

Comparative Analysis of Distillation Column Control Structure & Remote Controlling of a Robotic System

A Thesis submitted in partial fulfillment of the requirement for the degree of master of technology

In

Electronics and Communication Engineering
(VLSI & Embedded system)

By

Setty Harsha Vardhan

Roll No. 710EC2140



NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA

PIN – 769008

ODISHA, INDIA

Comparative Analysis of Distillation Column Control Structure & Remote Controlling of a Robotic System

A Thesis submitted in partial fulfillment of the requirement for the degree of master of technology

In

Electronics and Communication engineering

(VLSI & Embedded system)

By

Setty Harsha Vardhan

Roll No. 710EC2140

UNDER THE SUPERVISION OF

PROF. TARUN KUMAR DAN



NATIONAL INSTITUTE OF TECHNOLOGY, ROURKELA

PIN – 769008

ODISHA, INDIA

DEDICATION

This thesis is dedicated to my parents, faculties and friends whose support is enabled to write this thesis



Department of Electronics and Communication Engineering

National Institute of Technology, Rourkela

Rourkela-769008, Odhisha, INDIA.

CERTIFICATE

This is to certify that the Thesis entitled, "**Comparative analysis of Distillation column Control Structure & Remote controlling of a Robotic system**" submitted by "**Setty Harsha Vardhan**" bearing Roll No. **710EC2140** to the National Institute of Technology Rourkela is a bonafide research work carried out by him under my guidance and is in partial fulfilment of the requirements for the award of the degree of "**Master of Technology**" in Electronics and Communication Engineering specializing in "**VLSI Design and Embedded System**" from this institute.

PROF. TARUN KUMAR DAN

DEPARTMENT OF E.C.E

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

ACKNOWLEDGEMENT

There are many people who has gave me support and valuable guidance about different aspects of the project. First of all, I want to thank my veteran guide **Prof. Tarun Kumar Dan** for his valuable ideas, support and guidance for the project and freedom that he gave me to explore new aspects in the project. Due to him, I gained the careful research attitude

My sincere gratitude to Prof. Ayaskanta Swain for his valuable support and guidance during the project and the moral support he gave me is really helped me to explore more in the project and helped me to try solve new challenges.

I want to thank Prof. K. K. Mahapatra Head of the Electronics and Communication Engineering department and Prof. U.C. Pati for providing me the tools and accessories to complete my project.

I want to thank Mr. Sankata Bhanjan Prusty & Mr. Sudeendra, Ph.D Scholars who gave me their valuable advice and support during the haywire conditions that I encountered during the project.

Finally, I would also like to thank my parents for their love and affection and especially their courage which inspired me and made me to believe in myself.

S Harsha Vardhan

710EC2140

Table of Contents

ACKNOWLEDGEMENT	v
List of Figures	viii
List of Abbreviations:	ix
Abstract	x
Chapter 1	
Introduction	1
1.1 Motivation:	2
1.2 Objective:	2
1.3 Literature Survey:.....	2
1.4 Thesis Layout:.....	3
Chapter 2	
Distillation column and decoupling of the interacting systems	4
2.1 Distillation column:	5
2.2 Working of the distillation column:	6
2.3 Over view of the work flow:.....	8
2.4 NIEDERLINSKI index:	9
2.5 Decoupling:	11
Chapter 3	
Controlling strategies	13
3.1 63.2 % method of approximation:	14
3.2 Feedback control:.....	16
3.3 Classic PID Controller:	17
3.4 Internal model control:	19
3.5 IMC based PID control:	20
3.6 Feedforward control system:.....	22
Chapter 4	
Robotics control using LabVIEW	24
4.1 Introduction:	25
4.2 Connection between the host computers to the processor on the SBRIO:	26
4.3 Ultrasonic sensor:	27
4.3 FPGA Configuration:.....	29
4.4 Motor control:.....	30

4.5 Motor speed sensing:	31
4.6 Controlling of the speed of motor:	32
4.7 Remote control of the robot using mobile:	33

Chapter 5

Results	34
5.1 The calculation of RGA matrix:	35
5.2 Decoupling:	35
5.3 Schematic of the Lab view model for calculating the disturbance	40
5.4 Calculated characteristics of the output response of the step input	41
5.5 Results of robotic control:.....	44
5.6 Calibrating the motor and optical encoder.....	45
5.7 Obstacle avoidance by the robot:.....	46
5.8 Wireless control of robot using mobile:	47

Chapter 6

Conclusion and Future work	48
6.1 Conclusion:.....	49
6.2 Suggested future work:.....	49
Reference:.....	50

List of Figures

Figures	Page no.
Figure 1: Schematic of distillation column -----	5
Figure 2: Work flow of distillation column -----	8
Figure 3: Work flow of robotic control -----	8
Figure 4: Schematic of TITO-----	9
Figure 5: Schematic of decoupler of TITO -----	11
Figure 6: 63.2 % approximation-----	14
Figure 7: Feedback control loop -----	16
Figure 9: feedback with IMC controller -----	20
Figure 10: IMC controller in terms of model T.F-----	20
Figure 11: Schematic of the robot-----	26
Figure 12: The link between the host and SBRIO-----	27
Figure 13: Working of Ultra sonic sensor-----	27
Figure 14: FPGA chip internal building blocks -----	29
Figure 15: motor control-----	30
Figure 16: PWM different duty cycles-----	30
Figure 17: Optical encoder -----	31
Figure 18: Signal generated by optical encoder -----	31
Figure 19: Feedback loop in robot -----	32
Figure 20: Backend of the decopler -----	36
Figure 22: Result after decoupling -----	37
Figure 23: Output of the PID controller-----	40
Figure 24: Output of the Internal Module controller-----	41
Figure 25: Output of ULTRASonic sensor -----	44
Figure 26: Output of Ultra Sonic sensor with 2d calibration -----	44
Figure 27: Output of the motor-----	45
Figure 28: Output of the PID controller-----	45
Figure 29: Labview Schematic Lab code -----	46
Figure 30: Front panel for robotic control-----	46
Figure 31: Mobile control front panel for robotic control -----	47
Figure 32: Mobile control back end program schematic -----	47

List of Abbreviations:

SISO: Single-Input-Single-Output

TITO: Two-Input-Two-Output

MIMO: Multi-Input-multi-Output

RGA: Relative Gain Array

PI: Proportional Integral

PID: Proportional Integral Derivative

IMC: Internal Model Control

FPGA: Field Programmable Gate Array

Abstract

Now a days, everything is became automated. So, It is important to have a smart design which can control the system with in tolerance range and able to perform tasks in the timely manner

In this project the process control of the entire plant by designing the different control systems PID, IMC and feed forward system and analyzing their performance. The advantages of the each system as individual have their own advantages like PID controller gives control over system without having much knowledge about the plant. While IMC controller gave superior response then the PID controller but IMC requires the insight of the system model

Later part of the project involves the discrete control of the plant(robot) with the help of the data acquisition system, electrical drive & sensors in the system during this part the robot is controlled using a mobile phone using android OS with the help of the team viewer software this a mini version of the wireless control of the plant.

The control of whole plant by understanding different aspects of the plant i.e..... controllers, Actuators & sensors. All the simulations are done in the LabVIEW 2013 environment and SBRIO- 9632 with other tools altogether acted as a robot, Team viewer software is used for communicate between robot and mobile(Remote control).

Chapter 1:

Introduction

1.1 Motivation:

Everything, now a days is automated & self-sustained. Mainly in the energy Industries the Safety and products quantity are the characteristics which are going to decide the Industries success rate. So, it is very crucial to understand the different control strategies and why they are choose for those conditions so that we can design control systems more complex process. Understanding a robotic control gives the insight of the controlling the plant from the operator point of view and also Robotics are becoming part of the industries now. Understanding and designing them to work for a particular operation is paramount

1.2 Objective:

To Analyze and compare different control strategies used in the industries of Energy –Oil & Gas. And with the proper understanding design a control system to a Robotic system using LABVIEW modules and Robot which can be programmed by using a FPGA

1.3 Literature Survey:

In the literature, there are a range of published studies on the control system and process control and plant wide controls.

- Norman nice (Control system) is provided great insight about control system analysis, stability criteria, designing and optimizing the control system.
- Wayne Banquette (Process Control modeling, Design and Simulation) provide the basics about the whole plant wide control with the clear understanding about the control system we can design complex plant automation using simple PID and IMC control techniques.
- Plant wide control and design by Prof.Nitin Kaistha explains every tiny detail about distillation column and it's control structure.

- Mobile Robotics Experiments with DaNI by DR. Robert king, Colorado school of mines, gave a crystal clear insight of working of a robot, programming aspects in LABVIEW and chances to explore new programs.

1.4 Thesis Layout:

In Chapter 2, there is a brief explanation about working of the distillation column and the wood and berry transfer function model of distillation column which is basically a Two Input Two Output system and then the discussion goes about the interacting systems, problems dealing with it and about the Decoupling techniques and RGA matrix for pairing the inputs with outputs

In Chapter 3, we discuss about the controlling techniques that can be used in a process controlling loop, a brief theory about working philosophy about PID controllers, and Internal model controls, the tradeoffs between them and then about the Feed forward controller for the better performing aspects.

In Chapter 4, We now discuss about the working of the robot, The programming modules for controlling the individual module of the robot , The communication techniques between controller and sensors, Actuators, FPGA programming and its advantages.

In Chapter 5, Results, The discussions about the results obtained and the analysis of the calculation take place behind them, it goes both for the PID and IMC part and Robotics part. And in the final 6th chapter the conclusion and future work are discussed.

Chapter 2:

Distillation column and decoupling of the interacting systems

2.1 Distillation column:

Distillation column are basic building block of a refinery which basically separates a mixture of chemical products based on the respective boiling points.

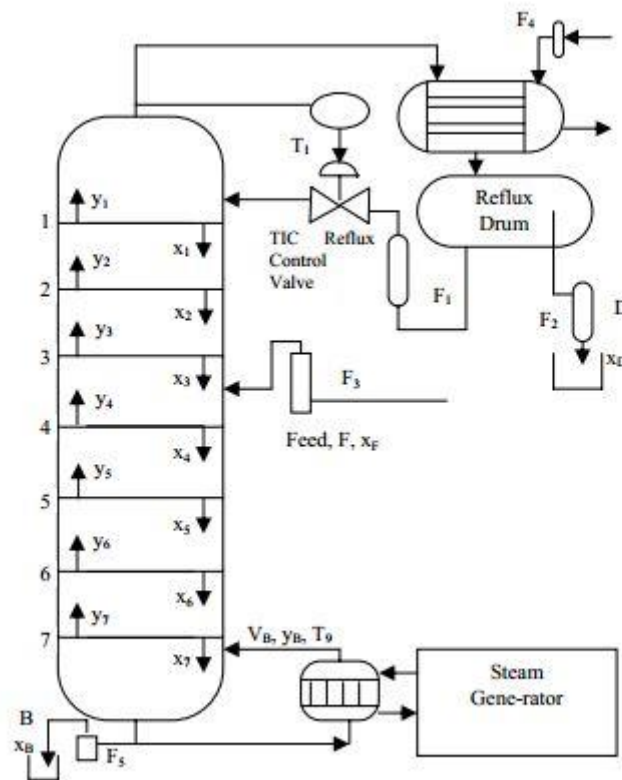


Figure 1: Schematic of distillation column

As you can see above we have a schematic of a distillation column. Distillation column contains basically have a long tower (Much long if a super fractionators consider). With trays at different levels of the tower separated by constant length throughout the tower. And each tray has a bubble caps through which mass transfer can be take place.

And then there is a feed which is in general of mixture of the different chemicals in this case binary distillation (only two components in the mixture) which have to be separated with different temperature profiles. Then comes the most important parts of distillation column reflux and reboiler.

If we consider the working of the reflux it has a surge tank which stores the light key of the mixture and there is a condenser which indeed control the pressure of the distillation column the surge tank controls the level of the light key and acts as a input to the reflux which introduce the light key to the process and the level of the surge tank is controlled by the output valve. In this particular example we are considering only the quality of purity of the light key and heavy key of the chemical mixture.

Then comes the re boiler has same structure of operation as did the reflux except that the re boiler has a boiler which heat up the hot key and act as the heat source to the distillation column.

2.2 Working of the distillation column:

Let say there is a mixture of A and B in a chemical plant and A is a heavy key and B is the light key in the sense the A element have higher boiling point and B has the lower boiling point and when the mixture is entered into the column, the light key will heat up due to high temperature and go in to the form of the gas step by step and tray by tray in the distillation column and ends up at the highest point in the column in a gas phase then it got condense by the condenser and got store into the surge tank and due to high boiling point B stuck up at bottom and purity of the product depends on the boiler temperature of the distillation column.

Work of the reflux is to send back the chemical which got stored in the re flux surge tank which has light key B along with small part of heavy key A and due to continues process of the Reflux feedback we can have the pure light key up to the standards and within the tolerance level and now, we see the process from point of Heavy key.

Heavy key A as the boiling point is very high the A will drop down in the distillation column and got collected at the bottom of the Distillation column which goes through the re

boiler to heat up send back to the column which helps to maintain the column at desired temperature during the separation process. Now, we can control the purity of the Heavy key using feed flow or rate of reflux flow or the re boiler temperature or the feedback to the re boiler system also. And the same conditions is applicable for the Light key also. This is the problem of coupling takes place where you want to control the multiple outputs using the multiple inputs of the system. This type of systems are called as coupled systems where the output depends upon more than one input.

The whole distillation column can be taken in the mathematical model which is

$$\begin{bmatrix} x_d \\ x_s \end{bmatrix} = \begin{bmatrix} \frac{12.8e^{-s}}{16.9s+1} & \frac{-18.9e^{-3s}}{21s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} R \\ S \end{bmatrix} + \begin{bmatrix} \frac{3.8e^{-8.1s}}{14.9s+1} \\ \frac{-4.9e^{-3.4s}}{13.2s+1} \end{bmatrix} [F] \quad 2.2.1$$

- x_d is the purity percentage of the Distillate of the mixture
- x_s is the purity percentage of the bottoms of the mixture
- R is the Reflux flow rate
- S is the stem flow rate at the re boiler
- F is the Feed flow rate.

As you can see from the above equation we can have disturbance to the process as the feed is entering into the system and also there is a problem whenever the feed enters, we shall have to recycle more as the purity level goes down and also we have another problem of the interaction between inputs as we can see the purity of distillate depends both on the Temperature of the distillation column and as the rate of the reflux. The same problem occurs in the case of bottoms too, so we need to pair the outputs with the inputs and for that we have RGA matrix.

2.3 Over view of the work flow:

Controlling using IMC and PID Technique

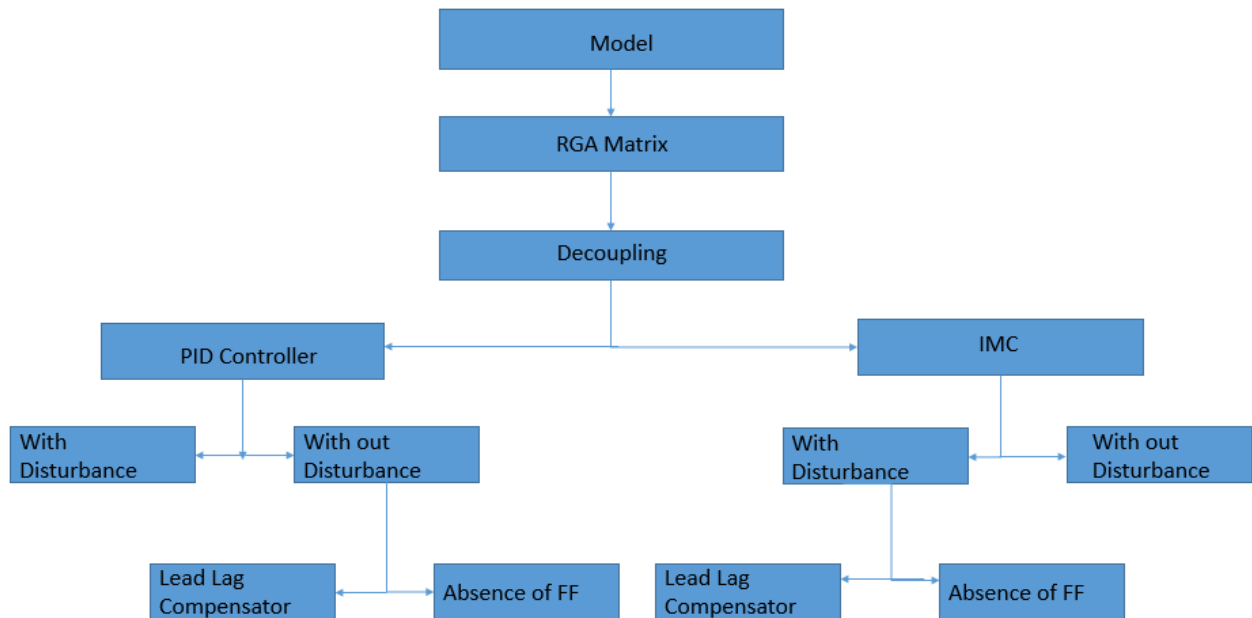


Figure 2: Work flow of distillation column

Robotics control using mobile

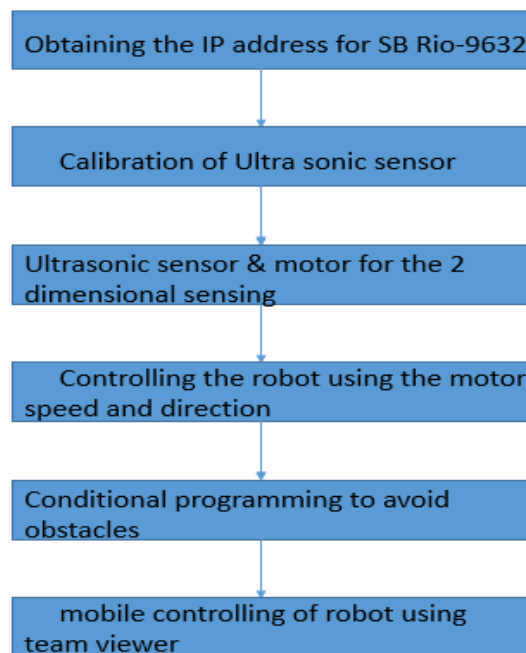


Figure 3: Work flow of robotic control

2.4 NIEDERLINSKI index:

Niederlinski index is the tool which help to identify the pairing of the output with inputs in multiple input multi output system. The logic behind this strategy is as follows

Let's take a system two input and two output system.

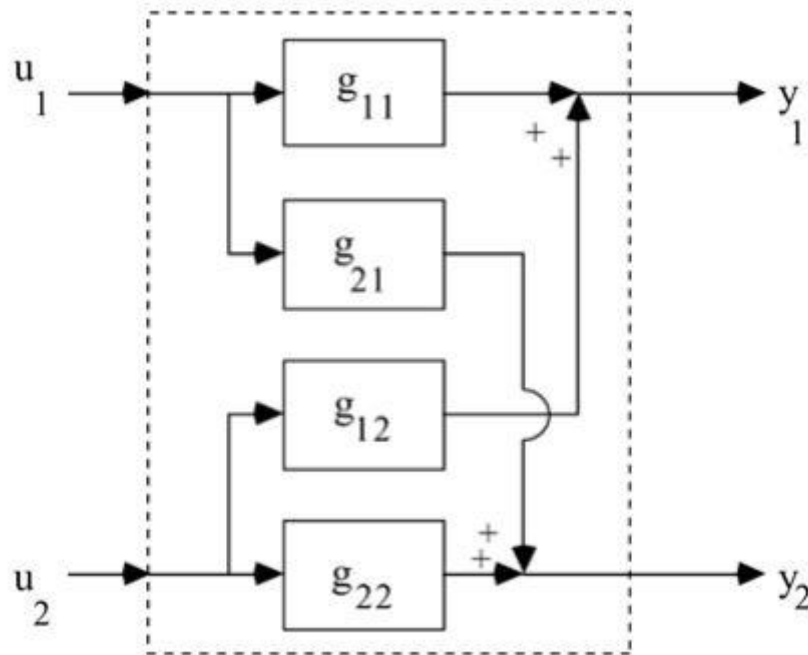


Figure 4: Schematic of TITO

Let's just say a disturbance is entered into the system and y_1 is changed by 1. To compensate that we are approximating the above to final steady state gain only. u_1 has to be change by $\frac{(-1)}{g_{11}}$ as it is interacting it changes the value of y_2 to compensate that u_2 has to change by some other amount due to that y_1 is changed and this process goes on and all the compensation efforts of u_1 got add up. Those sum ups should die out in the system during the process reaches a certain stage.

This is the logic behind the Niederlinski index, now if we come to the mathematical form of the equation we have

These following constraints

$$\lambda_{ij} = \frac{g_{ij}}{g^*_{ij}} = \frac{\frac{dy}{dx} (\text{all loops open})}{\frac{dy}{dx} (\text{only loop } i \text{ is open})} \quad 2.4.1$$

If the value of

- $\lambda = 1$ then it is perfect condition of coupling
- $\lambda = 0$ then it is poor condition of coupling
- $\lambda = -1$ then it is bad condition of coupling

We can try out all the form of the conditions of pairing and the whole matrix is said to be the RGA Matrix (Relative Gain Array) and it is defined as follows

$$\Lambda = \begin{bmatrix} \lambda_{11} & \cdots & \lambda_{1n} \\ \vdots & \ddots & \vdots \\ \lambda_{n1} & \cdots & \lambda_{nn} \end{bmatrix} \quad 2.4.2$$

$$\Lambda = \frac{1}{k_{11}k_{22} - k_{12}k_{21}} \begin{bmatrix} k_{11}k_{22} & -k_{12}k_{21} \\ -k_{12}k_{21} & k_{11}k_{22} \end{bmatrix} \quad 2.4.3$$

As we figure out the value of individual values of the matrix, we can compare the results with the specified format and we have the elements which has to get paired and which are not. We are only considering the final gain of the transfer function of each transfer function and are assuming that the input won't change until the whole process comes to a steady state due to the disturbance entered, That is the reason behind the taking the k_{11} instead of g_{11} for calculating the RGA matrix.

2.5 Decoupling:

Once we figure out which pair of inputs and outputs are to be paired we have the Decoupling process that has to be done, the insight of the decoupling process is described as below

Let's have a system as below:

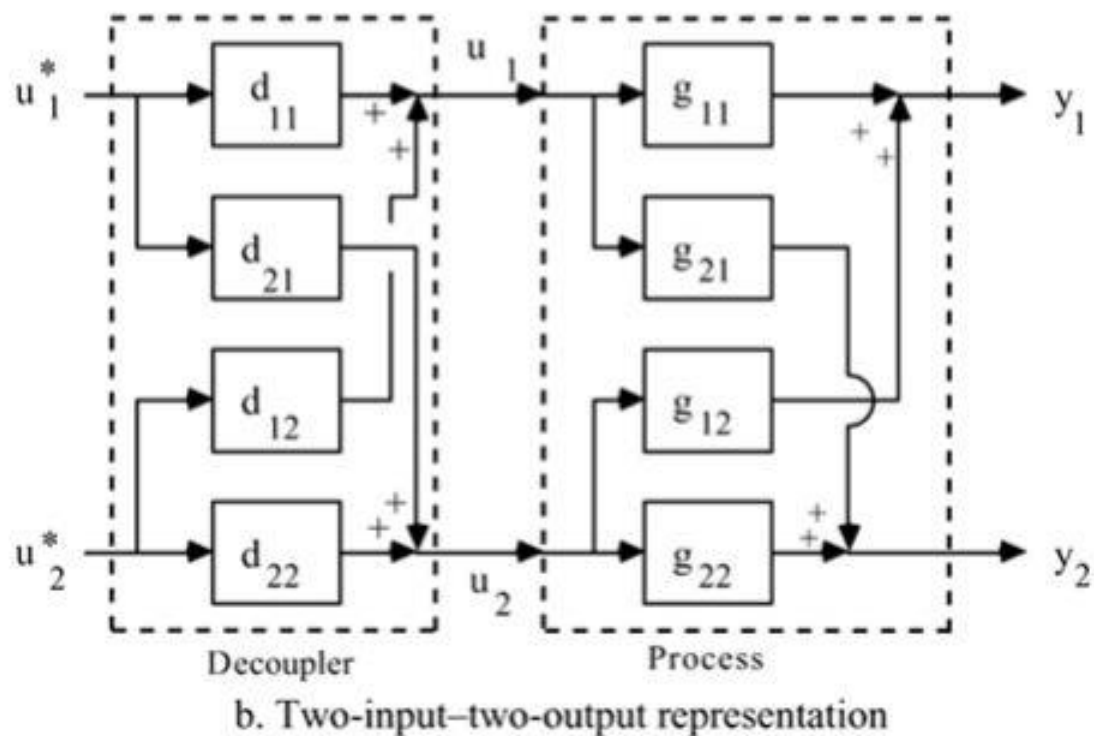


Figure 5: Schematic of Decouple of TITO

As shown in the above, consider we have a TITO system with the respective process elements. For decouple the process we should design the values of Decouplers based on the process transfer functions.

$$G_p(s)D(s) = \begin{bmatrix} g_{11}^*(s) & 0 \\ 0 & g_{22}^*(s) \end{bmatrix} \quad 2.5.1$$

$$\begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix} \begin{bmatrix} 1 & d_{12}(s) \\ d_{21}(s) & 1 \end{bmatrix} = \begin{bmatrix} g_{11}^*(s) & 0 \\ 0 & g_{22}^*(s) \end{bmatrix} \quad 2.5.2$$

Which implies:

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

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

$$g_{11}^*(s) = g_{11}(s) - \frac{g_{12}(s)g_{21}(s)}{g_{22}(s)} \quad 2.5.3$$

$$g_{22}^*(s) = g_{22}(s) - \frac{g_{12}(s)g_{21}(s)}{g_{11}(s)} \quad 2.5.4$$

By implementing the Decouple for the whole system, it will act as single input single output system. Which can be tuned very easily for the purpose of the tight control. This will make sure that the interaction between the both systems are almost negligible.

Chapter 3

Controlling strategies

3.1 63.2 % method of approximation:

As the decoupling is over now we should control the whole system by braking it into pieces for that analysis of the overall transfer function we used the approximation Numerical Application of 63.2% Method which is like this

To approximate a system



To know the transfer function $G(s)$. We give a unit impulse function just like off to on and we analyze the output response and approximate the $G(s)$ value to certain transfer function.

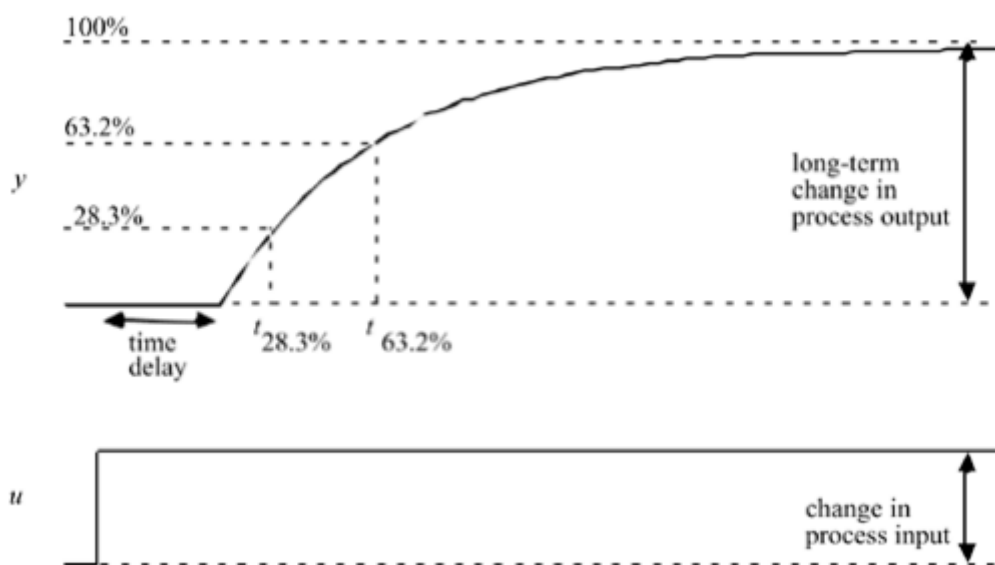


Figure 6: 63.2 % approximation method

Now from the above graph we can approximate the $Y(s)$ as follows

$$Y(s) = \frac{k_p e^{-td}}{\tau s + 1} \quad 3.1.1$$

Where

- k_p is the gain of the $G(s)$ value of long term change value
- td is the delay time
- τ is the time constant of the $G(s)$ value at 63.2%

Now we have the transfer function of the whole system along with decouple, so we have the basically a first order time delay system to which we should design the control strategies.

For that we have different control strategies there is the classical PID control model and then there is the IMC control strategy which can give tighter control of the things compared to PID but have a trade of knowing the exact transfer function $G(s)$ of the system.

To improve the system there is a feedforward control strategy which gives much tighter control of the system along with one of the feedback control strategies that are there. We will examine about the PID control and it's working in the following section of control strategies.

3.2 Feedback control:

Before studying the PID controller we should know about the Feedback control loop and its schematic is something like this:

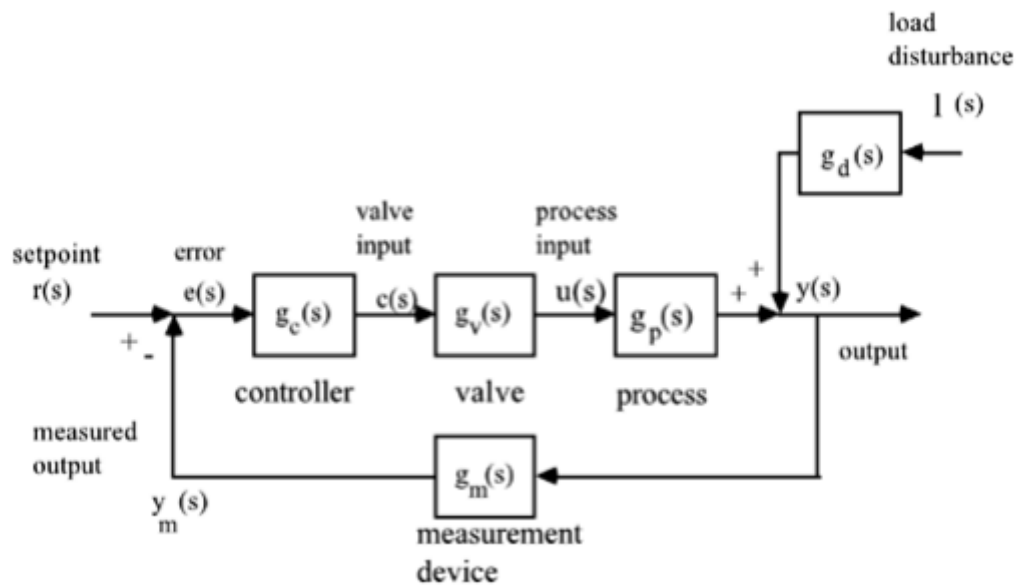


Figure 7: Feedback control loop

As you can see above we have the process and control variables and feedback loop coming from the measured output (measuring device).

In the colloquial way of saying let's just say you are driving a car and it is deviating from the path and goes off the road due to a turn a head (which is an analogy of the changing of the process output due to a disturbance) then the immediate response from you would be sensing that car is off the path (analogy of the measuring the output and comparing with set point) then you turn the steering which in turn turns the car wheels which controls the path of the car.(analogy of the control variables and their effect on the process).

The same exact strategy goes in the above block diagram and now we discuss the intuition part of the turning the steering (control strategies).

3.3 Classic PID Controller:

Before taking about PID controller we just have a look at the each controller individually. Let's just have the basic understanding.

PID controller in the position form as follows

$$u(t) = Kc \left[e(t) + \frac{1}{\tau_i} \int e(t) dt + \tau_d \frac{d(e(t))}{dt} \right] + \text{Bias} \quad 3.3.1$$

- $U(t)$ is the output of the controller part.
- $e(t) = y^{sp} - y(t)$, Error term
- Kc is the gain of the controller
- τ_i is the integral time constant
- τ_d is the derivative time constant

To have better understanding of how controlling takes place in the PID we should look at the velocity form of the PID controller strategy

$$\frac{d(u(t))}{dt} = Kc \left[\frac{d(e(t))}{dt} + \frac{e(t)}{\tau_i} + \tau_d \frac{d^2 e(t)}{dt^2} \right]$$

As you know at the steady state there won't be any change due to the time so the derivative term of the equation becomes zero in the above equation that leaves us with the equation

$$Kc \frac{e(t)}{\tau_i} = 0$$

Which implies the integral action gives us zero offset and we can have seen the proportional action function, It try to change the output multiple times to the error change which gives the option of the faster response.

And due to the derivative action we have some anticipation of the system as you can have from the position form of PID controller we have the equation as follows

$\frac{d(e(t))}{dt}$ Which means it can see in which direction the response going and try to compensate the oscillation values in the system. So it allows to have higher K_c value to have the faster response with the less oscillation values in the system. But the disadvantages with the derivative action is that it has nature of amplifying the things of noise which got introduce in the system.

To compensate that filters and signal processing shall take place before the derivative action and most importantly they are precisely used where ever tight control is necessary like controlling the Reactor temperature in the process plant 90 % of the plants are like have PI controllers only and most of the level control is take care by the P controllers only.

The K_c and other tuning parameters of the PID controller can be adjusted using many methods Ziegler-Nichols method, Cohen-Coon method of the tuning parameters are very popular and we considered Ziegler-Nichols method while tuning the parameters in the simulations.

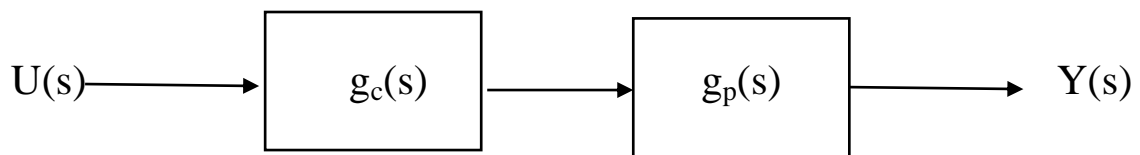
Controller type	K_c	τ_I	τ_D
P-only	$\frac{1}{k\theta}$ or $\frac{\tau_p}{k_p\theta}$	—	—
PI	$\frac{0.9}{k\theta}$ or $\frac{0.9\tau_p}{k_p\theta}$	3.3 θ	—
PID	$\frac{1.2}{k\theta}$ or $\frac{1.2\tau_p}{k_p\theta}$	2 θ	0.5 θ

Tuning parameters of Ziegler-Nicholas method

3.4 Internal model control:

As we saw before about one of the controller strategy PID where we have very little knowledge over the process model but for the designing the Internal model control we should have the process model so that we can control the whole process optimistically. Let's get back to the classic example of the car model. In the analogy of the controlling we just saw that car is off the path and then we controlled to make it come back to the path.

We didn't take the aspects whether the path is steep, bumpy or car is small or big. If we take those aspects also while controlling the car then we can have better control over the car than just turning the steering. Same goes with the IMC control where we take the process model into the consideration during the controlling the process and design the controller on the bases of process model.



We have the equation

$$U(s) * g_c(s) * g_p(s) = Y(s) \quad 3.4.1$$

$$\text{If } g_c(s) = \frac{1}{g_p(s)} \quad 3.4.2$$

Then we have $U(s) = Y(s)$, so we output following the input in open loop.

3.5 IMC based PID control:

Let's take the discussion to closed loop control than we have the following equation and strategies.

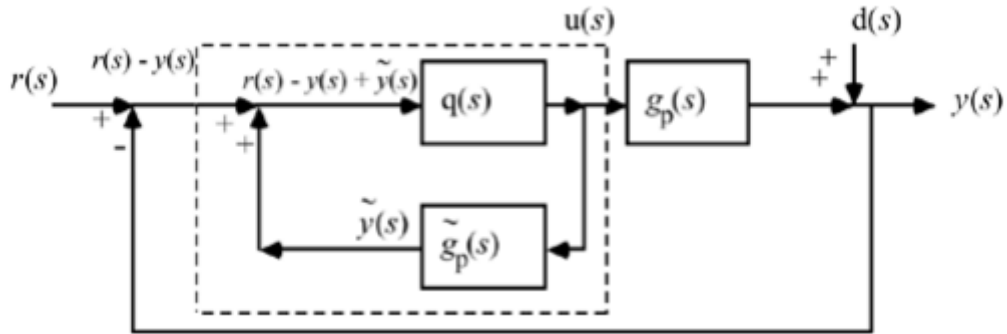


Figure 8: feedback with IMC controller

We can write the whole part as a controller and replace it with following block diagram.

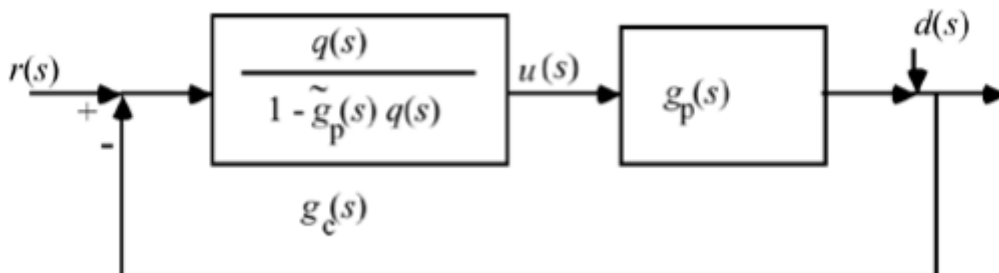


Figure 9: IMC controller in terms of model T.F

Now we have a controller $g_c(s)$ which is indeed in the form of PID controller. With the tuning parameters as follows for the First order delay process. We have estimated process transfer function as

$$g_{\tilde{p}}(s) = \frac{k_p e^{-\theta s}}{\tau_p s + 1} \quad 3.5.1$$

According to the Padé approximation we can write

$$e^{-\theta s} = \frac{-0.5s + 1}{0.5s + 1} \quad 3.5.2$$

$$g_{-p}^{-1}(s) = \frac{k_p}{(\tau_p s + 1)(0.5\theta + 1)}$$

Substituting the values in the above controller equation we will end up with the equation

$$g_c(s) = \frac{(\tau_p s + 1)(0.5\theta + 1)}{k_p s(\lambda + 0.5\theta)}$$

$$k_c = \frac{\tau_p}{k_p(\lambda + 0.5\theta)} \quad 3.5.3$$

$$\tau_I = \tau_p + 0.5\theta \quad 3.5.4$$

$$\tau_D = \frac{\tau_p \theta}{2\tau_p + \theta} \quad 3.5.5$$

Where we have

θ Is the delay of the process

τ_p Is the time constant of the process

k_p Is the gain of the process

k_p Is the gain of the PID controller

τ_I Is the time constant of the Integral constant

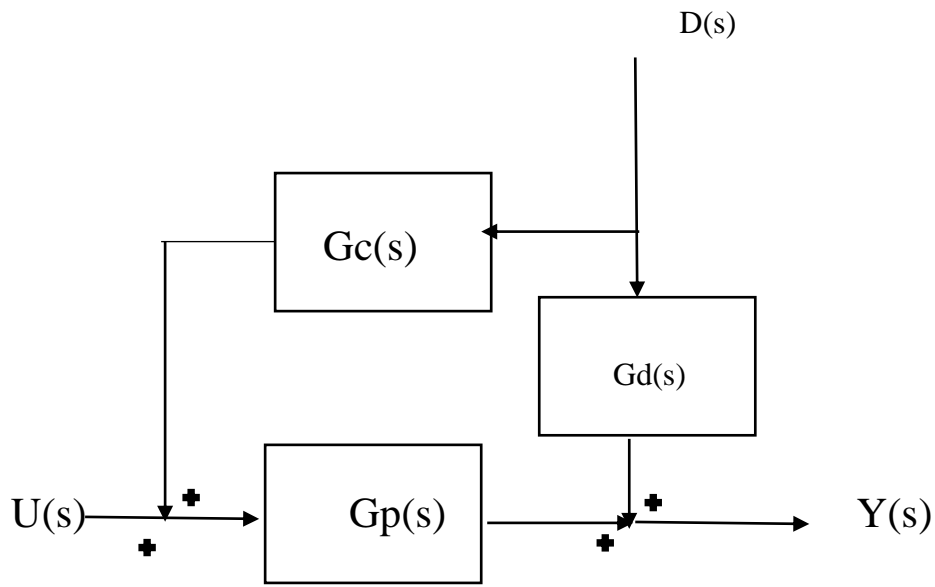
τ_D Is the time constant of differential constant

With this we can have a controllers which controls the process with insight of the process transfer function. IMC controller have better performance than the PID controller and the tradeoff here is that we should have the process transfer function which takes lot of calculation to figure out the process transfer function.

The problem with these controllers are either PID or IMC, They start acting once the disturbance entered in the system, to anticipate the feedforward is used it is fairly as an open loop control system and the working is as follows.

3.6 Feedforward control system:

The schematic of the feedforward control system goes like this, let's have a Process Transfer function $G_p(s)$ and with respect to the disturbance coming to the system the transfer function $G_d(s)$



Where $D(s)$ is the disturbance entering into the system and $U(s)$ is the input to the system. $G_c(s)$ is the feed forward controller that compensate the disturbance variable in the system and it is equal to the

$$G_c(s) = \frac{-G_d(s)}{G_p(s)} \quad 3.6.1$$

For the first order time delay system we have the following method to calculate the feedforward controller transfer function.

$$g_p(s) = \frac{k_p e^{-\theta_p s}}{\tau_p s + 1}$$

$$g_d(s) = \frac{k_d e^{-\theta_d s}}{\tau_d s + 1}$$

$$g_{cf}(s) = -\frac{g_d(s)}{g_p(s)} = -\left(\frac{k_d}{k_p}\right) \frac{(\tau_p s + 1) e^{-(\theta_d - \theta_p)s}}{\tau_d s + 1} \quad 3.6.2$$

From the above equation we can calculate the value of the feed forward controller. There are many reason to adopt the feed forward compensations like the controller ought to be stable and physically it must be feasible. For immaculate feedforward compensation, the time deferrals of the deviations ought to be more noteworthy than the time delay of the. On the off chance that it isn't possible then the ideal compensation can't be accomplished.

As of now we see the designing of the control system to the individual process. we are now focus on controlling the plant (Robot) which involves much of data acquisition, sensors, motors, Controllers and most importantly the whole aspects working together.

Chapter 4

Robotics control using LabVIEW

4.1 Introduction:

A robot is an analogy of the whole process plant. By means it has sensors which can detect the different parameters and it has a controller which is a brain of the system and has motors which can be controlled by the controller to make the robot work in the way we want. So, by studying the working of robot we are indeed understanding the whole working of a plant. Robots are become very essential in the field of the marine, defense & in the application of the space too. In this project we controlled the robot remotely and also made the robot take its own path if an obstacle came in the path.

Robotics and automation are becoming an essential component of engineering and scientific systems and consequently they are very important topics for study by engineering and science students. Furthermore, robotics is built on fundamentals like transducer characterization, motor control, data acquisition, mechanics of drive trains, network communication, computer vision, pattern recognition, kinematics, path planning, and others that are also fundamental to other fields, manufacturing, for instance. The National Instruments (NI) LabVIEW Robotics Kit and LabVIEW are the two main important parts of the project one is hardware with FPGA and other is the software through which we can develop the remote controlling and automated control of the systems. The NI LabVIEW robotics kit includes DaNI: an assembled robot with frame, wheels, drive train, motors, transducers, computer, and wiring.

In the below picture we can see the whole picture of how the robot is assembled.

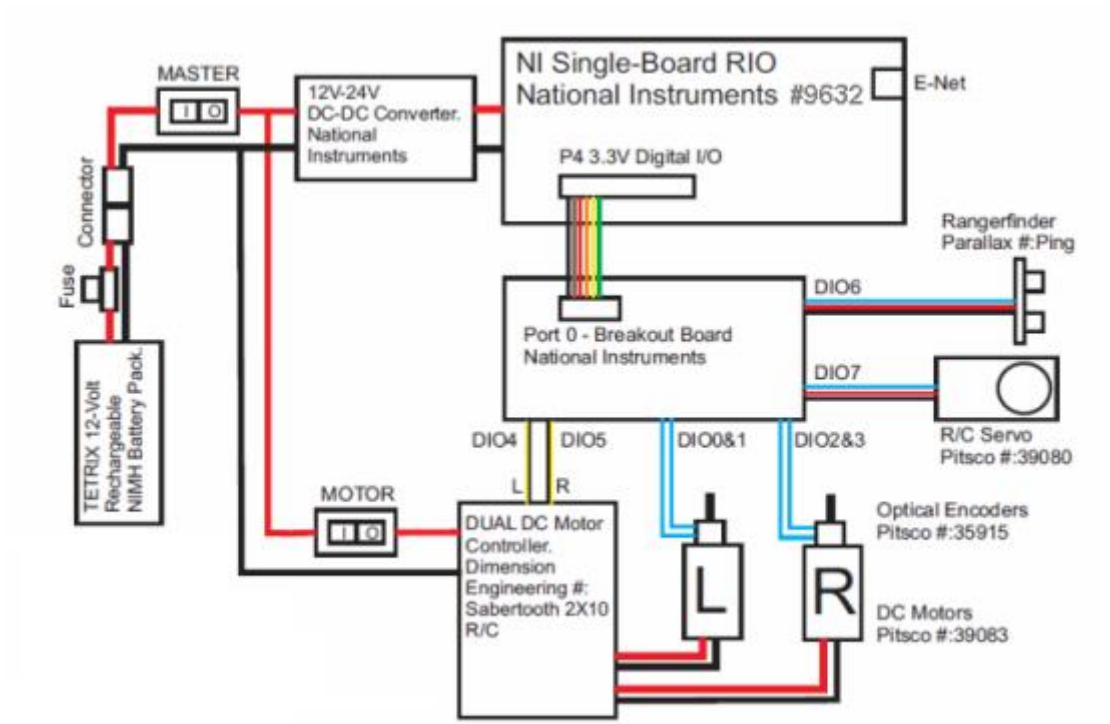


Figure 10: Schematic of the robot

4.2 Connection between the host computers to the processor on the SBRIO:

By using the project wizard we can identify the IP address of the SBRIO. Thus we can have address of the system to which we can communicate through the LAN cable. Which will generate an interface to the system with in the following manner:

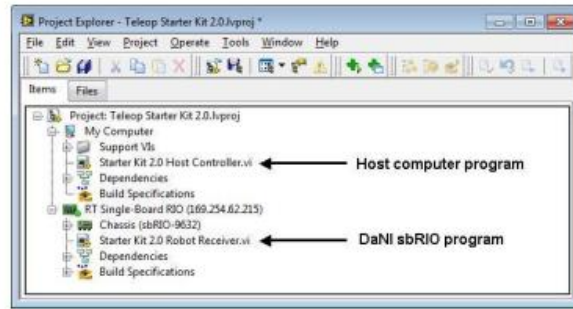


Figure 11: The link between the host and SBRIO

In the above interface we have all the files which can be used successfully interface with the robot. Supported VI, the sub programs which helps the sub VI's as dependencies and Build specifications

In normal these programs will take care of changing the program from G-language (the basis of the LabVIEW code) to the bit code which can be able to dump into the FPGA.

4.3 Ultrasonic sensor:

Then we have sensors in this case we have an Ultrasonic sensor which can detect any obstacle in the range of 0.2m to 3m.

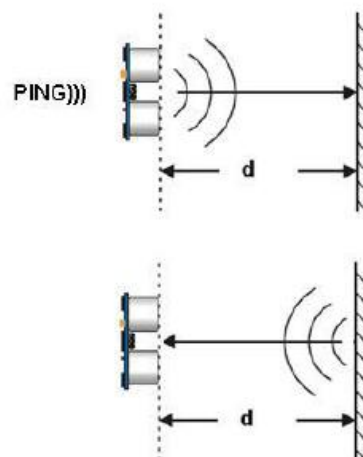
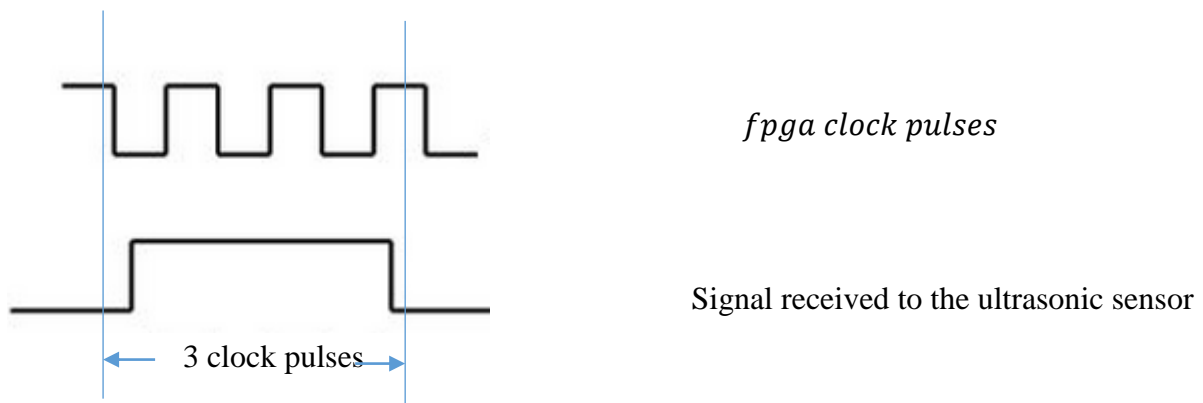


Figure 12: Working of Ultra sonic sensor

As from the above figure we have the ultrasonic sensor which can transmits the ultrasonic wave it waits for the signal to come back from the obstacle. In general when a wave comes across the obstacle during the wave propagation it get reflects. So the sensor waits for the signal to come back and based on the time it get back the signal it calculates the distance between the obstacle and sensor itself.



As you can see the time period of the 3 clock pulses, it is used to calculate the distance using Two Way Travel Time. The Processor counts the pulse to get back the result of the signal so we have

$$X = \frac{v \cdot t}{2} \tag{4.3.1}$$

Where, X is the distance of the obstacle and the Ultra sonic sensor.

V is the velocity of the sound in open medium (330 m/s)

T is the time calculated by the processor.

The accuracy of the sensor depend upon the many factors like the reflection index of the surface and orientation of the sensor and medium between the ultra-sonic sensor and obstacle. We have a servo motor along with the sensor which can give a full circular monitoring of the path in which robot is going

4.3 FPGA Configuration:

Unlike the hard wired print board systems the FPGA is in between the real time processor which can controls and computes with I/O ports. As a program is written in the LabVIEW we have a read VI which reconfigure the FPGA [Field programmable gate Array] and connected with the processor with help of configured FPGA through this we can have ability to program and FPGA configure itself to replicate the software model written on the LabVIEW.

As mentioned above the FPGA is rewired according to the logic given by the LabVIEW, Logic blocks and I/O blocks make the FPGA so fast, The internal PID control is so fast and the loop is so fast in the range of the ns, so the performance is restricted by the actuators and sensors not by the FPGA performance.

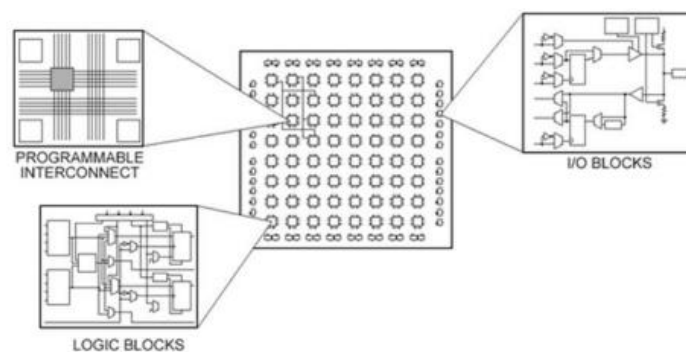


Figure 13: FPGA chip internal building blocks

When a signal comes from the let's say an ultrasonic sensor the data will pass through the FPGA, and set on the field data bus to the real time processor to process. When the Ultrasonic VI calls the FPGA program for it automatic loads and convert into the bit file by default.

4.4 Motor control:

The signal from the processor to the motor is very low of about 3 ma. Dc motor is 10 amp working machines and the speed is controlled by the Pulse Width Modulation Technique.

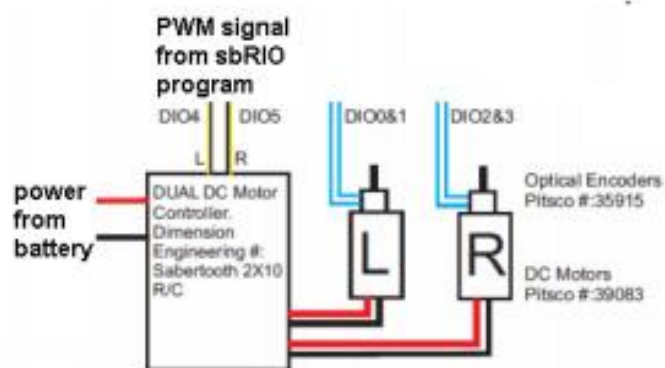


Figure 14: motor control

The dc motor speed depends up on the field excitation. Here the processor sends a signal and it changes in to pulse, for more velocity we should have the more duty cycle and for the less speed we should be excited the field with less duty cycle.

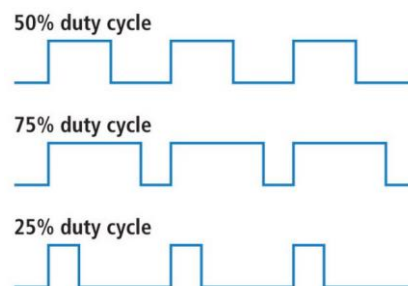


Figure 15: PWM different duty cycles

The word —set point in the name of the Velocity Set point VI, used in the previous section of this experiment, means that the inputs to the VI are target velocities of the two

motors. When the target velocities are the same, the motors should have the same rotational velocities. Therefore, the control system needs to know the value of the velocities and automatically send signals to the motors to maintain the target values

4.5 Motor speed sensing:

There is an optical encoder which gives the direction and speed of the motor in the pulse form then after analyzing the data pulse from the optical encoder for the analyzing processor.

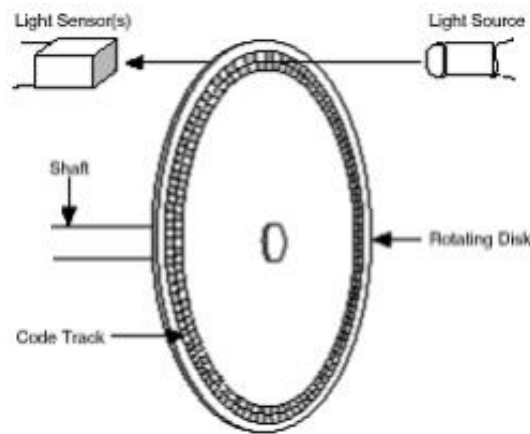


Figure 16: Optical encoder

As the motor shaft speed increases the frequency of the output pulse also changes and there are two discs which are concentric (A & B), As the light passes through let's say if A leads B then it is clockwise directional and If B lead A then it is anti-clockwise directional.

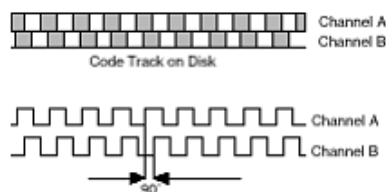


Figure 17: Signal generated by optical encoder

4.6 Controlling of the speed of motor:

The Internal logic in the FPGA of the controlling a speed of the dc motor or the rotating shaft of ultrasonic sensor which gives it 2D view of the things is a PID controller, we have discussed about PID controller in the previous chapter and here is its real time application .

The loop goes like the following.

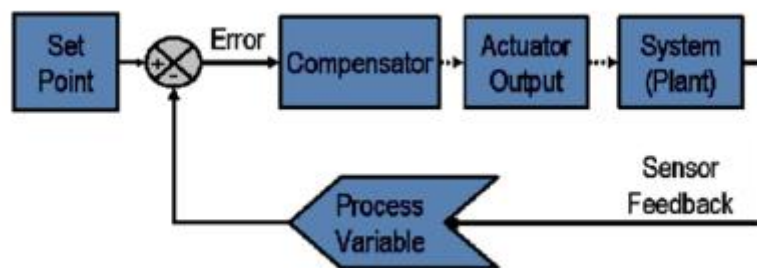


Figure 18: Feedback loop in robot

Set point is given from the processor, let's say we want to control the speed of the vehicle say 10 rev/min. Then the processor set the value of the motor speed as 10 rev/sec

But as the robot is in rest state the process variable shows 0 rev/sec, then the compensator that is inbuilt in FPGA comes to picture adjust the Pulse duty cycle to excite the motor speed. And the actuators speed will bit increase and that value is in turn compare with the set point and this process goes on till the error is in tolerance value or as we are using a PID controller in this case, the error is almost nullified.

But as the sensing elements or the actuators are compromises or have certain limits in their performance there exists a steady state value error in this robot and it is very small.

4.7 Remote control of the robot using mobile:

For this purpose there should be a channel through which commands can be passed to the Robot from the Mobile. So, a remote controlling software “Team Viewer” is used for this purpose.

Here is the working we have installed the TeamViewer in the both computer and the mobile so that they can communicate through World Wide Web, The Software identifies each system with unique ID. So we have two unique ID and just like a postal address to homes there can be communication between the both ends.

For that first we should agree to receive anything from the other end. So, after the agreement the communication will be start between the host computer and remote controller. As the packets are transfer between the both ends we can control what’s going on with host computer. Now, we run the VI that has implemented for the purpose of moving around, we can give command to move right then that will go to the host computer and as the host computer and robot are connected that command is implemented in the robot and it will move around.

There is another way we can have the program that can calculate the distance travelled by the SBRIO, we can dump the program in to the robot and the connection can be removed because as the robot itself is a computer it acts as per the program suggested even though there is no connection between the robot and host computer.

Chapter 5

Results

5.1 The calculation of RGA matrix:

As discussed before we have:

$$\begin{bmatrix} xd \\ xs \end{bmatrix} = \begin{bmatrix} \frac{12.8e^{-s}}{16.9s+1} & \frac{-18.9e^{-3s}}{21s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} R \\ S \end{bmatrix}$$

We are considering the gain of the final steady state value

$$\Lambda = \frac{1}{k_{11}k_{22} - k_{12}k_{21}} \begin{bmatrix} k_{11}k_{22} & -k_{12}k_{21} \\ -k_{12}k_{21} & k_{11}k_{22} \end{bmatrix}$$

$$k_{11}k_{22} - k_{12}k_{21} = 12.8(-19.4) - 6.6(-18.9) = -123.58$$

$$\Rightarrow \Lambda = \frac{1}{-123.58} \begin{bmatrix} -248.32 & 124.74 \\ 124.74 & -248.32 \end{bmatrix}$$

$$\Rightarrow \Lambda = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

From the above equation the

X_d should be controlled by Reflux and the X_s should be controlled by bottoms.

5.2 Decoupling:

As mentioned before of decoupling technique we have

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

Which implies:

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

$$\Rightarrow \frac{18.9e^{-3s}}{21s + 1} * \frac{16.9s + 1}{12.8e^{-s}} = e^{-3s} \frac{50.1s + 3}{42s + 2}$$

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

$$\Rightarrow \frac{6.6e^{-7s}}{10.9s + 1} * \frac{14.4s + 1}{19.4e^{-3s}} = e^{-4s} \frac{14.4s + 1}{33s + 3}$$

And

$$d_{22}(s) = 1 \text{ \& } d_{11}(s) = 1$$

The results are as follows before and after decoupling:

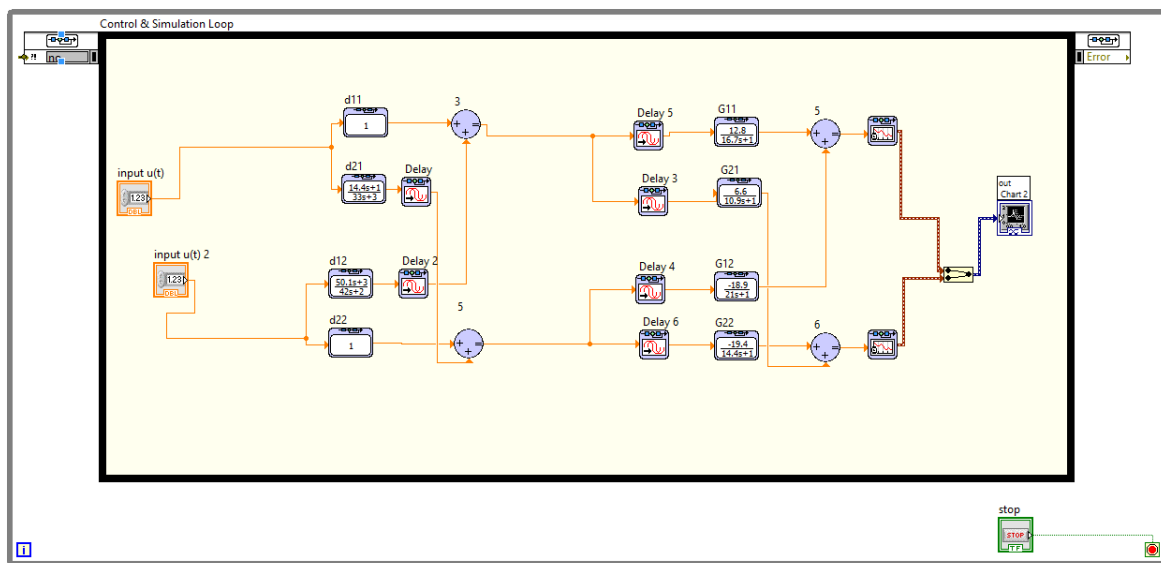


Figure 19: Backend of the decoupler

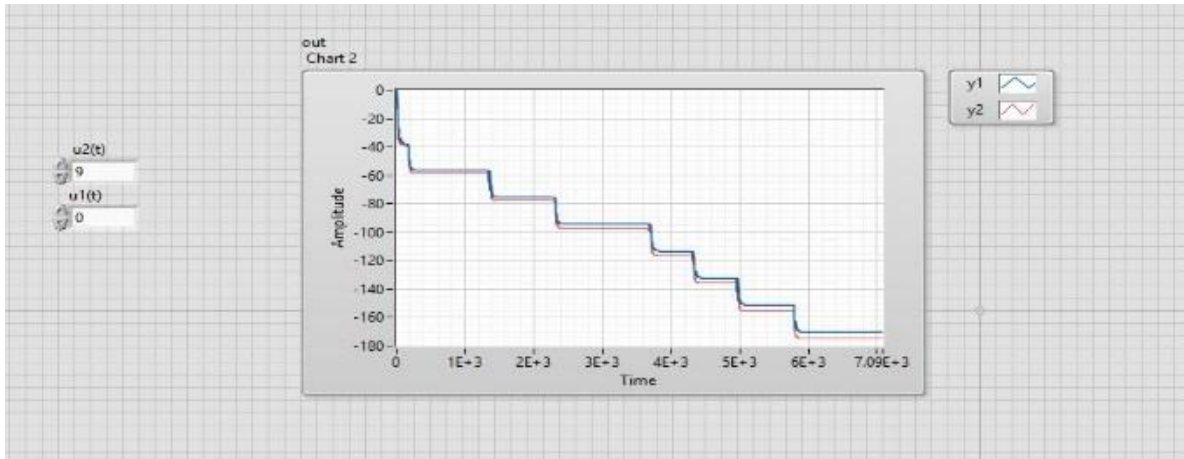


Figure 22: result before decoupling

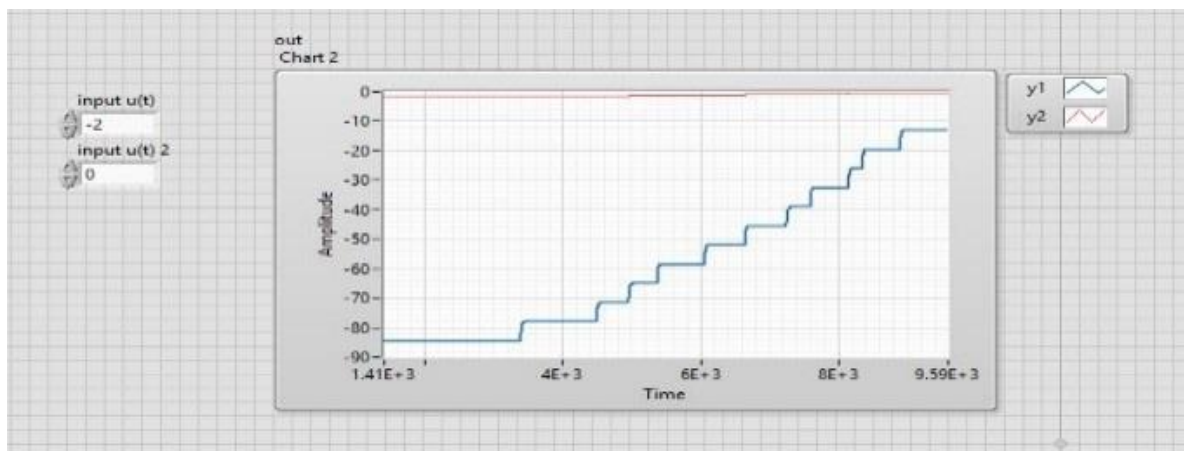
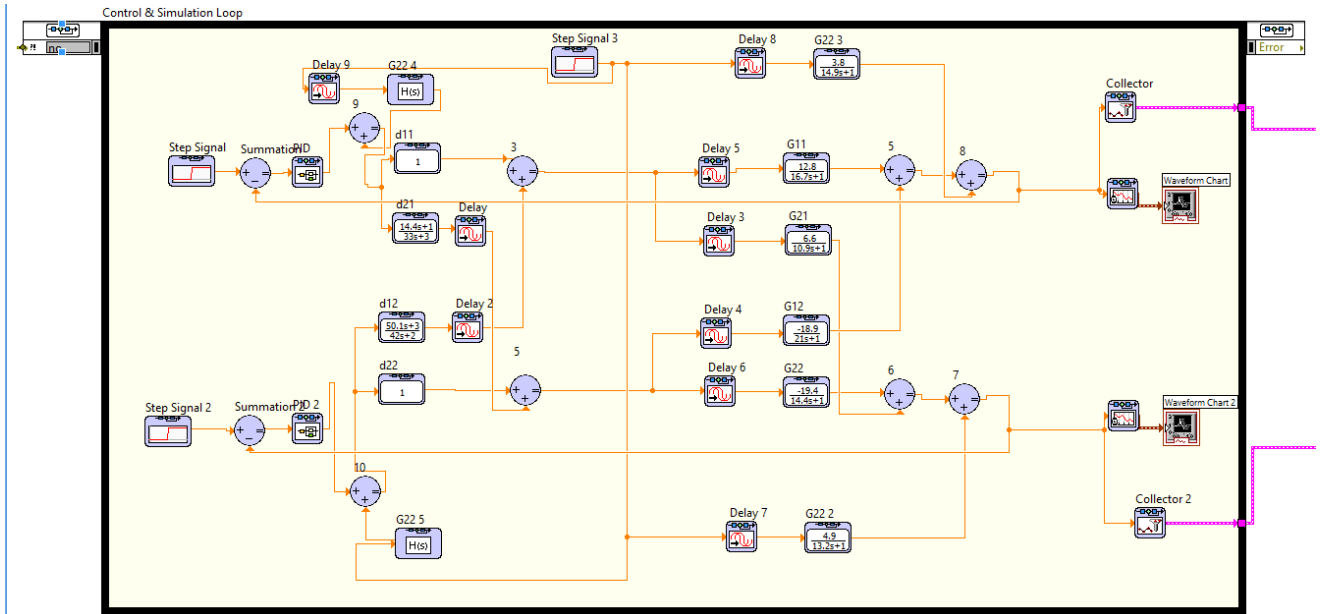


Figure 20: Result after decoupling

As you can clearly see that in the first diagram change in the one input gives the change in both the outputs, But in the second condition we can see that change in the first input is only changes the value of the first output only and this leads to the successful decoupling of the outputs.



By using the 63.2% of approximation we have the transfer function of process 1 and 2 after decoupling are

$$g_{p1}(s) = \frac{4.1e^{-2s}}{8s+1}$$

$$g_{p2}(s) = \frac{-9.5e^{-4s}}{6s+1}$$

Calculation of the IMC and PID tuning parameters as mentioned in the above chapter the tuning parameters will be as follows and the results Feed forward tuning parameters are calculated as below:

$$g_{d1}(s) = \frac{-g_{f1}(s)}{g_{p1}(s)}$$

$$\Rightarrow (-1) \frac{3.8e^{-8.1s}}{14.9s+1} * \frac{8s+1}{4.1e^{-2s}} = e^{-6.1s} (-0.926) \frac{8s+1}{14.9s+1}$$

$$g_{d2}(s) = \frac{-g_{f2}(s)}{g_{p2}(s)}$$

$$\Rightarrow \frac{4.9e^{-3.4s}}{13.2s + 1} * \frac{6s + 1}{9.5e^{-4s}} = \frac{(0.515)6s + 1}{13.2s + 1}$$

$$g_{d2}(s) = \frac{(0.515)6s + 1}{13.2s + 1}$$

The Tuning value are:

	Gain	Lead	Lad	Delay
Process1(dis)	-0.926	8	14.9	6.1
Process2(dis)	0.515	6	13.2	-

Table 1: Tuning parameters of feedforward controller

The tuning value of the IMC and PID controller:

	KC	Ti	Td
Process1(PID)	0.146	4	1
Process2(PID)	-0.0315	8	2
Process1(IMC)	0.2	7.5	1.375
Process2 (IMC)	-0.15	9	0.88

Table 2: Tuning parameters of IMC and PID

5.3 Schematic of the Lab view model for calculating the disturbance

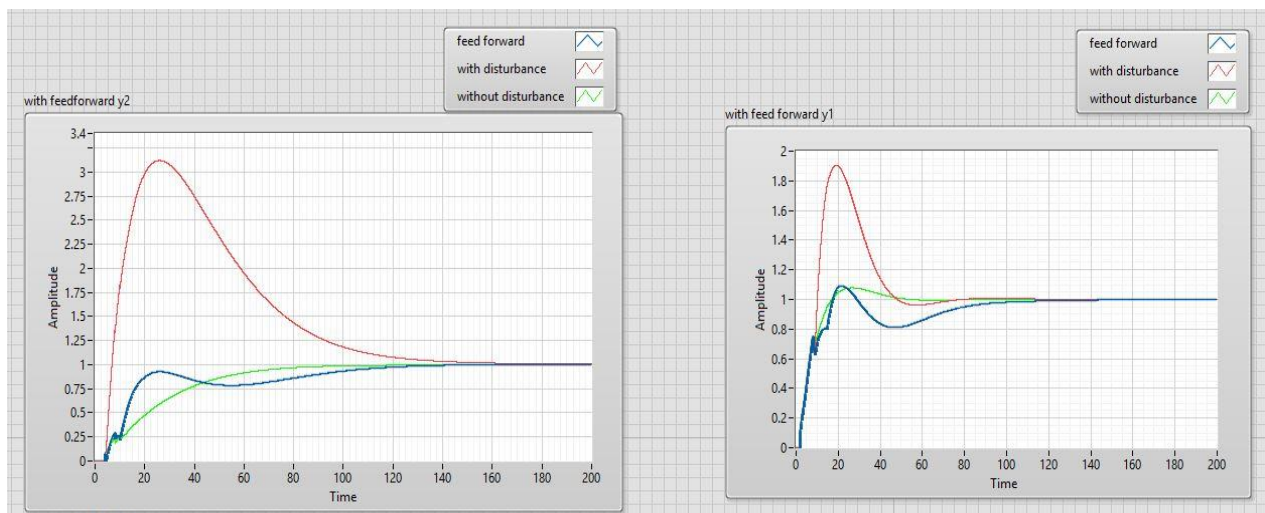
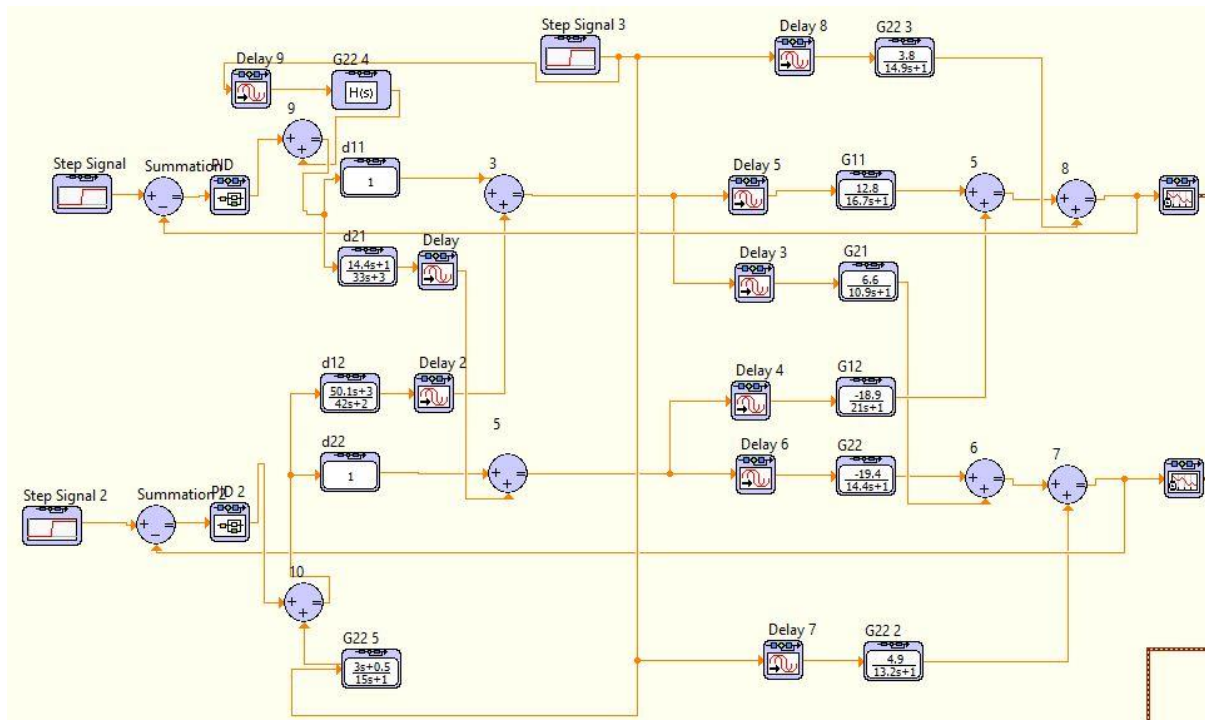


Figure 21: Output of the PID controller

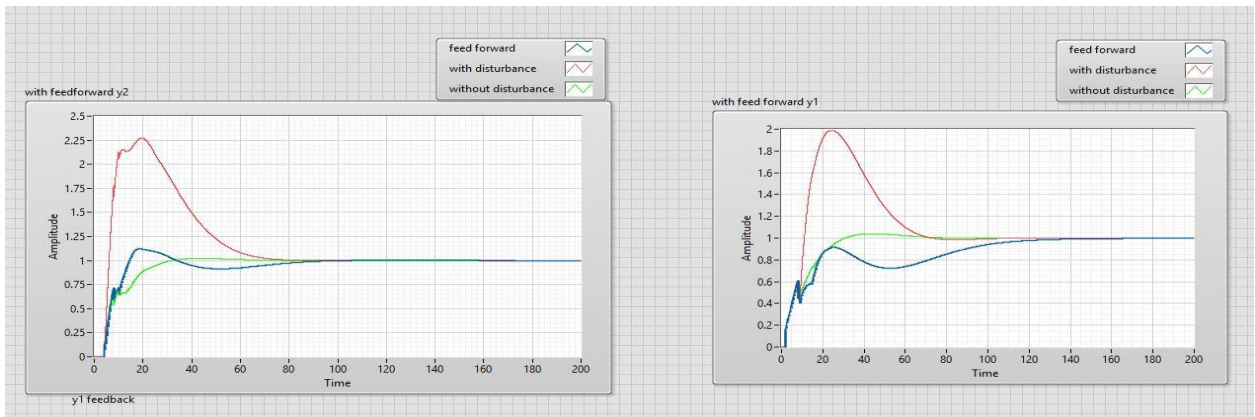


Figure 22: Output of the Internal Module controller

5.4 Calculated characteristics of the output response of the step input

Non Disturbance	Settling time	Overshoot(%)	Peek response
PID-Process1	39.09	8.73	1.08
PID- Process2	69.450	0	0.9994
IMC- Process 1	25.738	2.41	1.02
IMC-Process 2	27.23	0	1.0005

Table 3: output without disturbance

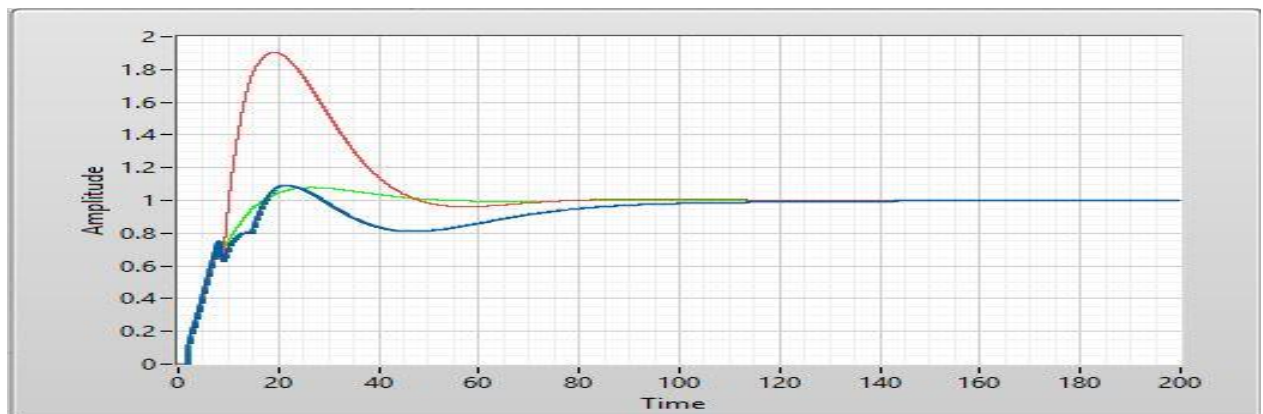
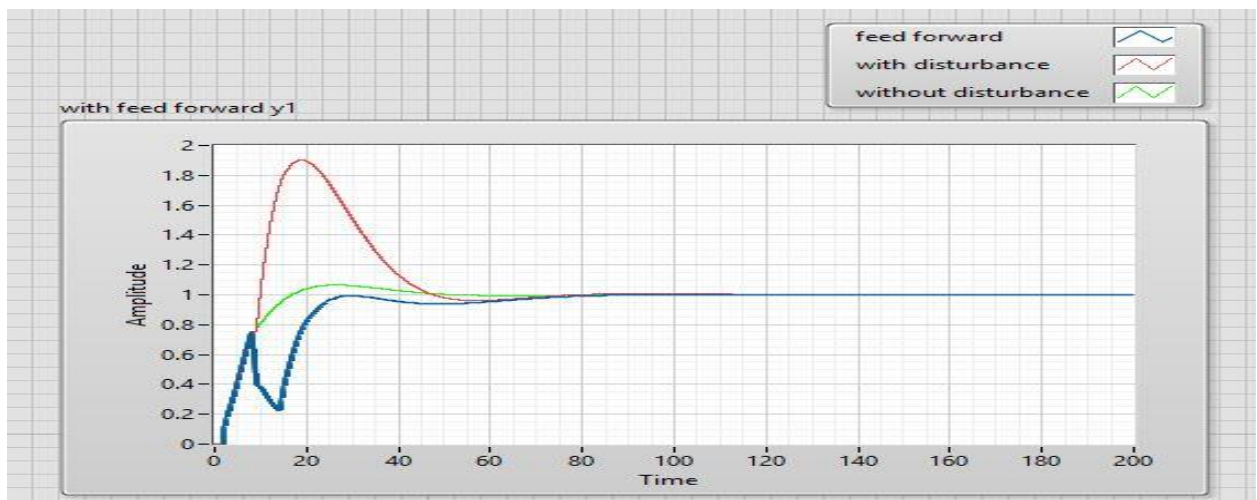
With disturbance	Settling time	Overshoot(%)	Peek response
PID-Process1	43.9	90.32	1.90
PID- Process2	128.341	210.59	3.1211
IMC- Process 1	63.63	98.6	1.98
IMC-Process 2	63.218	127.02	2.27

Table 4: Output with disturbance

With lead lag	Settling time	Overshoot(%)	Peek response
PID-Process1	58.839	0.11	1.0001
PID- Process2	99.915	0.09	1.0015
IMC- Process 1	70.24	0	0.9989
IMC-Process 2	69.935	12.35	1.125

Table 5: Output with feed forward

Effect of lead lag compensator and what if the feed forward estimate the process even faster than process and the compensation method:



Tuning parameters of the lead lag compensator are as follows:

Lead lag	Gain	Tn	Tp	Delay
Process 1	-0.926	8	14.9	6.1
Process 2	0.515	6	13.2	---
Modified P1	-0.926	8	25	5.9
Modified P2	0.5	6.2	15	---

Table 6: Modified feed forward variable

5.5 Results of robotic control:

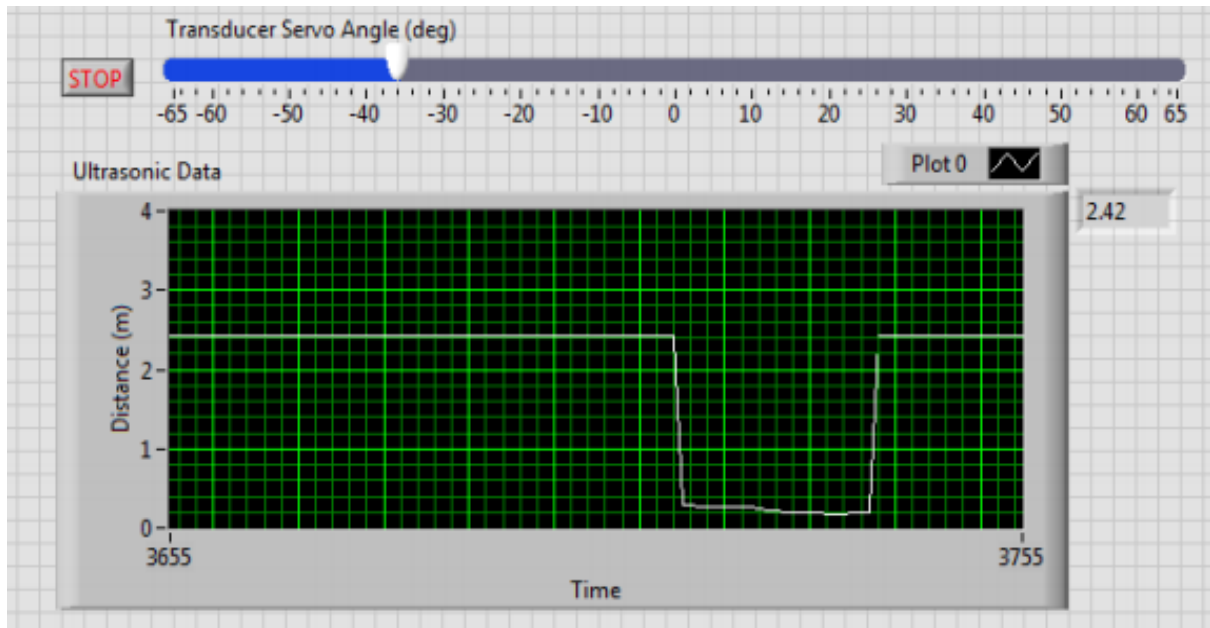


Figure 23: Output of ULTRASonic sensor

We have result of the sonic sensor in uni -directional we can specify the angle at which transducer with respect of robot and it shows the object at different distances

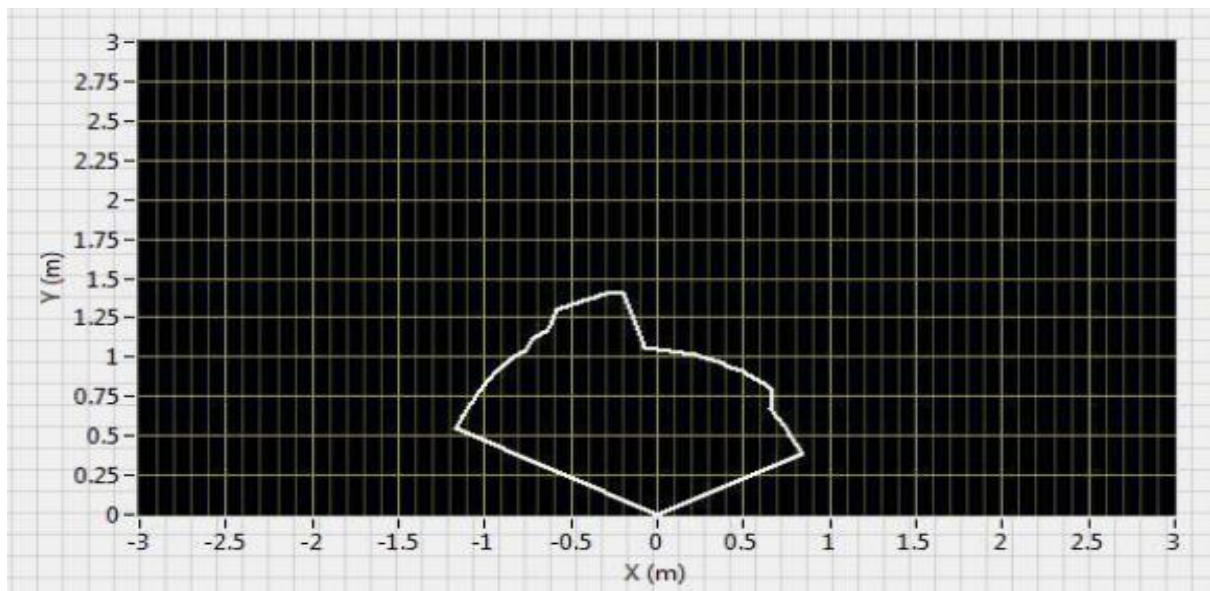


Figure 24: Output of Ultra Sonic sensor with 2d calibration

We have used servo motor which rotates the sonic sensor which creates the ability to detect the objects in the 2 Dimensional in front of the robot.

5.6 Calibrating the motor and optical encoder

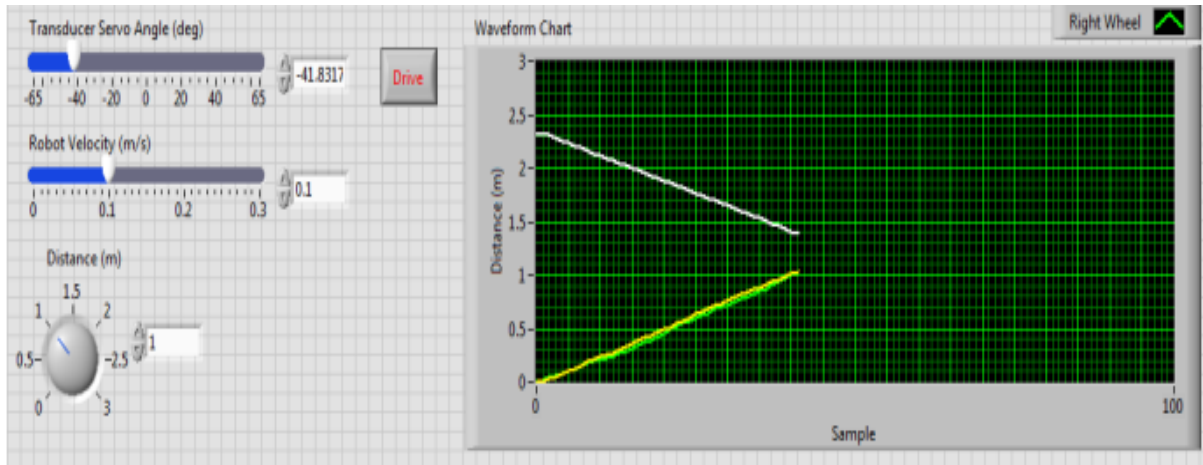


Figure 25: Output of the motor

We adjust motor velocity and the robot is moving towards an obstacle the above white line shows the sonic sensor value and the below line shows the distance moved by the robot with respect to time

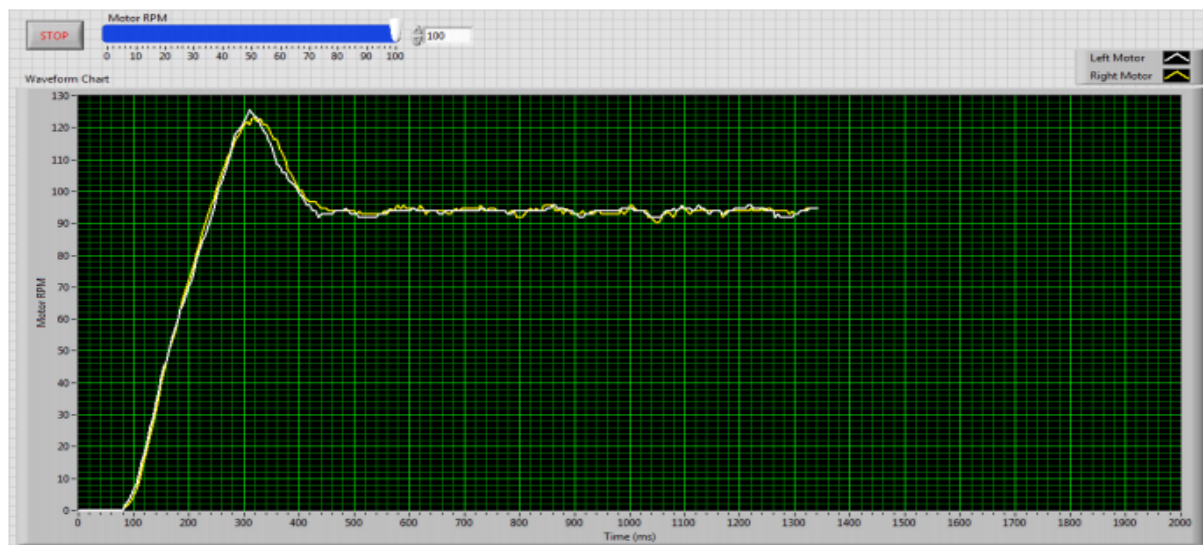


Figure 26: Output of the PID controller

The above graph is a PID controller which controls the speed of the motor as per specified by the user.

5.7 Obstacle avoidance by the robot:

Front end and back end of the mobile robot control is With help of the acquired knowledge I had implemented two new programs that the robot can perform:

Obstacle avoidance by the robot travelling specific distance that a user can specify.

Controlling the robot manually by using a mobile phone using the Team viewer software.

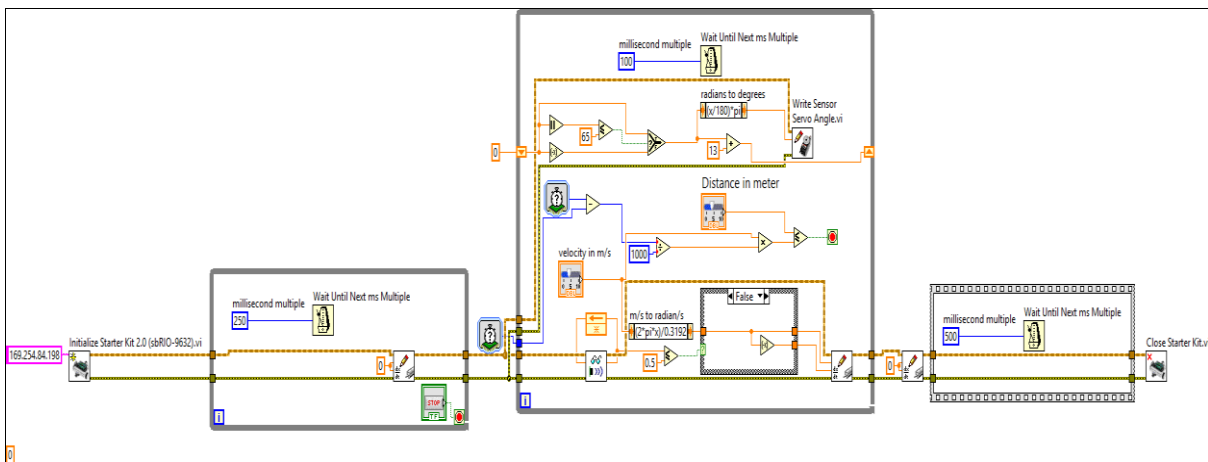


Figure 27: Labview Schematic Lab code

Front end of the robotics control which can move for a certain specified distance that is given by the user with the velocity that is also specified by avoiding the obstacles.

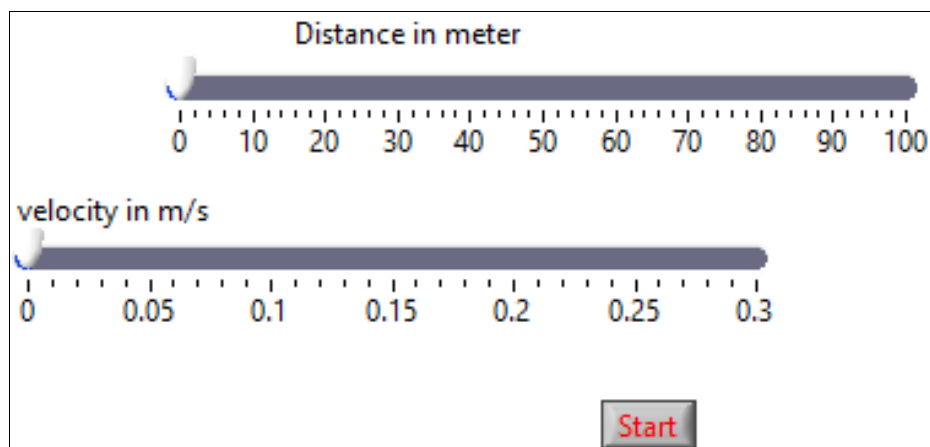


Figure 28: Front panel for robotic control

5.8 Wireless control of robot using mobile:

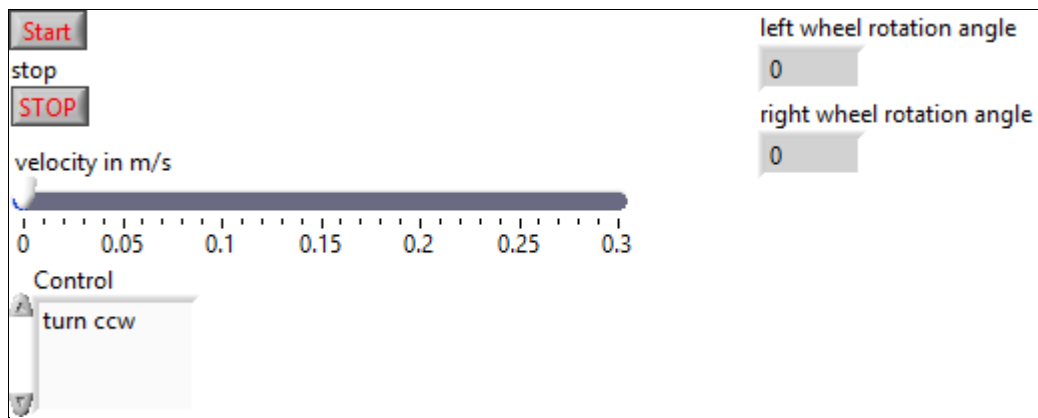


Figure 29: Mobile control front panel for robotic control

The above diagram is the front end of the remote control of the robot using mobile which can move in the direction specified using the mobile.

Backend program looks like this.

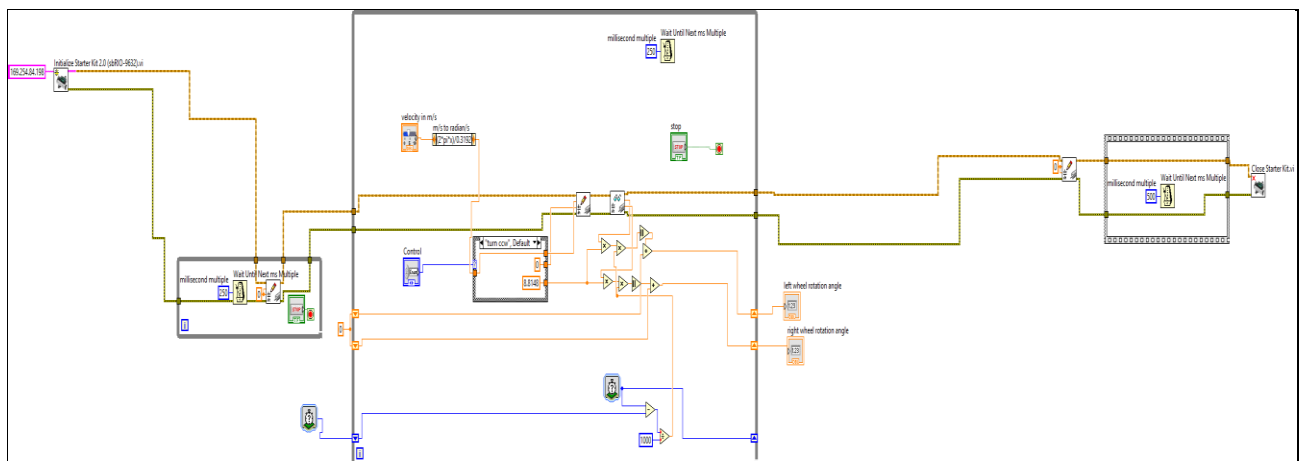


Figure 30: Mobile control back end program schematic

Chapter 6

Conclusion and Future work

6.1 Conclusion:

The peak overshoot of system increases when we introduce disturbance in process and by using feedback & feed forward combination overshoot is minimized. Peak overshoot of interacting system is less than that of non-interacting system.

IMC has better performance characteristics than a PID controller whether in the presence of disturbance or not. But PID is the dominant controller in the industries because the design of IMC requires the mathematical model of the process where PID is just like a Plug in.

The Lead-Lag compensator should have a slower response than the process in order to compensate the disturbance in the process without having miscellaneous action as we have faced in the above response.

As per the robotics, we analyze how the sensor calibration, data acquisition, and controlling actuators are combined to make the robot work in the way we want to work, just like a whole process plant works and for the future work we can develop the .

6.2 Suggested future work:

Further internal model control is designed for MIMO distillation processes such as 2×2 wood and berry process, 3×3 Ogunnaike and Ray Distillation Process, Dokuas and Luyben 4×4 Process. Lead-Lag based internal model control and modified internal model control are used for several process uses in process industries given as heat exchanger system, boiler drum system, CSTR etc. Comparison between IMC based PID controller, Lead-Lag based IMC and Modified IMC is also a very good task for MIMO distillation column. For robotics part with the help of wireless router through which we can wirelessly contact the robot and could send the instruction to the robot. With the help of the camera interface with the robot we can analyze the object and can be used for both surveillance and space technology.

Reference:

1. Bequette, B.W., Process control: modeling, design, and simulation. 2003: Prentice Hall Professional.
2. Grosdidier, P., M. Morari, and B. Holt, "Closed-Loop Properties from Steady-State Gain Information," *Ind. Eng. Chem. Fund.*, 24(2), 221–235 (1985).
3. Alatiqi, I. M., and W. L. Luyben, "Control of a Complex Side stream Column/Stripper Distillation Configuration"
4. Hulbert, D. G., and E. T. Woodburn, "Multivariable Control of a Wet Grinding Circuit,"
5. Visioli, A., A new design for a PID plus feedforward controller. *Journal of Process Control*, 2004.
6. Nordfeldt, P. and T. Hägglund, DE coupler and PID controller design of TITO systems. *Journal of process Control*, 2006. 16(9): p. 923-936.
7. Gargari, E.A., et al., Colonial competitive algorithm: a novel approach for PID controller design in MIMO distillation column process. *International Journal of Intelligent Computing and Cybernetics*, 2008. 1(3): p. 337-355.
8. "Effect of tuning Parameters of a Model Predictive Binary Distillation Column" by Rakesh Kumar Mishra, Tarun Kumar Dan, National Institute of Technology-Rourkela
9. "Mobile Robotics Experiments" with DaNI By DR. ROBERT KING COLORADO SCHOOL OF MINES
10. LabVIEW robotics programming by National Instruments.
11. "USER GUIDE NI sbRIO-961x/963x/964x and NI sbRIO-9612XT/9632XT/9642XT Single-Board RIO OEM Devices" By NI
12. "Getting Started with the LabVIEW™ Robotics Module" BY NI.