. From the step test identification methods at low flows (<55% flow) the gain was 0.45, whereas at high flow rates (>75% flow) the gain was much larger at 1.76. This represents a change in gain of nearly four times its size over the full range of flow rates. The process time constant varied in a lesser manner; it became smaller at higher flows and larger at low flows. This suggested that the flow process in general performs faster at high flow rates. The time delay seemed to maintain some amount of consistency over the range in comparison to the other parameters, only varying by about 8% around an average of one second. It would be expected that the time delay would decrease as flow increased because of the distance-velocity relationship (i.e.  $\tau = \text{dist}/\text{velocity}$ ). That is assuming the flow is incompressible and

hence the velocity of the air and the flow rate vary linearly. This is broadly true, but because of the complexities involved in the transition from laminar to turbulent flow it is difficult to say for sure if the time delay should get smaller as the flow rate increases. It is possible that the identification technique somewhat skewed the results because of signal noise. The frequency response model (based on low-to-medium flow, i.e. <75% flow) yielded a time constant twice as large as that of the step response model (based on an average process model for low-to-medium flow). These comparisons highlight the challenges in developing reliable models. Frequency response modelling takes a great deal longer than step response modelling; furthermore, the margin for error is higher, as discussed in section 3.2.2. Moreover, the step response models are based on a wider range of flows and therefore are the basis for which controller settings were subsequently determined.

It was found that the temperature process characteristics depended on the conditions of the flow process. Therefore, models of the temperature process were determined over a range of temperatures (low, medium, and high) and at three different flow rates (i.e. nine models were obtained). The temperature process models obtained from chapter 3 are shown in table 6-2 below.

|                                   |   |   |  |
|-----------------------------------|---|---|--|
| Step Test<br>Models<br>(30% Flow) | $G_{m_{LOW-TEMP}}(s) = \frac{0.32e^{-32s}}{1+124s}$ | $G_{m_{MED-TEMP}}(s) = \frac{0.61e^{-22s}}{1+153s}$ | $G_{m_{HIGH-TEMP}}(s) = \frac{0.45e^{-24s}}{1+150s}$ |
| Step Test<br>Models<br>(50% Flow) | $G_{m_{LOW-TEMP}}(s) = \frac{0.32e^{-16s}}{1+123s}$ | $G_{m_{MED-TEMP}}(s) = \frac{0.43e^{-16s}}{1+109s}$ | $G_{m_{HIGH-TEMP}}(s) = \frac{0.30e^{-17s}}{1+113s}$ |
| Step Test<br>Models<br>(70% Flow) | $G_{m_{LOW-TEMP}}(s) = \frac{0.30e^{-22s}}{1+99s}$  | $G_{m_{MED-TEMP}}(s) = \frac{0.41e^{-23s}}{1+101s}$ | $G_{m_{HIGH-TEMP}}(s) = \frac{0.33e^{-19s}}{1+119s}$ |

**Table 6-2 All temperature process models obtained**

Three sets of tests were carried out, i.e. at 30%, 50%, and 70% flow. The process time constant remained relatively consistent over the range of temperatures for a particular flow rate. Apart from the tests at 50% flow, the time constant generally became larger at higher temperatures. The time constant tended to get smaller as the flow rate increased, hence decreasing the time required for the element to make a change in temperature. This is due to the thermodynamics of the heating element. High flow rates imply higher forced convection and therefore a higher heat transfer coefficient.

A common feature evident at all flow rates was that the process gain was larger at medium temperatures than at lower *and* higher temperatures. The static temperature process tests in chapter 4

endorsed this result. The input-output curve has its greatest slope, and therefore the greatest gain, at medium temperatures. The reason for this is that the heating element can only get so hot or so cool, i.e. the maximum temperature obtainable is limited by the maximum power output of the element, and the ambient temperature of the space being heated limits the minimum temperature obtainable.

As mentioned previously, a constant flow rate implies that the time delay should be constant. The time delay remained relatively constant for each flow rate apart from the 30% flow case. The noise in the step response plots may be attributable to the variation in time delay. Estimating the position of the tangent through the point of inflection on each plot was quite difficult for some plots, which affected the model time delay determined. Nevertheless, the resulting matrix of process model transfer functions shown makes sense from an intuitive point of view.

The results of chapter 3 complimented the findings of chapter 4. It was found that both processes were indeed non-linear in nature, i.e. the process model parameters varied under different operating conditions. It was possible to obtain a static characteristic (input-versus-output) curve for each process. This was done by inputting a suitably designed staircase function to the process, in open loop, recording the response, and subsequently plotting the input staircase against the output response. The flow process curve (figure 4-3) tells us that the process is non-linear and that limits exist on its maximum and minimum operating region. At flows less than 15% very little change in output occurs for a change in input. This is effectively a dead-band region of the flow. The maximum flow rate obtainable is 75%. Figure 4-3 compliments the work detailed in chapter 3, i.e. the slope of the characteristic curve is greater at high inputs therefore implying high process gain at high inputs.

A procedure was carried out to measure the flow measurement accuracy of the rig. The flow rate as measured by the rig was compared to the actual flow rate as measured by a Pitot tube. It was found that the orifice plate (the rig's flow measurement device) had a co-efficient of discharge ( $C_d$ ) of approximately 0.55, i.e. the rig continuously overestimates the actual flow rate by a factor of approximately two ( $1/C_d$ ). Thus, it seems likely that the flow rate information available to the closed loop flow control system contains significant inaccuracies.

Static testing was carried out on the temperature process in a similar way to the flow process. The temperature process has an infinite number of characteristic curves. This is because the process behaviour depends on the infinite number of possible flow rates. Characteristic curves at three flow rates were determined (figure 4-13). It is clear that the temperature process is non-linear. The higher the flow rate the lower the maximum temperature achievable. This is sensible as the cooling effect of the airflow would be greater at high flow rates. At high temperature inputs, each curve tended to level

off or saturate. This is because the heating element reaches its maximum power output at 5Volts, or 100% input. Each curve has a lower limit consistent with, and because of, the ambient room temperature (0.23 approximately equivalent to 23°C). A simple test that compares the displayed temperature sensor reading to the actual temperature, as given by a mercury thermometer, revealed the rig consistently underestimates actual temperature by approximately 3°C, see figure 4-11.

To examine the possibility of process interactions between the temperature and flow processes, two simple tests were carried out. In each case the input to both processes were held constant and allowed to settle. Then one of the process inputs would undergo a step change. The output of the other process was observed to see if this change had any effect on it. The tests confirmed that an interaction exists between the output temperature and the input flow. The interaction was modelled using a range of step response tests, and three models were obtained corresponding to three temperature ranges, as shown in table 6-3

|                           |   |   |   |
|---------------------------|---|---|---|
| <i>Interaction Models</i> | $G_{12-LowTemp}(s) = \frac{-0.02e^{-16s}}{1 + 63s}$ | $G_{12-Med.Temp}(s) = \frac{-0.18e^{-8s}}{1 + 85s}$ | $G_{12-HighTemp}(s) = \frac{-0.24e^{-7s}}{1 + 70s}$ |
|---------------------------|---|---|---|

**Table 6-3 Interaction models obtained**

As expected, it was found that the model gain was negative, i.e. increasing flow leading to decreased temperature. The effect of the interaction was more extreme at high temperatures. For example, the model gain was over ten times higher at 70% temperature input (gain=-2.4) than at 30% (gain=-0.2). This established that a control strategy to eliminate or minimise process interactions would have to be found.

In chapter 5, the Ideal PI and Classical PID controllers were chosen to control the processes because of the relatively low time delay to time constant ratio of both processes, and also because of their wide use in industry and relatively simple implementation. Suitable tuning rules were chosen for these controllers that were based on minimising IAE. It was noted that continuous time controllers would be implemented in the discrete time environment without adjusting the settings. This was because of the relatively small sample time used. Preliminary tests were carried out using the selected controllers and tuning rules. It was found that, while noise in the feedback signal could be a problem (see figures 5-2 and 5-4), the flow process controllers produced satisfactory performance, in regulator and servo mode.

The temperature process proved more difficult to control. It seemed that a disturbance to the process did not have any effect on the output (see figure 5-6). It was later found that this was because



the test was carried out near the temperature limit imposed by the flow rate at which the test was run. The servo tests displayed an inverse like response, the cause for which was unknown. It was established that the observed behaviour could not be a true inverse response, but rather due to the way in which the simulation package operates, and/or due to the interacting effect of the flow process. The problem was only evident at the beginning of simulations, hence leading to this conclusion. It was proposed that a decoupler be designed and tested before any inverse response compensation methods were considered. The appropriately designed decoupler made a noticeable reduction in the so-called inverse response effect, and clear improvement in the servo and regulator performance of the process. These tests are thoroughly detailed and discussed in sections 5.3.2, 5.3.3, and 5.4.2. An investigation into inverse response methods yielded two possible solutions; recalculate controller settings with the extra delay due to the inverse response added to the process model, or use a particular inverse response compensator design. The improvement in performance after decoupling meant that these methods would not be implemented.

A simple gain scheduler design was developed for the flow process that provided a way of skipping between controllers with fixed settings particular to given operating conditions. The design was extended to the temperature process also. This method was not tested. Nevertheless, it provided a suitable infrastructure for implementing a more involved design. This design incorporates look-up tables to continuously interpolate controller settings as process operating conditions vary. Three-dimensional plots of the variance in temperature controller parameters were obtained that illustrate the interpolation method. It was found that the integral time and gain varied more than the derivative time. Two-dimensional plots of flow controller parameter variance showed similar results. An alternative method of implementing an Ideal PI controller was found that can utilise a controller setting signal from a look up table. Extending this design to a Classical PID controller proved to be more difficult because of the effect of a pure derivative on the error signal, leading to derivative kick. A compromise of the simple gain scheduler design that interpolates for gain and integral time, but skips between specific 'D' implementations, was chosen as the best way forward. The static decoupler was integrated into the design, the gain for which was determined in a similar manner to the PI controller by means of look-up tables. The final design was named an 'advanced gain scheduler' (AGS).

Tests were carried out using the AGS separately on both processes in regulator and servo mode. The appropriate 'test region' for each process was determined from the results of the static tests from chapter 4. The AGS was compared to an Ideal PI and Classical PID controller with fixed settings based on an average model (average model controller or AMC). A range of tests were carried out

comparing both controllers in regulator and servo mode, using PI and PID set-ups. The basis for performance comparison was the integral of absolute error (IAE). The results show that in some cases the AGS outperformed the AMC. For regulator PI control of the flow process, the AGS offers a 13% reduction in IAE over the AMC, and 50% reduction in IAE for PID control. In other cases the AMC was clearly preferable to the AGS. In servo control of the flow process, the AMC gave a 14% reduction in IAE over the AGS, in PI control, and 5% reduction in PID control. The temperature AGS controller was inferior to the AMC in regulator PI and PID control. However, the difference in IAE was smaller in the case of PI control (2.5% decrease in IAE) than in PID control (25.8% decrease in IAE). Contrary to this, in servo mode the PID AGS temperature controller offered a 10.3% improvement in IAE over the AMC, but an 11.7% increase in IAE when in PI control. Therefore it is difficult to say, overall, if the AGS is superior to the AMC. It is obvious that, for any non-linear system, the gain scheduling control strategy should provide enhanced performance over an average model controller.

It is very possible that the method of testing was flawed. For example, in many of the servo and regulator responses, for the temperature process, the output had not fully settled by the end of the simulation. Longer tests would have solved this problem, but unfortunately project time restrictions did not allow for this, especially with the large number of tests that had to be carried out. This does not answer the question about why the AGS servo responses, of the flow process, offered no performance increase. However, the quality of process identification would have an effect on the performance of the AGS design. It may be the case that discrete time identification would yield more consistent results.

Nonetheless, it should be noted that initially neither the flow nor temperature processes had been adequately controlled using the local controllers and simple control techniques. Whereas, both the AMC and the AGS controller designs managed to adequately control the flow process, and the particularly troublesome temperature process. In general, the performance of the AMC was consistently good for both processes, and in some case the AGS displayed performance that was even better.

## **6.2 Further Work and Recommendations**

It would be possible to apply the advanced gain scheduler design to other 2x2 MIMO processes with relative ease. For example, a suitable MIMO process would be a liquid level-temperature process. The design would require adequate process identification information to be effective, and obviously the more information obtained, the more effective the design.

If the process had more than two inputs and outputs the design could be adapted to suit. However, it should be kept in mind that the amount of process information required would increase substantially. In general, assuming all processes interact with each other, an  $n \times n$  MIMO process with  $y$  number of operating regions per process would require  $y^n$  models for implementation of this design. For example, a  $3 \times 3$  process would require 27 models — nine models per process with three operating ranges for each process, i.e. low, medium, and high. Similarly, considering again a  $2 \times 2$  MIMO process with five operating regions (for example very low, low, medium, high, and very high) the number of models required would be 25. Thus, the simple process identification methods used in this project would no longer be feasible, as they would be too time consuming.

The solution to the forgoing would be to use discrete time identification methods, such as correlation testing and the least squares algorithm. The least squares algorithm and its variants can be implemented in Matlab using the ‘system identification toolbox’. So, the need for physical modelling would be eliminated and large amounts of information could be dealt with at once. It is very likely that discrete time identification would yield enhanced results in the AGS performance. However, this task is not always simple to carry out, as the quality of the process feedback signal would be an issue when noise is present, and setting up the appropriate programs/algorithms can take some time.

The design is similarly applicable to other continuous non-linear processes that are not necessarily part of a MIMO processes, such as the control of pH levels in wastewater processes or flow rate in pneumatic flow control valves. The quality of performance will again depend on the accuracy and extent of the process identification.

In terms of the existing process and controller design, some small but interesting possibilities remain. For reasons already given, more detailed identification would be sensible. Further experimentation with Matlab could yield a solution to the problem of derivative kick when implementing the Classical PID controller with look-up tables. Considering the level of noise in the feedback signal of both processes this would be a fruitful activity. More detailed identification of the process interaction could subsequently improve the decoupler’s effectiveness. For example; tests could be carried out at different ranges of flow rate step change, and the interacting effect on the temperature measured at three different temperatures, so that a two dimensional look-up table is used to estimate the interaction gain ( $K_{12}$ ), as opposed to a one dimensional table. Further testing where both process inputs are varied may reveal additional advantages of the decoupling design, apart from minimising the so-called inverse response of the temperature process. It would be interesting to investigate the effect of introducing disturbances in the output signal of both processes. This could be done by

partially closing the load vane on the flow process or by varying the ambient room temperature. The possibility of replacing/modifying the existing flow measurement device should be considered.

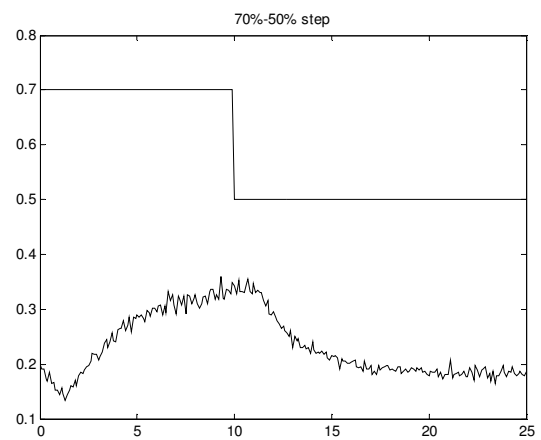
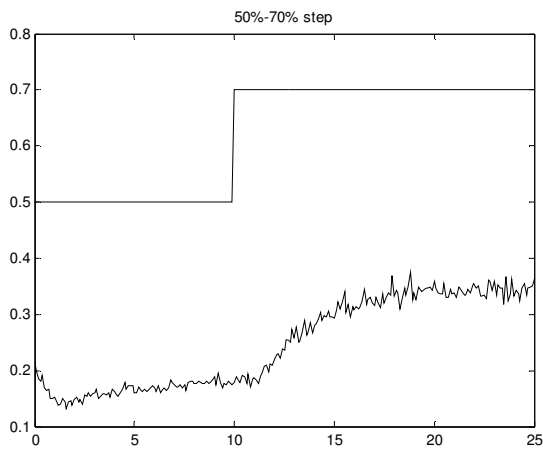
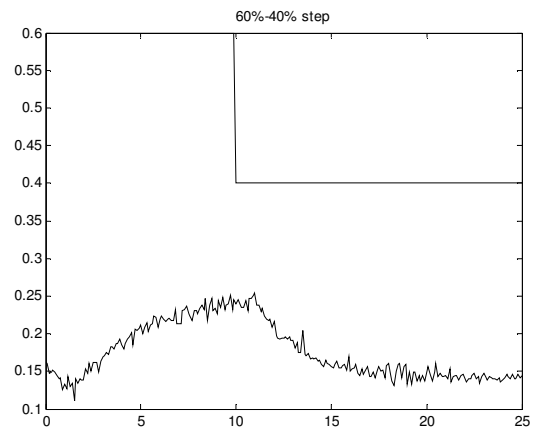
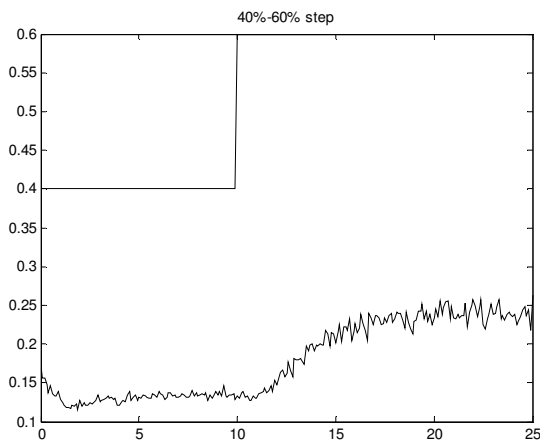
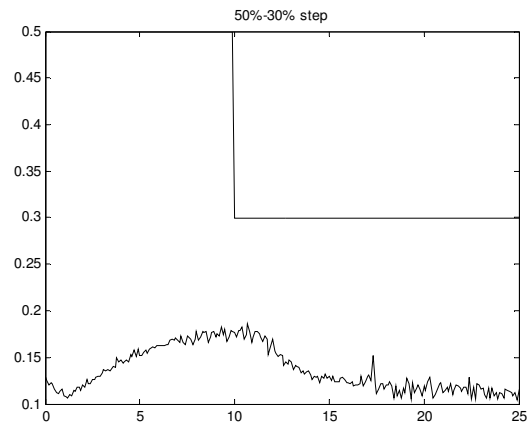
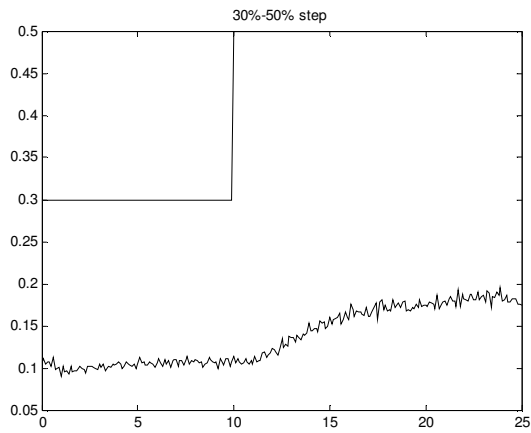
In a broader context, there is a possibility that the controller design could be applied to a reduced scale model of a building with heating and ventilation. The work carried out in this project provides much of the information/design approaches to attempt this. The control problem becomes more complex because of the shape of the building interior. Turbulent flow conditions can arise providing a particularly taxing situation for the controller. The problem crosses over into the mechanical/building services engineering area and would be a worthwhile exercise for someone of this background. In relation to this, it would be possible to concurrently carry out a Finite Element Analysis (FEA) of the fluid flow and heat transfer around the building. This would provide a means of designing the building interior *and* the control strategy with a view to optimising the heating and ventilation of the building.

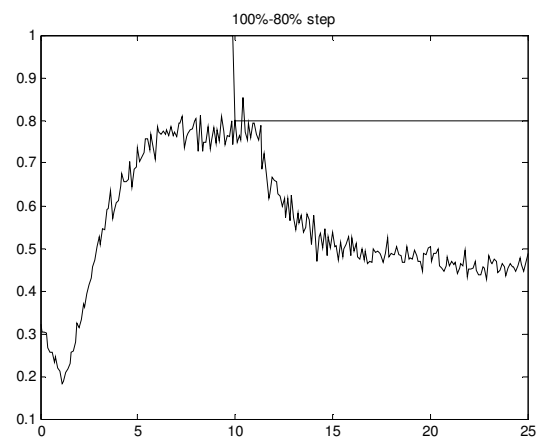
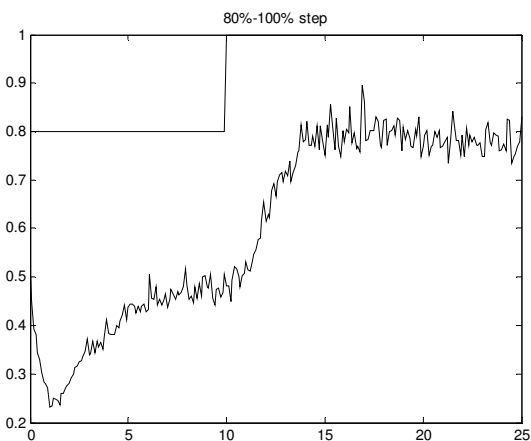
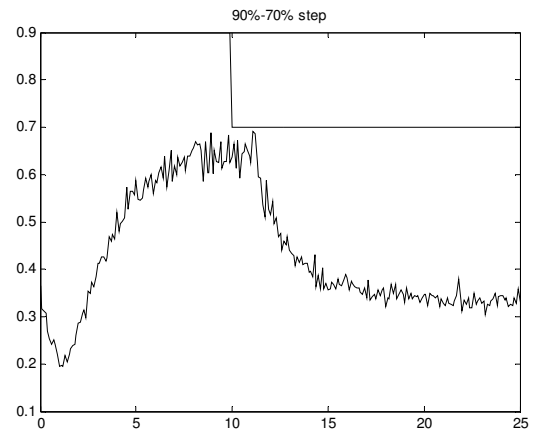
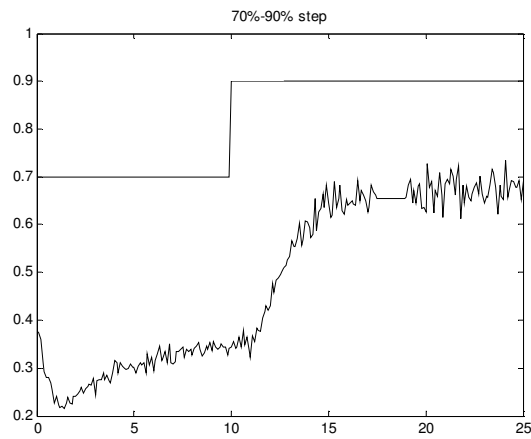
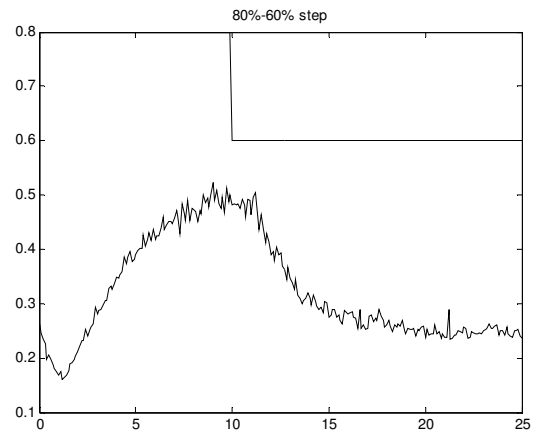
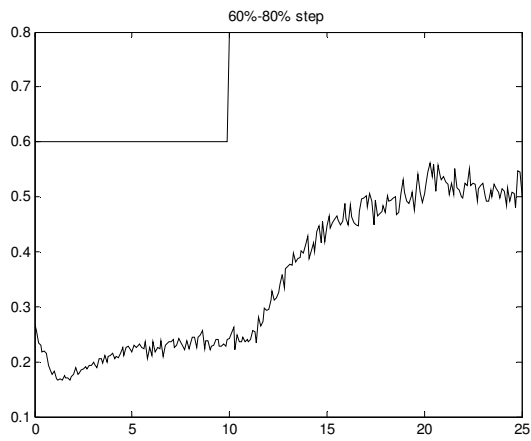
## 7. Appendix

|  |            |
|--|------------|
| <b>APPENDIX A – FLOW PROCESS OPEN LOOP STEP TEST PLOTS</b>                   | <b>92</b>  |
| <b>APPENDIX B – FLOW PROCESS FREQUENCY RESPONSE PLOTS</b>                    | <b>94</b>  |
| <b>APPENDIX C – FLOW PROCESS FREQUENCY BODE PLOT – POSSIBLE MODEL</b>        | <b>97</b>  |
| <b>APPENDIX D – TEMPERATURE PROCESS OPEN LOOP STEP TEST PLOTS (30% FLOW)</b> | <b>98</b>  |
| <b>APPENDIX E – TEMPERATURE PROCESS OPEN LOOP STEP TEST PLOTS (50% FLOW)</b> | <b>102</b> |
| <b>APPENDIX F – TEMPERATURE PROCESS OPEN LOOP STEP TEST PLOTS (70% FLOW)</b> | <b>106</b> |
| <b>APPENDIX G – PROCESS INTERACTION STEP TEST PLOTS</b>                      | <b>110</b> |
| <b>APPENDIX H – PROJECT TIMETABLE</b>  | <b>111</b> |
| <b>APPENDIX I – SCHEMATIC OF RIG</b>   | <b>112</b> |
| <b>APPENDIX J – MATLAB/SIMULINK CONTROLLER SETTINGS</b>                      | <b>113</b> |
| <b>APPENDIX K – RELATIVE GAIN ARRAY METHOD</b>                               | <b>114</b> |
| <b>APPENDIX L – PLOTS OF ADVANCED GAIN SCHEDULER CONTROLLER SETTINGS</b>     | <b>118</b> |
| <b>APPENDIX M - ALL ADVANCED GAIN SCHEDULER CONTROLLERS</b>                  | <b>122</b> |

**Appendix A – Flow Process Open Loop Step Test Plots**

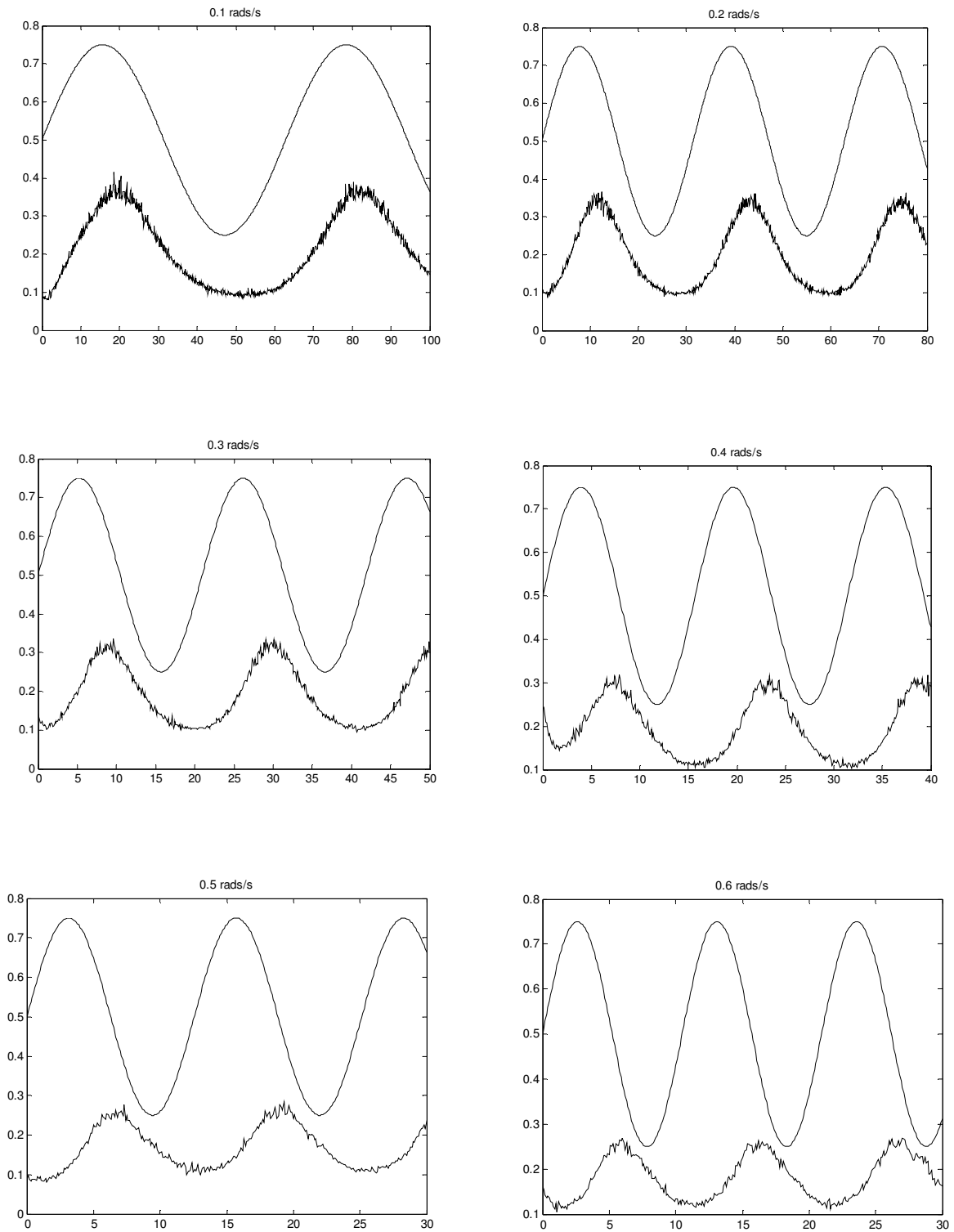
*[For all plots: the x-axis measures time and the y-axis measures process input and output as a portion of 5Volts]*



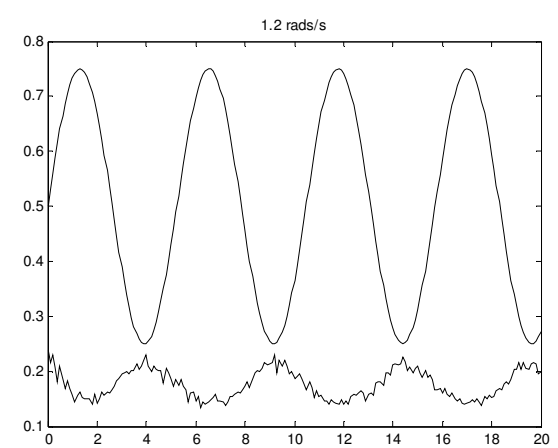
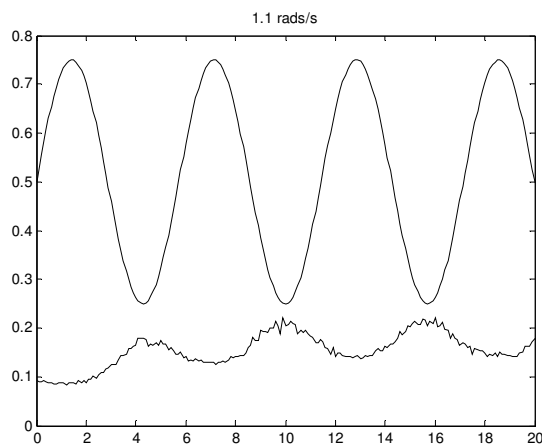
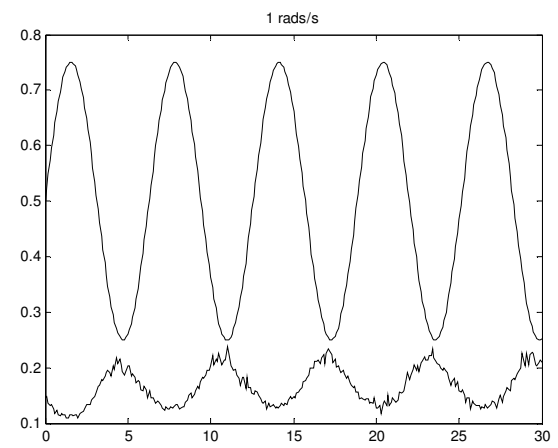
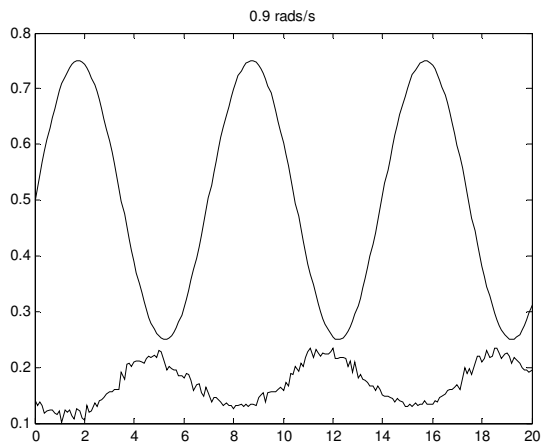
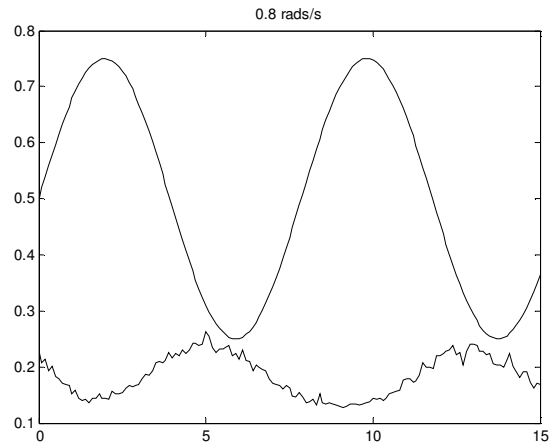
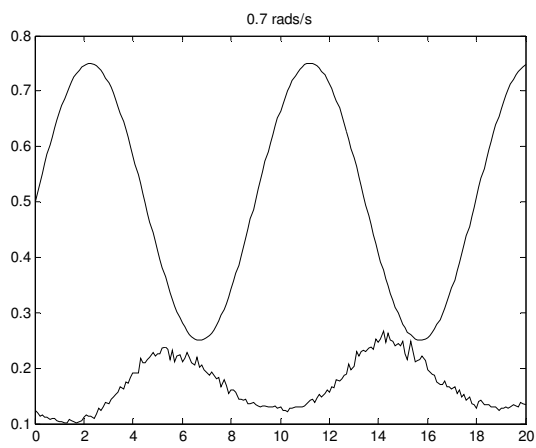


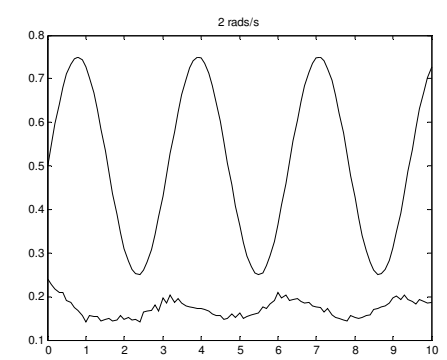
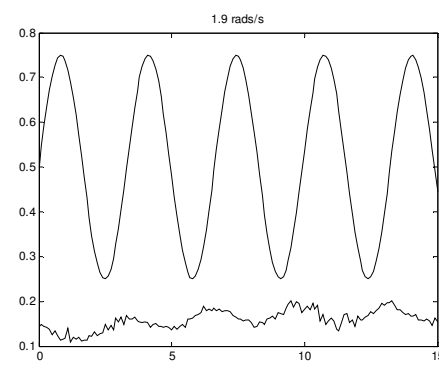
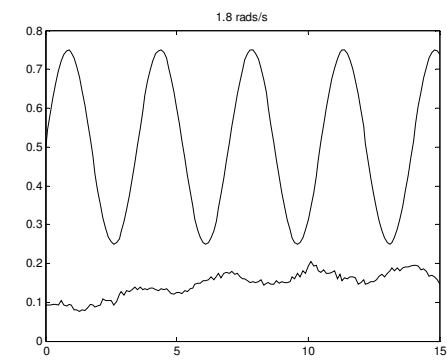
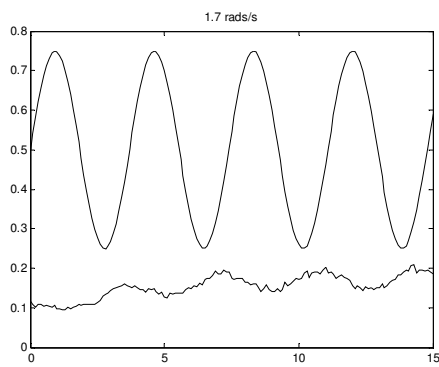
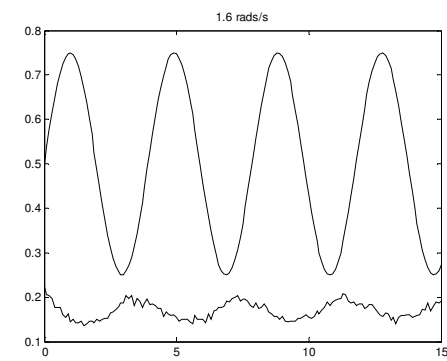
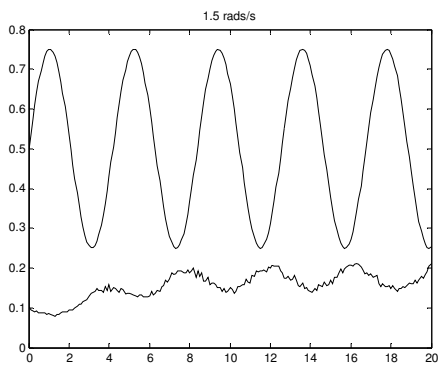
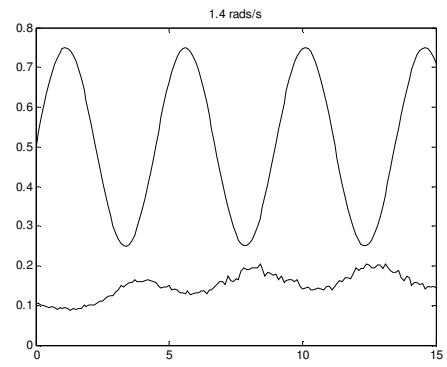
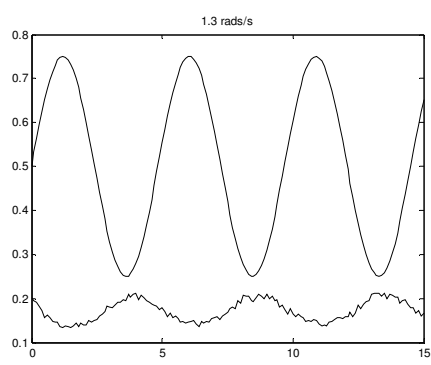
**Appendix B – Flow Process Frequency Response Plots**

*[For all plots: the x-axis measures time and the y-axis measures process input and output as a portion of 5Volts]*

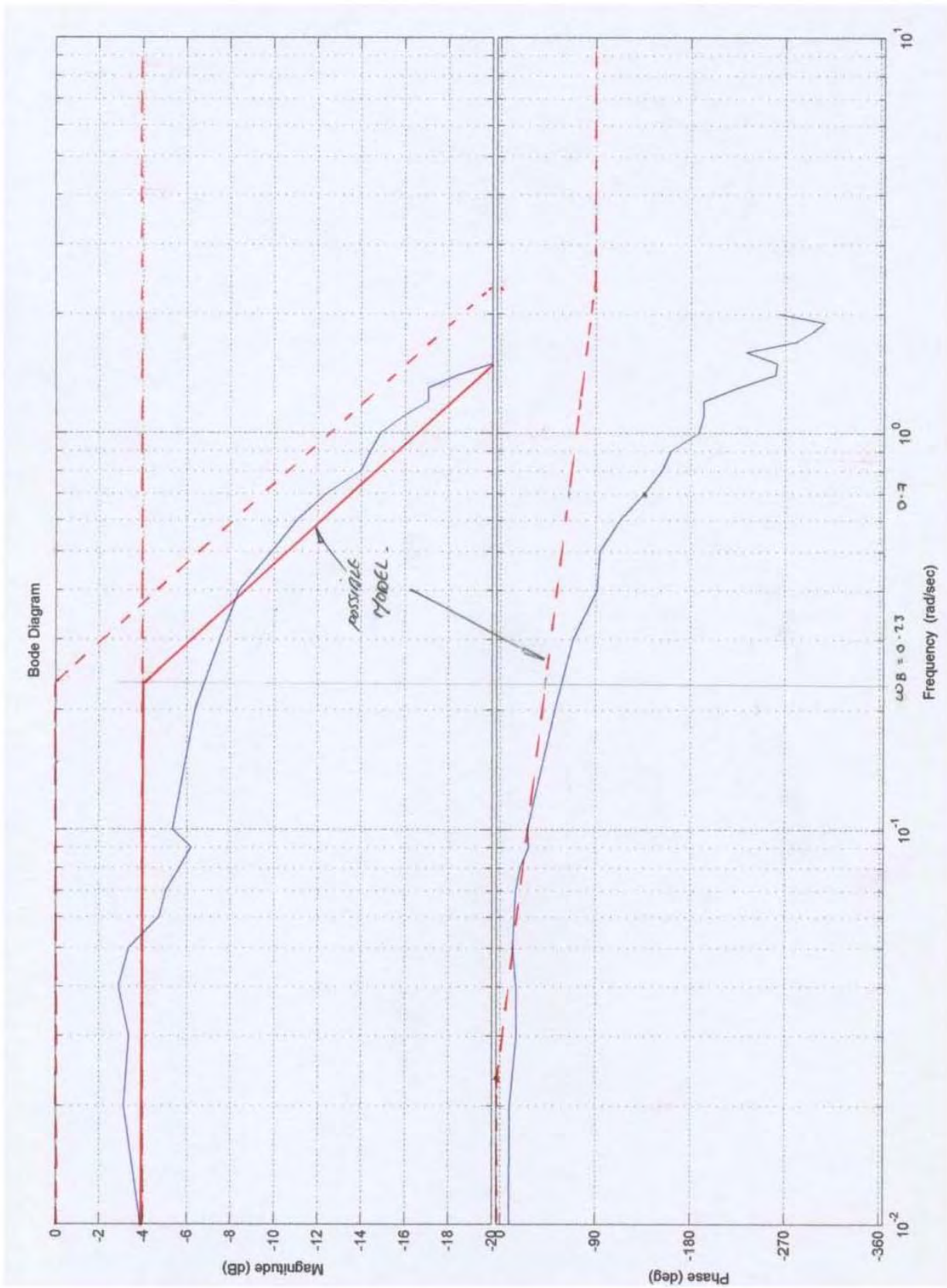






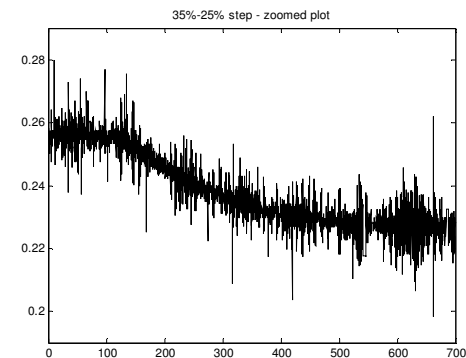
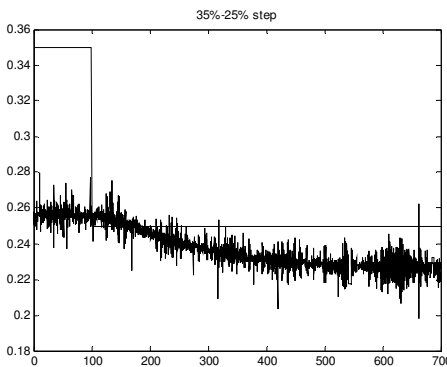
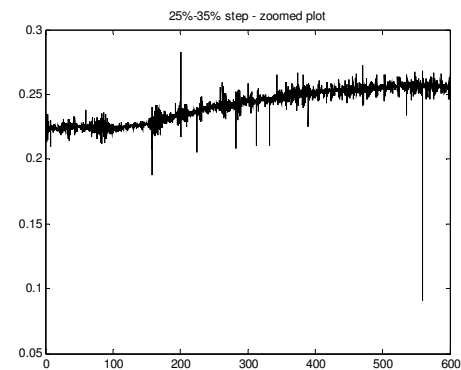
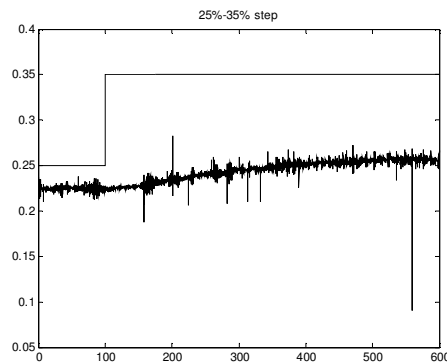
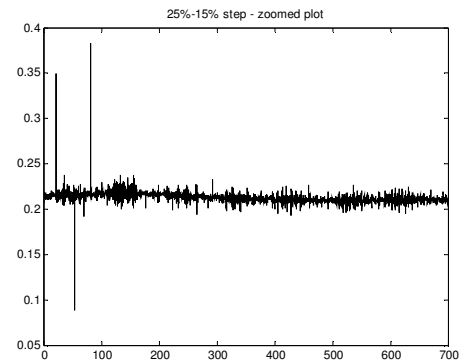
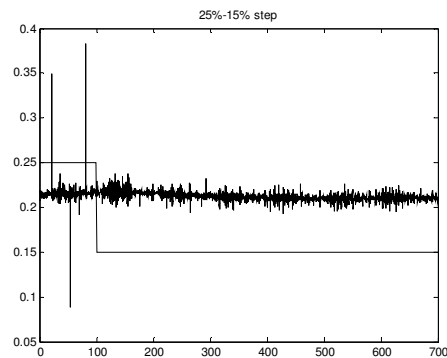
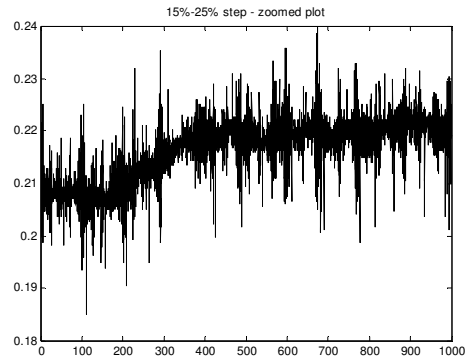
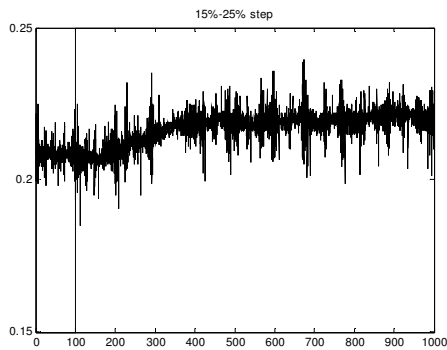


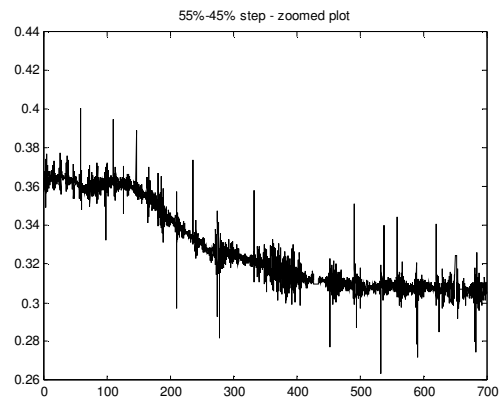
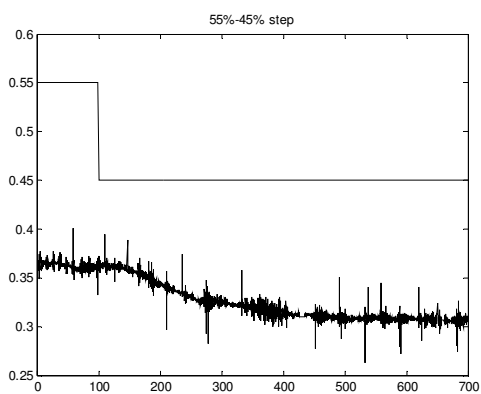
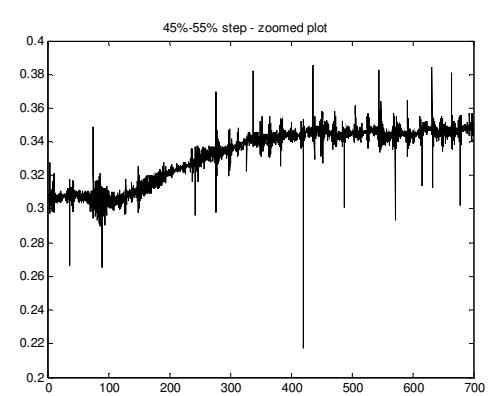
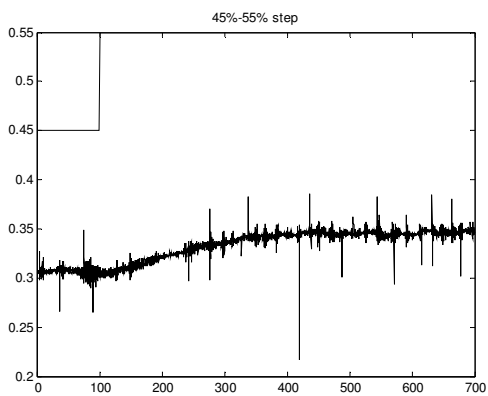
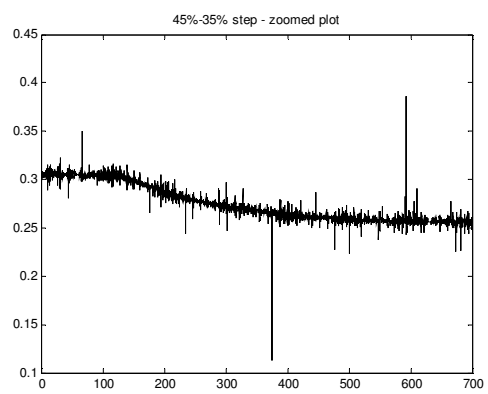
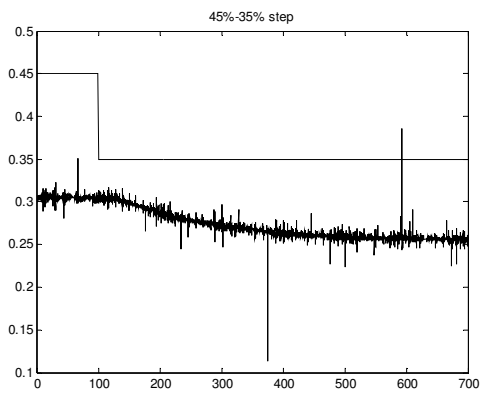
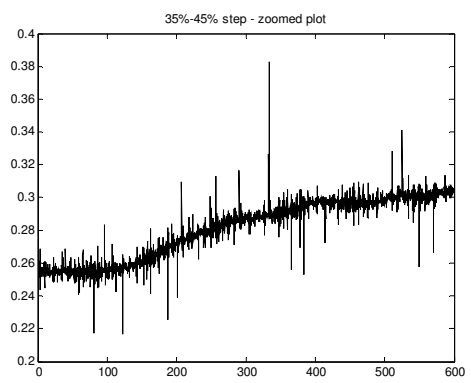
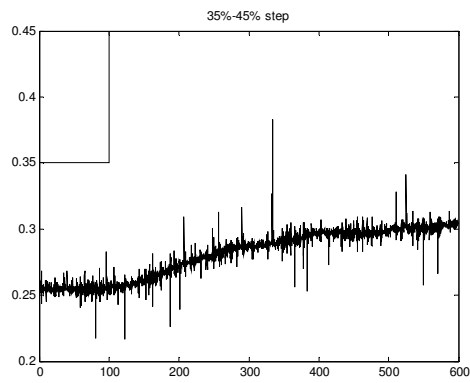
**Appendix C – Flow Process Frequency Bode Plot – Possible Model**

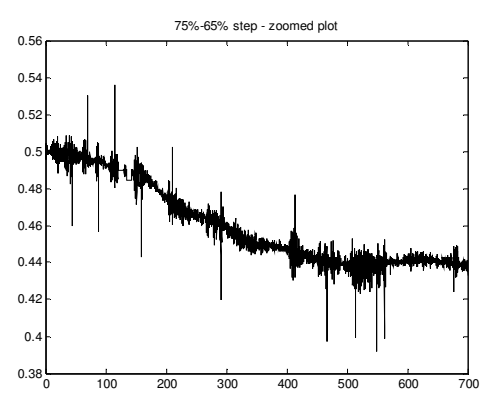
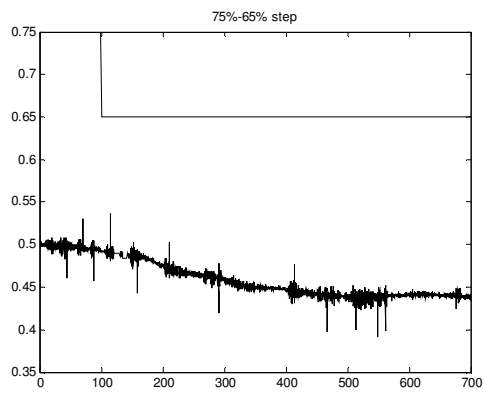
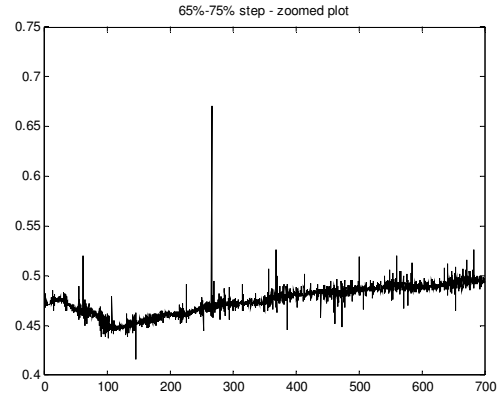
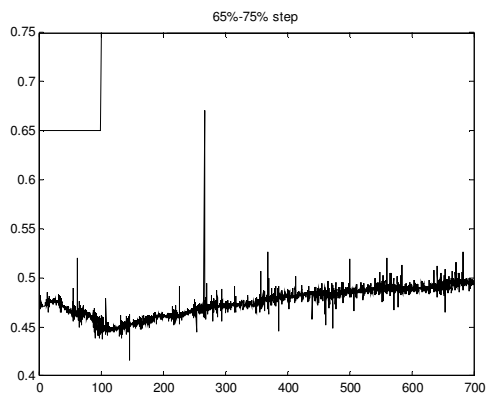
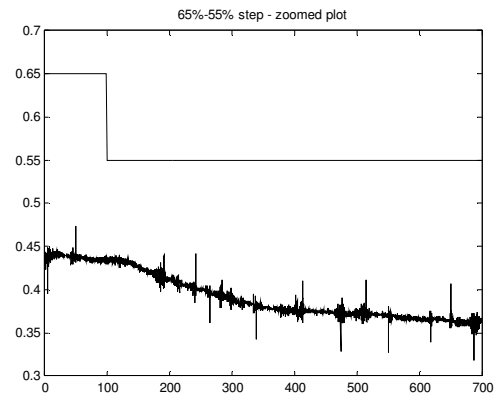
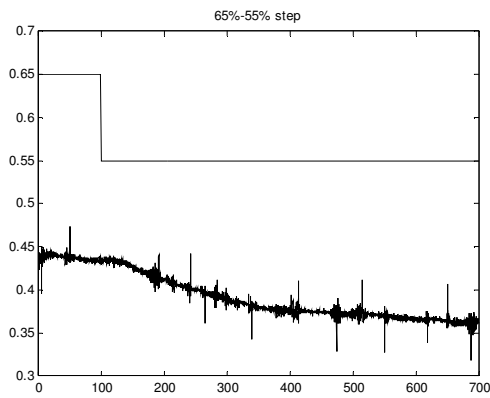
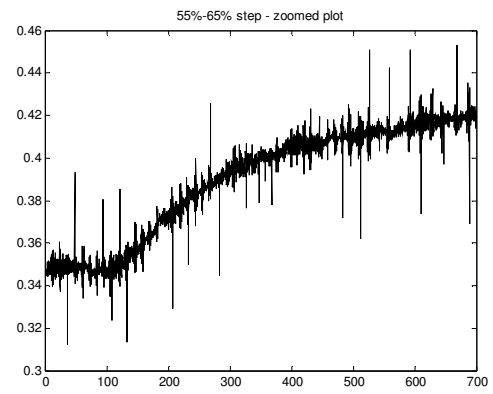
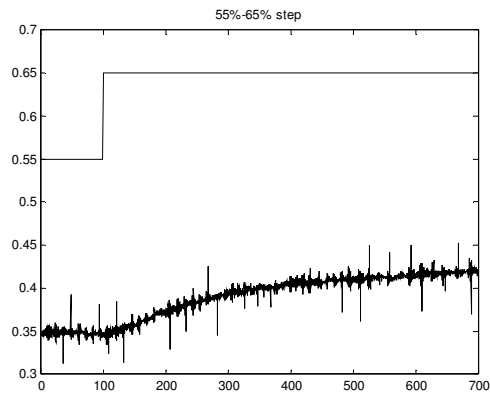


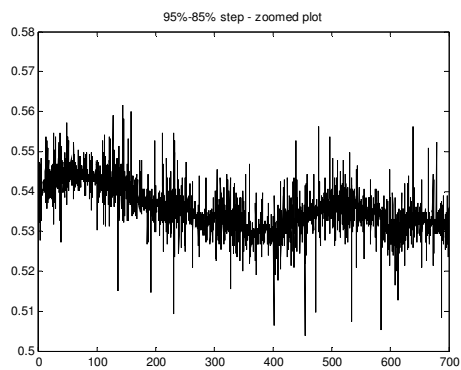
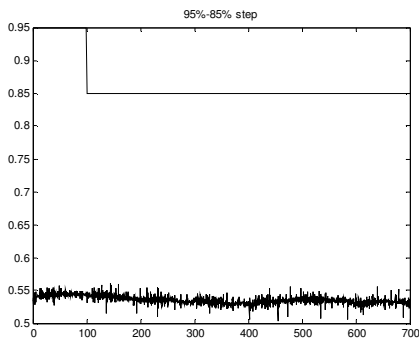
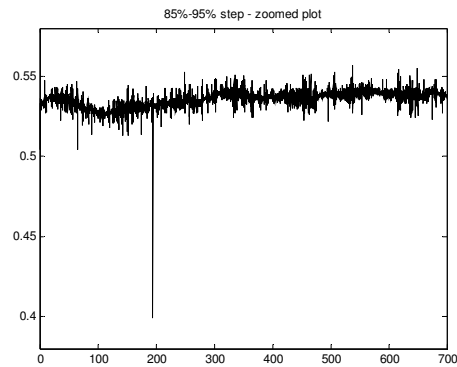
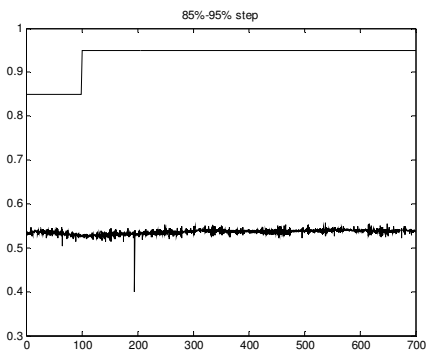
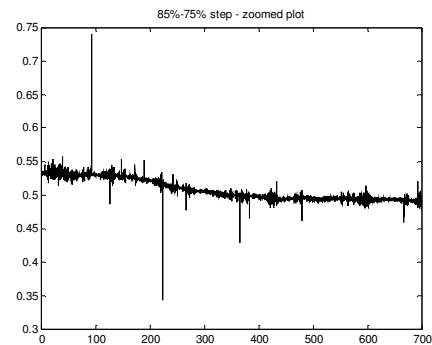
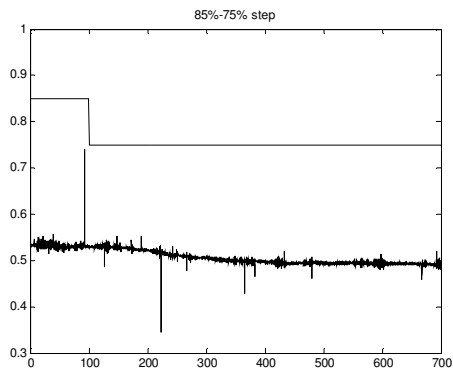
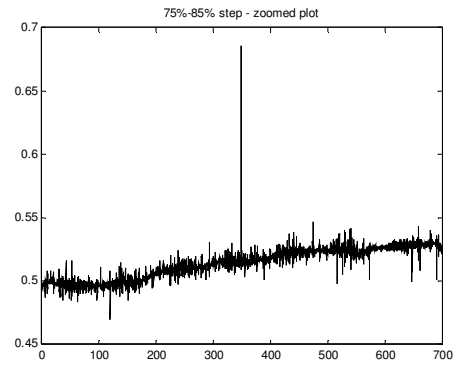
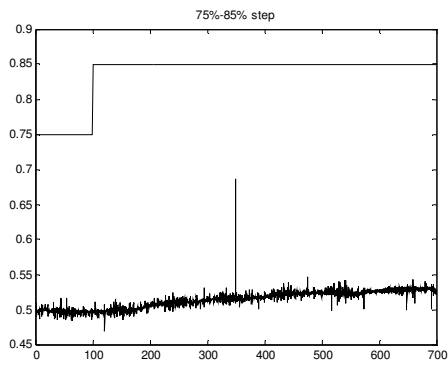
**Appendix D – Temperature Process Open Loop Step Test Plots (30% flow)**

*[For all plots: the x-axis measures time and the y-axis measures process input and output as a portion of 5Volts]*



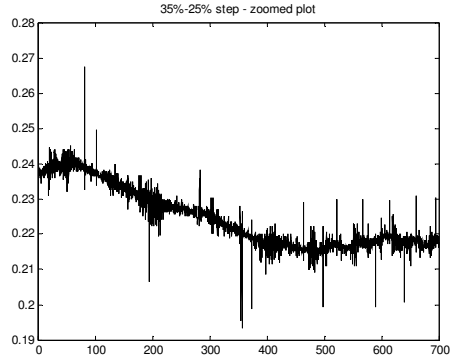
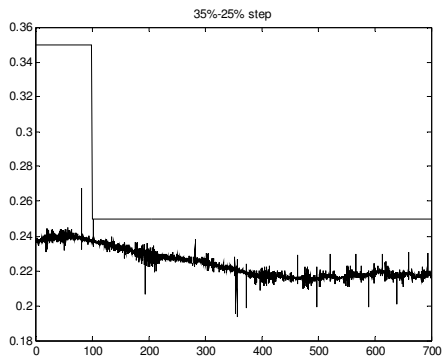
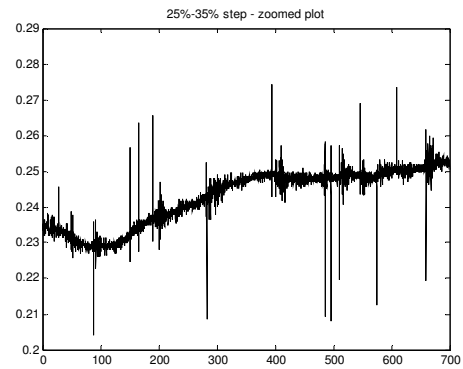
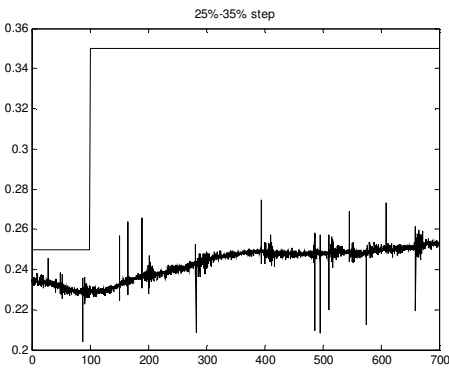
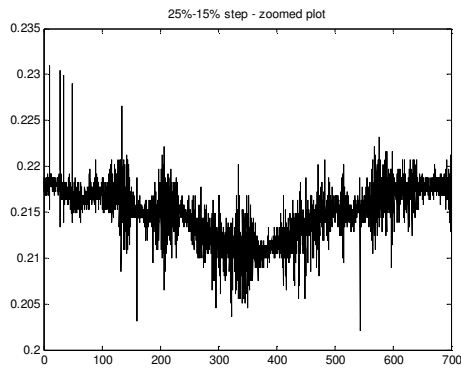
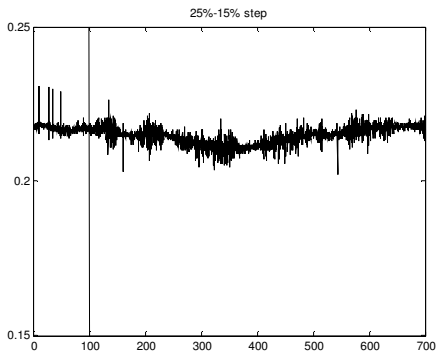
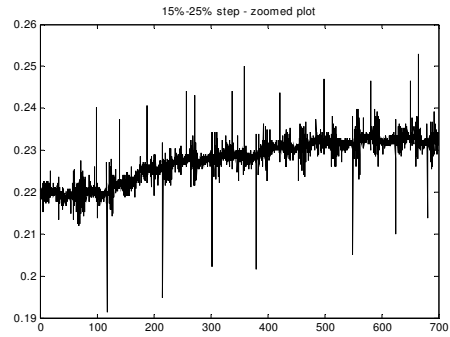
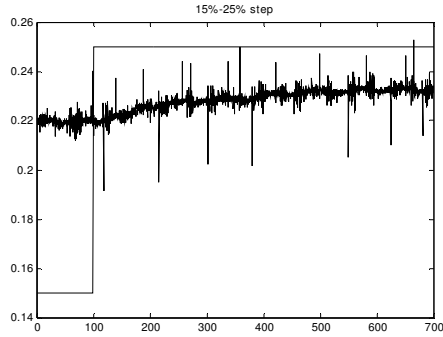




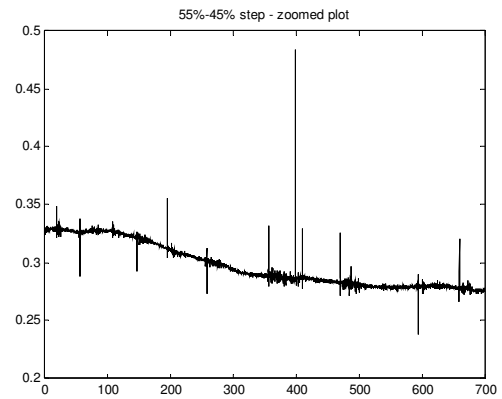
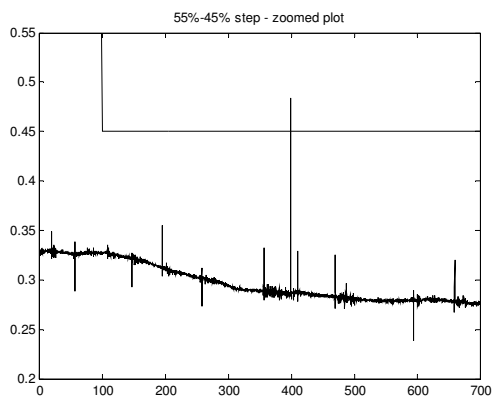
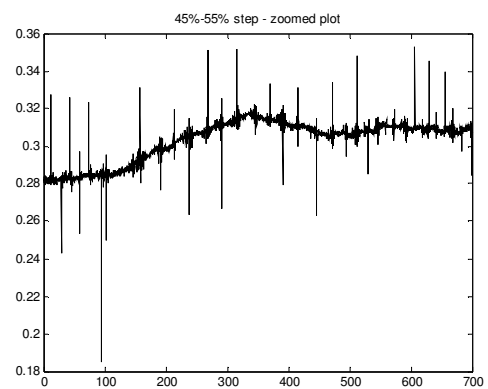
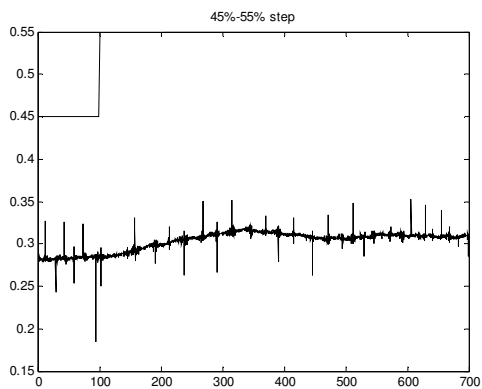
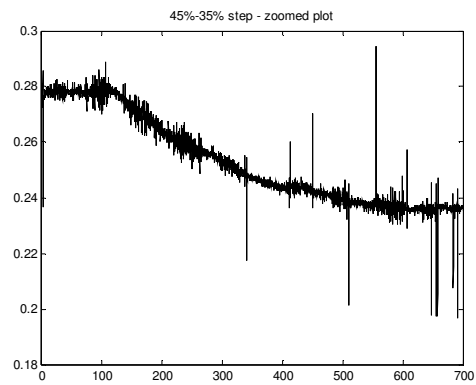
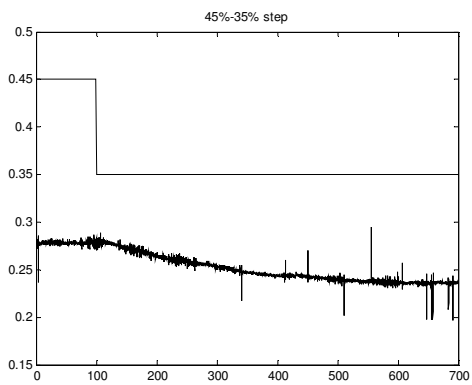
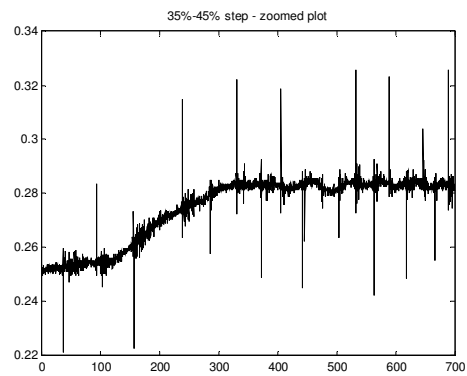
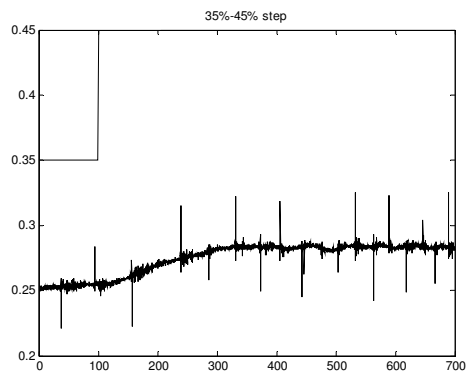


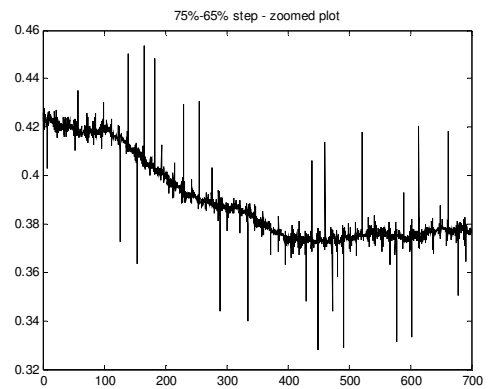
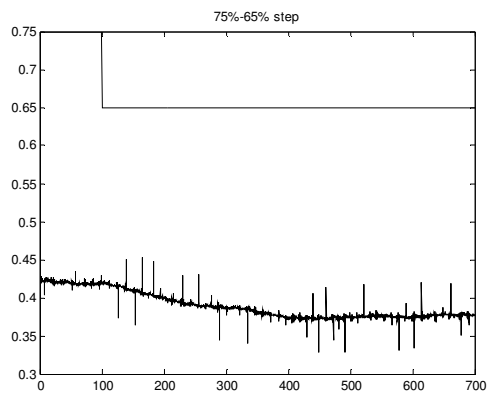
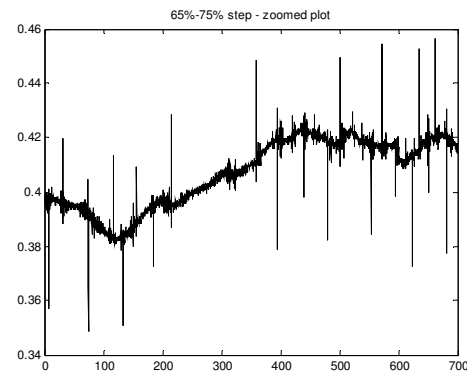
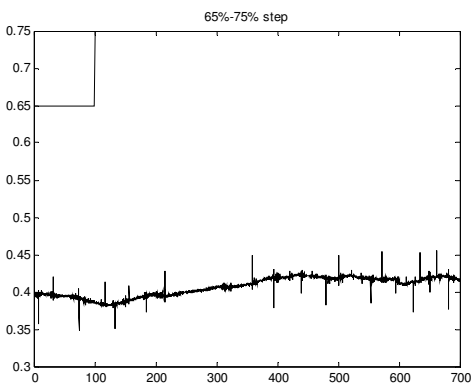
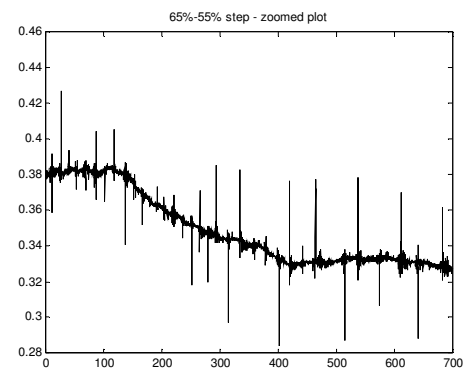
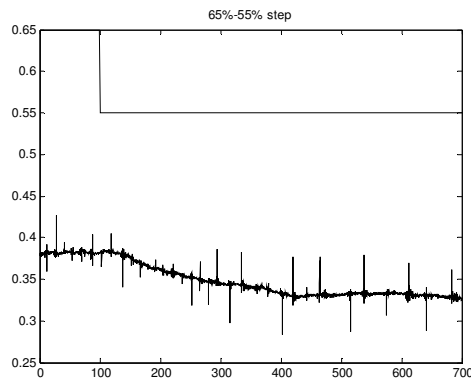
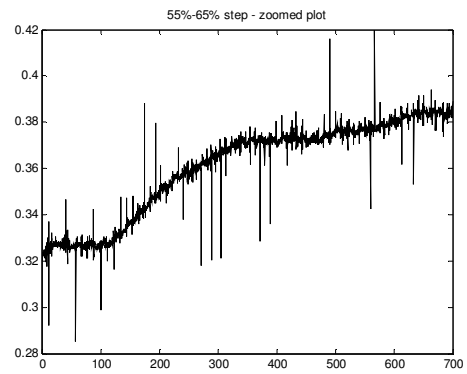
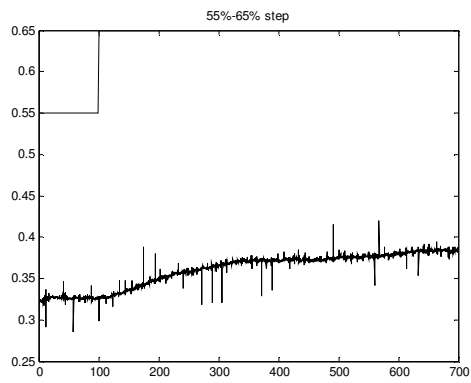
**Appendix E – Temperature Process Open Loop Step Test Plots (50% flow)**

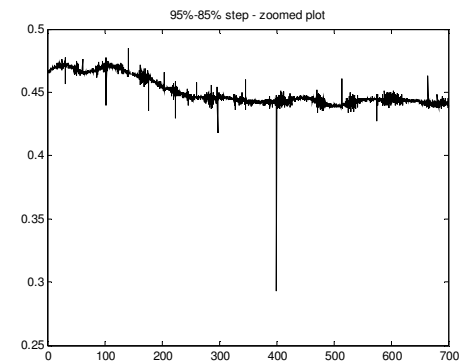
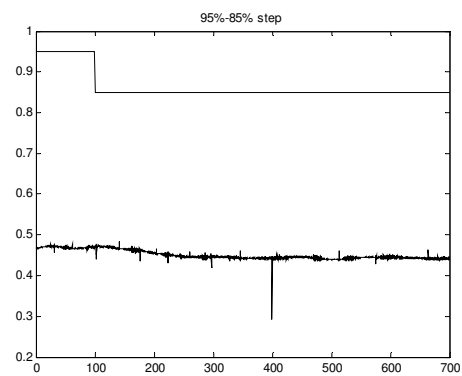
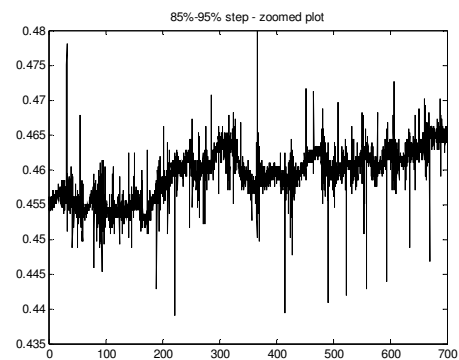
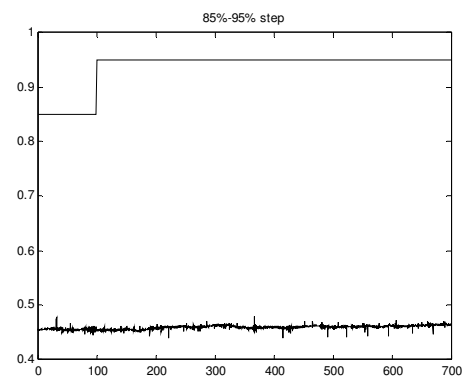
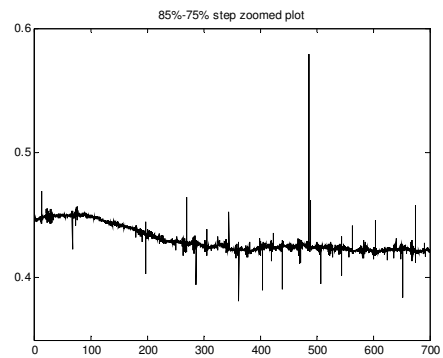
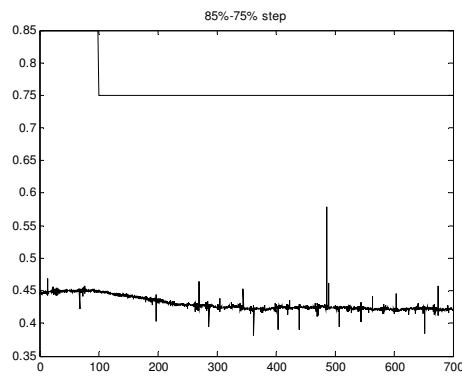
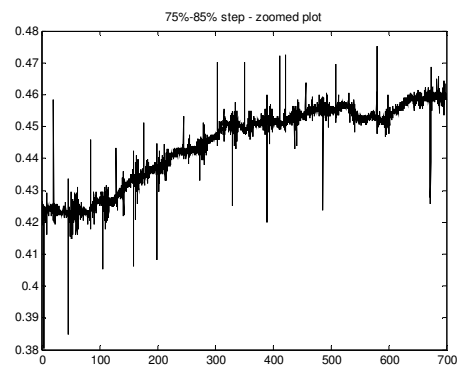
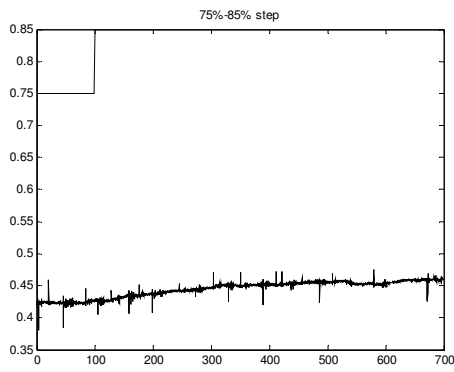
*[For all plots: the x-axis measures time and the y-axis measures process input and output as a portion of 5Volts]*





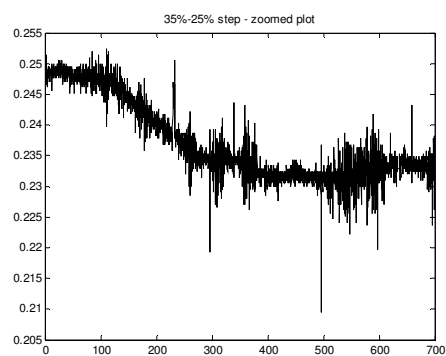
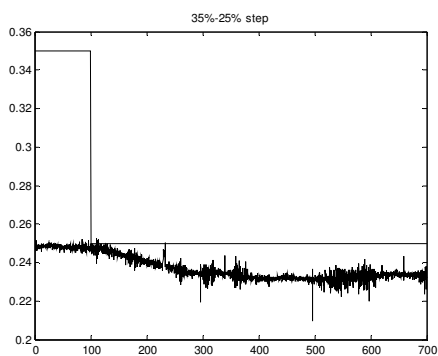
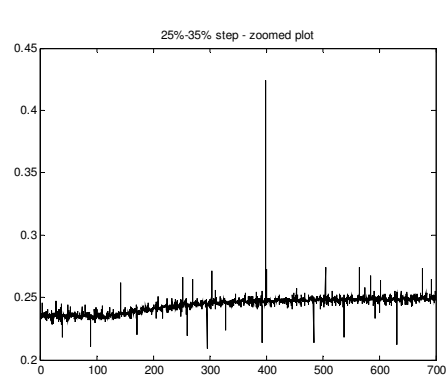
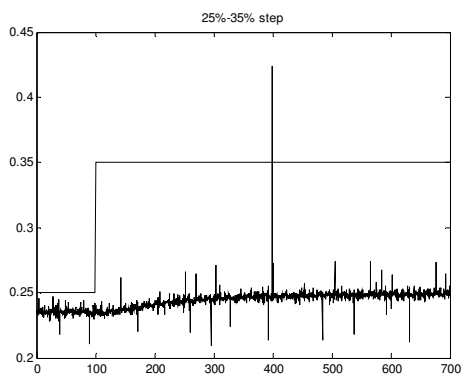
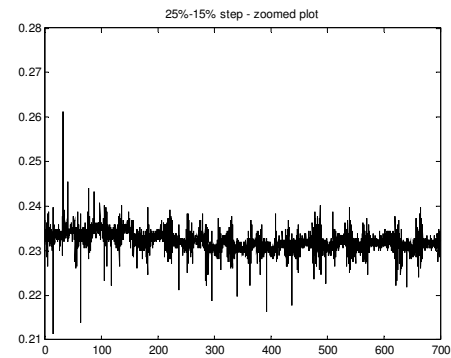
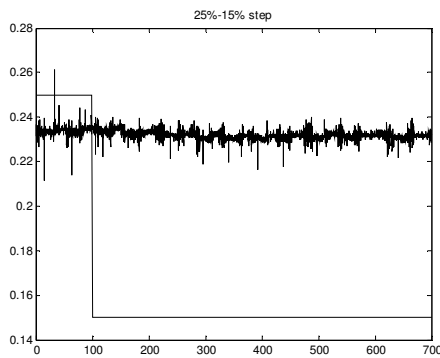
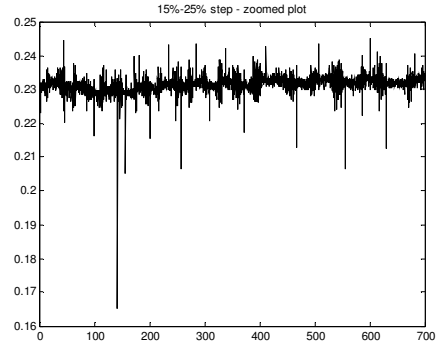
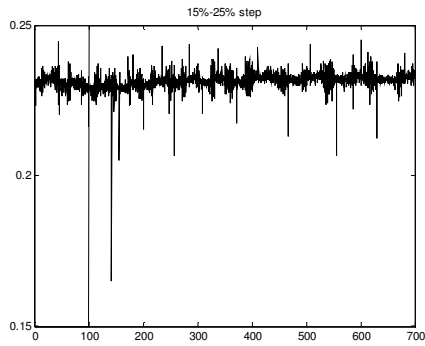


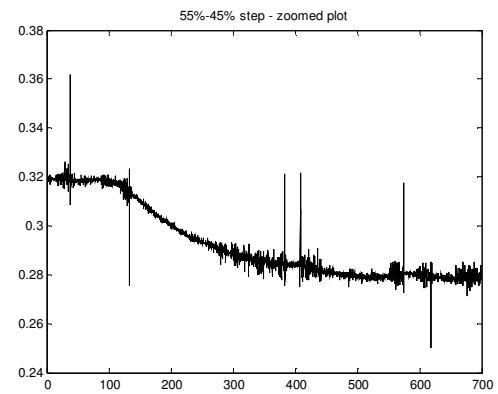
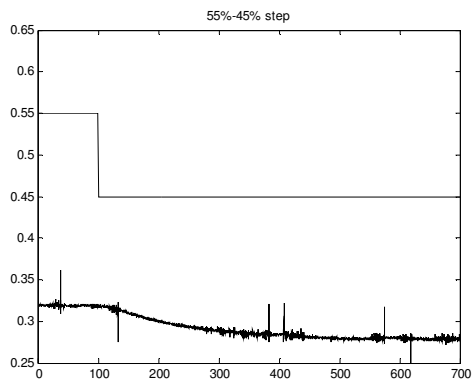
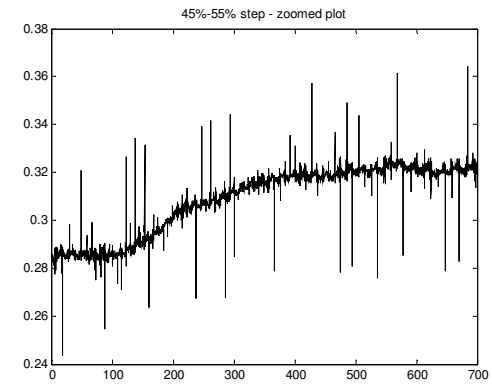
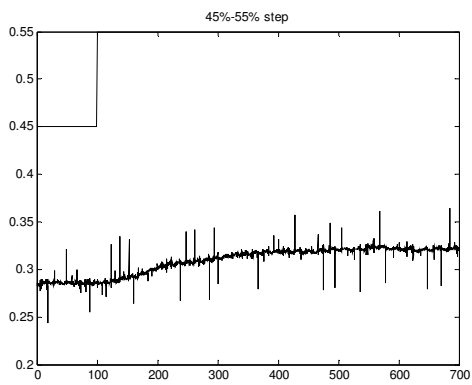
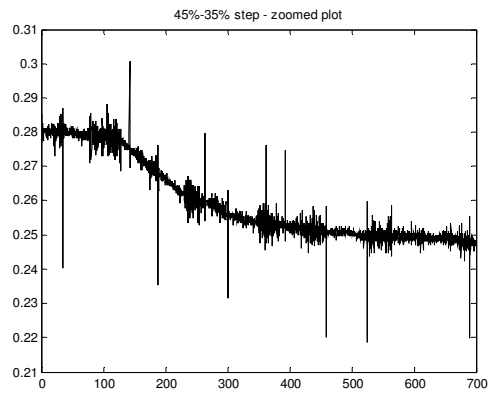
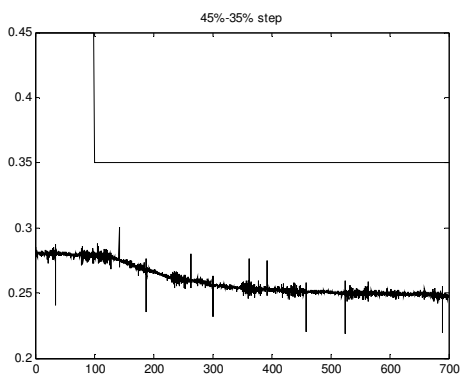
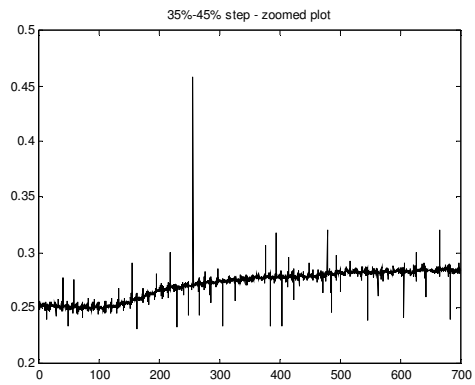
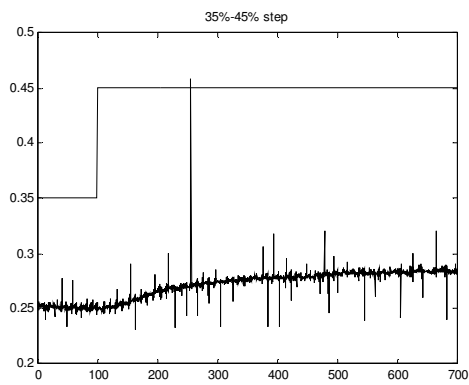


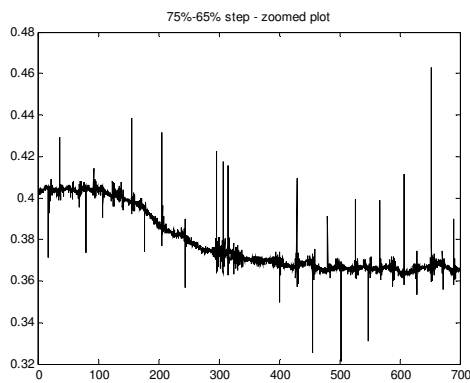
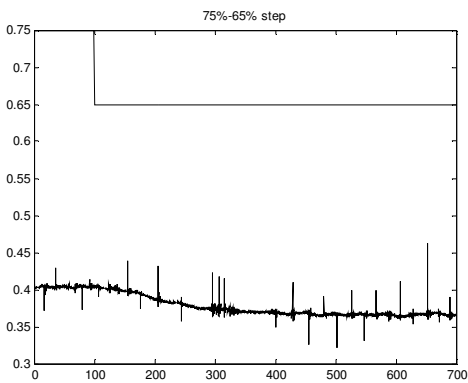
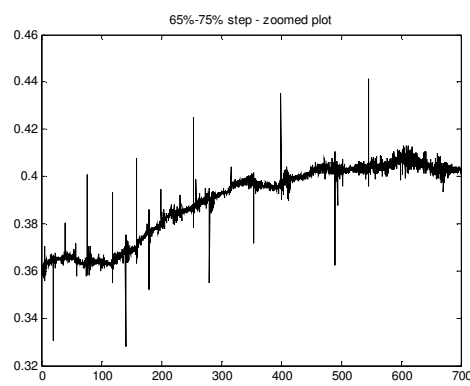
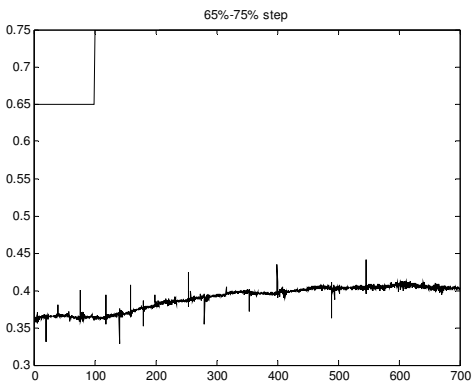
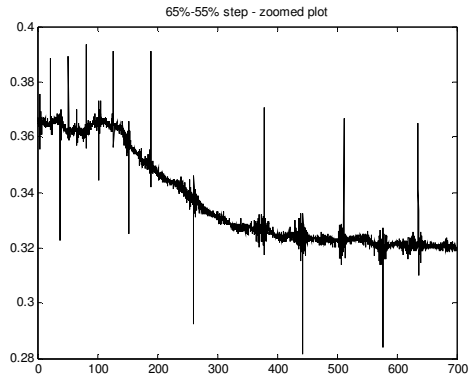
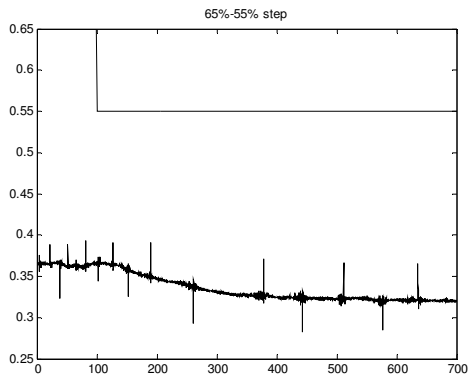
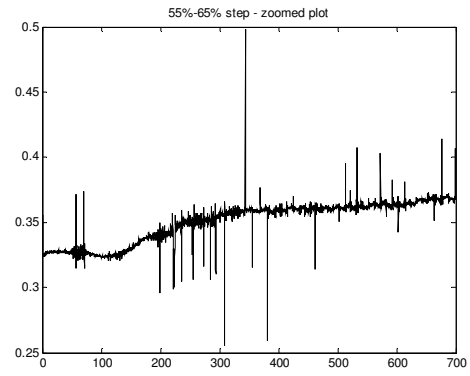
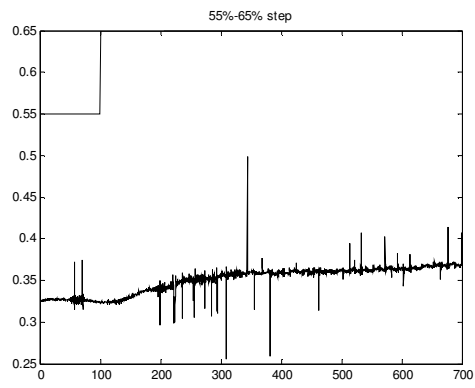


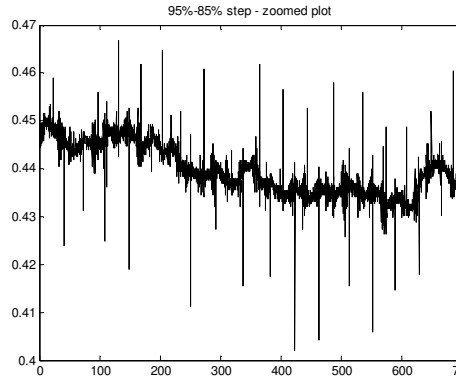
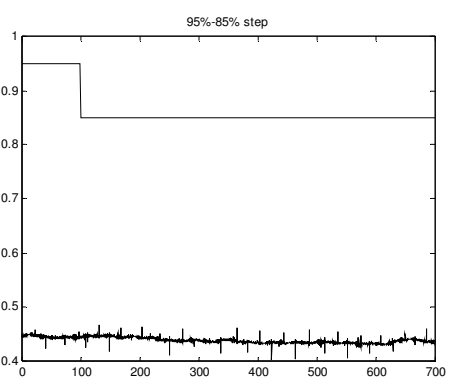
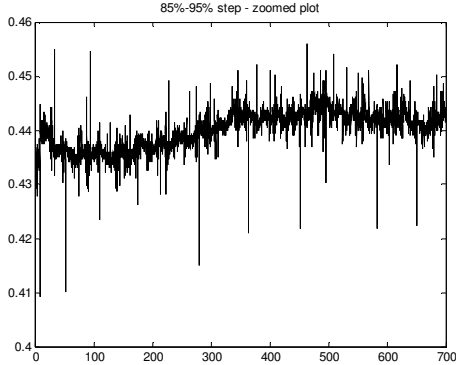
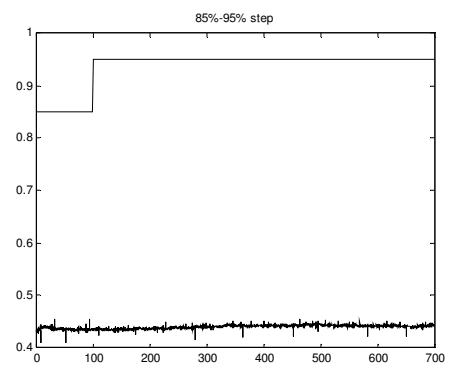
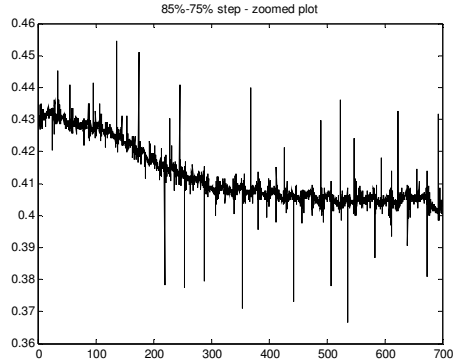
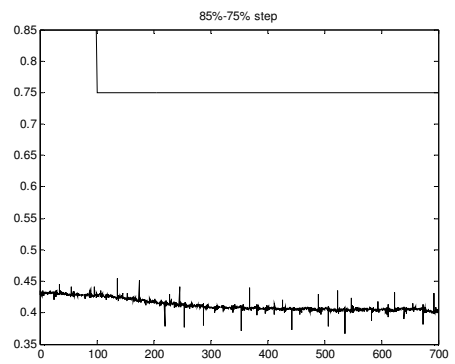
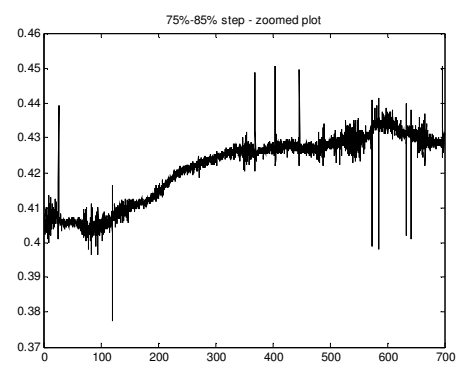
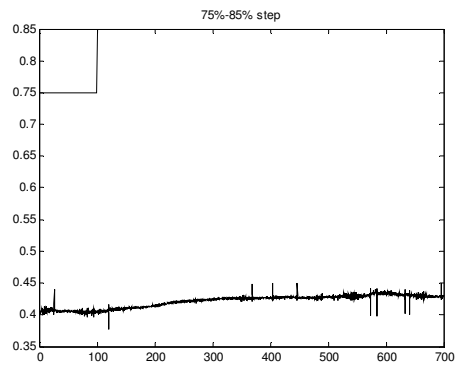
**Appendix F – Temperature Process Open Loop Step Test Plots (70% flow)**

[For all plots: the x-axis measures time and the y-axis measures portion process input and output as a portion of 5Volts]

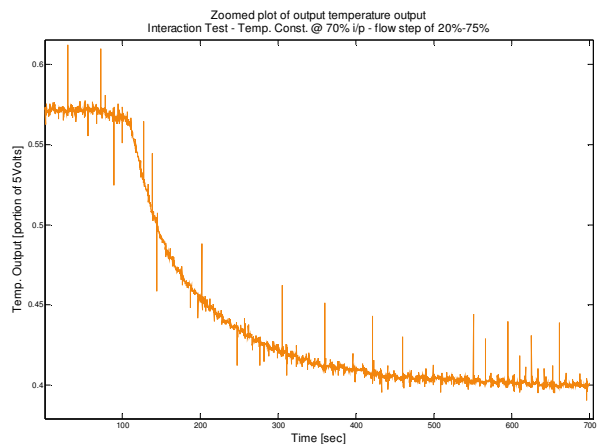
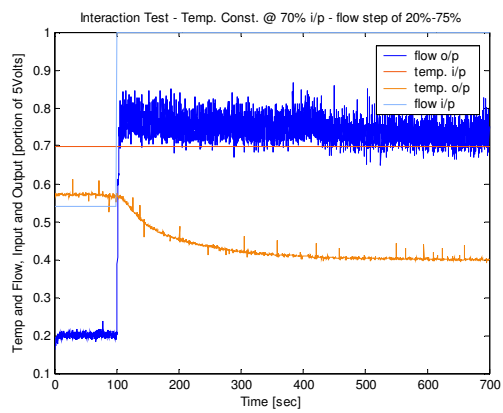
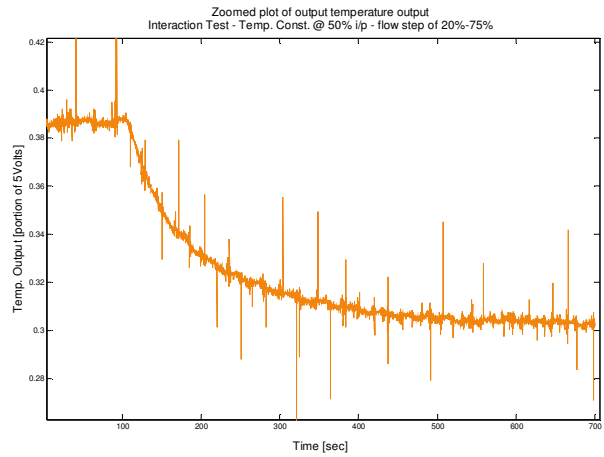
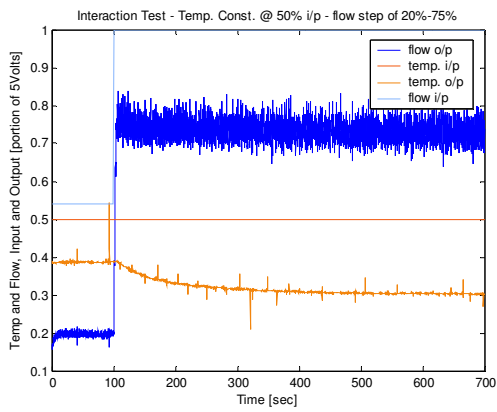
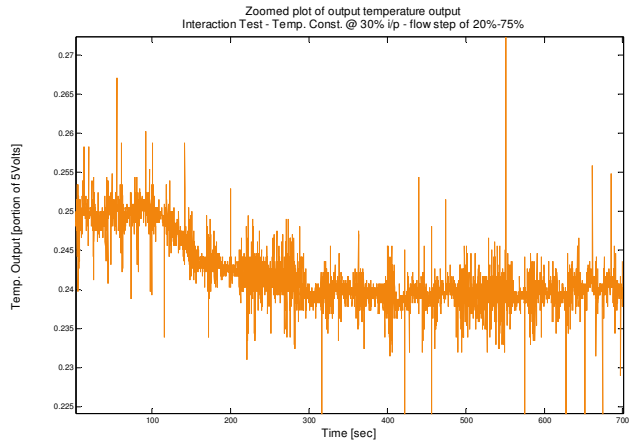
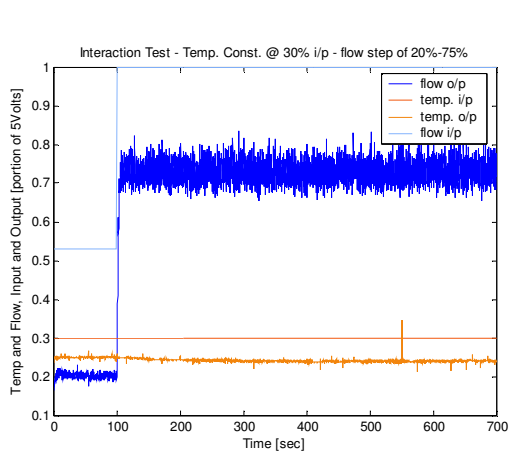








**Appendix G – Process Interaction step test plots**

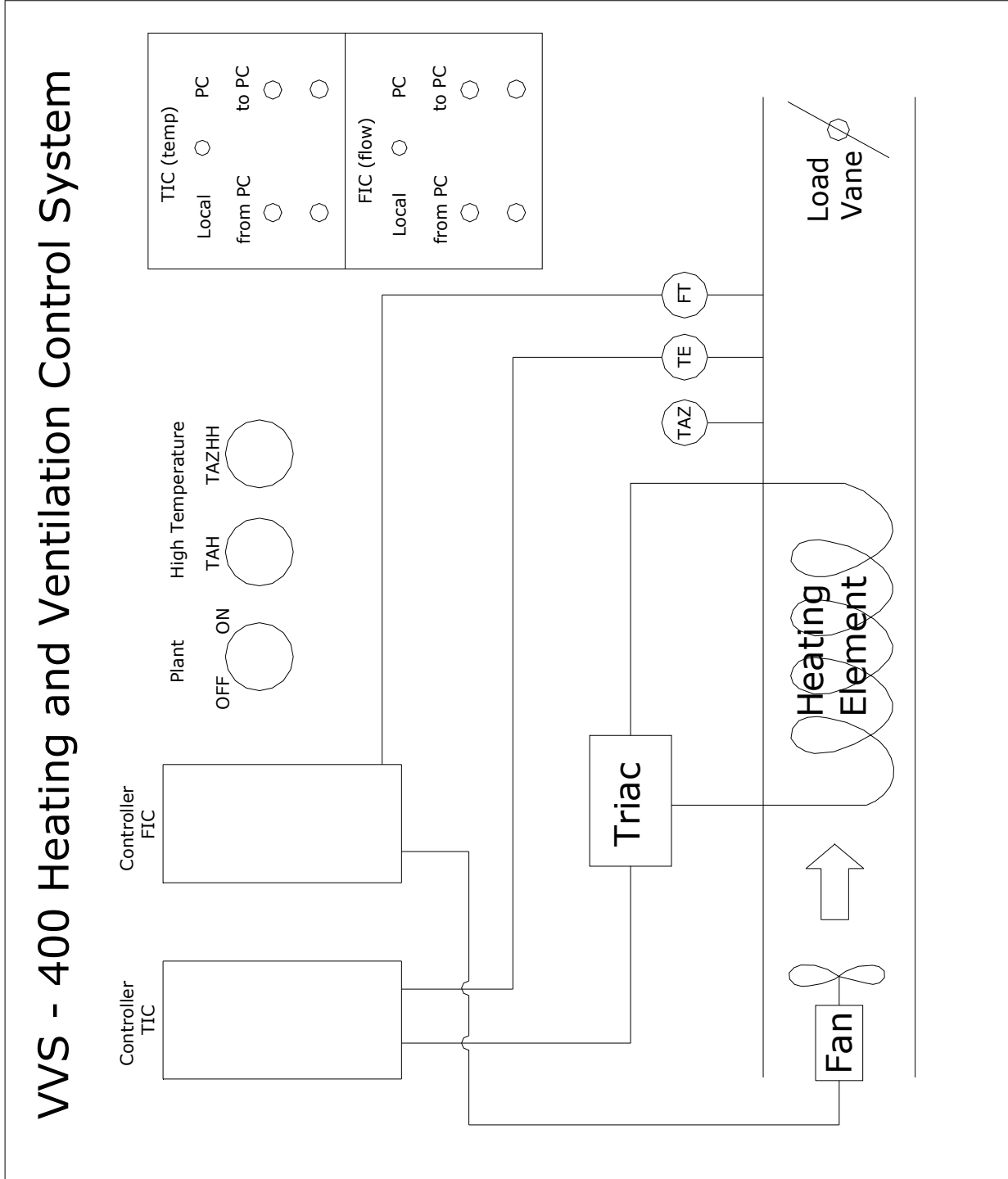




**Appendix H – Project Timetable**

|  | January  |   |   | February |   |   |   | March |   |    |    | April        |    |   |                      | May                         |    |    |    | June |   |    | July |    |    | August |    |    | September |    |    |    |    |    |    |    |    |  |  |  |
|--|--|---|---|----------|---|---|---|-------|---|----|----|--------------|----|---|----------------------|-----------------------------|----|----|----|------|---|----|------|----|----|--------|----|----|-----------|----|----|----|----|----|----|----|----|--|--|--|
| Week   | 1  | 2 | 3 | 4        | 5 | 6 | 7 | 8     | 9 | 10 | 11 | 12           | 13 | 14  | 15                   | 16                          | 17 | 18 | 19 | 20   | 21                                      | 22 | 23   | 24 | 25 | 26     | 27 | 28 | 29        | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 |  |  |  |
| <b>Literature Survey:</b>  | ongoing  |   |   |          |   |   |   |       |   |    |    |              |    |   |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
| <b>Familiarisation:</b>  | general investigation into process behaviour<br>initial testing / carrying out existing lab experiments<br>controller design - using local controller<br>controller design - using PC (PRO-REG) software<br>interface with Humusoft/Matlab<br>process identification - using simple techniques - step/freq response<br>investigation into possible interactions / MIMO process |   |   |          |   |   |   |       |   |    |    |              |    |   |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
| <b>Preliminary Report:</b>   |  |   |   |          |   |   |   |       |   |    |    | write report |    |   | prepare presentation |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
| <b>Exams:</b>  |  |   |   |          |   |   |   |       |   |    |    |              |    |   |                      | preparation and completion: |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
| <b>Detailed Identification &amp; Investigation into Non-Linearities:</b> |  |   |   |          |   |   |   |       |   |    |    |              |    | further and more complete temperature process identification          |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
|  |  |   |   |          |   |   |   |       |   |    |    |              |    | further static tests on temperature and flow processes                |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
|  |  |   |   |          |   |   |   |       |   |    |    |              |    | flow and temperature measurement accuracy tests                       |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
|  |  |   |   |          |   |   |   |       |   |    |    |              |    | process interaction tests   |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
| <b>Controller Design:</b>  |  |   |   |          |   |   |   |       |   |    |    |              |    | choosing PI & PID controller type and tuning rules - implement & test |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
|  |  |   |   |          |   |   |   |       |   |    |    |              |    | investigation into inverse response compensation methods              |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
|  |  |   |   |          |   |   |   |       |   |    |    |              |    | design, implement, and test static decoupler                          |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
|  |  |   |   |          |   |   |   |       |   |    |    |              |    | design simple gain scheduler  |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
| <b>Final Design: Implementation &amp; Validation Testing</b>             |  |   |   |          |   |   |   |       |   |    |    |              |    | design and implement advanced gain scheduler                          |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
|  |  |   |   |          |   |   |   |       |   |    |    |              |    | validation testing  |                      |                             |    |    |    |      |   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
| <b>Final Report:</b>   | concurrently written throughout the project  |   |   |          |   |   |   |       |   |    |    |              |    |   |                      |                             |    |    |    |      | complete dissertation                   |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |
|  |  |   |   |          |   |   |   |       |   |    |    |              |    |   |                      |                             |    |    |    |      | prepare presentation & technical report |    |      |    |    |        |    |    |           |    |    |    |    |    |    |    |    |  |  |  |

**Appendix I – Schematic of Rig**



**Appendix J – Matlab/Simulink Controller Settings****FLOW CONTROLLER SETTINGS**

| Operating Condition | <b>PI Controllers</b>    |      |                                |      | <b>PID Controllers</b>           |      |                |                                  |      |                |
|---------------------|--------------------------|------|--------------------------------|------|----------------------------------|------|----------------|----------------------------------|------|----------------|
|                     | Min. IAE (Murrill, 1967) |      | Min. IAE(Rovira et. al., 1969) |      | Min. IAE (Kaya and Scheib, 1988) |      |                | Min. IAE (Kaya and Scheib, 1988) |      |                |
| Flow                | K                        | I    | K                              | I    | K                                | I    | T <sub>d</sub> | K                                | I    | T <sub>d</sub> |
| L                   | 5.97                     | 2.76 | 4.05                           | 1.35 | 4.74                             | 4.66 | 0.65           | 4.18                             | 1.59 | 0.45           |
| M                   | 1.63                     | 0.78 | 1.16                           | 0.51 | 1.42                             | 1.24 | 0.68           | 1.11                             | 0.60 | 0.52           |
| H                   | 0.87                     | 0.50 | 0.63                           | 0.36 | 0.78                             | 0.79 | 0.58           | 0.59                             | 0.43 | 0.45           |
| Av.                 | 1.82                     | 0.90 | 1.28                           | 0.54 | 1.54                             | 1.47 | 0.64           | 1.25                             | 0.64 | 0.47           |
|                     | <i>Regulator Tuning</i>  |      | <i>Servo Tuning</i>            |      | <i>Regulator Tuning</i>          |      |                | <i>Servo Tuning</i>              |      |                |

Table J-1 Flow controller settings for Matlab/Simulink implementation

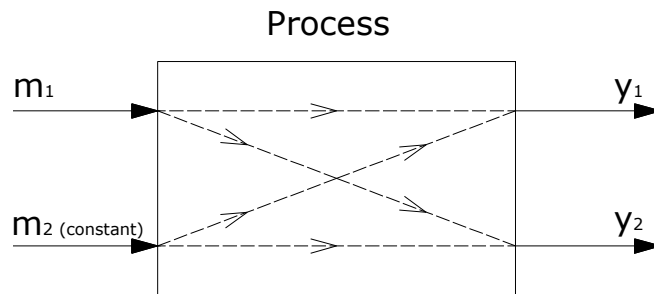
**TEMPERATURE CONTROLLER SETTINGS**

| Operating Conditions | <b>PI Controllers</b>    |       |                                |       | <b>PID Controllers</b>           |       |                |                                  |       |                |       |
|----------------------|--------------------------|-------|--------------------------------|-------|----------------------------------|-------|----------------|----------------------------------|-------|----------------|-------|
|                      | Min. IAE (Murrill, 1967) |       | Min. IAE(Rovira et. al., 1969) |       | Min. IAE (Kaya and Scheib, 1988) |       |                | Min. IAE (Kaya and Scheib, 1988) |       |                |       |
| Flow Temp.           | K                        | I     | K                              | I     | K                                | I     | T <sub>d</sub> | K                                | I     | T <sub>d</sub> |       |
| 30 %<br>flow         | L                        | 11.88 | 0.15                           | 7.72  | 0.06                             | 8.74  | 0.27           | 22.03                            | 8.49  | 0.07           | 14.51 |
|                      | M                        | 10.84 | 0.17                           | 6.56  | 0.04                             | 7.01  | 0.32           | 16.24                            | 8.01  | 0.05           | 9.61  |
|                      | H                        | 13.78 | 0.21                           | 8.42  | 0.05                             | 9.06  | 0.39           | 17.02                            | 10.14 | 0.07           | 10.21 |
| 50 %<br>flow         | L                        | 23.49 | 0.50                           | 13.97 | 0.11                             | 14.71 | 0.96           | 11.48                            | 17.51 | 0.14           | 6.61  |
|                      | M                        | 15.60 | 0.34                           | 9.43  | 0.08                             | 10.07 | 0.64           | 11.50                            | 11.53 | 0.11           | 6.80  |
|                      | H                        | 21.82 | 0.46                           | 13.22 | 0.11                             | 14.13 | 0.86           | 12.03                            | 16.12 | 0.14           | 7.13  |
| 70 %<br>flow         | L                        | 14.56 | 0.26                           | 9.29  | 0.09                             | 10.34 | 0.47           | 15.22                            | 10.50 | 0.11           | 9.73  |
|                      | M                        | 10.44 | 0.18                           | 6.68  | 0.06                             | 7.47  | 0.32           | 16.02                            | 7.51  | 0.08           | 10.31 |
|                      | H                        | 18.47 | 0.35                           | 11.30 | 0.09                             | 12.18 | 0.65           | 13.62                            | 13.58 | 0.11           | 8.19  |
| Av.                  | 11.62                    | 0.17  | 7.23                           | 0.05  | 7.90                             | 0.31  | 18.38          | 8.48                             | 0.06  | 11.34          |       |
|                      | <i>Regulator Tuning</i>  |       | <i>Servo Tuning</i>            |       | <i>Regulator Tuning</i>          |       |                | <i>Servo Tuning</i>              |       |                |       |

Table J-2 Temperature controller settings for Matlab/Simulink implementation

**Appendix K – Relative Gain Array Method**

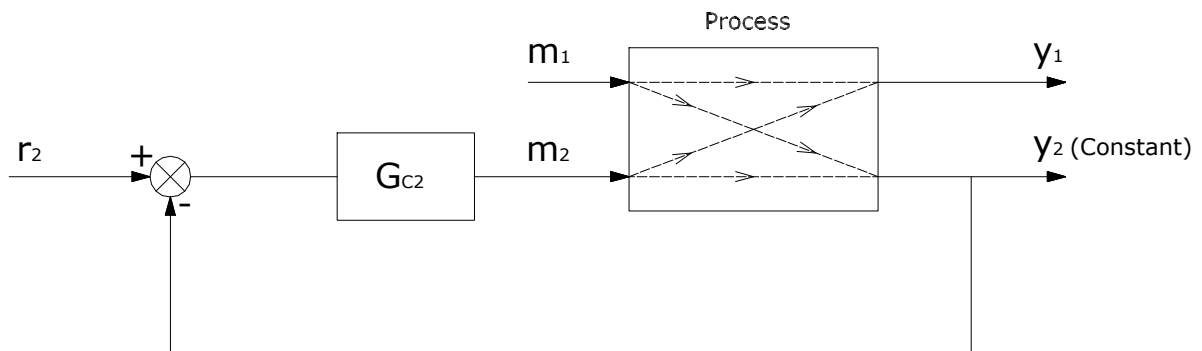
This method, developed by Bristol (1966), is used to determine the best pairing of controlled and manipulated variables for MIMO processes. The method is based solely on steady state information of the system. The VVS-400 heating and ventilation system has two inputs and two outputs, i.e. two processes – flow and temperature. Take  $m_1$  and  $m_2$  to be the inputs (or manipulated variables) and  $y_1$  and  $y_2$  to be the outputs (or controlled variables) of the temperature and flow processes respectively.



Assuming  $m_2$  is constant, a step change in input of magnitude  $\Delta m_1$  is introduced.  $\Delta y_1$  is the corresponding output change. So then the open loop gain between  $m_1$  and  $y_1$  is;

$$\left( \frac{\Delta y_1}{\Delta m_1} \right)_{m_2 \text{ constant}}$$

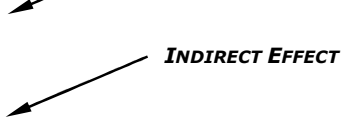
Now consider the loop gain between  $y_1$  and  $m_1$  when  $m_2$  can vary (in a feedback loop controlling the other output,  $y_2$ );



The controller  $G_{C2}$  attempts to hold  $y_2$  constant;  $m_2$  must change for this to happen, provoking a change in  $y_1$  ( $y_1'$ ); this change is a result of the “direct” effect (from  $m_1$ ) and the “indirect” effect (from  $m_2$ ). The open loop gain is then;

$$\left( \frac{\Delta y_1'}{\Delta m_1} \right)_{y_2 \text{ constant}}$$

Now, the “relative gain” between  $y_1$  and  $m_1$ ,  $\lambda_{11}$ , is defined by;

$$\lambda_{11} = \frac{(\Delta y_1 / \Delta m_1)_{m_2 \text{ const.}}}{(\Delta y_1' / \Delta m_1)_{y_2 \text{ const.}}}$$


Similarly, the remaining relative gains may be defined as follows;

Relative gain ( $m_2 - y_1$ ):

$$\lambda_{12} = \frac{(\Delta y_1 / \Delta m_2)_{m_1 \text{ const.}}}{(\Delta y_1' / \Delta m_2)_{y_2 \text{ const.}}}$$

Relative gain ( $m_1 - y_2$ ):

$$\lambda_{21} = \frac{(\Delta y_2 / \Delta m_1)_{m_2 \text{ const.}}}{(\Delta y_2' / \Delta m_1)_{y_1 \text{ const.}}}$$

Relative gain ( $m_2 - y_2$ ):

$$\lambda_{22} = \frac{(\Delta y_2 / \Delta m_2)_{m_1 \text{ const.}}}{(\Delta y_2' / \Delta m_2)_{y_1 \text{ const.}}}$$

The relative gains are arranged into a “relative gain array” (RGA):

$$\underline{\lambda} = \begin{pmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{pmatrix}$$

It can be shown that;  $\lambda_{11} + \lambda_{12} = 1$ ,  $\lambda_{11} + \lambda_{21} = 1$ ,  $\lambda_{12} + \lambda_{22} = 1$ ,  $\lambda_{21} + \lambda_{22} = 1$

Then, for a 2x2 MIMO system;  $\underline{\lambda} = \begin{pmatrix} \lambda & 1 - \lambda \\ 1 - \lambda & \lambda \end{pmatrix}$ ,  $\lambda = \lambda_{11}$

Possible cases for  $\lambda$ :

$\lambda = 1$ ;  $\underline{\lambda} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$  i.e. *direct effect = direct + indirect effect*, Pairings: ( $y_1, m_1$ ); ( $y_2, m_2$ ).

$\lambda = 0$ ;  $\underline{\lambda} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$  i.e. *no direct effect*, Pairings: ( $y_1, m_2$ ); ( $y_2, m_1$ ).

$0 < \lambda < 1$ ; i.e. the control loop interacts

$$\text{e.g. } \underline{\lambda} = \begin{pmatrix} 0.75 & 0.25 \\ 0.25 & 0.75 \end{pmatrix} \quad \dots \text{Pairings: } (y_1, m_1); (y_2, m_2).$$

$$\text{e.g. } \underline{\lambda} = \begin{pmatrix} 0.25 & 0.75 \\ 0.75 & 0.25 \end{pmatrix} \quad \dots \text{Pairings: } (y_1, m_2); (y_2, m_1).$$

$$\text{e.g. } \underline{\lambda} = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix} \quad \dots \text{Most severe interaction.}$$

$$\lambda > 1; \text{ e.g. } \underline{\lambda} = \begin{pmatrix} 1.2 & -0.2 \\ -0.2 & 1.2 \end{pmatrix}$$

i.e. closing the second loop reduces the gain between  $y_1$  and  $m_1$ ; as  $\lambda$  increases, the degree of interaction becomes more severe.

$$\lambda < 1; \text{ e.g. } \underline{\lambda} = \begin{pmatrix} -0.5 & 1.5 \\ 1.5 & -0.5 \end{pmatrix}$$

i.e. opening loop two gives a negative gain between  $y_1$  and  $m_1$ ; closing loop two gives a positive gain between  $y_1$  and  $m_1$ , i.e. the control loops interact by trying to “fight each other”.

So in general,  $y_1$  should be paired with  $m_1$  when  $\lambda \geq 0.5$ ; otherwise  $y_1$  should be paired with  $m_2$ .

The value for  $\lambda$  can be worked out as follows:

The steady state system outputs are;  $y_1 = k_{11}.m_1 + k_{12}.m_2$

$$y_2 = k_{21}.m_1 + k_{22}.m_2$$

Where  $k_{11}$ ,  $k_{12}$ ,  $k_{21}$ , and  $k_{22}$  are the steady state gains; i.e.:

$$k_{11} = \left( \frac{\Delta y_1}{\Delta m_1} \right)_{m_2 \text{ constant}}$$

$$\text{Putting } y_2 = 0; \quad 0 = k_{21}.m_1 + k_{22}.m_2 \quad \therefore \quad m_2 = -\frac{k_{21}}{k_{22}}.m_1$$

$$\therefore y_1 = k_{11}.m_1 - \frac{k_{12}k_{21}}{k_{22}}.m_1 \quad \therefore y_1 = k_{11} \left( 1 - \frac{k_{12}k_{21}}{k_{11}k_{22}} \right) m_1 \quad \therefore \left( \frac{\Delta y_1}{\Delta m_1} \right)_{y_2 \text{ const.}} = k_{11} \left( 1 - \frac{k_{12}k_{21}}{k_{11}k_{22}} \right)$$

$$\therefore \lambda_{11} = \frac{1}{1 - \frac{k_{12}k_{21}}{k_{11}k_{22}}} = \lambda$$

The following table describes the process model steady state gains obtained from chapter 3 (process identification) and section 4.3 (investigation into process interactions) over a range of operating conditions;

| S.S.Gain | Operating Conditions | 30%flow |       |       | 50%flow              |      |      | 70%flow |      |      |
|----------|----------------------|---------|-------|-------|----------------------|------|------|---------|------|------|
|          | Temp. i/p            | L       | M     | H     | L                    | M    | H    | L       | M    | H    |
| $K_{11}$ |                      | 0.32    | 0.61  | 0.45  | 0.32                 | 0.43 | 0.30 | 0.30    | 0.41 | 0.33 |
| $K_{12}$ | Temp. i/p            | L       | M     | H     | over full flow range |      |      |         |      |      |
| $K_{21}$ |                      | -0.02   | -0.18 | -0.24 |                      |      |      |         |      |      |
|          | Flow i/p             | L       | M     | H     | all temperatures     |      |      |         |      |      |
| $K_{22}$ |                      | 0       | 0     | 0     |                      |      |      |         |      |      |
|          |                      | 0.45    | 1.08  | 1.76  |                      |      |      |         |      |      |

Table K Steady state gains obtained from process modelling

From section 4.3 it was clear that the temperature to flow interaction does not exist. Therefore  $K_{21}$  was zero in all cases. This means that;

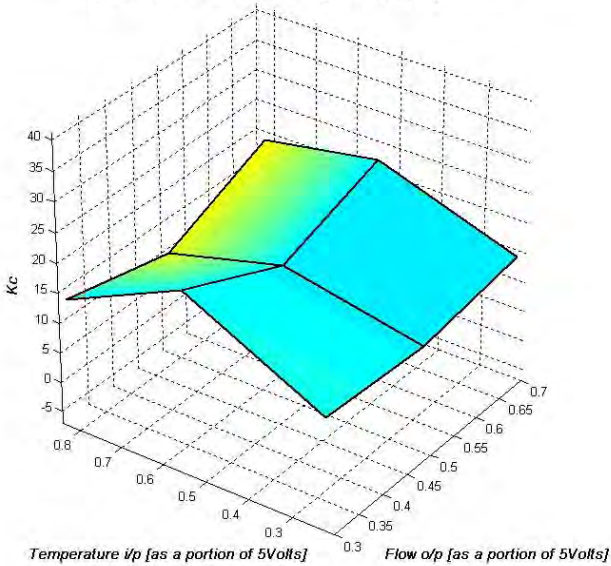
$$\lambda = \frac{1}{1 - \frac{k_{12}(0)}{k_{11}k_{22}}} = 1$$

Hence the appropriate pairings of controlled and manipulated variables should be:  $(y_1, m_1)$ ;  $(y_2, m_2)$ .

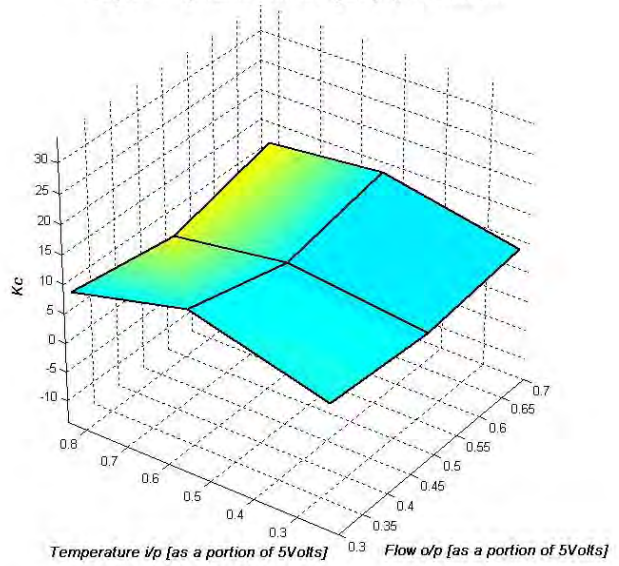
**Appendix L – Plots of Advanced Gain Scheduler Controller Settings**

**TEMPERATURE CONTROLLER GAIN PLOTS:**

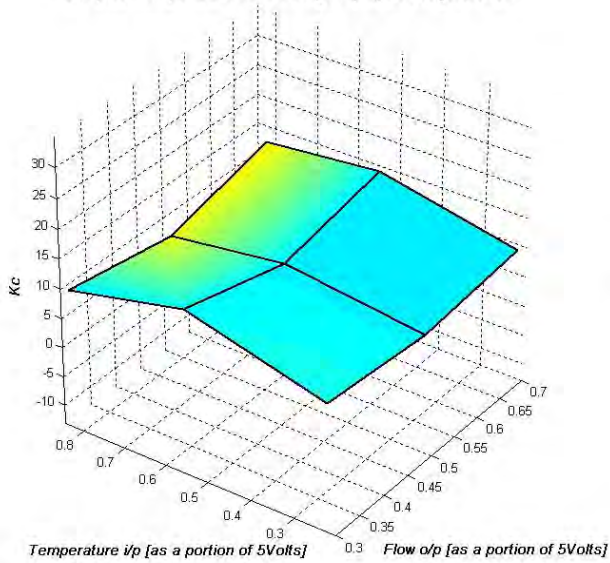
3-D plot: Temperature controller gain ( $K_c$ ) PI regulator ctr.



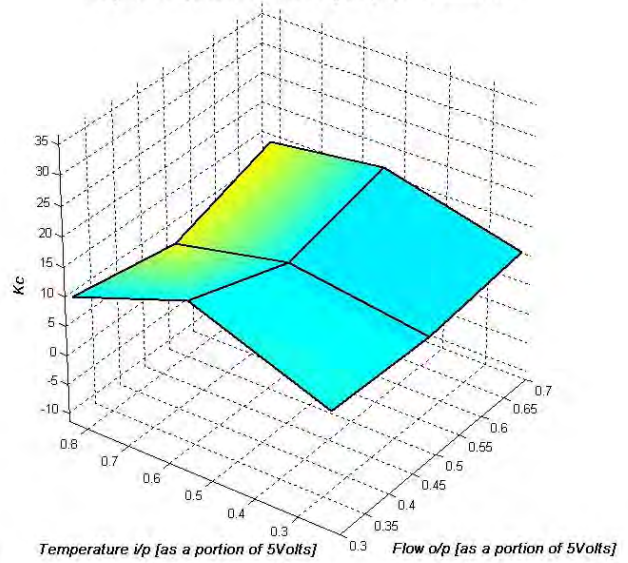
3-D plot: Temperature controller gain ( $K_c$ ) PI servo ctr.



3-D plot: Temperature controller gain ( $K_c$ ) PID regulator ctr.



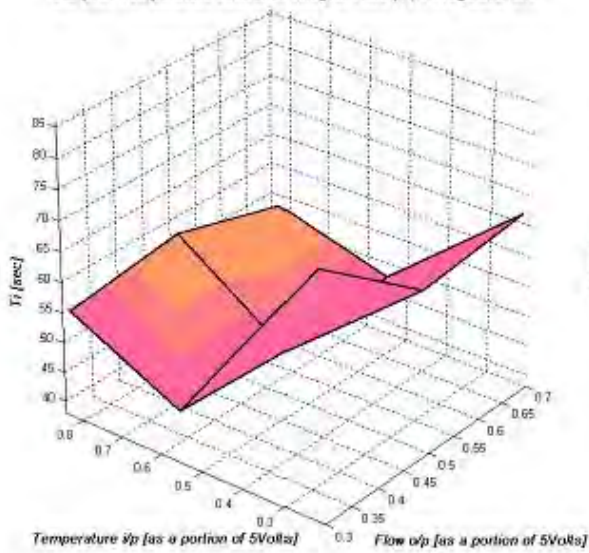
3-D plot: Temperature controller gain ( $K_c$ ) PID servo ctr.



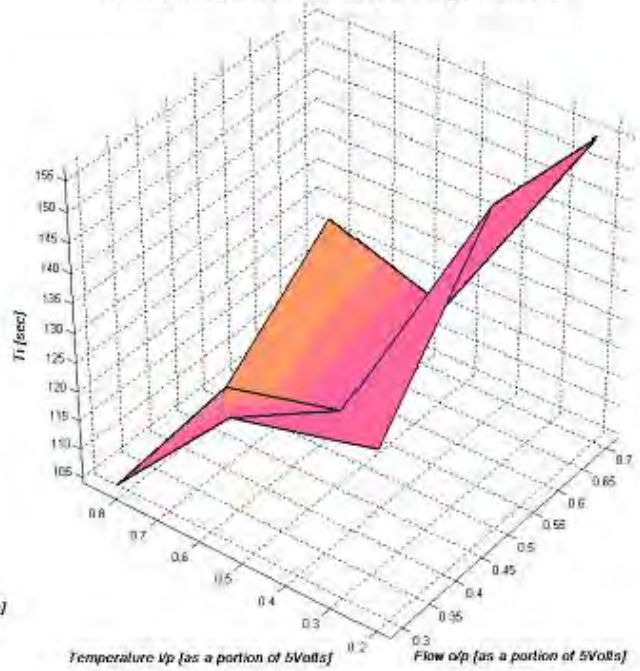


TEMPERATURE CONTROLLER INTEGRAL TIME PLOTS:

3-D plot: Temperature controller Integral Time (Ti) PI regulator ctr.

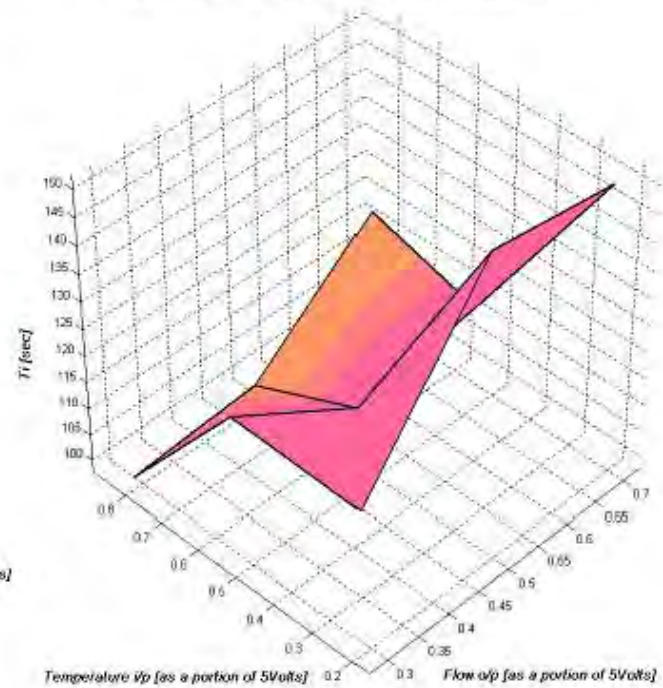
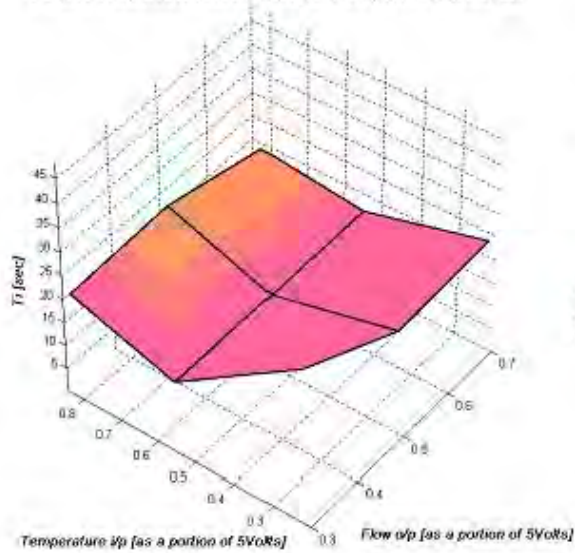


3-D plot: Temperature controller Integral Time (Ti) PI servo ctr.



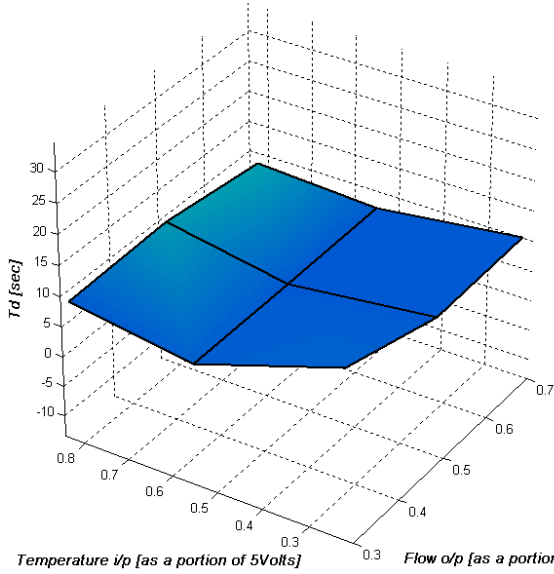
3-D plot: Temperature controller Integral Time (Ti) PID regulator ctr.

3-D plot: Temperature controller Integral Time (Ti) PID regulator ctr.

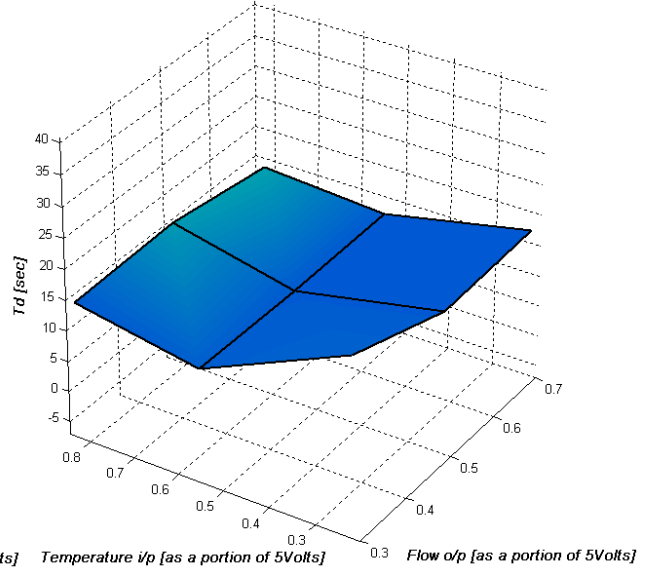


**TEMPERATURE CONTROLLER DERIVATIVE TIME PLOTS:**

3-D plot: Temperature controller derivative time ( $T_d$ ) PID servo ctr.

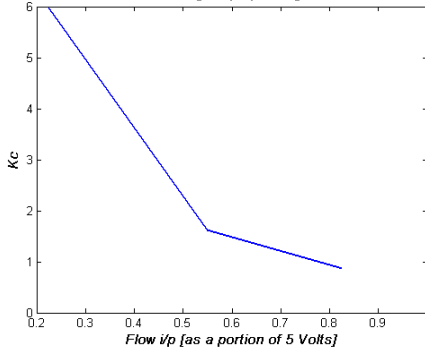


3-D plot: Temperature controller derivative time ( $T_d$ ) PID regulator ctr.

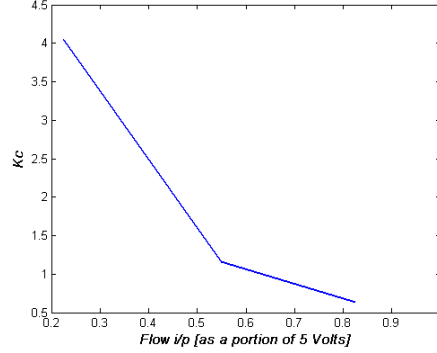


**FLOW CONTROLLER GAIN PLOTS:**

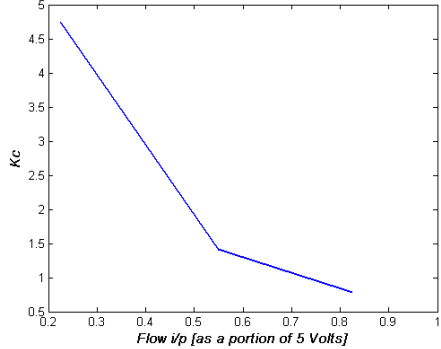
Flow controller gain ( $K_c$ ) PI regulator ctr.



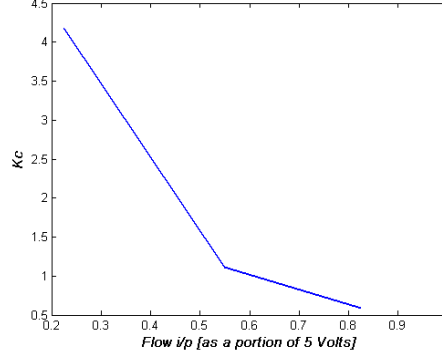
Flow controller gain ( $K_c$ ) PI servo ctr.



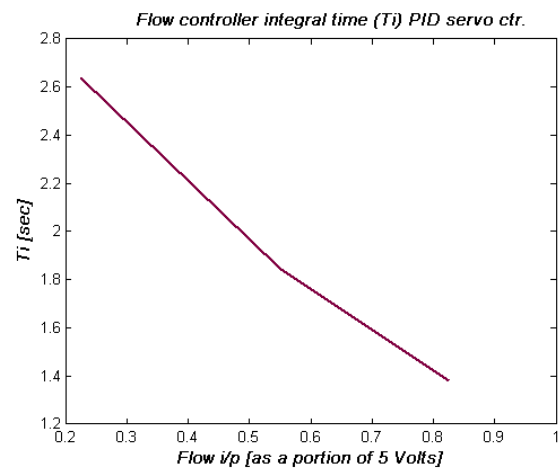
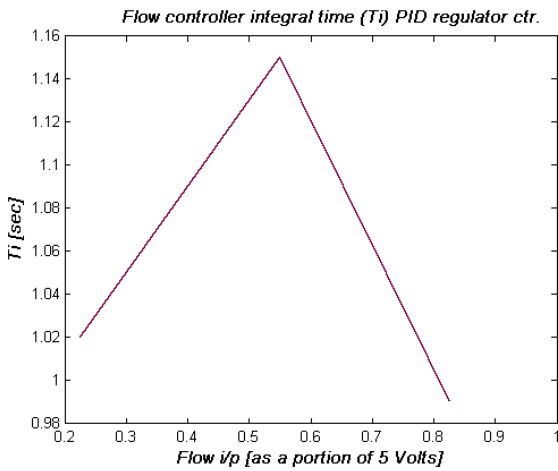
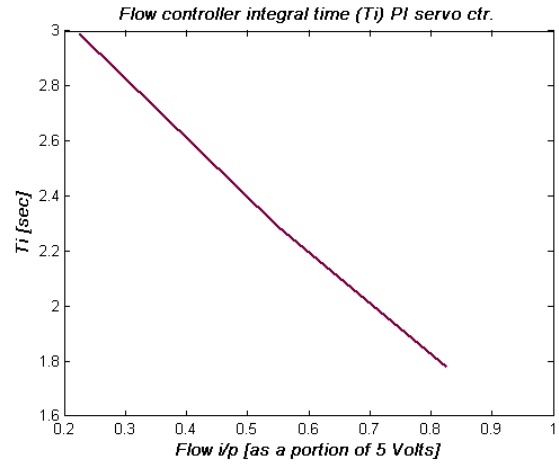
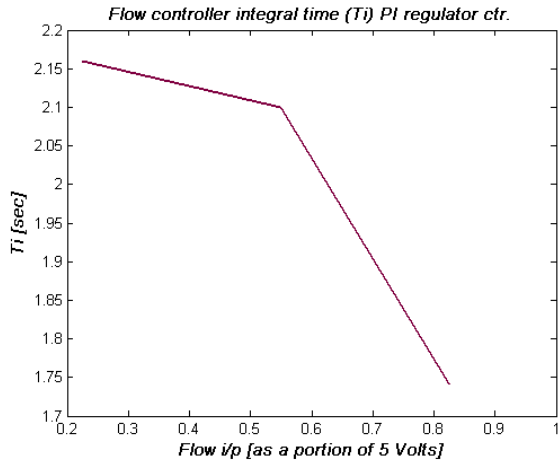
Flow controller gain ( $K_c$ ) PID regulator ctr.



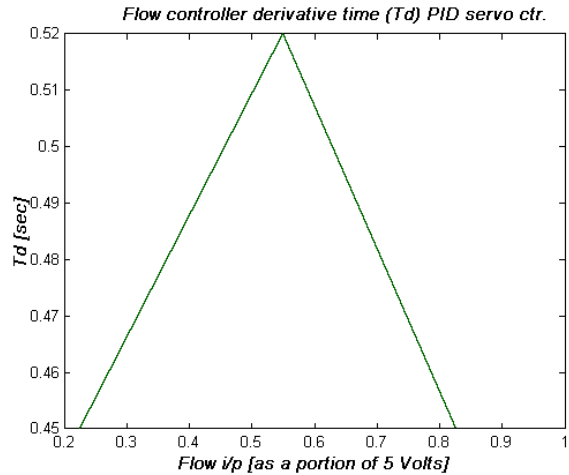
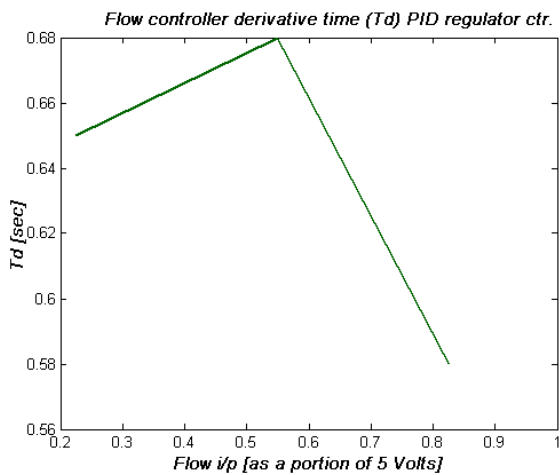
Flow controller gain ( $K_c$ ) PID servo ctr.



**FLOW CONTROLLER INTEGRAL TIME PLOTS:**



**FLOW CONTROLLER DERIVATIVE TIME PLOTS:**



Appendix M - All Advanced Gain Scheduler Controllers

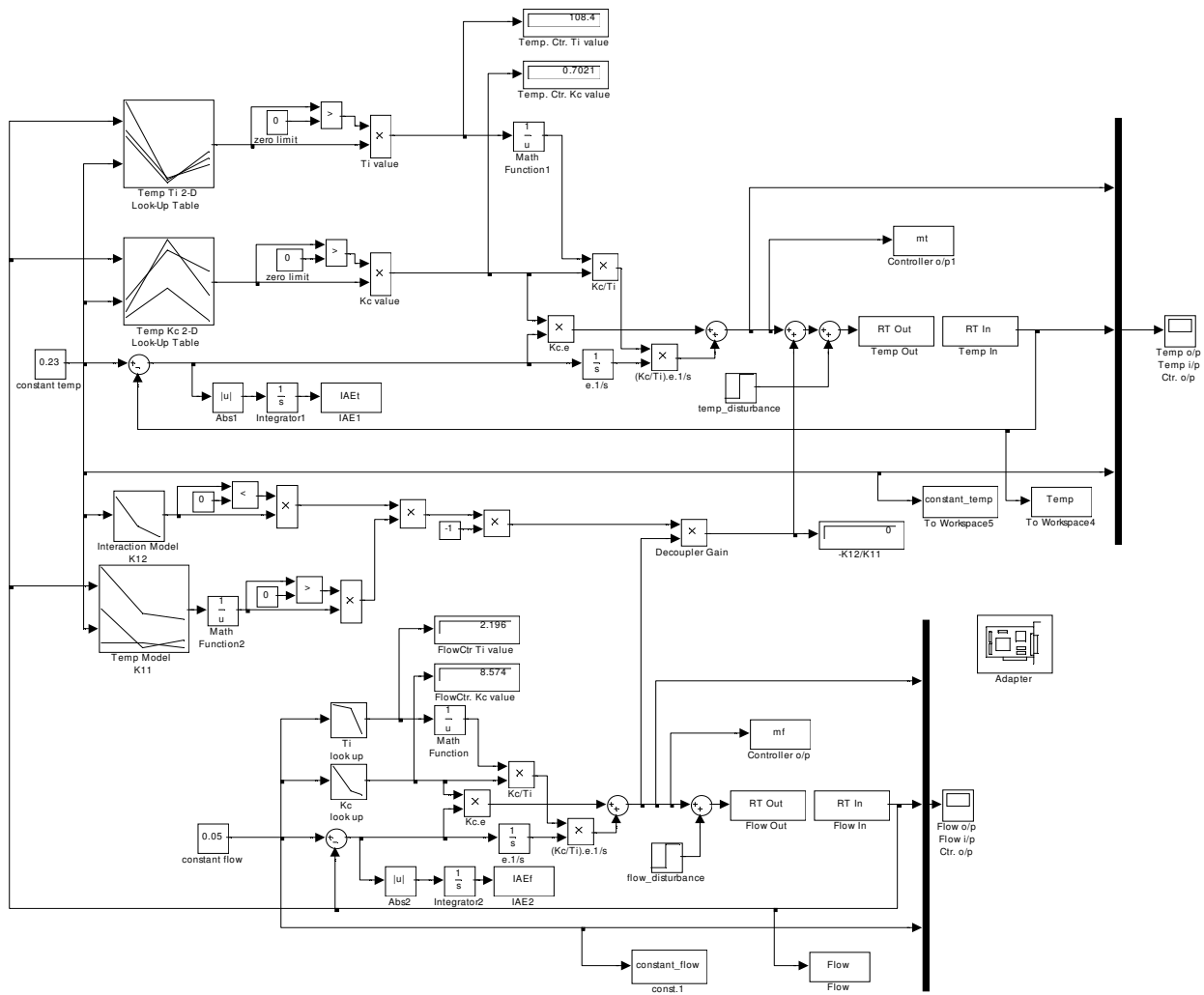


Figure M-1 Advanced gain scheduler – PI regulator

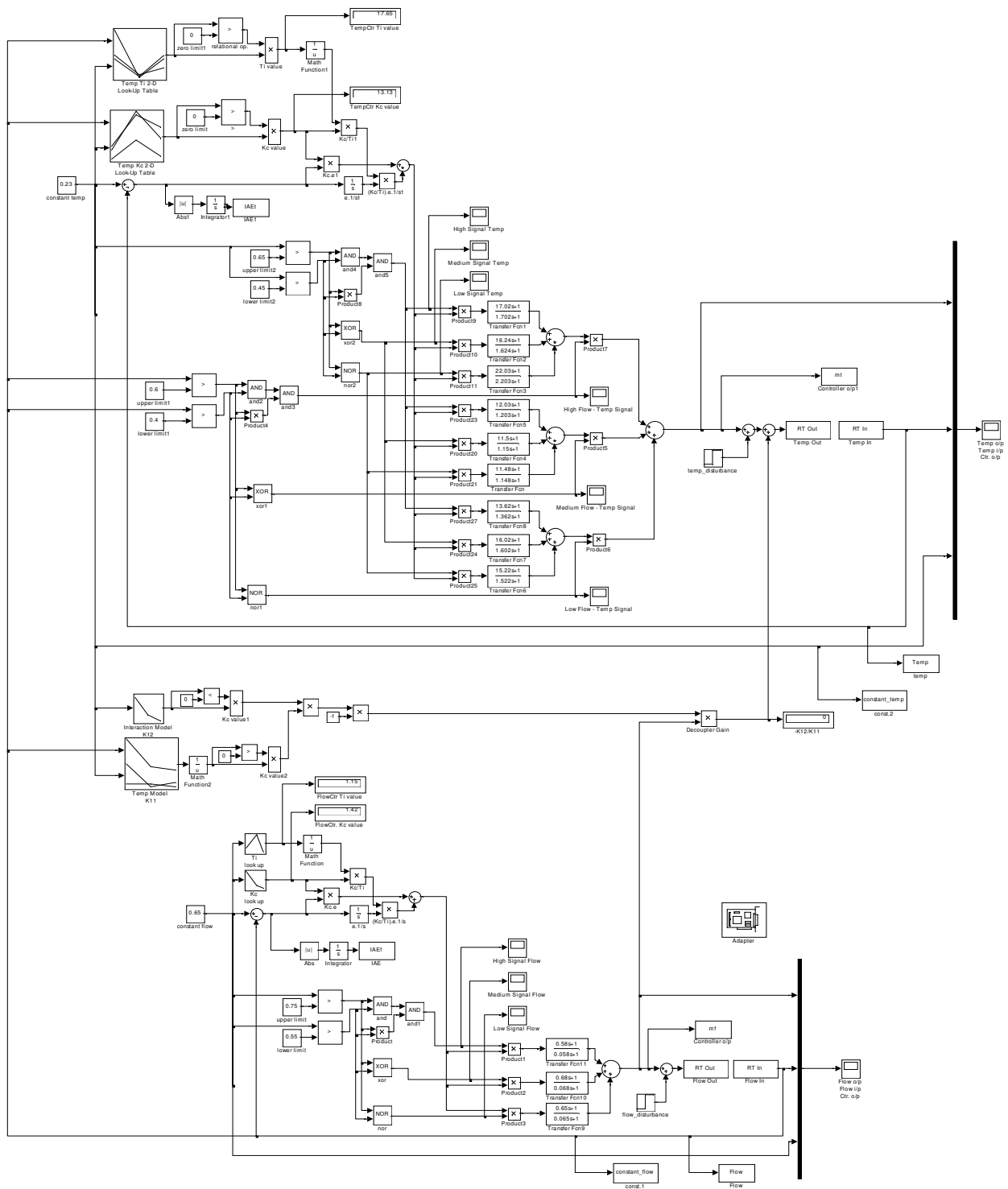


Figure M-2 Advanced gain scheduler – PID regulator

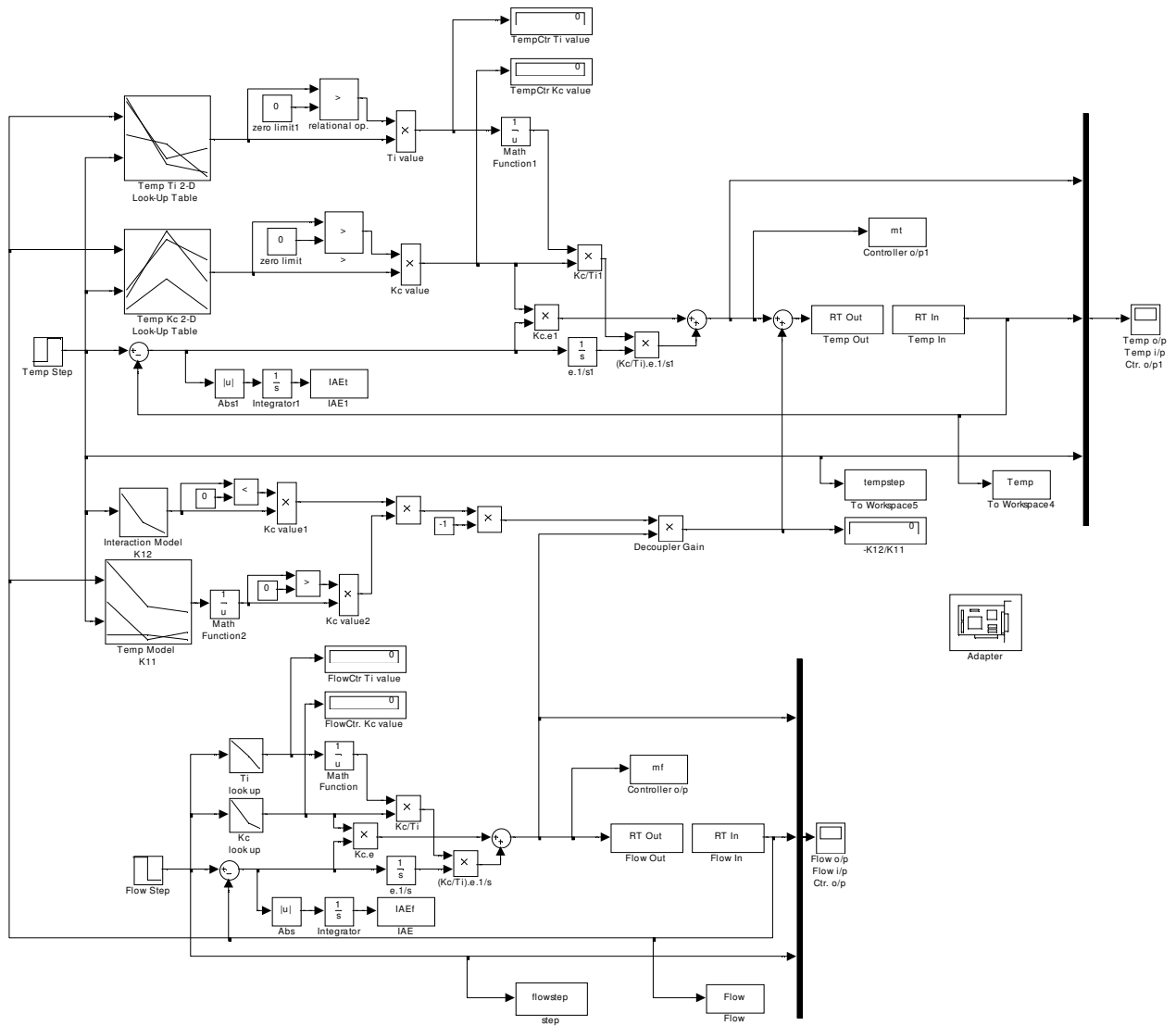


Figure M-3 Advanced gain scheduler – PI servo

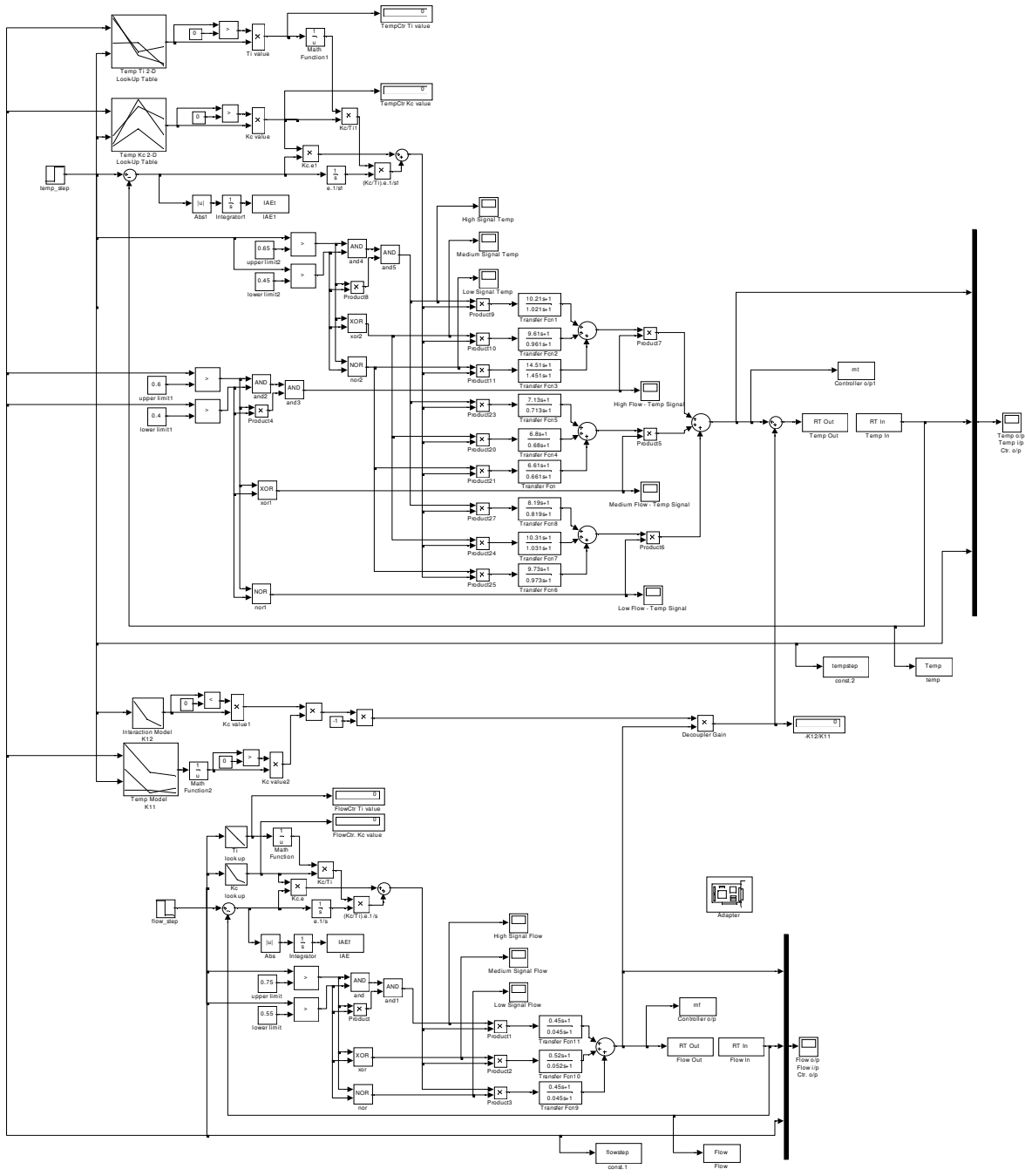


Figure M-4 Advanced gain scheduler – PID servo

## 8. Bibliography

- Bristol, E.H. (1966) “*On a new measure of interactions for multivariable process control*”, IEEE Trans. Automatic Control, Vol. AC-11, p.133.
- Douglas, J.F. and Mathews, R.D. (1996) *Solving Problems in Fluid Mechanics*, Vol.1, 3<sup>rd</sup> Edition, Longman.
- Evans, J. R. and Lindsay, W. M. (2002), *The Management and Control of Quality*, South-Western, Ohio,
- Iinoya, K. and Altpeter, R.J. (1962) “*Inverse Response in Process Control*”, I&EC, 54, (7), 39.
- Kaya, A. and Scheib, T.J., *Control Engineering* (July 1988), 62-65.
- McQuiston, F.C., Parker, J.D., and Spitler, J.D. (2002) *Heating, Ventilation, and Air Conditioning —Analysis and Design*, Wiley & Sons Ltd.
- Murrill, P.W. (1967) *Automatic control of processes*, International Textbook Company.
- O’Dwyer, A. (2003a) *Handbook of PI and PID Controller Tuning Rules*, Imperial College Press.
- O’Dwyer, A. (2003b) “*Adaptive and self-tuning control*”, Advanced Control Systems lecture notes, Masters in Advanced Engineering, DIT.
- O’Dwyer, A. and Ringwood, J.V. (1999) “*A classification of techniques for the compensation of time delayed processes. Part 1: Parameter optimised controllers*” Modern Applied Mathematical Techniques in Circuits, Systems and Control, World Scientific and Engineering Society Press, ISBN: 960-8052-05-X, pp. 176-186].
- Ogunnaike, B.A. and Ray, W.H. (1994) *Process dynamics, Modelling and Control*, Oxford University Press.
- Rovira, A.A., Murrill, P.W. and Smith C.L. *Instruments and Control Systems* 42, (December 1969), 67-69.
- Seborg, D. E., Edgar, T. F., and Mellichamp, D. A. (1989) *Process Dynamics and Control*, Wiley.
- Thompson, W. (Lord Kelvin) (1883). “*Electrical units of measurement*”, in Popular Lectures and Addresses, London, 1989, vol. 1.
- White, F. (1999) *Fluid Mechanics*, 4<sup>th</sup> Edition, McGraw Hill.
- Ziegler, J.G. and Nichols, N.B. (1942) *Optimum Settings for Automatic Controllers*, Trans ASME 64, 759.