

Instrumentation & Control Assignment 1

by Christopher Budden

FILE	24552_CHRISTOPHER_BUDDEN_INSTRUMENTATION__CONTROL_ASSI GNMENT_1_101701_442484273.DOCX (5.39M)		
TIME SUBMITTED	08-JUN-2017 10:57 PM (UTC+0100)	WORD COUNT	3185
SUBMISSION ID	73748966	CHARACTER COUNT	17943



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

Christopher Budden

P148323

Dr Stephen Mosey

LEVEL/ LEFEL 5

SEMESTER 2

MODULE CODE/ COD Y MODIWL:

M5X00940

Instrumentation and Control

ASSIGNMENT 1 PROCESS CONTROL SYSTEMS

ASSESSMENT TYPE/ MATH O ASESU: ASSIGNMENT

ASSESSMENT/COMPONENT/ ASESAD/CYDRAN: 1

Instrumentation & Control: Process Control Systems

By

Christopher Budden

Contents

Abstract	Page 1
Introduction	Page 2-3
Method	Page 3-15
- Task 1	
- Task 2	
Results	Page 16
Discussion	Page 17
Conclusion	Page 17
References, Bibliography & Figures	Page 18

Abstract

This paper is intended to solve the engineering problem of designing a control system capable of controlling an Industrial process that uses a transfer function. The control system is employed to automatically obtain real time input and output data into and out from the proportional integral derivative controller and adjust to follow an input command with minimal set point overshoot and in a quick time fashion.

The designed control system in this assignment will be used to follow a unit step input with an overshoot less than 5% and settling time less than 2 seconds, using MATLAB's Simulink tool, which will be used to build a control system and run the simulation.

In this paper, the plant transfer function is to have a stored variable coefficient, which is generated from the last three digits of the student's unique ID number. The PID controller is then tuned in order to cohere with the unit step input and the feedback loop from the output of the transfer function.

Three systems are constructed in Simulink®; the first simulated system is a closed-loop feedback that incorporates a PID block function. This is then condensed down into a control system that will consist just of a transfer function with a closed loop feedback mode, and one that only utilises a transfer function, in order to see if the latter two systems can function stably as that of the PID controller and compare all three system characteristics and critically analyse any variations and findings.

Introduction

The forms of control systems are currently known as;

- Proportional
- Proportional Integral
- Proportional Derivative
- Proportional Integral Derivative

As proportional being the basic form of control system. By definition 'proportional' means; ***'If two amounts are proportional, they change at the same rate so that the relationship between them does not change'*** [1]. In other words, if the error is large, then the effort or controller output is large in order to compensate. A proportional system simply measures the output compared to the input value and subtracts the input value from the measured output value to sum the error value. The fundamental characteristic of a proportional control system is that the output can never reach the desired input set point due to the finite limitations, as in the finite error will give a finite output. Although increasing the gain can give the desired output with a small error [2][3]. Proportional only accounts for the present values of error at that instance of time.

Proportional Integral controllers differ from that of proportional controllers, in that it integrates the error over time. Integral simply calculates the past values of error to which apply the controller effort at present time. The purpose of integral controllers is to plot a trend of past errors in order to predict current errors and respond to the input demand more efficiently. The downside to integral controllers is that there is a transient delay due to the past value calculations needed to sum the present error, denoted as $n-1$, where n being the number of samples used to integrate.

Proportional Derivative controllers calculates for future errors, it does this by deriving the current slope of the error, therefore plotting the trend line and extrapolating it to give an advance in the error value. Derivative process can have an added benefit to controller response as it can act quickly to change in error. As described by controleng.com ***'Derivative acts as a brake or dampener on the control effort'*** [4]. By this definition, the derivative portion of the control system inhibits the overshoot of the process output. The downside to this is that it can adversely amplify any noise that is on the process variable signal, therefore, any noise will need to be filtered [5].

Proportional Integral Derivative controller or PID is the accumulation of all three control methods listed above. They are in use in industry for various tasks and applications. An example of what they are used for and where they are found...

For example, a common application for a PID controller widely used for are temperature controllers for industrial heaters, to which require a temperature program that uses different levels of temperatures and different temperature ramping, soaking or a set point temperature mode.

Fig. 1 is a CAL 9500P® closed loop controller which is primarily used to control such industrial plant like high power heaters that require accuracy. This PID controller, in conjunction with the CALGrafix® software can enable the PID controller to 'auto tune' using the



Fig. 1

controller. This is detrimental if the controller is used to control high power heater thyristors that heats an oven. But depending on the configuration of the oven volume, heater pipe length and the heat transfer response, the 'auto tune' is vital so that the transfer function is calculated and the PID controller can heat the oven up efficiently and with minimum overshoot. This is highly important when conducting trials such as the IM (Insensitive Munitions) Slow Cook-off trials; STANAG 4382, to where the test asset needs to be slow heated at a rate of 3.3°C/hr [6][7]. Therefore the PID controller that has been 'tuned' makes these kind of munitions testing accurate and compliant.

Method

The aim of this assignment is to create a control system using MATLAB's Simulink® in order to control a transfer function industrial process to achieve a tolerance of 5% overshoot and settle time within 2 Seconds for a unit step input.

The method for designing a control system to automatically process the transfer function, using Simulink®, will be carried out in two stages. The first task will be to produce a variable in order to store the plant transfer function. The variable that is used to create this is assigned by the unique student ID numbers, from the last three digits.

The assigned digits will be applied to the transfer function;

$$G(s) = \frac{1}{as^2 + bs + c}$$

The unique student ID number coefficients are as follows;

Student ID	<i>a</i>	<i>b</i>	<i>c</i>
P148323	5	5	2

Fig. 2

And so, the transfer function that will be employed will be;

$$G(s) = \frac{1}{5s^2 + 5s + 2c}$$

The PID & Plant Transfer Function will be designed in Simulink® as illustrated in Fig. 3 below, which will include a closed loop feedback design, in order to accompany the error side of the control.

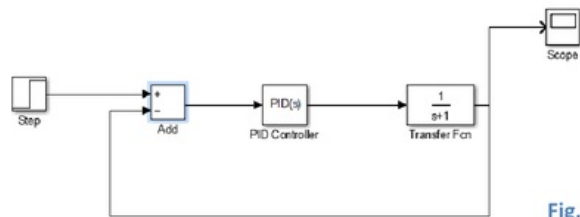


Fig. 3

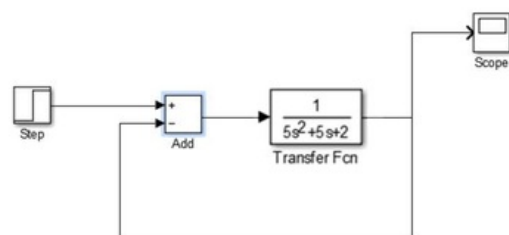


Fig. 4



Fig. 5

Then in **Task 2**, the system that is designed in **Task 1** will be condensed down into a single transfer function to replace the PID controller, then the performance of these will be compared to that of the initial system designed in **Task 1**. Fig. 4 will have the PID process removed, to which the new transfer function incorporates the initial PID operation. After Fig. 4 is carried out and simulated, the next procedure in this assignment is to further reduce down the block functions to which a newer version of Transfer function is created; this will be done in accordance to Fig. 5 illustrated above. This method makes the closed loop feedback redundant, reducing the steps taken to control the system.

Task 1

The first task is to construct a control system using closed loop feedback with the generated transfer function and a PID operator. As described in the method section, this system that is created will be a closed loop feedback, the proportional integral derivative will control the generated transfer function and the feedback loop will measure the error into the PID function. Once the system is designed in Simulink®, the PID is 'Tuned' to be within tolerance of 5% overshoot and settle time of 2 seconds of the unit step input. Nyquist and Pole Zero maps will be plotted to inspect the system stability and acknowledge the system characteristics that are determined by the plant transfer function and the PID operator.

Using MATLAB as the simulation tool for this assignment, the generated transfer function is inputted into MATLAB's command window to create the transfer function coefficient.

```
>> G= tf([1, [5 5 2]])
G =
      1
-----
 5 s^2 + 5 s + 2
Continuous-time transfer function.
```

Fig. 6

The transfer function (Fig. 6) is entered in order to store it for the next task. Simulink® is selected to design the control system for this task. From Simulink's® library, the PID controller, transfer function, 'add' function, 'step' function and scope are the components selected and connected as shown in Fig. 3 in the method section of this paper, and Fig. 7 shown below.

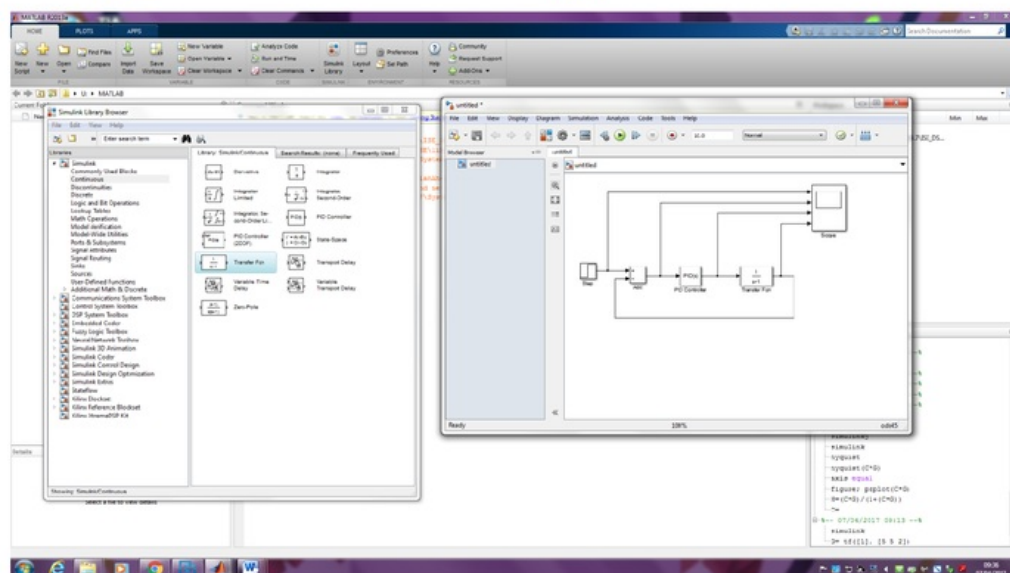


Fig. 7

The 'add' function block signs are then changed to '+' '-' in order to read the input error fed back from the closed loop and the transfer function block is edited for the assigned values given, shown here in Fig. 8 below.

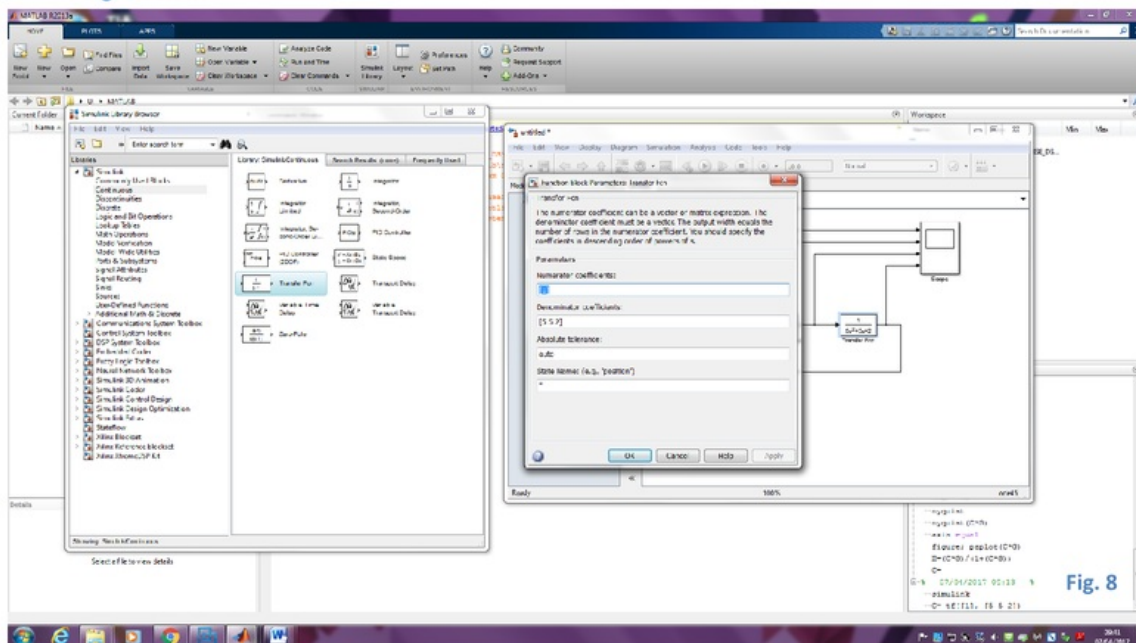


Fig. 8

The PID controller is then modified using the tune function in the PID settings on the block function, shown in Fig. 9.

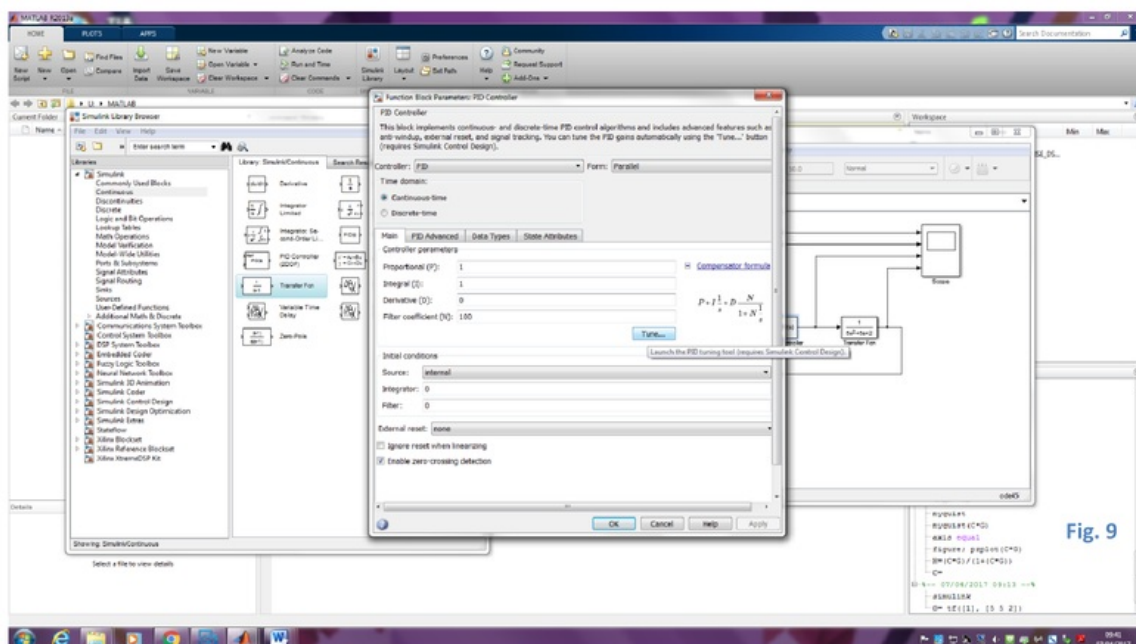


Fig. 9

Adjusting the settle time and overshoot sliders, labelled as 'response time' and 'transient behaviour' brings the system performance within specification of 5% overshoot and 2 seconds settling time, as stated in the assignment instructions. Here the tuner has given 0.798% overshoot and 1.03s settle time.

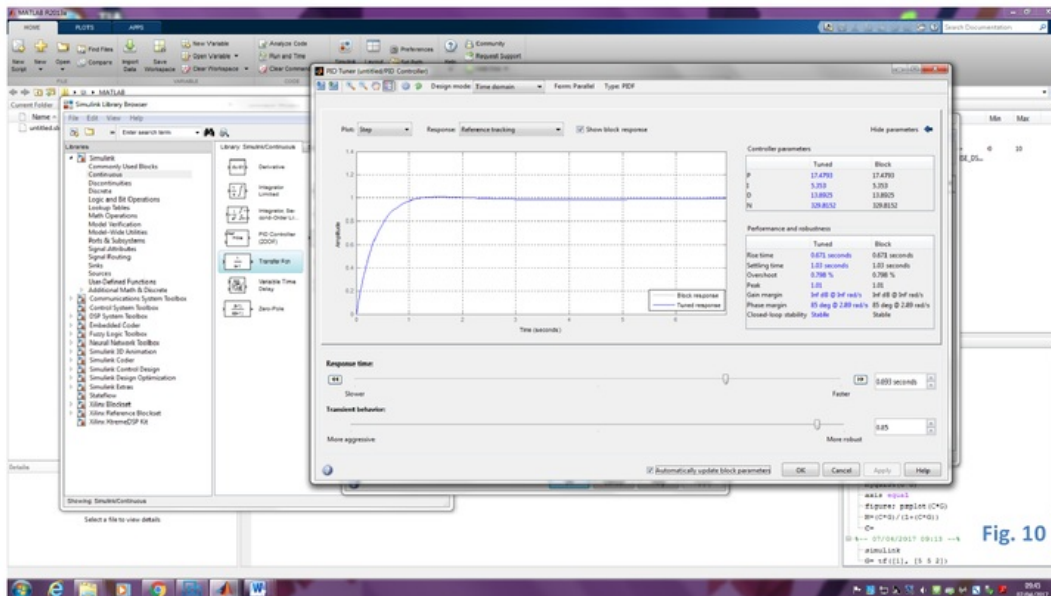


Fig. 10

Once the PID tuner has been adapted to suit specifications, the 'automatically update block parameters' is checked in order to store the settings in the simulation study.

Next phase is to export controller model to workspace, symbolised as:



Fig. 11

This feature exports the PID settings to MATLAB's workspace in order to perform calculations using the PID function, which will be used in Task 2.

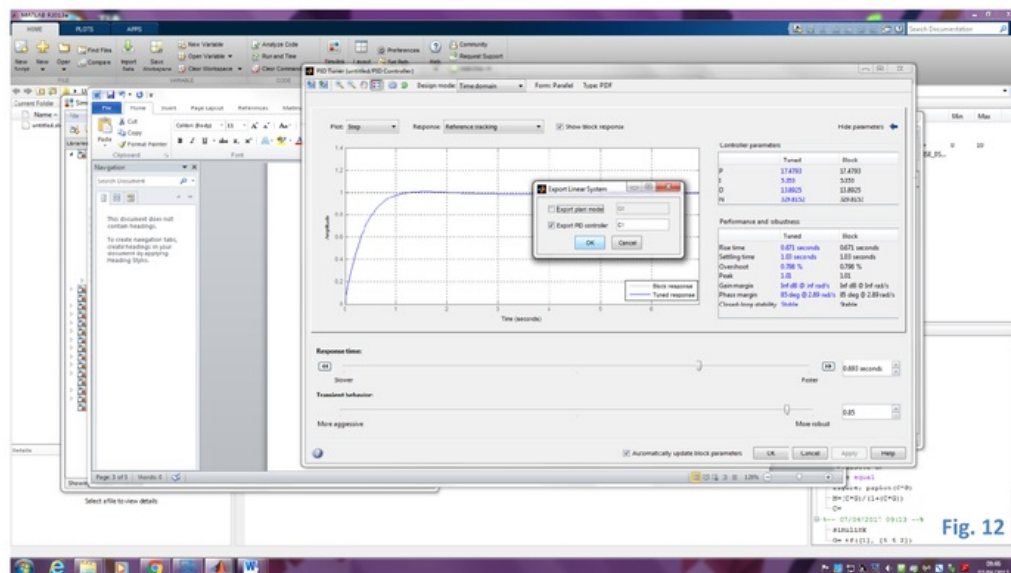


Fig. 12

Once the PID tuner settings are adjusted, updated and exported, the proportional, integral and derivative values are assigned from the tuner, therefore the coefficients are set.

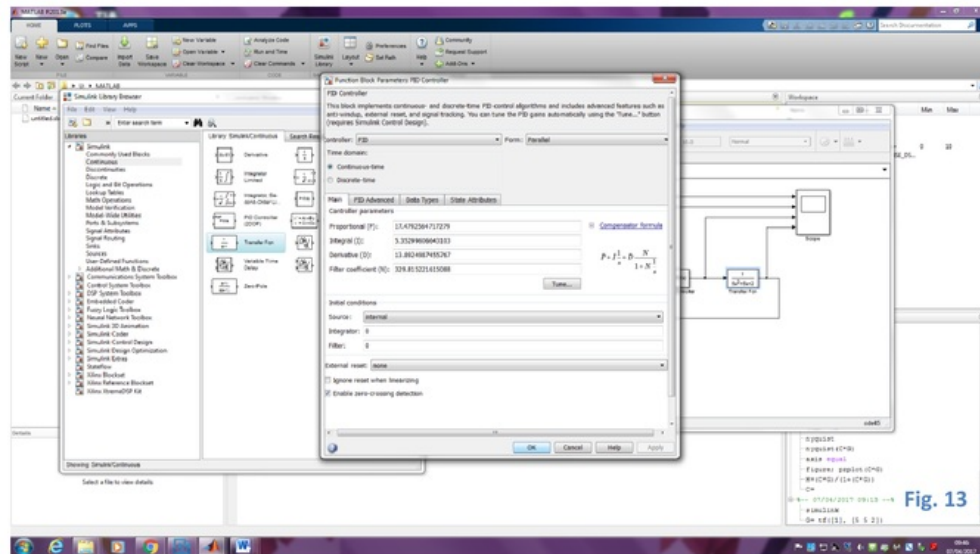


Fig. 13

The next step is to run the simulation and record the results that are plotted on the scope. The scope is recording multiple channels to graph the output from the; step function (Process value set point), add block (error), PID (Plant effort) and the transfer function (Plant output). Graphing this gives a pictorial indication of how the system is responding and to further compare the results from Task 2 to this.

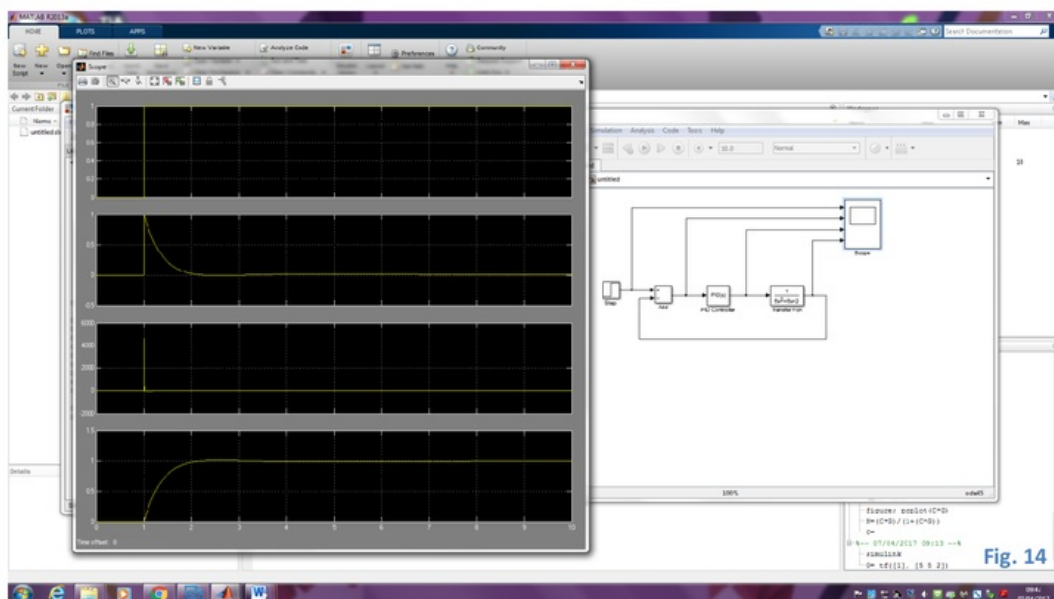


Fig. 14

The next process is to irradiate the PID function.

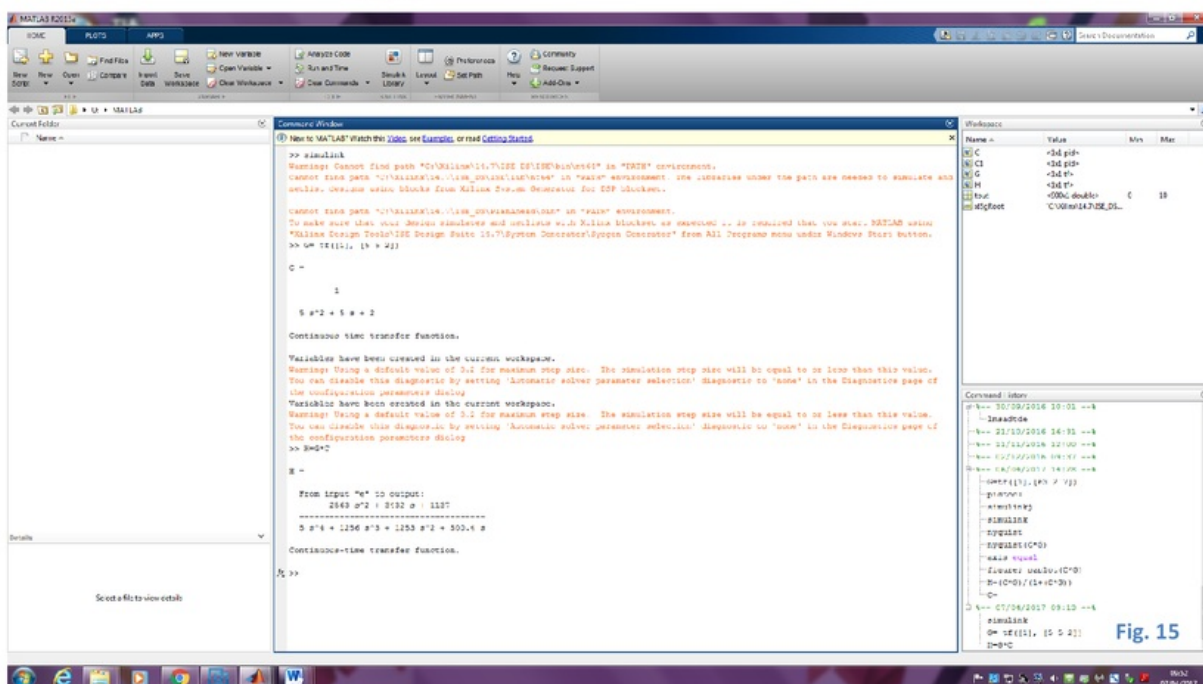
Another indication to evaluate the system performance is to plot a Zero Pole and Nyquist plot. These plots will give an overview of whether the system is stable or unstable in maintaining control.

Using MATLAB's command window to calculate a new transfer function, the PID equation is multiplied by the current transfer functions equation to give a new transfer function that incorporates the PID coefficients.

To gain a new transfer function that will replace the PID controller and the current plant transfer function, they are multiplied in MATLAB's command window. As the PID controllers coefficients settings were exported to the MATLAB's workspace using Simulink's PID tuner in the previous step denoted as the variable 'C'.

In MATLAB's command window, the new transfer function, in this case, denoted as 'H' is generated by typing the sum;

$$H=G*C$$



The new transfer function is now:

$$H = \frac{2563 s^2 + 3432 s + 1137}{5 s^4 + 1256 s^3 + 1253 s^2 + 500.4 s}$$

In order to ascertain whether the new transfer function is stable and capable of controlling the new system without the PID controller, a Pole Zero and Nyquist plot is to be graphed to view its response characteristic. Using MATLAB's command window, 'pzplot(H)' and 'nyquist(H)' is entered that MATLAB will automatically plot on a separate graph.

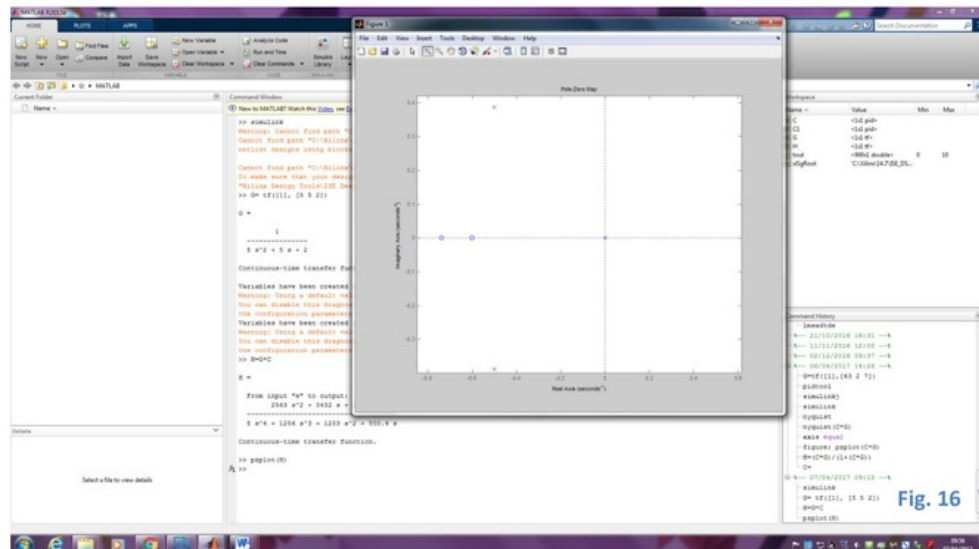
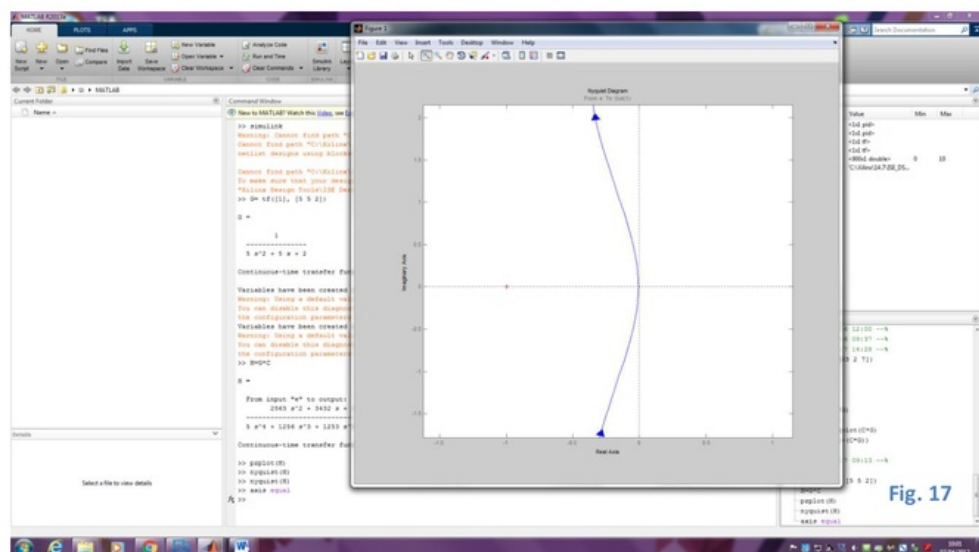


Fig. 16 is the plot of the Pole Zero map of the new transfer function; the zeroes appear to be plotted on the real axis line.



And shown here in Fig. 17, the Nyquist plot shows the phase and gain margin for the newly acquired transfer function for the second phase of the system design.

Task 2

Task 2 in this assignment is to apply the newly generated transfer function to the existing system model by editing the transfer function coefficient settings, using the values that MATLAB has calculated. Then remove the PID controller altogether to that the new transfer function takes control over what the PID controller did in the previous task.

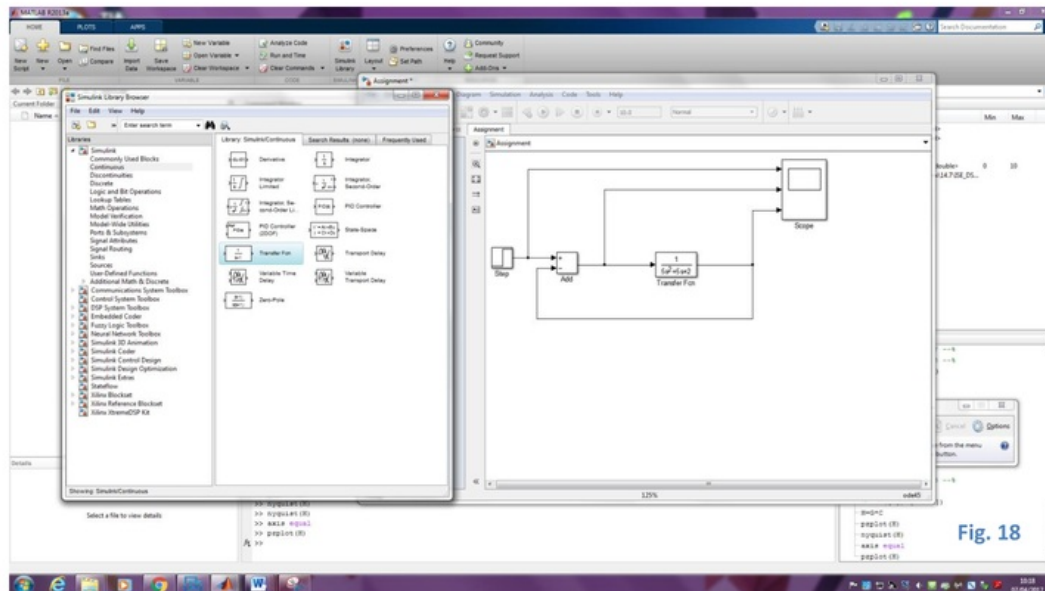


Fig. 18

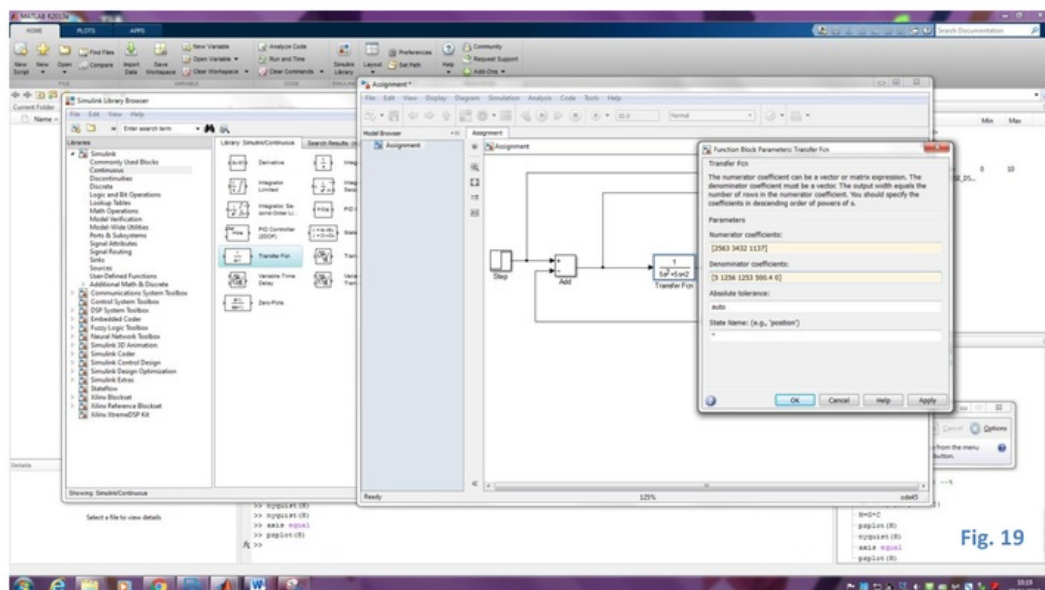


Fig. 19

Fig. 18 & Fig. 19 show the process of removing the PID controller & editing the transfer function block model with the new coefficients. When entering the denominator coefficients, a zero must be

added after where there is no digit after the 'S' variable. Note that the other coefficient values do not require s^2 or s^3 , as the block settings for the transfer function block is already in that format.

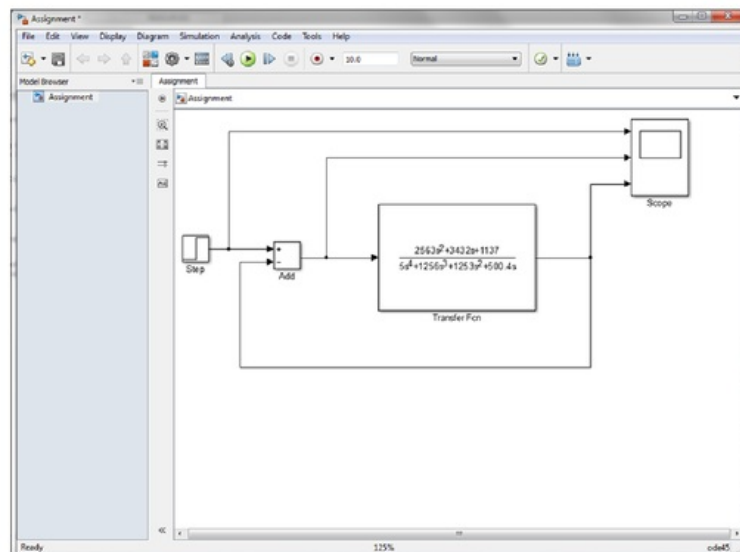


Fig. 20 here shows the overview of the second version of the control system. **Note;** the transfer function block contains the new coefficients.

Fig. 20

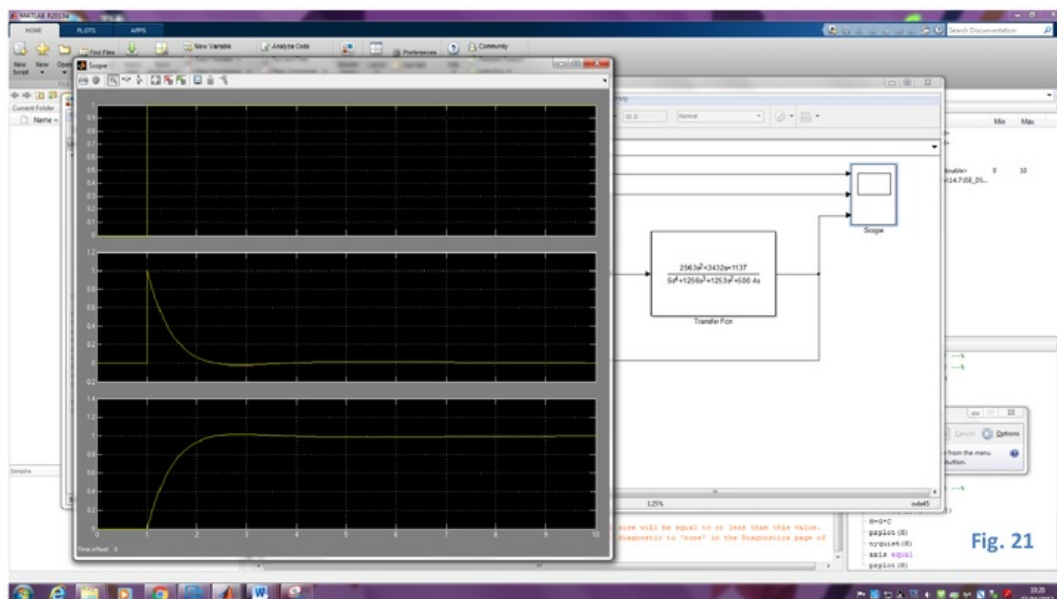


Fig. 21

Running the simulation shown here in Fig. 21 displays the system response to the unit step input. The top waveform shows the unit step input, the second is the proportional error and the third is the plant transfer response toward the unit step input. The waveform results of this version of controller system compared to the first in Task 1 appear to be identical, in that, they have an identical profile of the plant transfer function output.

The next task is to further reduce the system components in the model and compare its performance to that of **Task 1** and the previous system in **Task 2**. The method that will be carried out in this simulation will be working from the model in [Fig. 20](#). The method will be to remove the 'add' block function, making the closed loop feedback part of the system redundant, rendering this system an open loop system. For this, a new version of transfer function needs to be calculated again.

The transfer function for this simulation model will be labelled as '**H2**'. Using the transfer function from the previous model (**H**), **H2** will be generated from this by using the calculation shown below in order to remove the closed loop function. **H2** will be calculated in MATLAB's command window as follows:

$$H2 = H/(1+H)$$

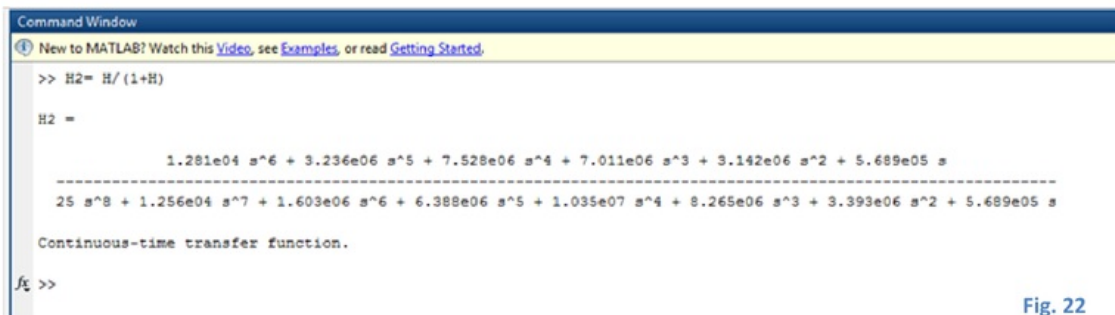


Fig. 22

Repeating the previous steps to obtain **H2** transfer function, a longer coefficient transfer function is noted down before the next step of modifying the simulation model, shown here in [Fig. 22](#).

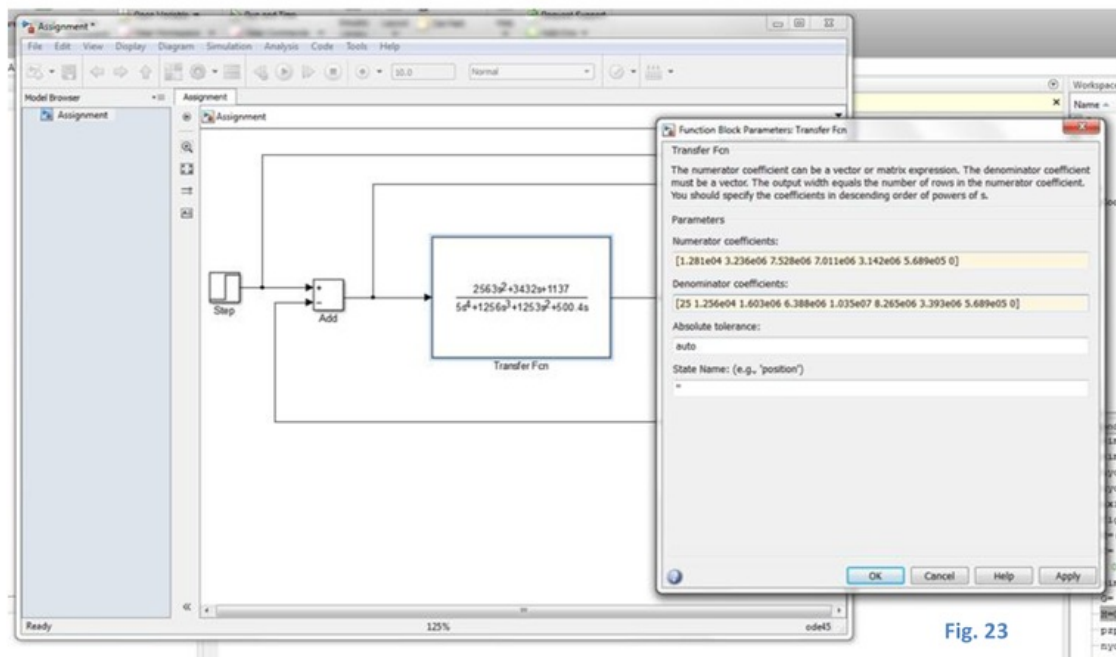


Fig. 23

Fig. 23 above is the same process as the previous model for entering the numerator and denominator coefficients.

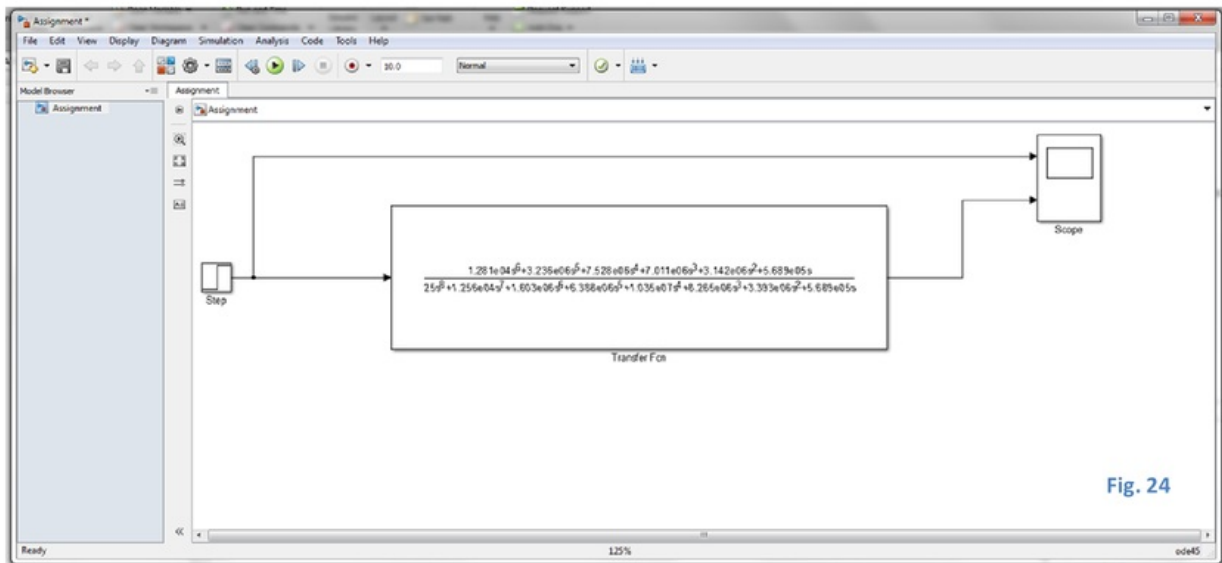


Fig. 24

Fig. 24 above is the layout of the simulation model containing the new transfer function 'H2' and the proportional error 'add' block and closed loop feedback deleted.

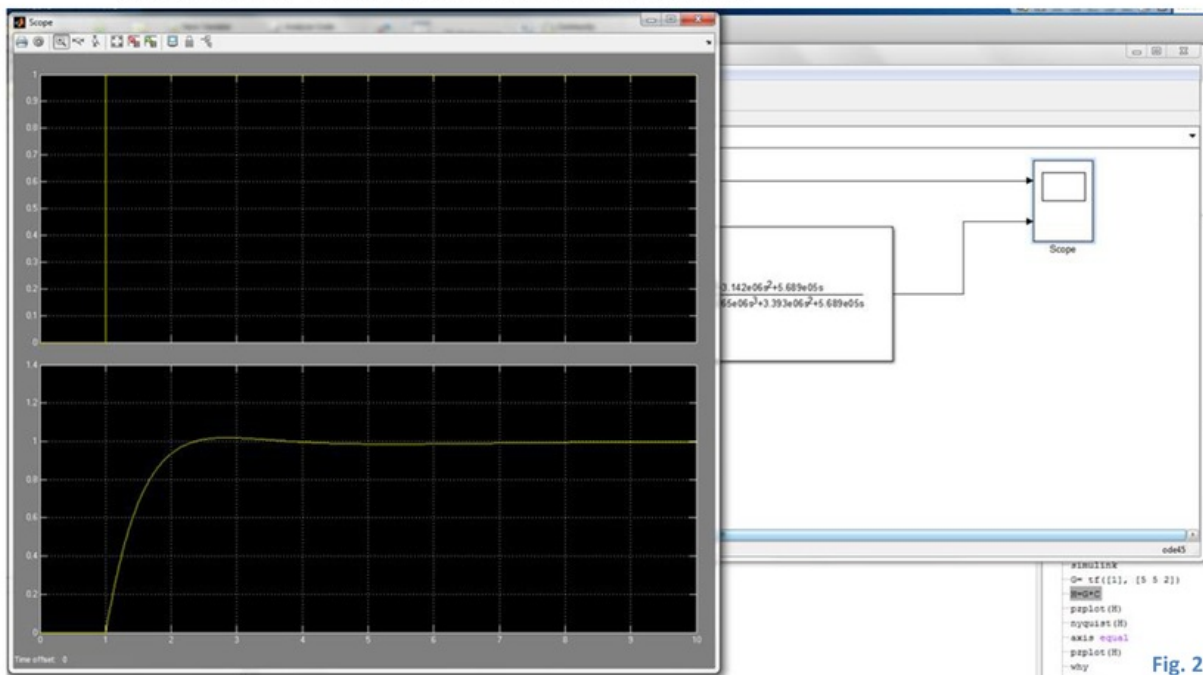


Fig. 25

Running the simulation shows the scoped results for **H2** transfer function. There are only two axes of waveform results now due to the redundancy of the closed loop feedback error margin.

To compare the performances of all three simulation model systems, a side-by-side comparison will need to be made to assess the similarities or variations between the three variants of the control system.

The next task conducted is to plot all three systems into one Simulink workspace in order to run the simulation simultaneously.

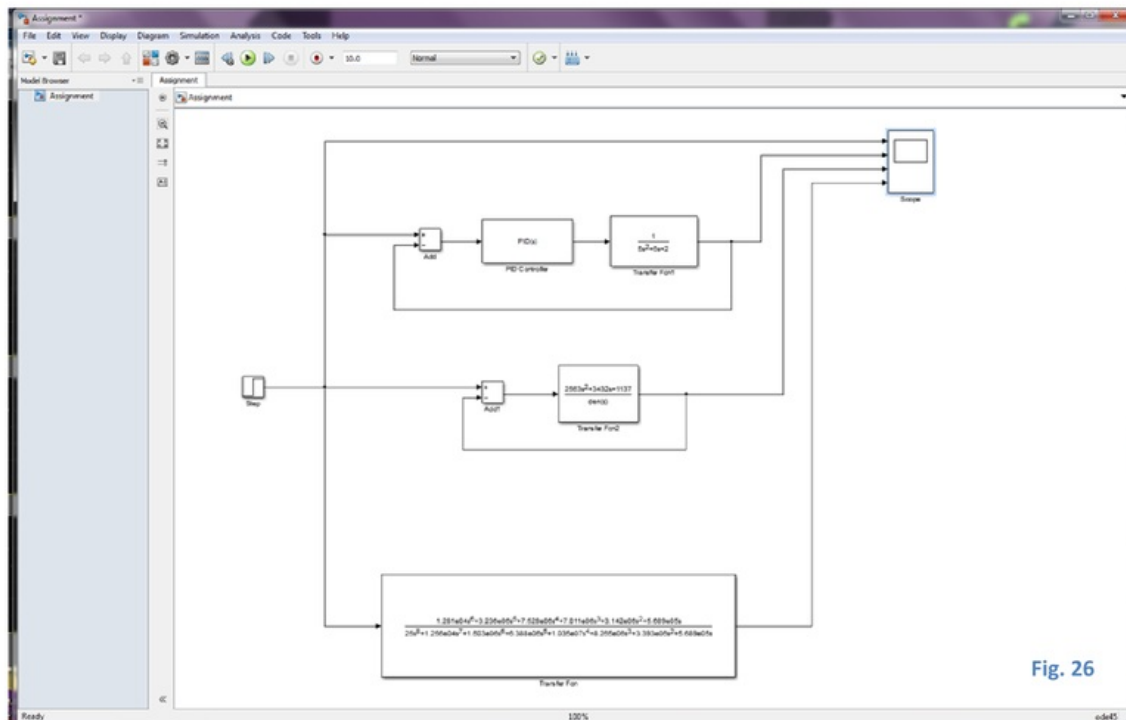
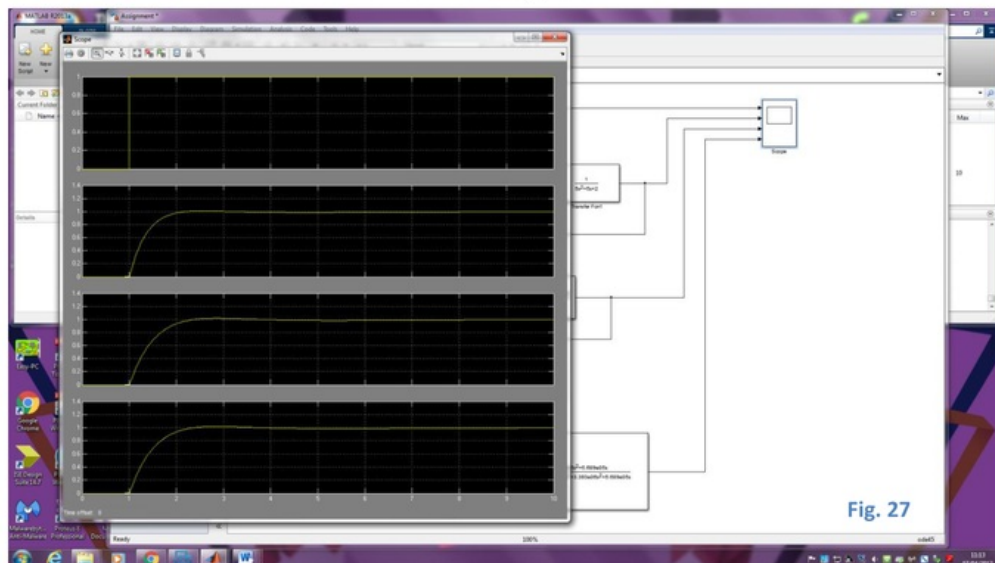


Fig. 26

Fig. 26, shown above is the model design of all three control systems that is controlled by one unit step input to all three systems. The scope will graph the step input, and all three outputs of each control systems.

Results

The results graphed here in Fig. 27 show at a glance that all three system appear to be responding in a similar fashion within the overshoot and settling time.



Comparing the results taken from the simulation, Fig. 28 graphically shown below are the outputs from each of the three variations of the control systems. The top pane displayed is the unit step input into each of the controllers for the desired setpoint; the second is the system that comprises of

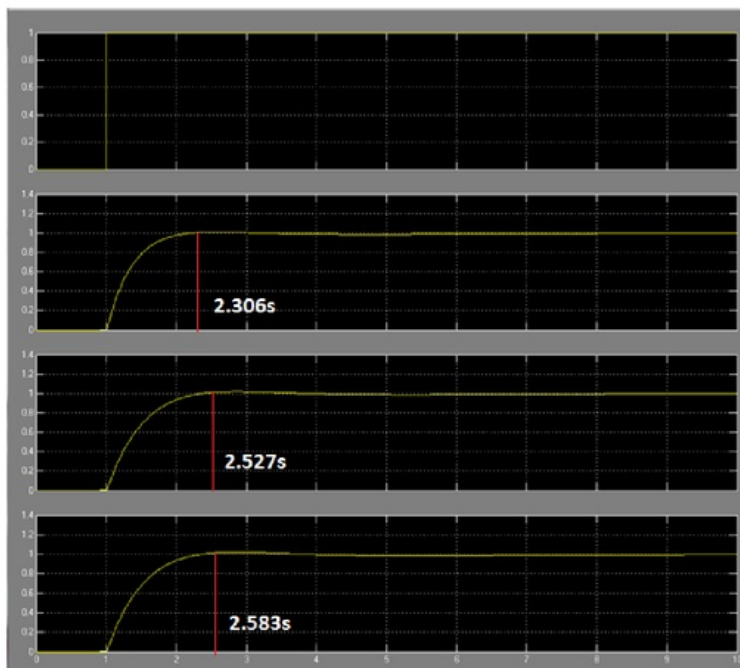


Fig. 28

the PID controller, transfer function and the proportional error from Task 1. The third pane is the control system to where the PID controller is redundant and the forth pane is the controller system with just the transfer function with no proportional error feedback.

The results shown below do vary in time to reach the desired setpoint. The times have been labelled on each system, and as the results show, removing the PID controller does affect the performance to reach the set point. It can be visibly seen that it takes longer to reach the set point value.

Discussion

The results taken from the Simulink experiment shows that the system is at its most efficiency when it includes the PID controller. The explanation for the faster settle and set point reach time can be factored down to the PID controller, which calculates the real time error (proportional) past errors (integral) and corrects the errors in advance by the derivative mode.

Although the control systems in **Task 2**, has the generated transfer functions to replace the PID controller & proportional feedback loop respectively, they are stable systems as proved using the pole zero and Nyquist plots. An indication to an unstable system would give results of sporadic oscillations and unnecessary overshoots, and not to mention long settling time. But the Nyquist plot proves the systems in **Task 2** are stable as the phase and gain margins are plotted within the -1 on the real x-axis line.

The overshoot stability of all three control systems are next to identical, with very little overshoot. The results given from the simulation are surprising and unexpected, as some form of oscillation and minor overshoot was expected, although the Nyquist and pole zero plots back the results in [Fig. 28](#) that prove they are stable systems.

Conclusion

To conclude this experiment, the results show that using the PID controller method decreases the time response to the program set point, although just using generated transfer function, which was generated from the system with the PID feedback loop works for a basic control system that is not reliable on time settling is adequate, the PID controller is beneficial for systems that have multiple process values or set points, as a simple PID derived transfer function would only be stable for a system with a singular set point value, as in the case of this experiment, the set point value consists of a singular set point, and the generated transfer functions were derived from this.

Looking at the results gained from the simulation, the maximum time deviation is 0.277s, or 277ms, which in the terms of the assigned tolerances is very minimal as the settling time is 2 seconds.

References

[1] Proportional Definition <http://dictionary.cambridge.org/dictionary/english/proportional> - Accessed: 23/5/17 11:30

[4] Derivative Dampener Quote <http://www.controleng.com/search/search-single-display/understanding-derivative-in-pid-control/4ea87c406e.html> - Accessed: 24/5/17 12:00

Figures

Fig. 1 – CAL 9500P Controller

<http://www.west-cs.com/products/models/9500-programmable-temperature-controller/> Accessed: 3/6/17 10:00

Fig. 2 – Fig. 28 'Instrumentation & Control' Assignment 1 Simulink® Screenshots by C.Budden

Bibliography

[2] Proportional literature research <http://www.controleng.com/search/search-single-display/understanding-derivative-in-pid-control/4ea87c406e.html> - Accessed: 30/5/17 09:28

[3] Proportional literature research <https://www.electrical4u.com/transfer-function/> Accessed: 30/5/17 10:45

[5] Derivative literature research <http://blog.opticontrols.com/archives/153> - Accessed: 31/5/17 11:15

[6] Slow Cook Off STANAG 4382 Test Procedure Standard

<http://standards.globalspec.com/std/357757/nato-stanag-4382> Accessed 08/06/17 22:00

[7] Slow Cook off STANAG Testing literature research

<https://webcache.googleusercontent.com/search?q=cache:3QfIR3jHZhYJ:https://www.sto.nato.int/publications/STO%2520Educational%2520Notes/STO-EN-AVT-214/EN-AVT-214-02.pdf+&cd=1&hl=en&ct=clnk&gl=uk> - Accessed: 3/6/17 13:40

Instrumentation & Control Assignment 1

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

54/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

GRADING FORM: SMU GENERIC_GJ

CHRISTOPHER BUDDEN**54****RESEARCH-INFORMED LITERATURE**

Inclusion of a range of research-informed literature, including sources retrieved independently.

2.75

Referencing conventions mostly consistently applied. Your reference list should not be split into three. You have cited items from your bibliography in your text - these items should have been in your references not your bibliography. The fig 1 you have listed in your 'figures' list should also be included in the references. The screenshots listed at the end should not be there at all - as it is your work, it is assumed you captured them.

KNOWLEDGE AND UNDERSTANDING

Knowledge is accurate with a good understanding of the field of study.

16.5**ANALYSIS**

Evidence of some logical, analytical thinking and



synthesis. Can analyse new and/or abstract data and situations without guidance. An emerging awareness of different stances and ability to use evidence to support the argument. Valid conclusions and recommendations, where relevant.

16.5

PRACTICAL APPLICATION AND DEP.



A good and appropriate application of standard methods, materials, tools and/or techniques. Good appreciation of the context of the application.

13.75

SKILLS FOR PROFESSIONAL PRAC.



Can communicate in a suitable format but with some room for improvement. - Figure captions should be placed directly beneath the figure and a brief description of the figure should be given.

4.5