STUDY OF TWO INPUT TWO OUTPUT SYSTEM WITH PID CONTROLLER AND DECOUPLER DESIGN

Thesis submitted in partial fulfilment of the requirements for the degree

of

Bachelor of Technology
in
Instrumentation Engineering
by
Sudhanshu Mishra (Gau-C-11/L-208)
Labanya Baruah(Gau-C-11/151)
RimpyChetia (Gau-C-11/L-205)

Under the guidance of
Mr.Rajesh Kondareddy
Assistant Professor
Dept. of IE, CIT Kokrajhar

DEPARTMENT OF INSTRUMENTATION ENGINEERING

केन्द्रीय प्रौद्योगिकी संस्थान कोकराझार

CENTRAL INSTITUTE OF TECHNOLOGY KOKRAJHAR

(A Centrally Funded Institute under Ministry of HRD, Govt. of India)

BODOLAND TERRITORIAL AREAS DISTRICTS :: KOKRAJHAR

:: ASSAM :: 783370

Website: www.cit.ac.in

MAY 2015

DECLARATION

We hereby declare that the work that is being presented in the B. Tech project report entitled "**STUDY OF TWO INPUT TWO OUTPUT SYSTEM WITH PID CONTROLLER AND DECOUPLER DESIGN**" in the partial fulfillment of requirement for the award of Bachelor of Technology in Instrumentation Engineering department, C.I.T Kokrajhar is an authenticated work carried out by our group under the guidance of **Mr.Rajesh Kondareddy** sir and this work is nowhere submitted for similar purpose accepting in Instrumentation Department, Central Institute of Technology Kokrajhar.

Date:........................

(SUDHANSHU MISHRA)
University Roll: Gau-C-11/L-208
University Registration No: 081876of
the year 2011-12

(LABANYA BARUAH)
University Roll: Gau-C-11/151
University Registration No: 014792 of
the year 2011-12

(RIPY CHETIA)
University Roll: Gau-C-11/L-205
University Registration No: 081859 of
the year 2011-12

DEPARTMENT OF INSTRUMENTATION ENGINEERING
CENTRAL INSTITUTE OF TECHNOLOGYKOKRAJHAR
May 2015

CERTIFICATE OF APPROVAL

This is to certify that the work embodied in this project entitled "**STUDY OF TWO INPUT TWO OUTPUTSYSTEM WITH PID CONTROLLER AND DECOUPLER DESIGN**" submitted by **Sudhanshu Mishra**, **Labanya Baruah,** and **Rimpy Chetia** to the Department of Instrumentation Engineering, is carried out under our direct supervisions and guidance.The project work has been prepared as per the regulations of Central Institute of Technology, Kokrajhar and I strongly recommend that this project work be accepted in partial fulfillment of the requirement for the degree of B. Tech.

HoD Incharge
I.E. Department,
CIT, Kokrajhar

(Mr.R.Kondareddy)
Assistant Professor
Department of Instrumentation Engineering

DEPARTMENT OF INSTRUMENTATION ENGINEERING
CENTRAL INSTITUTE OF TECHNOLOGYKOKRAJHAR
May 2015

CERTIFICATE BY THE BOARD OF EXAMINERS

This is to certify that the project work entitled "**STUDY OF TWO INPUT TWO OUTPUTSYSTEM WITH PID CONTROLLER AND DECOUPLER DESIGN**" submitted by **Sudhanshu Mishra**, **Labanya Baruah** and **Rimpy Chetia** to the Department of Instrumentation Engineering of Central Institute of Technology Kokrajhar has been examined and evaluated. The project work has been prepared as per the regulations of Central Institute of Technology and qualifies to be accepted in partial fulfilment of the requirement for the degree of B. Tech.

Project Co-ordinator

Board of Examiners

ACKNOWLEDGEMENT

It is high privilege for us to express our deep sense of gratitude to all those faculty members who helped us in the completion of the project, especially our guide **Mr. Rajesh Kondareddy** who was always there at hour of need.

Our special thanks to **Mr. Jeet Dutta**, Asst. Professor of Instrumentation Engineering Department for giving us a lots of suggestion in the time of project work.

We also extend our gratitude to **Mr Ganesh Roy**, Asst. Professor of Instrumentation Engineering Department for helping us in the completion the project.

We also thank to our family and all of our friends who helped us by various ways for completing this project.

Sudhanshu

Sudhanshu Mishra

Labanya Baruah.

Labanya Baruah

Rimpy Chetia

RimpyChetia

ABSTRACT

The main aim of the project is to determine the mathematical model of interacting TITO system, decoupler design and PID controller designing of the system. The step response of the interacting TITO system can be can be analyzed using MATLAB Simulink software .Here we compare the transfer function step response of the interacting TITO system with linearized step response. Here it also compare different method of PID controller tuning and compare the step response of the interacting TITO system with different PID value. In interacting TITO system there is a loop interaction between input and output. The loop interaction can minimized by using inverting and non-inverting decoupler. The decoupler is design to minimize the effect of interaction between input1 to output2 and vice versa. .The main advantage of the decoupler with PID controller is to minimize the peak overshoot due to loop.

Keywords –TITO, MATLAB, PID, Interacting, Decoupler, Inverting, Non-Inverting.

vi

CONTENTS

LIST OF FIGURES

TOPIC **PAGE NO**

CHAPTER 1:

CHAPTER 2:

CHAPTER 3:

CHAPTER 4:

CHAPTER 5:

LIST OF TABLES

ABBREVIATIONS

MIMO	Multi-Input-Multi- Output
SISO	Single-Input-Single-Output
RGA	Relative Gain Array
PID	Proportional Integral Derivative Controller
PI	Proportional Integral Controller
det	Determinant
ss	State Space
tf	Transfer Function
PB	Proportional Band
LPH	Litre per Hour
PVC	Polyvinyl Chloride
DAQ	Data Acquisition

CHAPTER 1

INTRODUCTION

1.1. GENERALINTRODUCTION

The basic problem in the process industries is to control the liquid level in tanks and flow between tanks. In vital industries such as petro-chemical industries, paper industries,water treatment industries have the interacting tanks which the processes of chemical or mixing treatment takes place in the process tanks. Hence, the level of fluid in the tanks and interaction between tanks must be controlled.The main objective of this project is to determine the mathematical model of interacting TITO system, decoupler design and PID controller designing of the system. The step response of the interacting TITO system can be analyzed using MATLAB Simulink software .Here we compare the transfer function step response of the interacting TITO system with linearized differential equation step response. This project also compare different method of PID controller tuning and compare the step response of the interacting TITO system with different PID value. In interacting TITO system there is a loop interaction between input and output. The loop interaction can minimized by using inverting and no invertingdecoupler.

For controlling a system first step is to design the mathematical model of the system and study the step response of the system. After that design the controller.The Two input Two output system is a two interacting tank system with two manipulating variable and two controlled variable. In the interacting TITO system the Process dynamics of tank1 affects the dynamics of tank2 and vice versa, because flow rate depends upon the difference between the liquid levels. The decoupler is design to minimize the effect of interaction between input1 to output2 and vice versa. The conventional closed loop tuning method is Ziegler-Nicholas method for determining PID value. But in case such as underdamp system this tuning method is fail to determine the ultimate gain and ultimate period. So one quarter decay ratio modified method can be implemented to determine ultimate period and ultimate gain value. Here Autotuning method and also used for PID controller design and compared the closed loop response of all the different tuning PID value in TITO system.The main advantage of the decoupler with PID controller is to minimize the peak overshoot due to loop interaction in the TITO system.

1.2: OBJECTIVE

The main objective

- To determine the mathematical model of interacting TITO system.
- To decoupler design and PID controller designing of the system

- To compare different method of PID controller tuning and compare the step response of the interacting TITO system with different PID value.

1.3:SYSTEM DESIGN:

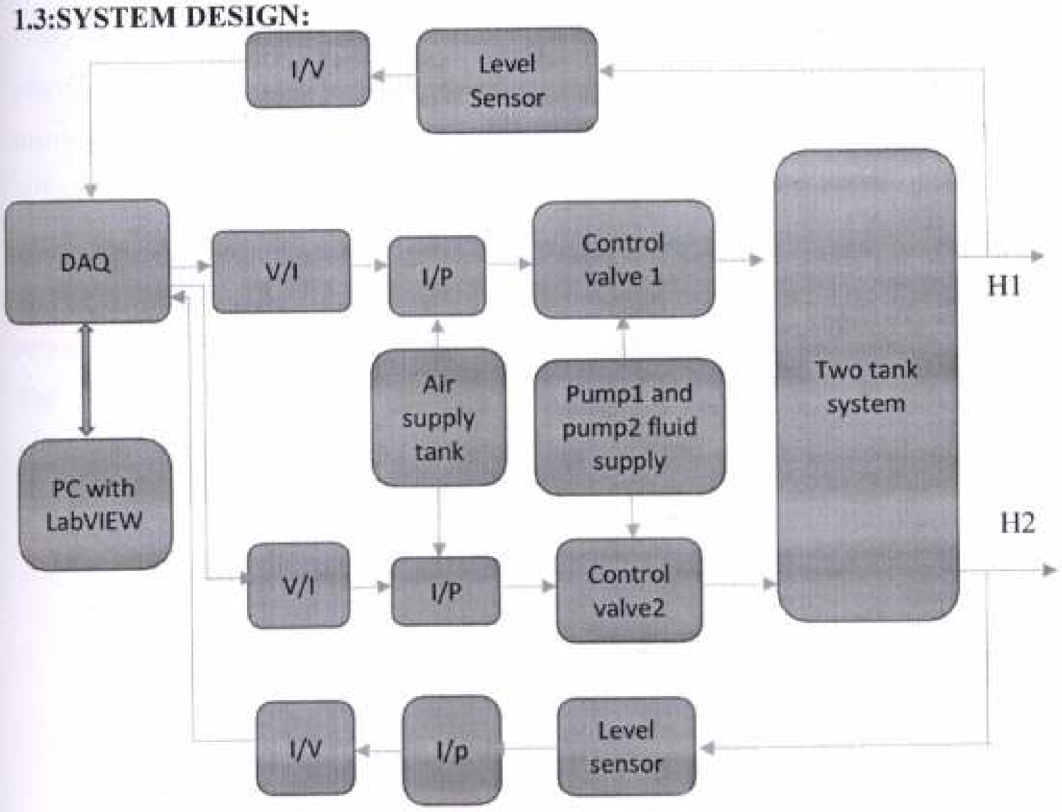


Figure 1.1:Block diagram of Two tank system

Level Sensor Output=4 to 20 mA
I/V Converter Output=5.05 to 7.02 V
V/I Converter Output=4.2 to 20 mA
I/P Converter Output=3 to 15 psig
Air Supply Output= 20 psi
Control Valve Pneumatic Input=3 to 15 psig
Pump Supply=100 lph

In our project we use LabVIEW as a system design platform and development environment for the TITO system. Here it is used for data acquisition. The PC is connected to the DAQ. DAQ is use as interfaces between the signal and a PC. DAQ measures the real world physical condition and convert the data into digital numeric value, which can be manipulated by the PC. It converts the analog values into digital values for processing. The V/I converter convert the output of the DAQ into current in the range of 4.2 to 20 mA. Again the I/P converter converts the current signal into pneumatic pressure, which gives output in the range of 3-15 psig. This pressure is used to control the control valve, then the pressure is used to control the control valve which further controls the two tank system. The output of the system is the level of the tank which is senses by the piezoresistive sensor.

CHAPTER 2
BACKGROUND THEORY

2.1LITERATURE REVIEW:

The overall goal of this report is to establish the significance to control the level of the water and to study the interaction between the tanks in many process industry. The bulk of this report is on critically evaluating the different methodologies used in this field so as to identify the appropriate approach for investigating the project. Here in this report we have collect the study materials from different sources like internet, journals, books.

2.2TITO SYSTEM:

In TITO system there typically Two number of process variables which must be controlled and Two number of variables which can be manipulated.

Let's consider general block diagram of TITO system or 2x2 systems.

Figure 2.1: Block diagram of TITO system

Here $U_1$and $U_2$are the two controlled variables or inputs and $Y_1$and Y_2 are the two outputs or manipulated variables.

According to the above diagram two possible controller pairings can be obtained i.e.

U_1With Y_1,U_2withY_2(1-1 / 2-2 pairing)

U_1WithY_2 , $U_2$with Y_1(1-2 / 2-1 pairings)

2.3: INTERACTING TANK SYSTEM:

Tank1 affects the dynamic behavior of tank2, because the flow rate of tank1 depends on the difference between liquid levels h_1 and h_2 .

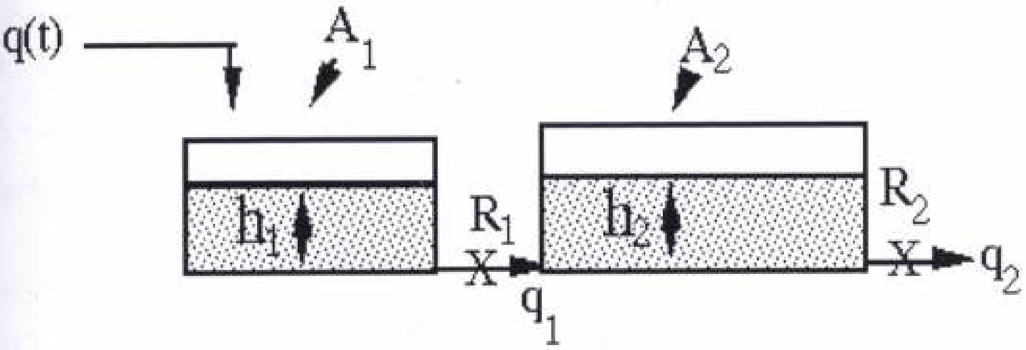


Figure2.2: Schematic Diagram of Interacting tank

System Equations:

For tank 1:

$q - q_1 = A_1 \frac{dh1}{dt}$(2.3.1)

For tank 2:

$q_1 - q_2 = A_2 \frac{dh2}{dt}$(2.3.2)

At tank1:

$q_1 = \frac{h1 - h2}{R1}$(2.3.3)

$q_2 = \frac{h2}{R2}$(2.3.4)

At steady state eq^n (2.3.1) & (2.3.2) can be written as

$qs - q_1 s = A_1 \frac{dh1(s)}{dt}$(2.3.5)

$q_1 s - q_2 s = A_2 \frac{dh2(s)}{dt}$(2.3.6)

now,(2.3.5)-(2.3.1)=>

$q - q_1 - (qs - q_1 s) = A_1 \frac{dh1}{dt} - A_1 \frac{dh1(s)}{dt}$

$\Rightarrow q - q_1 - qs + q_1 s = A_1 \frac{d(h1 - h1s)}{dt}$

$\Rightarrow (q - qs) - (q_1 - q_1 s) = A_1 \frac{d(h1 - h1s)}{dt}$

$\Rightarrow Q - Q_1 = A_1 \frac{dH1}{dt}$(2.3.7)

Again,

(2.3.6)-(2.3.2)=>

$Q_1 - Q_2 = A_2 \frac{dH2}{dt}$ (2.3.8)

Applying L.T in eq^n (2.2.3),(2.3.4),(2.3.7) &(2.3.8) we get,

$Q(s)R_1 = H_1(s) - H_2(s)$

$Q(s)R_2 = H_2(s)$

$Q(s) - Q_1(s) = A_1SH1(s)$

$Q_1(s) - Q_2(s) = A_2SH2(s)$

$$\frac{H2(s)}{Q(s)} = \frac{R2}{A1A2R1R2S^2+(A1R1+A2R2+A1R2)s+1}$$

The transfer functions are:

$$\frac{dh1}{dt} = [Q_{in} - \frac{h1-h2}{R1}]\frac{1}{A1}$$

$$\frac{dh2}{dt} = [\frac{h1-h2}{R1} - \frac{h2}{R2}]\frac{1}{A2}$$

2.4: NON-INTRACTING TANK SYSTEM:

Tank1 feeds tank2 and thus it affects its dynamic behavior, whereas the opposite is not true. Such a system is called non-interacting system.

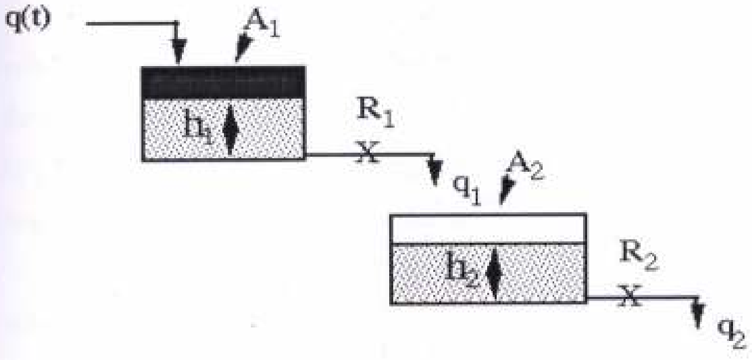


Figure2.3: Schematic Diagram of non-Interacting tank

System Equations:

For tank1:

$q-q_1= A_1\frac{dh1}{dt}$(2.4.1)

For tank2:

$q_1-q_2= A_2\frac{dh2}{dt}$(2.4.2)

$q_1=\frac{h1}{R1}$(2.4.3)

$q_2=\frac{h2}{R2}$(2.4.4)

From eq^n (2.4.1),(2.4.2),(2.4.3)&(2.2.4) we have

$$\frac{dh1}{dt}=\frac{1}{A1}\left(q-\frac{h1}{R1}\right)$$

$$\frac{dh2}{dt}=\frac{1}{A2}\left(\frac{h1}{R1}-\frac{h2}{R2}\right)$$

Combining (2.4.1) & (2.4.3) we get,

$\frac{Q1(s)}{Q(s)}=\frac{1}{R1A1s+1}$ where, $Q_1= q_1-q_2s$, $Q= q-qs$

Combining (2.4.2) & (2.4.4)

$\frac{H2(s)}{Q1(s)}=\frac{R2}{A2R2s+1}$, $H_2=h_2- h_2s$

$$\frac{H2(s)}{Q(s)}=\frac{1}{R1A1s+1}\frac{R2}{R2A2s+1}$$

2.5: TWOINPUTTWO OUTPUTSYSTEM DESCRIPTION:

The TITO system can be shown in the figure 2.4.It consist of two pumps, two control valve, and two tank with level transmitter. The flow of liquid to the tank can be control by the two control valve. The valve coefficient of the valve can be K_1 and K_2. The outlet of the two tank having ball valve having valve coefficient β_1,β_2 and β_{12}. The Q_{in1} andQ_{in2} in the inlet flow rate to the tank1 and tank2 respectively,

$$Q_{in1}= K_1F_1 \text{ and } Q_{in2}= K_2F_2$$

Where,

F1and F_2 flow delivered by the pump 1 and pump 2.

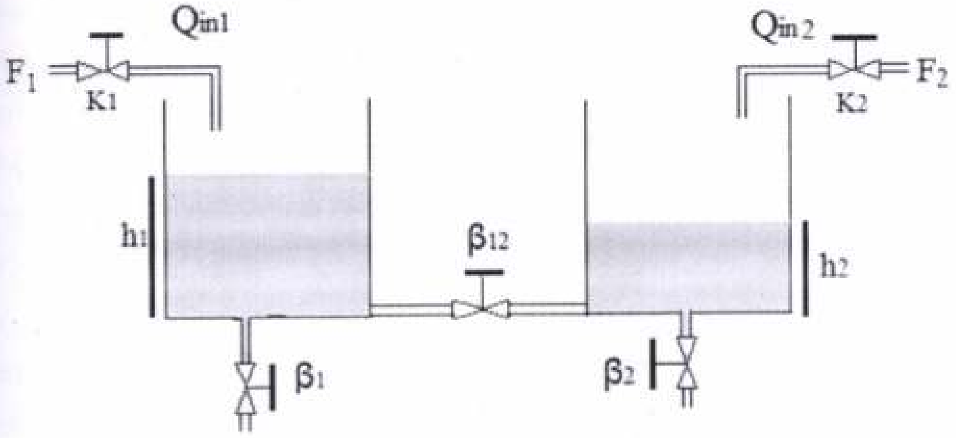
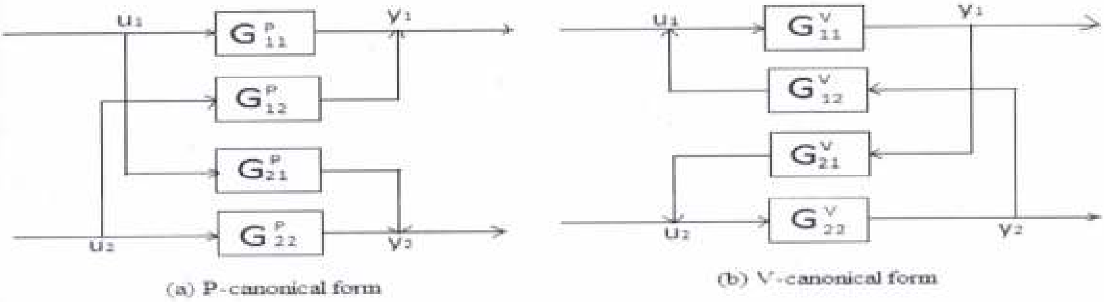


Figure 2.4: Two Input Two Output system

The PI controller is used to control the level of the tank1 and tank2. The feedback signal from the level transmitter is given to the controller by subtracting the set point. The PI controller generate a control signal as per error signal. This control signal control the final control element i.e. Control valve. The main objective of this experiment is to control the level of the tank1 and tank2 i.e. $h_1(t)$ and $h_2(t)$. The manipulating variable of the system is the flow rate of the two pump (F_1 and F_2) . The two PI controller is required for control the level of the tanks of TITO system.

2.6: REPESENTATION OF MULTI VARIABLE SYSTEM:

Input-output models may assume a number of structured forms. Two common (2×2) input output Models of multivariable system are- P and V canonical representation and are shown in figure bellow-



(a) P-canonical form

(b) V-canonical form

Figure2.5:Block diagram of MIMO System Representation

The difference between the two forms is clear from the diagrams with the P-canonical structure, loop interactions are regarded as feed forward coupling whereas in V-canonical structure,loopinteraction are regarded asfeedbackcoupling.

P-Canonical representation:

On a loop basis, the outputs are related to inputs according to-

$Y_1=U_1 G_{11}{}^P+U_2G_{12}{}^P$

$Y_2=U_1 G_{21}{}^P+U_2G_{22}{}^P$

Where the Y_i are the system outputs while the U_i are the manipulative inputs. The above relationship can be expressed more compactly in matrix-vector notation as-

$Y=G^Pu$

Where $y=[y1, y2]^T$ and $u=[u1, u2]^T$ while $G^P=\begin{bmatrix} G11 & G12 \\ G21 & G22 \end{bmatrix}$

V-Canonical representation:

This MIMO representation, in contrast, have a mathematical description given by

$y1=[y2G12+u1]G11$

$y2=[y1G21+u2]G22$

or in matrix vector notation

$y=[I-GmGi]^{-1}Gmu$

with, $Gm=\begin{bmatrix} G11 & 0 \\ 0 & G22 \end{bmatrix}$ and $Gi=\begin{bmatrix} 0 & G12 \\ G21 & 0 \end{bmatrix}$

Relationship between P-and V- representation:

If a system can be modelled using both P- and V-structures, the transfer functions of both structures must be related. The transfer function matrix G^P of the P-canonical form is related to the V-canonical form according to-

$G^P=[I-GmGi]^{-1}Gm$

Provided that the inverse exist.

Choice of representation:

The following factors should be taken into account-

a. It should be possible to determine the parameters of the model from experiments.
b. The model should be simple.

c. The model should be able to provide the relevant information for control system design.

d. The model must be representative of the process, and preferably general enough to encompass other process.

2.8: Loop interaction:

If the loop 2 is closed, then the output depends on input1 (u1).There is an additional effect of imput1 on input2 owing to the action of controller 2.Input 1 affects output2,which then changes input 2 through second controller. This change in input 2 then has an additional effect on the output 1. This effect can positive. So the whole system can show in the figure

$$G_{11,eff}(s)=G_{11}(s)-G_{12}(s)G_{21}(s)G_{c21}(s)+G_{22}(s)G_c(s)$$

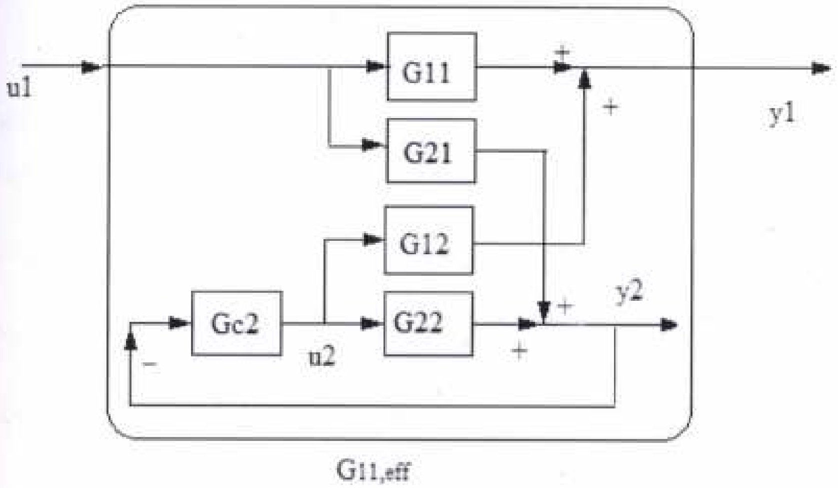


Figure 2.6: loop interaction

Similar affect is obtain when loop 1 is closed.

$$G_{22,eff}(s)=G_{22}(s)-G_{21}(s)G_{12}(s)G_{c1}(s)1+G_{11}(s)G_{c1}(S)$$

2.9: Decoupler Design:

The purpose of de couplers is to cancel the interaction effects between the two loops and thus render two no interacting control loop. This idea is to develop 'synthetic' manipulated inputs that effect only one process output each.

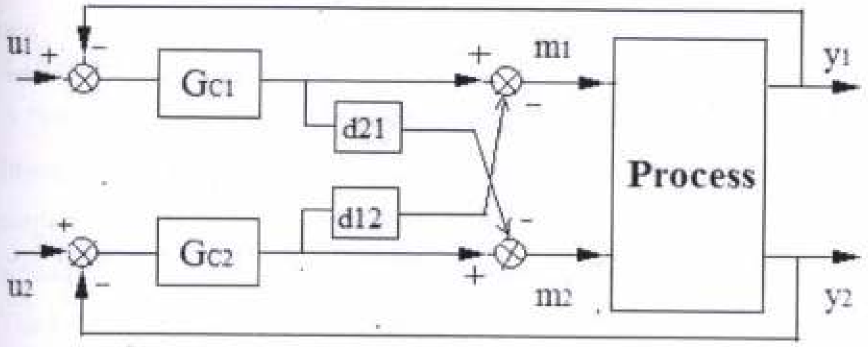


Figure2.7:Decoupler design

The relationship between the synthetic input vector and the process output vector is:-

y(s) = Gp(s)D(s)u(s)

For two input-two output process,

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = G_p(S)D(S)\begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix}$$

Where,

Gp(s)D(s) is a 2 X 2 transfer function matrix.

$$G_p(s)D(s) = \begin{bmatrix} g_{11}(s) & 0 \\ 0 & g_{22}(s) \end{bmatrix}$$

Now, the decoupling matrix is restricted to the form as:

$$D(s) = \begin{bmatrix} 1 & d_{12}(s) \\ d_{21}(s) & 1 \end{bmatrix}$$

Now, from equation (1) we get,

$$\begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix}\begin{bmatrix} 1 & d_{12}(s) \\ d_{21}(s) & 1 \end{bmatrix} = \begin{bmatrix} g_{11}(s) & 0 \\ 0 & g_{22}(s) \end{bmatrix}$$

From the above equation we have,

$$d_{12}(s) = -\frac{g_{12}(s)}{g_{11}(s)}$$

$$d_{21}(s) = -\frac{g_{21}(s)}{g_{22}(s)}$$

2.10 :I/P CONVERTER:

A "current to pressure" transducer(I/P)converts an analog signal(4-20 mA)to a proportional linear pneumatic output(3-15 psig).Its purpose is to translate the analog output from a control system into a precise, repeatable pressure value to control pneumatic actuators/operators, pneumatic valves, dampers, vanes etc.

The I/P converter provides a reliable, repeatable, accurate means of converting an electrical signal into pneumatic pressure in many control system. Models of this device are usually available in direct and reverse action and are field selectable with full or split range input and output as the case may be.

The most common application of an I/P transducer is to receive an electrical signal from a controller and produce a proportional pneumatic output for operating a control valve or positioner. The device can be mounted on the wall or a pipe stand or directly on the valve actuator. Where the device can withstand vibrations, they are directly mounted on the valve actuator. In many instances, the device is remotely mounted on instrument pipe stands to reduce vibration.

Figure 2.8: I/P converter

2.11: DAQ:

Data acquisition is the process of sampling signals that measure real world physical conditions and converting the resulting samples into digital numeric values that can be manipulated by a computer. Data acquisition systems (abbreviated with the acronym DAS or DAQ) typically convert analog waveforms into digital values for processing. The components of data acquisition systems include:

- Sensors that convert physical parameters to electrical signals.
- Signal conditioning circuitry to convert sensor signals into a form that can be converted to digital values.
- Analog-to-digital converters, which convert conditioned sensor signals to digital values.

Data acquisition applications are controlled by software programs developed using various general purpose programming languages such as LabVIEW, BASIC, C, Fortran, Java,Lisp, Pascal. Stand-alone data acquisition systems are often called data loggers.

DAQ hardware is what usually interfaces between the signal and a PC. It could be in the form of modules that can be connected to the computer's ports (parallel, serial, USB, etc.) . Usually the space on the back of a PCI card is too small for all the connections needed, so an external breakout box is required. The cable between this box and the PC can be expensive due to the many wires, and the required shielding.

DAQ cards often contain multiple components (multiplexer, ADC, DAC, TTL-IO, high speed timers, RAM). These are accessible via a bus by a microcontroller, which can run small programs. A controller is more flexible than a hard wired logic, yet cheaper than a CPU so that it is permissible to block it with simple polling loops.

Specialized DAQ software may be delivered with the DAQ hardware. Software tools used for building large-scale data acquisition systems include EPICS. Other programming environments that are used to build DAQ applications include ladder logic, Visual C++, Visual Basic, and LabVIEW and MATLAB.

2.12: PIEZORESISTIVE SENSOR:

A transducer which converts variations in mechanical stress into an electrical output, it consists of an element of piezoresistive materialthat is connected to a Wheatstone bridge circuit and is placed on a highly stressed part of a suitable mechanical structure, usuallyattached to a cantilever or other beam configuration. Pressure sensors can also be used to indirectly measure other variables such as fluid/gas flow, speed, water level, and altitude. Pressure sensors can vary drastically in technology, design, performance, application suitability and cost. A conservative estimate would be that there may be over 50 technologies and at least 300 companies making pressure sensors worldwide.

2.11:Mathematical Model of TITO system:

Consider the two tank having cross sectional area A cm^2. The outlet of the tanks having cross sectional area is unity cm^2.The flow rate to the tank Q_{in1} cm^3/min and Q_{in2} cm^3/min to tank1 and tank2 respectively.

Now mass balance equation:

Rate of change of total mass of fluid inside the tank = Mass flow rate of fluid in to the tank – Mass of flow rate of fluid out of the tank

$$\rho A\frac{dh1(t)}{dt}=\rho Q_{in1}(t)-pq_{12}(t)-pq_1(t)$$

$$=> A\frac{dh1(t)}{dt}=Q_{in1}(t)-q_{in1}(t)-q_1(t)$$

Now

$$q(t)=\frac{cv\pi v^2}{4}\sqrt{2g\Delta h}$$

$$q(t)=\beta\sqrt{h(t)}$$

β = constant $h_1(t)$

For tank 1

$$A\frac{dh1(t)}{dt}=Q_{in1}(t)-\beta_{12}\sqrt{h_1(t)-h_2(t)}-\beta_1\sqrt{h_1(t)}$$

For tank2

$$A\frac{dh2(t)}{dt}=Q_{in2}(t)+\beta_{12}\sqrt{(h1(t)-h2(t))}-\beta_2\sqrt{h2(t)}$$

Now linearized the system,

$$Q_{in1}(t)=K_1F_1$$

$$Q_{in2}(t)=K_2F_2$$

For Tank

$$A\frac{dh_1(t)}{dt}=K_1F_1{}^{*}(t)-\frac{\beta_{12}}{2\sqrt{\left(\bar{h}1-\bar{h}2\right)}}\left\{h_1(t)-h_2(t)\right\}-\frac{\beta_1}{2\sqrt{h_1}}h_1(t)$$

$$\Rightarrow A\frac{dh_1(t)}{dt}=K_1F_1^{*}(t)-\left\{\frac{\beta_{12}}{2\sqrt{\left(\bar{h}1-\bar{h}_2\right)}}+\frac{\beta_1}{2\sqrt{h_1}}\right\}h_1(t)+\frac{\beta_{12}}{2\sqrt{\left(\bar{h}1-\bar{h}2\right)}}h_2(t)$$

$$\Rightarrow\frac{dh_1(t)}{dt}=\frac{k_1}{A}F_1^{*}(t)-\left\{\frac{\beta_{12}}{2A\sqrt{\left(\bar{h}1-\bar{h}_2\right)}}+\frac{\beta_1}{2A\sqrt{h_1}}\right\}h_1(t)+\frac{\beta_{12}}{2A\sqrt{\left(\bar{h}-\bar{h}_2\right)}}h_2(t)$$

For Tank 2

$$\frac{dh_2(t)}{dt}=\frac{k_2}{A}F_2^{*}(t)+\frac{\beta_{12}}{2A\sqrt{\left(\bar{h}1-\bar{h}2\right)}}\left\{h_1(t)-h_2(t)\right\}-\frac{\beta_2}{2A\sqrt{h_2}}h_2(t)$$

$$\Rightarrow\frac{dh_2(t)}{dt}=\frac{k_2}{A}F_2^{*}(t)+\frac{\beta_{12}}{2A\sqrt{\left(\bar{h}_1-\bar{h}2\right)}}h_1(t)-\left\{\frac{\beta_{12}}{2A\sqrt{\left(\bar{h}1-\bar{h}2\right)}}+\frac{\beta_2}{2A\sqrt{h_2}}\right\}h_2(t)$$

Now state space representation of the system :

$$\begin{bmatrix} \bar{h1} \\ \bar{h2} \end{bmatrix} = \begin{bmatrix} -\left(\frac{\beta_{12}}{2A\sqrt{(h1-h2)}}+\frac{\beta_1}{2A\sqrt{h1}}\right) & \frac{\beta_{12}}{2\sqrt{(h1-h2)}} \\ \frac{\beta_{12}}{2A\sqrt{(h1-h2)}} & -\left(\frac{\beta_{12}}{2A\sqrt{h1-h2}}+\frac{\beta_2}{2A\sqrt{h2}}\right) \end{bmatrix} \begin{bmatrix} h1\,(t) \\ h2\,(t) \end{bmatrix} + \begin{bmatrix} \frac{K1}{A} & 0 \\ 0 & \frac{K2}{A} \end{bmatrix} \begin{bmatrix} u1\,(t) \\ u2\,(t) \end{bmatrix}$$

The system can be linearized around the operating point and we get the transfer stste space of the system in equation 2.11.1

$h_1 = 15\text{cm}$, $h_2 = 8\text{cm}$, $r=5$ cm

$\beta_1 = 40$, $\beta_{12} = 55$, $\beta_2 = 80$, $K_1 = 20$, $K_2 = 15$

$A = \pi r^2$

$= 3.14 \text{ X } 5^2$

$= 78.5$

$$\begin{bmatrix} \bar{h1} \\ \bar{h2} \end{bmatrix} = \begin{bmatrix} -0.1981 & 0.1324 \\ 0.1324 & -0.31255 \end{bmatrix} \begin{bmatrix} h1(t) \\ h2(t) \end{bmatrix} + \begin{bmatrix} 0.254 & 0 \\ 0 & 0.19108 \end{bmatrix} \begin{bmatrix} u1(t) \\ u2(t) \end{bmatrix} \qquad (2.11.1)$$

It is a tuning method By using MATLAB command 'ss' and 'tf' we get the transfer function of the system as,

$$G_{11} = \frac{0.254s + 0.07937}{s^2 + 0.5107s + 0.04439}$$

$$G_{12} = \frac{0.03363}{s^2 + 0.5107s + 0.04439}$$

$$G_{21} = \frac{0.0253}{s^2 + 0.5107s + 0.04439}$$

$$G_{22} = \frac{0.19111s + 0.03785}{s^2 + 0.5107s + 0.04439}$$

CHAPTER 3:

SOFTWARE USED

3.1: MATLAB:

MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment and fourth-generation programming language. Developed by MathWorks , MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, Java, Fortran and Python.

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing capabilities. An additional package, Simulink , adds graphical multi-domain simulation and Model-Based Design for dynamic and embedded systems.

MATLAB was first adopted by researchers and practitioners in control engineering, Little's specialty, but quickly spread to many other domains. It is now also used in education, academic and research institutions as well as industrial enterprises. In particular the teaching of linear algebra and numerical analysis, and is popular amongst scientists involved in image processing.

The MATLAB application is built around the MATLAB language, and most use of MATLAB involves typing MATLAB code into the Command Window (as an interactive mathematical shell), or executing text files containing MATLAB code, including scripts and/or functions.

MATLAB can be used interactively and has an inventory of routines, called as functions, which minimize the task of programming even more.

In our project we use the command window of MATLAB to find the TF of the system by using the command – 'ss' and 'tf'. We also use the Simulink block for the simulation.

The proportional term produces an output value that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant K_p, called the proportional gain constant.

The proportional term is given by:

$P_{OUT}=K_p e(t)$

A high proportional gain results in a large change in the output for a given change in the error. If the proportional gain is too high, the system can become unstable. In contrast, a small gain results in a small output response to a large input error, and a less responsive or less sensitive controller. If the proportional gain is too low, the control action may be too small when responding to system disturbances. Tuning theory and industrial practice indicate that the proportional term should contribute the bulk of the output change.

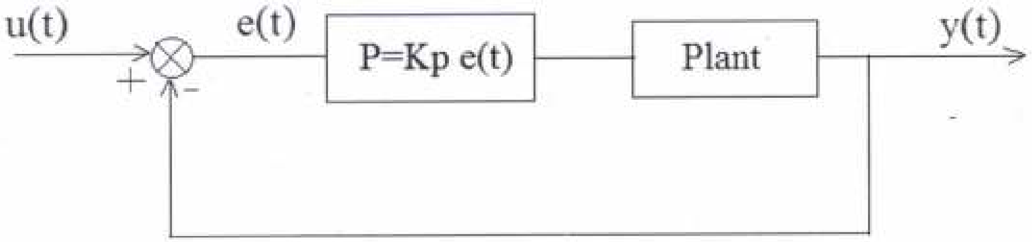


Figure 3.1: Block Diagram of Proportional Controller

3.1.2: INTEGRAL CONTROLLER (I):

The contribution from the integral term is proportional to both the magnitude of the error and the duration of the error. The integral in a PID controller is the sum of the instantaneous error over time and gives the accumulated offset that should have been corrected previously. The accumulated error is then multiplied by the integral gain (K_i) and added to the controller output.

The integral term is given by:

$$I_{out} = k_i \int_0^t e(\tau)d(\tau)$$

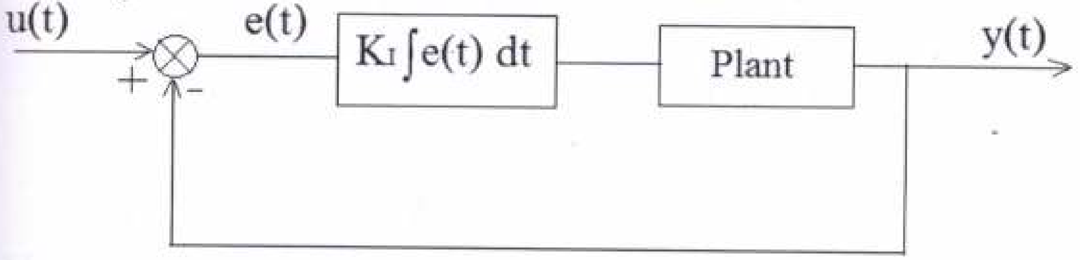


Figure 3.2: Block Diagram of Integral Controller

The integral term accelerates the movement of the process towards setpoint and eliminates the residual steady-state error that occurs with a pure proportional controller. However, since the integral term responds to accumulated errors from the past, it can cause the present value to overshoot the setpoint value.

3.1.3: DERIVATIVE CONTROLLER (D):

The derivative of the process error is calculated by determining the slope of the error over time and multiplying this rate of change by the derivative gain k_d. The magnitude of the contribution of the derivative term to the overall control action is termed the derivative gain, k_d.

The derivative term is given by:

$$D_{out} = K_d \frac{de(t)}{dt}$$

Derivative action predicts system behavior and thus improves settling time and stability of the system. An ideal derivative is not causal, so that implementations of PID controllers include an additional low pass filtering for the derivative term, to limit the high frequency gain and noise. Derivative action is seldom used in practice though - by one estimate in only 20% of deployed controllers - because of its variable impact on system stability in real-world applications.

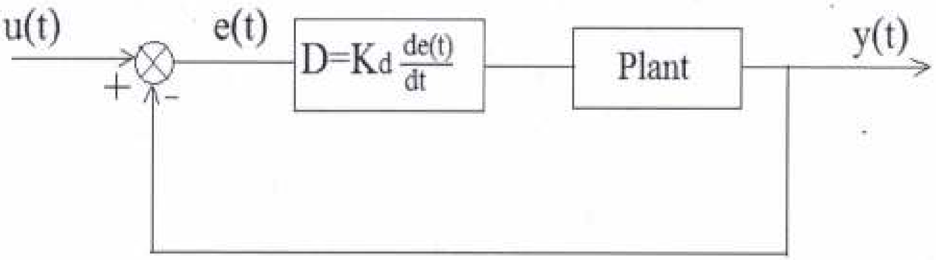


Figure3.3: Block Diagram of Derivative controller

3.1.4: PI-CONTROLLER:

A PI Controller (proportional-integral controller) is a special case of the PID controller in which the derivative (D) of the error is not used.

The controller output is given by,

$K_P\Delta+K_I\int \Delta dt$

Where,

Δ is the error or deviation of actual measured value (***PV***) from the setpoint (***SP***).

$\Delta=SP-PV$

Where

K_P = proportional gain

K_I = integral gain

The lack of derivative action may make the system more steady in the steady state in the case of noisy data. This is because derivative action is more sensitive to higher-frequency terms in the inputs.

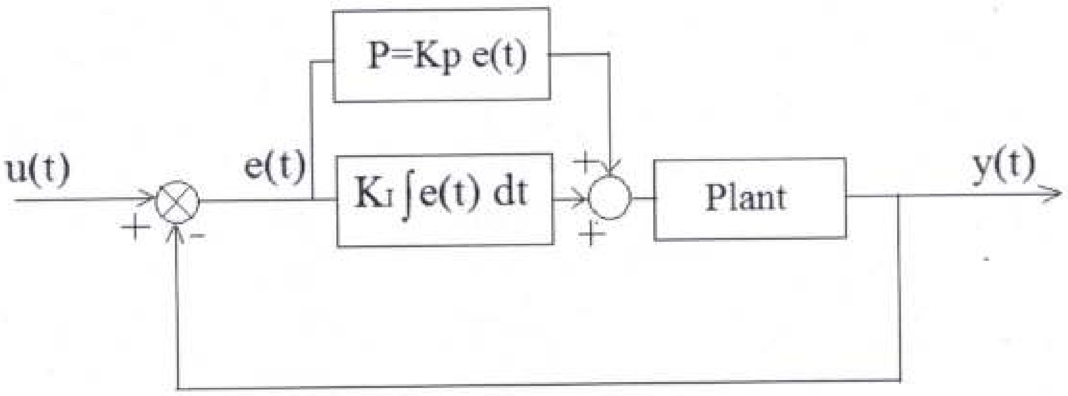


Figure 3.4: Block Diagram of PI Controller

Without derivative action, a PI-controlled system is less responsive to real (non-noise) and relatively fast alterations in state and so the system will be slower to reach set point and slower to respond to perturbations than a well-tuned PID system may be.

3.1.5: SIMULINK:

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

MathWorks and other third-party hardware and software products can be used with Simulink. For example, Stateflowextends Simulink with a design environment for developing state machines and flow charts.

MathWorks claims that, coupled with another of their products, Simulink can automatically generate C source code forreal-time implementation of systems. As the efficiency and flexibility of the code improves, this is becoming more widely adopted for production systems. In addition to being a tool for embedded system design work because of its flexibility and capacity for quick iteration.Embedded Coder creates code efficient enough for use in embedded systems.

Simulink Real-Time (formerly known as xPC Target), together with x86-based real-time systems, is an environment for simulating and testing Simulink and Stateflow models in real-time on the physical system. Another MathWorks product also supports specific embedded targets. When used with other generic products, Simulink and Stateflow can automatically generate synthesizable VHDL andVerilog

Simulink Verification and Validation enables systematic verification and validation of models through modeling style checking, requirements traceability and model coverage analysis. Simulink Design Verifier uses formal methods to identify design errors like integer overflow, division by zero and dead logic, and generates test case scenarios for model checking within the Simulink environment.

The systematic testing tool TPT is marketed as a way to perform a formal verification and validation process to stimulate Simulink models but also for use during the development phase where the developer generates inputs to test the system. By the substitution of the Constant and Signal generator blocks of Simulink, MathWorks claim that the stimulation becomes reproducible.

We also use the Simulink of the MATLAB to simulate our system for different tuning methods. By this simulation we try to control our system for different tuning values. From this simulation result we find some responses.

3.2: LabVIEW SOFTWARE:

LabVIEW (short for Laboratory Virtual Instrument Engineering Workbench) is a system-design platform and development environment for a visual programming language from National Instruments.

The graphical language is named "G" (not to be confused with G-code). Originally released for the Apple Macintosh in 1986, LabVIEW is commonly used for data acquisition, instrument control, and industrial automation on a variety of platforms including Microsoft Windows, various versions of UNIX, Linux, and Mac OS X. The latest version of LabVIEW is LabVIEW 2014, released in August 2014.

3.2.1: Dataflow programming:

The programming language used in LabVIEW, also referred to as G, is a dataflow programming language. Execution is determined by the structure of a graphical block diagram (the LabVIEW-source code) on which the programmer connects different function-nodes by drawing wires. These wires propagate variables and any node can execute as soon as all its input data become available. Since this might be the case for multiple nodes simultaneously, G is inherently capable of parallel execution. Multi-processing and multi-threading hardware is automatically exploited by the built-in scheduler, which multiplexes multiple OS threads over the nodes ready for execution.

Graphical programming:

LabVIEW ties the creation of user interfaces (called front panels) into the development cycle. LabVIEW programs/subroutines are called virtual instruments (VIs). Each VI has three components: a block diagram, a front panel and a connector panel. The last is used to represent the VI in the block diagrams of other, calling VIs. The front panel is built using controls and indicators. Controls are inputs – they allow a user to supply information to the VI. Indicators are outputs – they indicate, or display, the results based on the inputs given to the VI. The back panel, which is a block diagram, contains the graphical source code. All of the objects placed on the front panel will appear on the back panel as terminals. The back panel also contains structures and functions which perform operations on controls and supply data to indicators. The structures and functions are found on the Functions palette and can be placed on the back panel. Collectively controls, indicators, structures and functions will be referred to as nodes. Nodes are connected to one another using wires – e.g. two controls and an indicator can be wired to the addition function so that the indicator displays the sum of the two controls. Thus a virtual instrument can either be run as a program, with the front panel serving as a user interface, or, when dropped as a node onto the block diagram, the front panel defines the inputs and outputs for the given node through the connector panel. This implies each VI can be easily tested before being embedded as a subroutine into a larger program.

The graphical approach also allows non-programmers to build programs by dragging and dropping virtual representations of lab equipment with which they are already familiar. The LabVIEW programming environment, with the included examples and documentation, makes it simple to create small applications. This is a benefit on one side, but there is also a certain danger of underestimating the expertise needed for high-quality G programming. For complex algorithms or large-scale code, it is important that the programmer possess an extensive knowledge of the special LabVIEW syntax and the topology of its memory management. The most advanced LabVIEW development systems offer the possibility of building stand-alone applications. Furthermore, it is possible to create distributed applications, which communicate by a client/server scheme, and are therefore easier to implement due to the inherently parallel nature of G.

CHAPTER 4:

THEORETICAL STUDY

CONTROLLER TUNNING METHOD:

- Ziegler-Nichols Methods(Ultimate Cycle Method)
- Tyreus and Luyben Method
- Autotunning Method
- Modified Tuning Method
- Decoupler design

4.1:ZIEGLER-NICHOLS METHOD (ULTIMATE CYCLE METHOD):

It is a tuning method of a PID controller. This method is implemented by setting the value of integral I and derivative D gains to zero. Then the value of proportional gain P is increased until the system oscillates continuously.

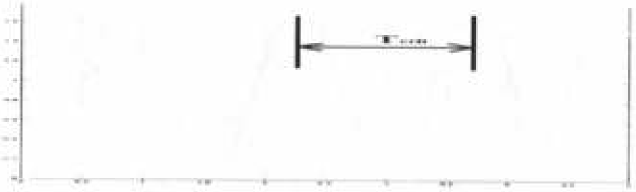


Figure 4.1:oscillation

From the certain oscillation we get the value for $\mathbf{P_{crit}}$and $\mathbf{T_{crit}}$. From the $\mathbf{P_{crit}}$and $\mathbf{T_{crit}}$**PID** parameter can be calculated from the following table

Controller	P	Ti	Td
P	0.5 Pcrit		
PI	0.45 Pcrit	0.833 Tcrit	
PID	0.6 Pcrit	0.5 Tcrit	0.125 Tcrit

Table 4.1: tunning parameter for ultimate cycle method

In the quadruple tank system if we apply the Z-N close loop tuning method, for high value of proportional gain value system does not get oscillation. So this method do not applicable for tuning of quadruple tank system or MIMO system. We apply the modified tuning method.

3.2:TYREUS AND LUYBEN METOHD:

The Tyreus-Luyben tuning method is based on oscillation as in Z-N methods, but with modified formulas for the controller parameters to obtain better stability in the control loop compared with Z-N method. This method is based on ultimate gain (Ku) and period (Pu).

Controller Type	P	Ti	Td
P I	Ku/3.2	2.2 Pu	
P I D	u/2.2	2.2 Pu	Pu/6.3

Table 4.2: tunning parameter for Tyrelus and Luyben Method

for high value of proportional gain value system does not get any oscillation.for high value of proportional gain value system does not get oscillation.

4.3:MODIFIED Z-N METHOD:

This method is same as Ziegler-Nichols closed loop cycling method but the difference is that it can applicable only if we do not get continuous oscillation. This process can be applied on overdamped system.

Procedure of Tuning:

1. Keep the closed loop of the system with PID controller.
2. Make derivative value zero, and integral value 100, or 1000.
3. Increase proportional gain and reached ¼ decay ratio.
4. Measure the Time between the two peak.

Now we get two value

Kcu=Critical Gain

Pc=Period (if Integral vale is 1000 then multiply 10 with Pc)

Modified Z-N PID tuninng Parameter :

Controller	Kp	Ti	Td
P I	0.5Kcu	1.5Pc	
PID	Kcu/1.5	2.5Pc	Pc/4

Table 4.3 Tuning parameter for modified Z-N method

4.4:AUTOTUNING METHOD:

Many process control system have an automatic tuning feature. The operator can simply push the auto tune button and have the controller tune itself, that is determine the values of the tuning parameter.

The basic control for a controller with an autotune relay switch is shown in the fegure. For normal operation, the switch is connected to the PID controller as shown in the figure. When the autotune function is operating, the switch is set to the output of the relay block, as shown in the figure.

The relay block represents a nonlinear function as:

If e < 0, then u = umin

If e > 0, then u = umax

Where, umin = h umax = - h

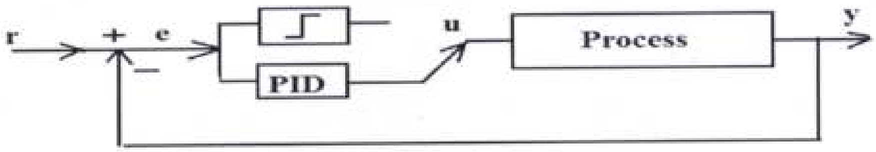


Fig-1

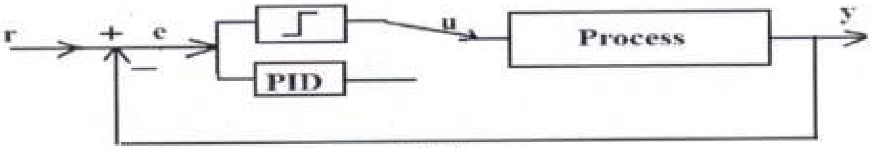


Fig-2

Figure 4.2: Block diagram of autotuning method

In autotune mode, the closed-loop system oscillates and the manipulated variable action is on-off as shown in the figure. In this example h = 0.05.

There are two parameters that results from this autotune test. One is the period P, and the other is the amplitude of the process output a. The period has units of time; the ultimate frequency can be found from

$$\omega u = \frac{2\pi}{P}$$

And the ultimate gain can be found from the amplitude

$$Kcu = \frac{4h}{\pi a}$$

This value of *Kcu*and P can be used to determine the PID parameter from the Ziegler-Nichols and Tyreus-Luyben tuning formula.

3.5:Decoupler Design:

The relative-gain array indicates how the inputs should be coupled with the output to form loops with the smaller amount of interaction.The main objective of decoupling control is to elimanate complicated loop interaction so that a change in one process variable will not cause corresponding changes in other process variable. In decoupler scheme a compensation network called decoupler is used in the right before the process. Decoupler compensation network can be obtain by the following equation.

$$D_{12}(s) = -\frac{G_{12}(s)}{G_{11}(s)} \text{ and } D_{21}(s) = -\frac{G_{21}(s)}{G_{22}(s)}$$

Decoupler structure are two type-Noninverted and Inverted Decoupler. shown in the figure

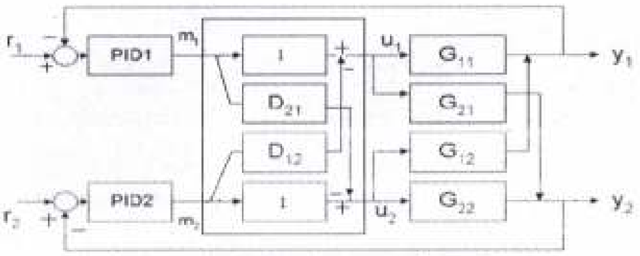


Figure 4.3: Inverted Decoupler

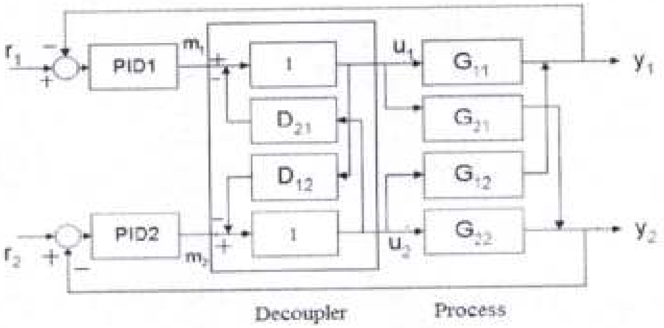


Figure 4.4: NonInverted Decoupler

LabVIEW DESIGN:

FRONT PANEL DIAGRAM

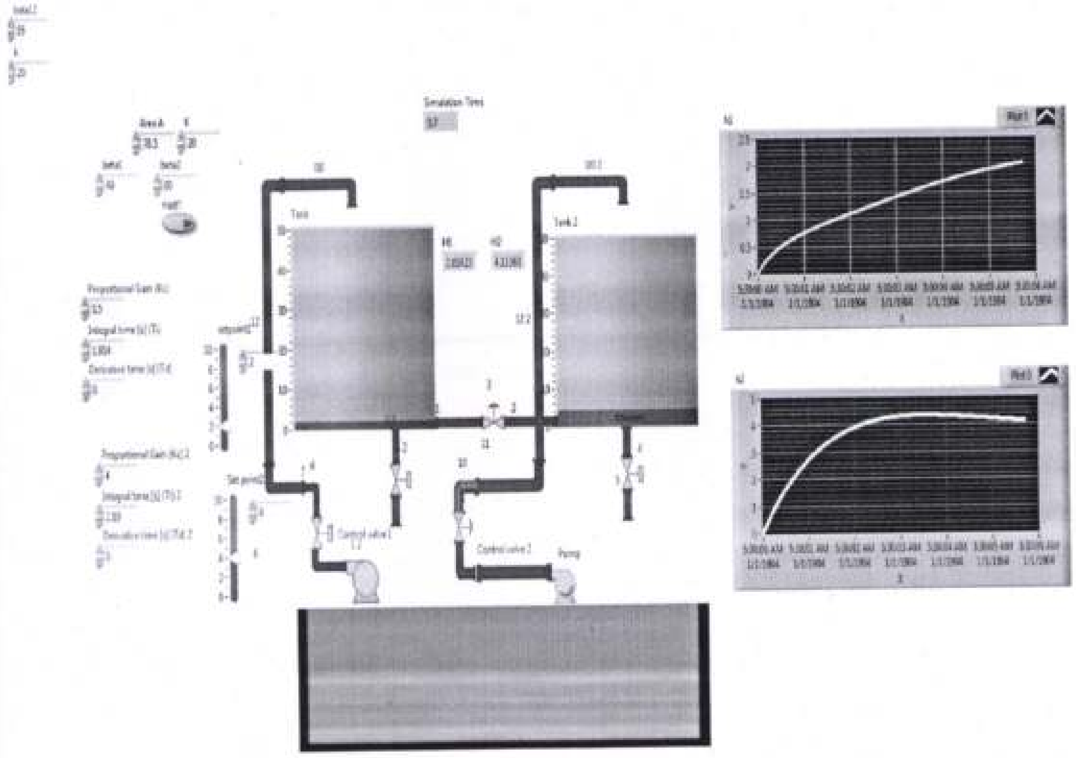


Figure 4.5: Block diagram in front panel

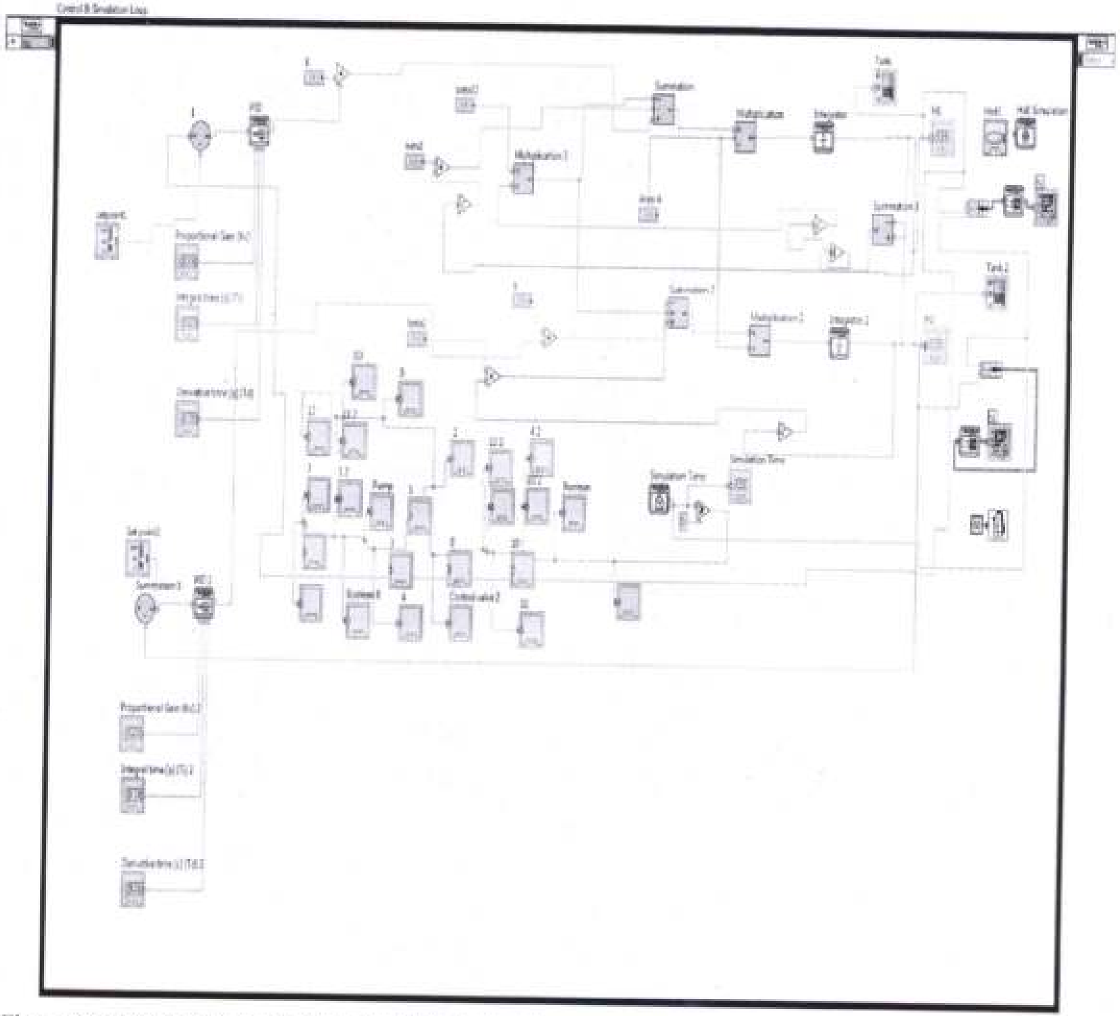
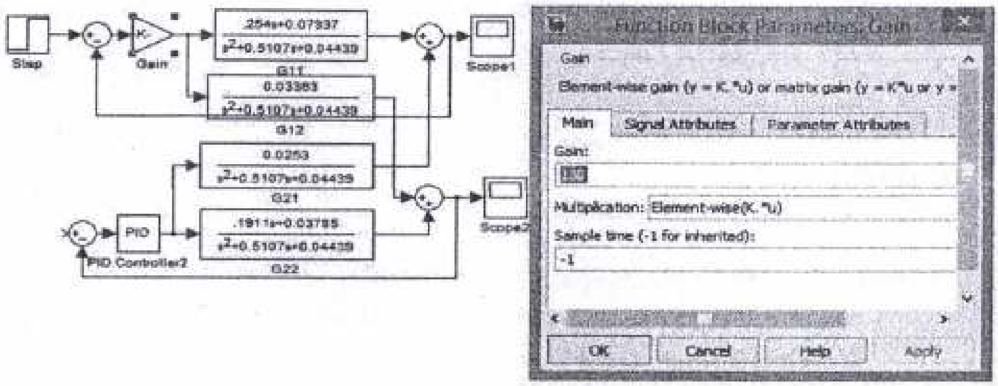


Figure 4.6: Block diagram in control and simulation loop

CHAPTER 5

EXPERIMENTAL RESULTS AND DISCUSSION

5.1:Ziegler-Nichols Method(Ultimate Cycle Method):



Gain

Element-wise gain (y = K.*u) or matrix gain (y = K*u or y =

Gain:

Multiplication: Element-wise(K.*u)

Sample time (-1 for inherited):

-1

Figure 5.1:block diagram of Z-N method

By increasing the value of proportional gain we do not get the oscillation in the overdamped system. So the Ziegler-Nichols Ultimate Cycle Method fail to determine the Kp and Tu value.

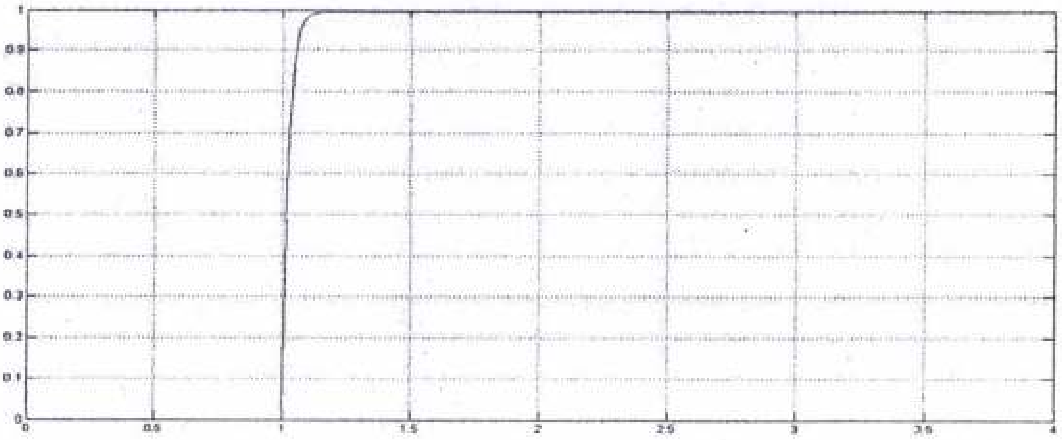


Figure 5.2: response in ultimate cycle method

5.2:MODIFIED Z-N METHOD

Controller1Tuining:

Here,

Kcu=7 and Pcu=1.276

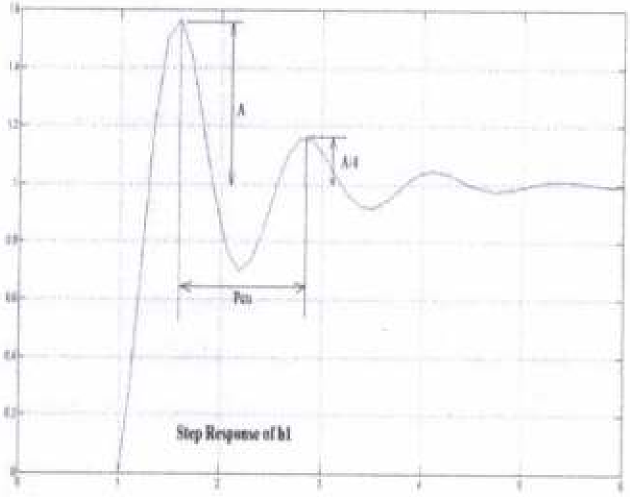


Figure 5.3: ¼ decay radio of controller1 for modified Z-N tuning method

Controller 2 Tuning:

Here,

Kcu=8 and Pcu=1.46

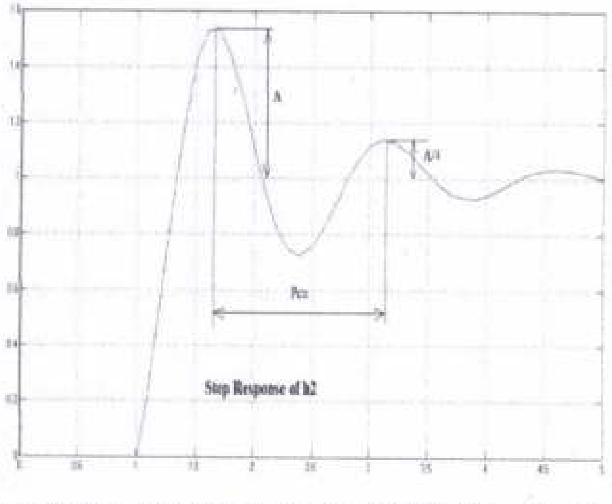


Figure 5.4: ¼ decay radio of controller2 for modified Z-N tuning method

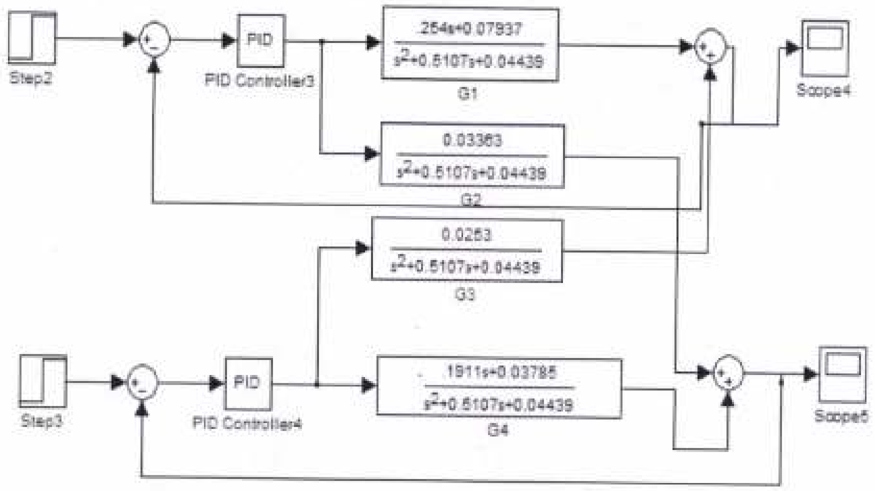


Figure 5.5: Block diagram of modified Z-N methods

Controller	Kp	Ti	Td
P I	3.5	1.914	
PID	4.667	3.19	0.319

Table 5.1: PI, PID values for modified Z-N methods for controller1

Controller	Kp	Ti	Td
P I	4	2.19	
PID	5.33	3.65	0.365

Table 5.2: PI, PID values for modified Z-N methods for controller1

As we donot get any continuous oscillation and gives a overdamped response. So we get a good response by using modified Z-N method.

4.3:AUTOTUNING METHOD:

As per the procedure of autotuning method we obtain the response as shown on the figure.

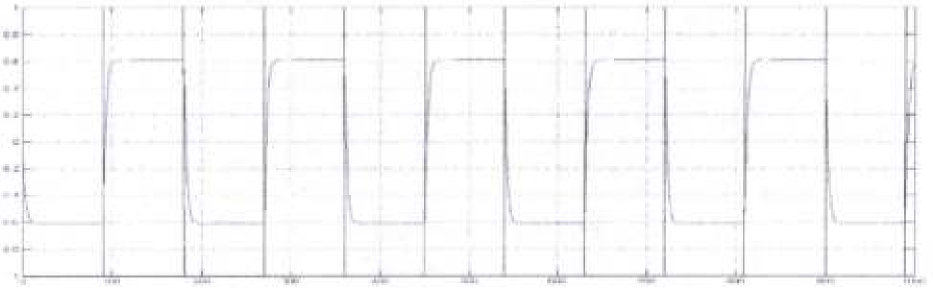


Figure 5.7 : response for autotuning method

From the figure we determine the Kcu and P value.

For controller1 Kcu= 1.019108 and P=3.

For Controller2 Kcu=2.123 and P=3

5.4:Comparison of PID values with different PID tuning method:

Controller 1

Method	Kp	Ti
Z-N closed loop	3.1818	1.0633
Tyreus-Luyben	2.1875	2.8072
Modified integral value Ziegler-Nichols Tuning	3.5	1.914

Table 5.3: PID values of different tuning method for controller1

Controller 2

Method	Kp	Ti
Z-N closed loop	3.636	1.2166
Tyreus-Luyben	2.5	3.212
Modified integral value Ziegler-Nichols Tuning	4	2.19

Table 5.4: PID values of different tuning method for controller2

Step Response of the System With Different PID Value:

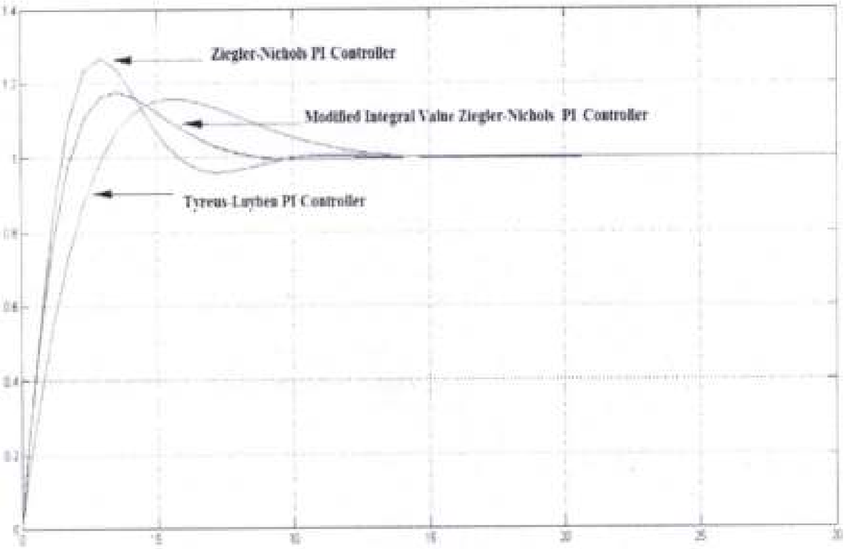


Figure 5.6: Response of the System With Different PID Values for h_1

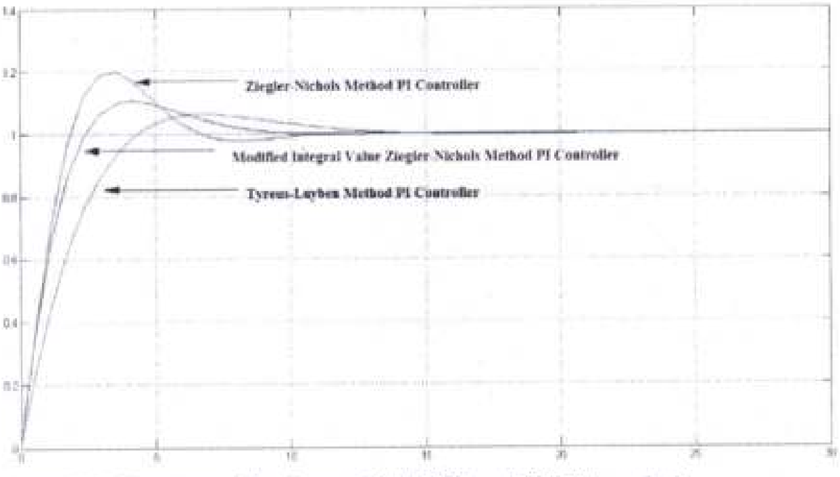


Figure 5.7 : Response of the System With Different PID Values for h_2

As shown in figure 5.6 and figure 5.7 in modified Z-N method we get better response compare to the other two methods.

5.5: DECOUPLER DESIGN:

$$d_{21} = -\frac{g_{21}(s)}{g_{22}(s)} = -\frac{0.0253}{0.1911s + 0.03785}$$

$$d_{12} = -\frac{g_{12}(s)}{g_{11}(s)} = -\frac{0.03363}{0.254s + 0.07937}$$

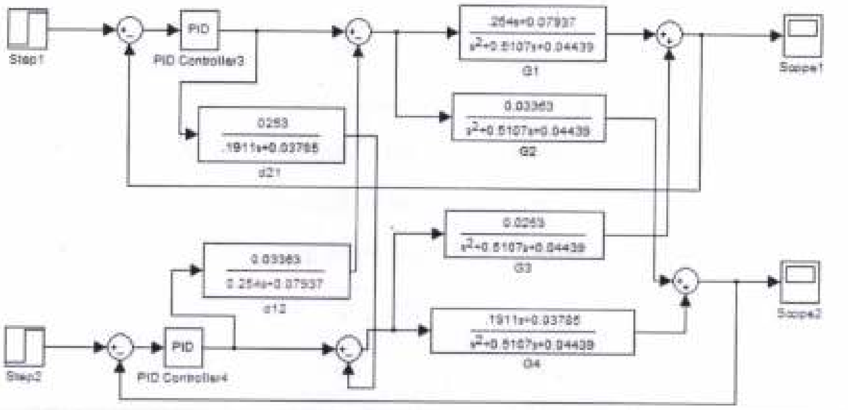


Figure 5.8:block diagram for decoupler design

Step Response Of TITO system PI Controller With Decoupler:

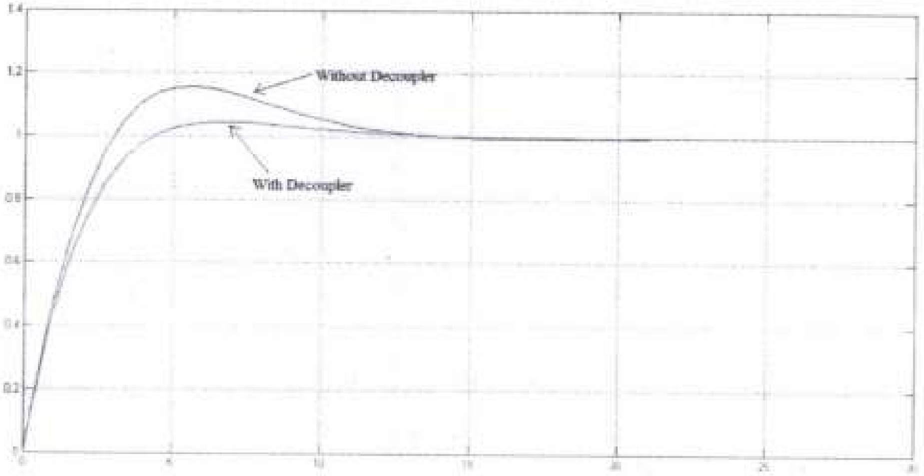


Figure 5.9: response for controller1 with decoupler

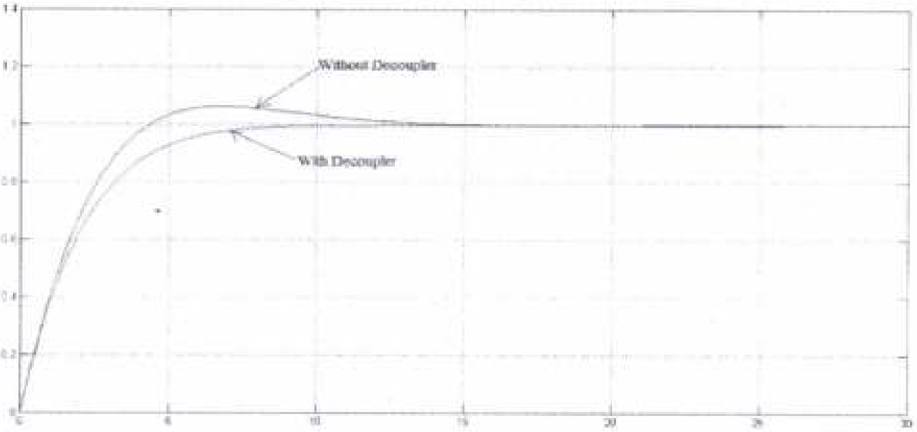


Figure 5.10: response for controller2 with decoupler

From the response it is clear that the overhoot which is present in our system is completely removed by using decoupler method of tuning.

5.6: COMPARISION BETWEEN PI TUNING IN MATLAB AND LABVIEW:

	SET POINT VALUE	MATLAB	LABVIEW	SIMULATION TIME
Controller 1	2	2.25	2.018	5.7
Controller 2	4	4.23	4.1136	
Controller 1	4	4.9	4.01812	4.15
Controller 2	6	6.5	6.4406	
Controller 1	6	7.2	6.0398	3.16
Controller 2	8	8.3	8.62008	
Controller 1	8	9.4	8.2	2.9
Controller 2	10	10.7	10.7	

Figure 5.5:comparision between MATLAB and LabVIEW

Here we compare between the PI tuning values in Matlab and labVIEW.By this comparison we found that the responses in both Matlab and labVIEW are almost equal.

CHAPTER 6

CONCLUSION

In our project we use various tuning method to control the tank level of interacting TITO systemFrom the experimental result is is conclude that for the overdamped system the Ziegler-Nichols closed loop ultimate cycling method fail to determine the Kcu and Pcu value. The Modified integral value Ziegler-Nichols Tuning is used for determine the Kcu and Pcu value in case of overdamped system. The increase in integral value make the system long oscillating. So as per the procedure we able to determine the Kcu and Pcu. The Decoupler with PID controller reduced the loop interaction and minimized the peak overshoot. . In this system loop interaction is present, which we eliminate by using Decoupler Method of tuning.

FUTURE ASPECTS:

Various controlling methods can be used to predict the output. We can consider two input and two output for more than four tank process. From different mathematical equations it can be expected to give variations in plant parameters. The step response of quadruple tank system using MPC with different control values can help to design a better system in future.

REFERENCES

Books:

1.DingyuXue, Yang Quan Chen, Derek P. Atherton "Linear Feedback Control' Chapter 6, Page 183.
2.K.Krishnaswamy 'PROCESS CONTROL' second edition.
3.George Stephanopoulos 'CHEMICAL PROCESS CONTROL, An Introduction to Theory and Practice'.

Journals:

1.P.Srinivas, K. Vijaya Lakshmi, V. Naveen Kumar. "A Comparison of PID Controller Tuning Methods for Three Tank Level Process' *International Journal of advanced research in Electrical, Electronics & Instrumentation Engineering.* Vol 3, Issue1, January 2014.
2.Profs.M.Morari, J.Lygeros Note on 'Automatic Control Laboratory'.
3. Francisco Vazquez, Fernando Morilla Block on "Tuning Decentralized PID controllers for MIMO system with Decouplers'.
4.D.AngelineVijula, Dr. N. Devarajan. 'Design of Decentralized PI controller using Model Reference Adaptive Control for Quadruple Tank Process' *International Journal of Engineering and Technology* Vol 5, No 6, December 2013 – January 2014.
5.Dr. Yeffry, Handoko Putra. Power point of 'Tuning of PID Controllers'.
6.Prof. D. Angeline Vijula, Anu K, Honey Mol P, PoormaPriya S. 'Mathematical Modeling of Quadruple Tank System' *International Journal of Engineering Technology and Advanced Engineering* Volume 3, Issue 12, December 2013.
7. Mr. Parvat B.J.,Mr.Deo S.A. and Mr. Kadu C.B. ' Mathematical Modeling of Interacting and Non interacting Tank System. *International Journal of Application for Innovation in Engineering & Management. Volume 4, Issue 1, January 2015.*
8. B. Wayne Bequette 'Process Control Modeling, Design, and Simulation' PHI Publication, Chapter 14 , Page 423-443.
9.Jayaprakash J. Davidson D. SubhaHency Jose P 'Comparison of Controller Performance for MIMO Process.' *International Journal of Emerging Technology and Advance Engineering* Volume 3, Issue 8, August 2013.
10.ArjinNumsomran, VittayaTipsuwanporn, ThanitTrisuwannawat and KittiTirasesth 'Desin of PID controller for the modified Quadruple Tank Process using Inverted Decoupling Technique' *11th International Conference on Control, Automation and System,*Korea.2011.