LabVIEW BASED ONLINE CONROL OF PMDC MOTOR

A THESIS SUBMITTED FOR PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

MASTER OF TECHNOLOGY
IN
CONTROL AND INSTRUMENTATION

SUBMITTED BY:

NITIN SINGH
(2K12/C&I/10)

UNDER THE SUPERVISION OF

PROF. NARENDRA KUMAR

Mr. A. R. KULKARNI

DEPARTMENT OF ELECTRICAL ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi, India

2014

Dedicated to my Brother

DEPARTMENT OF ELECTRICAL ENGINEERING

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042, INDIA

CERTIFICATE

This is to certify that the Thesis entitled, "**LabVIEW BASED ONLINE CONTROL OF PMDC MOTOR**" submitted by **NITIN SINGH** bearing roll no. 2k12/C&I/10 is a record of bonafide work carried out by him, in the department of Electrical Engineering of Delhi Technological University, New Delhi, under my supervision and guidance in partial fulfillment of requirement for the award of the degree of Master of Technology in Control and Instrumentation Engineering during session 2012-2014.

………………………………………… …………………………………….

Prof. Narendra Kumar
Supervisor
Electrical Engineering Department
Delhi Technological University
Delhi

Mr. A. R. Kulkarni
Astt. Professor, Supervisor
Electrical Engineering Department
Delhi Technological University
Delhi

ACKNOWLEDGEMENT

It is a great pleasure for me to acknowledge the assistance and contributions of few individuals to the successful completion of my Master's Thesis without whom it would be very difficult for me to complete the task.

I would like to especially to give my thanks to my adviser, **Prof. Narendra Kumar**, and **Astt. Prof A. R. Kulkarni** Department of Electrical engineering, Delhi Technological University (formerly Delhi College of Engineering) for his valuable time and inputs during the project. He always stood by me during the problem session and despite their very busy schedule, he always gave priorities to my queries and problems. It is for his help and support that I am able to successfully finish the project and understand the concepts clearly.

I would like to thanks **Mr. Karan Singh**, Foreman and **Mr. Bharat kasyap**, former Technical Assistant in Control system Laboratory, for providing the Lab facilities and their valuable technical assistance and time.

Finally, I want to express my deepest regards to my parents and friends for being very kind and supportive towards me throughout the course of thesis completion.

Date: ------------------------------------

Nitin Singh

ABSTRACT

The revolution of Internet-enables instrumentation is emerging as a revolution in Measurement and Automation. New standards are being developed for transmitting data and connecting instruments to the internet. The main purpose of this thesis is to design and develop a system to integrate various online experiments and able to monitor and control them over Internet using LabVIEW. LabVIEW is a graphical programming package capable of data acquisition, data analysis, data representation and real time remote control. Written communication in LabVIEW refers to communication via internet where the user can write messages to another user. Generally written communication is not found in any programming related software but LabVIEW makes this possible using internet protocol (IP).

In this thesis an online system is developed which integrates several real time experiments and remote control access over Internet is provided using LabVIEW. The latest remote panel technology is used to provide the remote control access. This thesis presents the implementation of online control of Speed of DC motor with the help of LabVIEW. In this software, a Build-in Web Server application which uses Internet Protocol (IP) for developing an online transmission process between client and server. Client has a control over the set point and Server has control over the speed. With the help of internet protocol, client provides the value of set point according to which the control actions are taken by the server. A system is developed to monitored and controlled the experiment remotely over Internet using LabVIEW.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABREVIATION

C

CMOS	Complementary Metal Oxide Semiconductor

D

DAQ	Data Acquisition

G

GUI	Graphical User Interface
GPIB	General Purpose Interface Bus

H

HTTP	Hypertext Transfer Protocol
HTML	Hyper Text Markup Language

I

I/O	Input/ Output
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
IMAQ	Image Monitoring Acquisition

K

K_P	Proportional Gain
K_d	Differential Gain
K_i	Integral Gain
K_U	Ultimate Gain

L

LabVIEW	Laboratory Virtual Instrument Engineering Workbench

N

NI	National Instrument

P

PC	Personal Computer
P	Proportional
PI	Proportional-Integral
PID	Proportional-Integral-Derivative
PXI	PCI-Extension for Instrument
P_U	Ultimate Period
PCI	Peripheral Component Interconnect

S

SSL	Secure Socket Layer

R

RS-232	Recommended Standard 232

T

TTL	Transistor-Transistor Logic

U

USB	Universal Serial Bus
URL	Uniform Resource Locator

V

VI	Virtual Instrument

CHAPTER 1

Introduction

1.1 Overview

Control system is a system that controls a variable by using error-sensing through a closed loop. The variable is known as controlled variable that must be maintained or controlled at some desired value. The desired value of the controlled variable is known as set point. The variable used to maintain the controlled variable at its set point is known as manipulated variable. A closed loop refers to the condition in which the controller is connected the process comparing the set point with the controller variable and determining corrective action. The objective of a control system is to use the manipulated variable and to maintain the controlled variable at its set point despite of disturbances. Disturbance is any variable that may cause controlled variable to deviate away from set point. Figure [1.1] represents the block diagram of a control system.

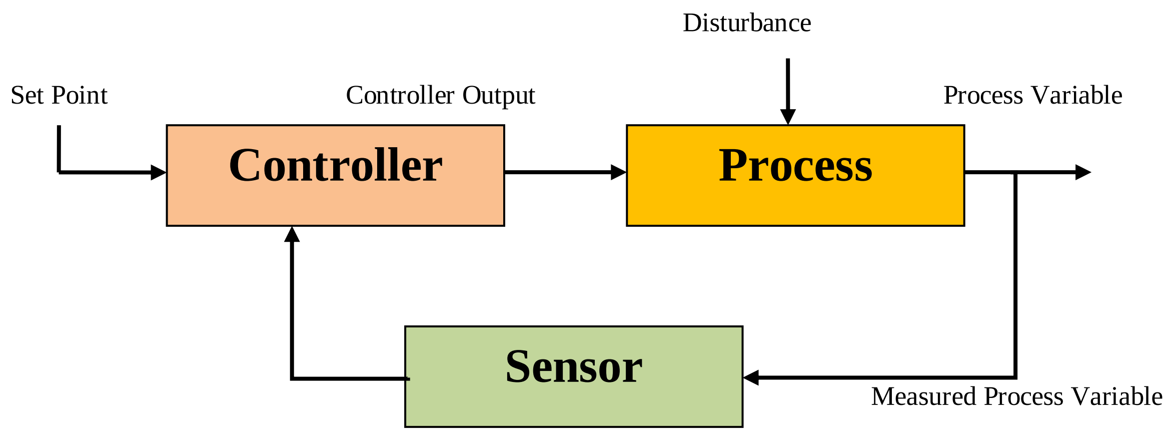


Figure 1.1 Block diagram of a Control system

The basic controllers used a process variable at its set point are P, PI, and PID. There are various networking protocols used in LabVIEW such as TCP/IP, Data Socket, and User datagram Protocol which provide online facilities to the users. There are the basic tools for communication over internet.

LabVIEW is a graphical programming environment which reduces the cost by giving a virtual existence of hardware, and hence termed as Virtual Instrumentation. This environment best suits for high-level or system-level design. The basic difference between "natural instrumentation" and "virtual instrumentation" is the software part of a virtual instrumentation. The software replaces complex and expensive equipment by simple and less expensive hardware. The program in this software consists of two parts: front panel and block diagram. Front panel shows the control and indicators while the main programming part is done in the block diagram which is usually kept up to user. Controls are the inputs and indictors are the outputs. Various operations have been done relating these controls and indicators.

1.2 Motivation of Work

In the academic and industrial communities, remote real-time control of processes is receiving considerable attention. In industries, when an instrument acquires some quantity and then measures it, its value has to be kept at some limit otherwise it may cause some problem. Hence the controllers are needed to solve the purpose. The controllers may function in controlling controlled variable such as temperature, pressure, flow level, humidity, etc. Using LabVIEW software it become easy to design a controller. In earlier year, much research has been done on the controller but not the various internet protocols found in LabVIEW. The internet protocols can be used in order to develop communication among users.

Data Acquisition (DAQ) is the process of measuring and processing any electrical or physical quantity such as voltage, temperature, humidity, pressure, etc. A "server-client" relationship is formed using the protocols. A server is a user who acquires and measures the variable, performs operations and if requires any help then sends a request to the client. Client is a user who accepts the request from server and provides service in return. The protocols help in long distance communication by making it very easy and fast. The combination of controllers and protocols is more helpful in industries. The client and server are connected using DAQ.

Depending upon the system to be controlled, the user will choose the type of control for the process. Among the controllers, ON/OFF controller is the basic one that has been implemented following by PID in order to compare. PID controller is best suited for industrial application since it gives zero offset because of I-component and better stability because of other components. Various technologies are developed in order to perform real time control using internet based technology. LabVIEW is one of the software packages used in process control application. It uses various protocols such as TCP/IP, Data Socket, etc that allows remote control using internet. Many universities have developed internet based process control laboratories for the students that would be helpful for distance education.

1.3 Objective of Work

The objective intends to measure and control the speed of PMDC motor using controllers. This is performed by using LabVIEW software, by developed a "Client" and a "Server". They communicate using Build-in Web Server and TCP/IP protocol where the server controls the speed using PID controller and the client has an option to decide the set point and control other parameters.

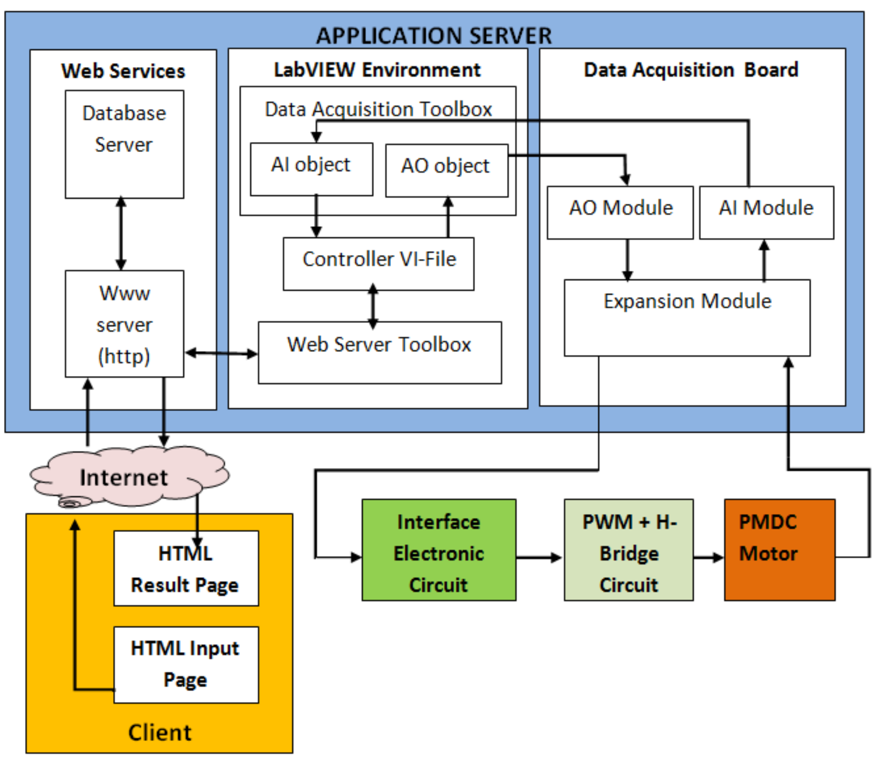


Figure 1.2: Block Diagram of the System Architecture designed

1.5 Literature Survey

The literature survey begins with the study of PID controller and self tuning methods. The tuning was first introduced by Ziegler-Nichols in 1940s, engineers finally found such practical and systematic way of tuning PID controller which improved performance and stability [1]. In 2007, J. H. Zhang and X. M. Liu, in 2009 have presented a self tuning methodology for PID controller based on recursive least square algorithm [2]. In 2009, J. Liu, P Zhang, and F. Wang, presented a real-time development system showing the control of position in a DC servo motor using PID controller. The paper also described the various algorithms PID algorithms to control position using data acquisition [3]. In 2010, F. Faizan, F. Farid, M. Rehan, S. Mughal, and M. T. Qadri, implemented a discrete PID on pendulum with an idea to balance an inverted pendulum using PIC microcontroller and discussed a various tuning algorithms of PID controller [4]. In 2011, S. K. Sahoo, discussed a modulus hugging approach for designing PI and PID controller in order to control the speed of DC motor using various algorithms [5].

In 2013, Sufendi, B. R. Trilaksono, S. H. Nasution, and E. B. Purwanto, designed an aircraft movement are controlled by embedded computer. The motion of aircraft is controlled by discrete PID controller and tuned using Ziegler-Nichols method. A structure and design of hardware was performed in Matlab platform [6]. In 2013, M. H. A. Jalil, M. H. Marzaki, N. Kasuan, and M. N. Taib proposed a paper on preventing the PID windup phenomenon. The first order model plus dead time is used for calculation of PID paaramter and tuned using Z-N method [7]. In 2014, Kim Jungsoo, M. M. Sabry, D. Atienza, and K. Vaidyanathan, have presented a PID control variable fan speed and eliminate the speed oscillation caused by temperature quanization [8].

In 2004, Jose Sanchez, Sebastian Dormido, Rafael Pastor, and Fernado Morilla, proposed a work on Web-based monitoring and control for remote laboratory. The motive of such research is to provide distance learning environment to students. The work is successfully implemented for inverted pendulum using Jave/Matlab environment. A structure is based on web-orientation and the remote experimentation is supervised over the internet [9]. In 2007, S. Bogosyan, M. Gokasan, A. Turan, and R. W. Wies have developed a remote accessible monitoring and control of electric drive machines. The Client-Server communication has buildup in C/C++ using WxWidgets. The electrical drives are placed in the laboratory can control over the internet using

TCP/IP protocal [10]. In 2010, H. Wang, J. Yao, and H. Cheng have developed a laboratory for distance education and industrial automation control. The web-based remote process control works efficiently for double holding water tank. Web-server is developed using ActiveX control and Java applet to transfer data between client and server. Client can control the water tank process plant remotely over the internet using Web Browser [11]. In 2011, Subhransu Padhee and Yaduvir Singh**,** have presented a paper on data acquisition, supervisory control and data logging for plant having multiple boilers. In the proposed plant, multiple variable and large data is logged in the database from the outfield for analysis and supervisory control [12]. In 2012, C. J. Hong, L. K. Luong, and S. Y. Chark, have developed an embedded server from 16- bit microcontroller by Microchip PIC24FJ256GB106 with the Ethernet controller. The designed system is used a Javascript to emulate http page to control AC power devices through web browser via LAN or the internet [13]. In 2013, I. Titov, and E. Titov, implemented a remote server using WebPager tool from Labicom.net. As the LabVIEW code runs with WebPager, the HTML file having front panel corresponding to the LabVIEW code is automatically generated on web pages [14].

CHAPTER 2

Process Plant

In this chapter, we will go through the process plant used in project and its characteristics and applications. In later part, the simulation circuit and hardware implementation of driving circuit and other will be discussed.

Almost in every engineering and automation industries we see electric motor. Basically it is a converting electrical energy to mechanical energy. Direct Current (DC) motor comes in wide range and in various sizes specifications, from 10's to 1000's horsepower. Some of the application of DC motor are, rolling mills, actuators, trains, control valve, and in small electronic experiments.

2.1 PMDC Motor

2.1.1 Principle of Operations

The basic principle of Permanent magnet DC motor is based on energy conversion, that is when a electrical energy is applied to a current carrying conductor lying perticular to a magnetic field produces a mechanical force or torque [16] [17] in figure 2.1. The exerted force can expressed as:

$$F = BIL \quad Newton \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

Where,

B = magnetic field flux density

I = current flowing through conductor

L = length of conductor

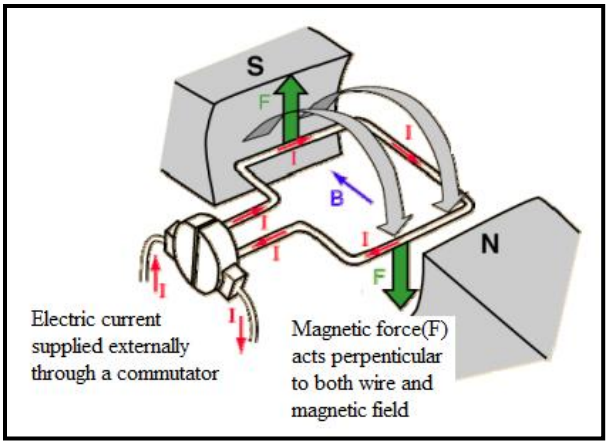


Figure 2.1: Torque produces in DC motor

In actual DC motor, numbers of coil are wounded together on the rotor, and all experiences the same force resulting in rotation. Lager the value of current, more will be magnetic field and greater force experienced by the conductor produced greater torque. Induced voltage in the armature winding and flux relation is shown in figure 2.2 at different instances of angle of rotation.

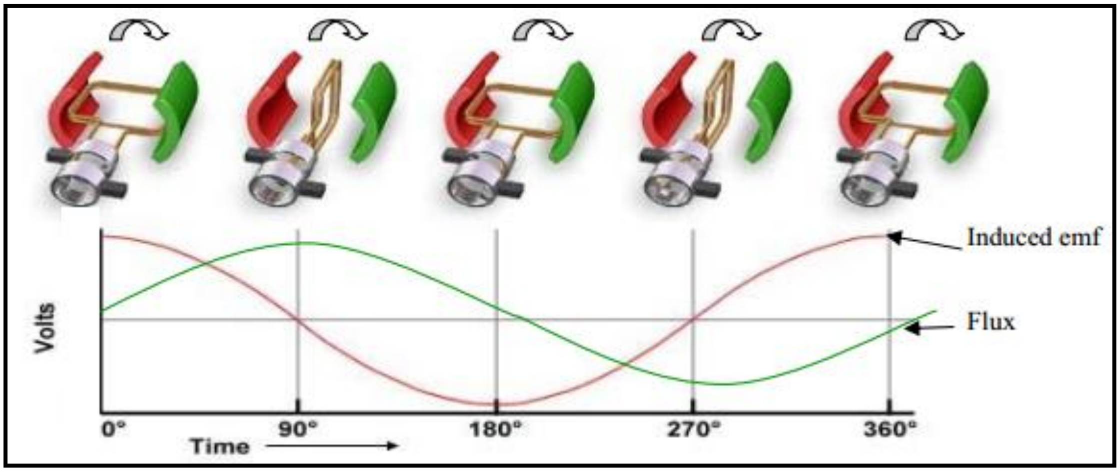


Figure 2.2: Induced voltage of DC motor

2.1.2 Speed Control of DC motor

Speed control means to drive the motor at variable speed, which can be attained by manual control, and or by using automatic control mechanism. There are several method of speed control as given below:

- Speed control by varying armature resistance
- Speed control by varying field current
- Speed control by armature voltage variation
- Ward Leonard method: combination of armature voltage and field current

In this thesis, the DC motor speed is controlled by using armature voltage variation method is implemented, so as discussed in upcoming section 2.1.3.

2.1.3 Speed Control by Armature Voltage Variation

In armature control method voltage source is applied to the armature terminals while keeping armature resistance and field current constant. The figure 2.4 below shoes the equivalent circuit diagram of armature voltage control.

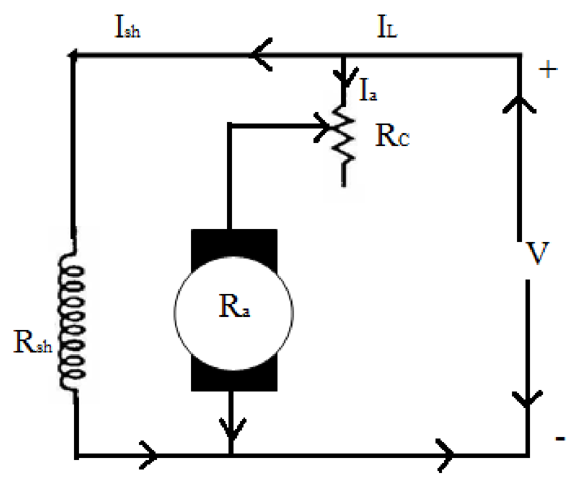


Figure 2.3: Eq. Circuit Diagram of armatur Control method

Form the circuit diagram, we have

$$E_b = V - I_a R_a \quad \text{..................................... (2)}$$

Also, $$E_b = \frac{PZN\emptyset}{60A} \qquad \text{.............................(3)}$$

Therefore, $$\frac{PZN\emptyset}{60A} = V - I_a R_a \qquad \text{.............................(4)}$$

Finally we have, $$N = K * \frac{(V - I_a R_a)}{\emptyset} \qquad \text{.............................(5)}$$

Or $$N = K * \frac{E_b}{\emptyset} \qquad \text{.............................(6)}$$

Where, N= Speed of DC motor

E_b = Back emf

Ø = Flux per pole

2.2 DC motor under Study

A PMDC motor manufactured by Tachno Instruments is used for control purposes for speed control as shown in figure 2.6. Permanent magnet DC motor is used as a process plant in this project. It is small rated motor generally used for position control purposes. It is coupled with a tachogenerator through a plastic coupling to sense the voltage signal proportional to the speed.

2.2.1 Speed Control

Since the given motor is a permanent magnet type, hence the field of the motor is fixed and armature voltage control method is used as the voltage across the motor terminals changes the speed of the motor is changed.

From the equation (1), (5) and (6) the speed of a D.C. motor is given by:

$$N \propto \frac{E_b}{\emptyset} \qquad \text{.............................(7)}$$

Or $$N = K * \frac{(V - I_a R_a)}{\phi}$$

As shown in above equation, as the voltage is changed the speed of the motor is changed.

2.2.2 Ratings

The motor specifications are [18]:

- Operating Voltage : 12 VDC
- Full Load Current : 1.2 A
- Rated Speed : Not mentioned
- Torque = 750 g/cm

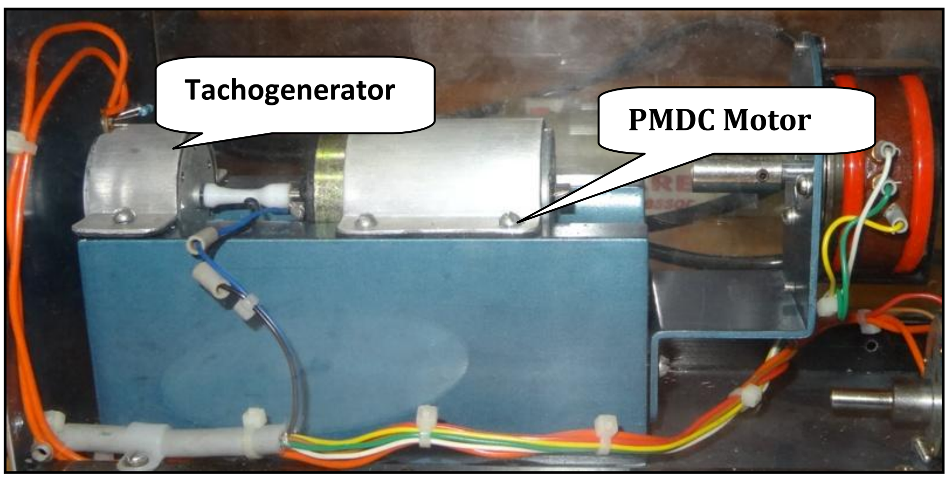


Figure 2.4 PMDC Motor-Tachogenerator Set up

2.3 Hardware Design

2.3.1 General

Before developing the hardware for the first time, it must simulate on the software. For this purpose Proteus software has been used. Proteus is a Simulation software for designing electronic circuit board designed by LabCenter Electronics. It consist all necessary electronics components and IC's packages.

2.3.2 Motor Driver Circuit

The control signal generated by the Data Acquisition Card is a analog voltage signal, but due to limited capability of the card, it cannot drive the motor hence a driver circuit is implemented which drives the motor. The driving signal is generated in the following steps:

- The analog voltage signal is converted to a equivalent PWM signal, as the voltage varies the duty cycle of the PWM signal is varied from 0 to 100 percent.
- The PWM signal generated in the step 1 is given to a H- bridge circuit which has power transistors in H configuration to power up motor in both reverse and forward direction. Due to variable duty cycle the H-bridge acts as a chopper and the average voltage output at its terminals changes with change in duty cycle.

2.3.3 Voltage to PWM converter

The control signal or analog voltage produced by the Data acquisition card of range 0-5V is converted to 0-100% duty cycle PWM signal is simulated in figure below.

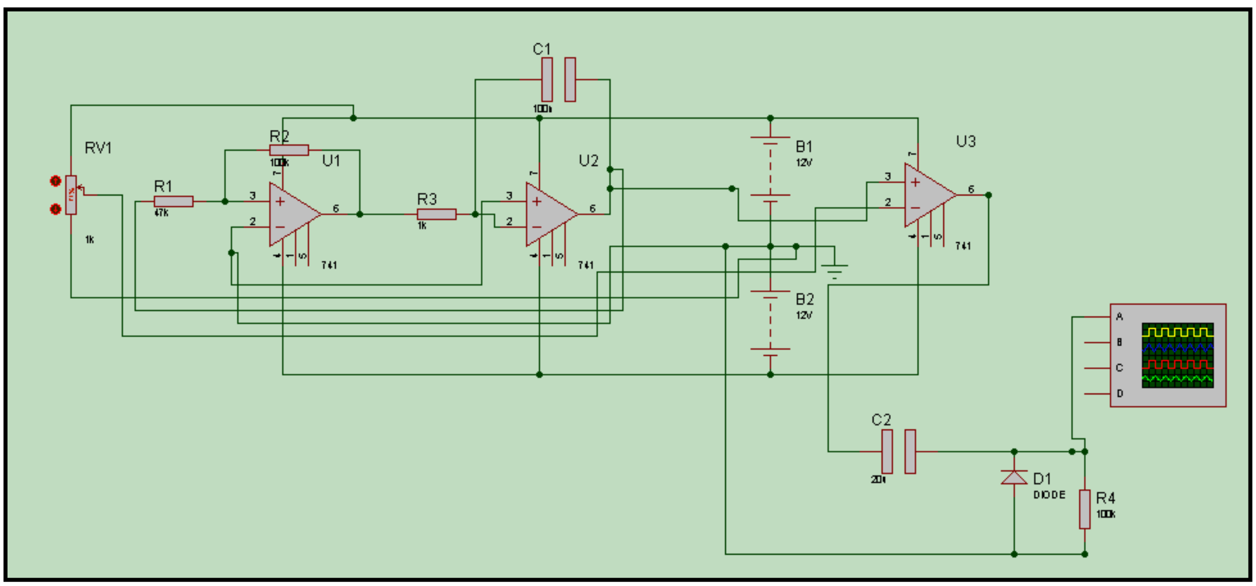


Figure 2.5 Simulated PWM generator

The circuit consists of three op-amps one is used for producing oscillation and one is used as integrator and last one is used in comparator mode. the output of the second is fed into positive terminal of first op-amp and a triangular waveform is generated at the output of the second op-amp, the third op-amp compares the triangular signal with the input analog signal, and a square wave is produced at the output, as the voltage is changed the duty cycle of the square waveform is changed and a PWM signal is obtained. Since the PWM signal is negative valued also, a diode-capacitor clamper is used to clamp it positively.

In the final fabricated PCB shown in figure 2.8 and in Appendix I, all the three op-amps are replaced by IC 3524 [19], who's function is also similar to convert 0-5V control signal to 0-100% duty cycle of PWM respectively.

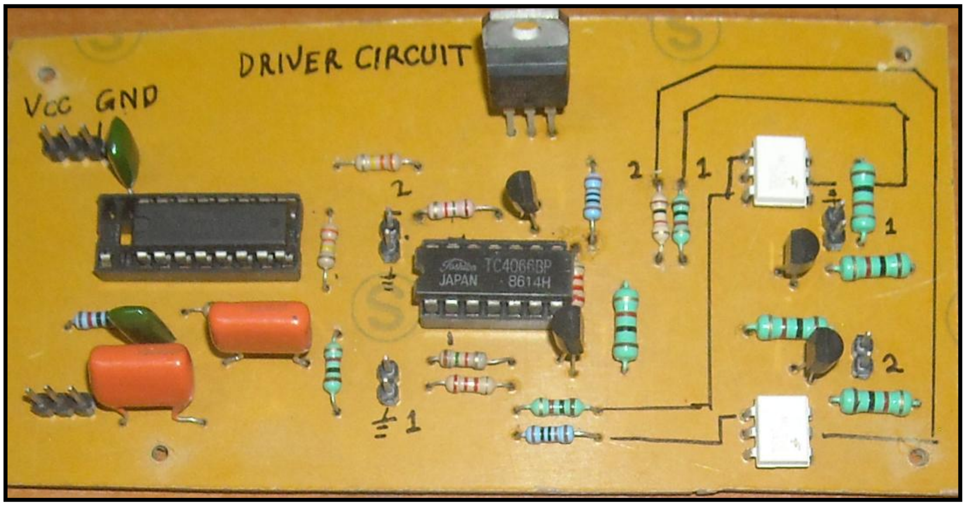
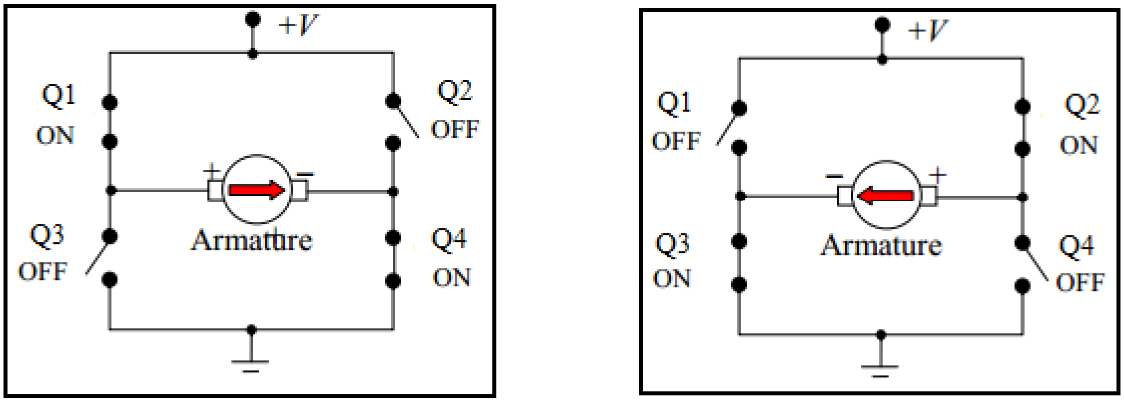


Figure 2.6: Driving Circuit PCB

2.3.4 H-Bridge

H- Bridge is a switching circuit consisting of power transistors, which are arranged in an H configuration, this circuit is used to give voltage to the motor in order to drive in both the directions. Hence two quadrant operation of the drive is obtained. The simulated H-bridge circuit shown in figure 2.10 consists of 2-npn transistor (TIP 122) and 2pnp transistors (TIP 127), the transistors have enough power rating to drive 12V DC motor without burning it [20]. At a time, Q1andQ4 combination or Q2andQ3 combination will be ON to drive the motor in either direction as shown below in figure 2.9 (A)(B). These transistor are digitally operated using IC CD4066, it's a 14-pin dual package having 4 digital input and 4 digital outputs [21].



A B

Figure 2.7: (A) shows Forward motion and (B) shows Reverse Motion

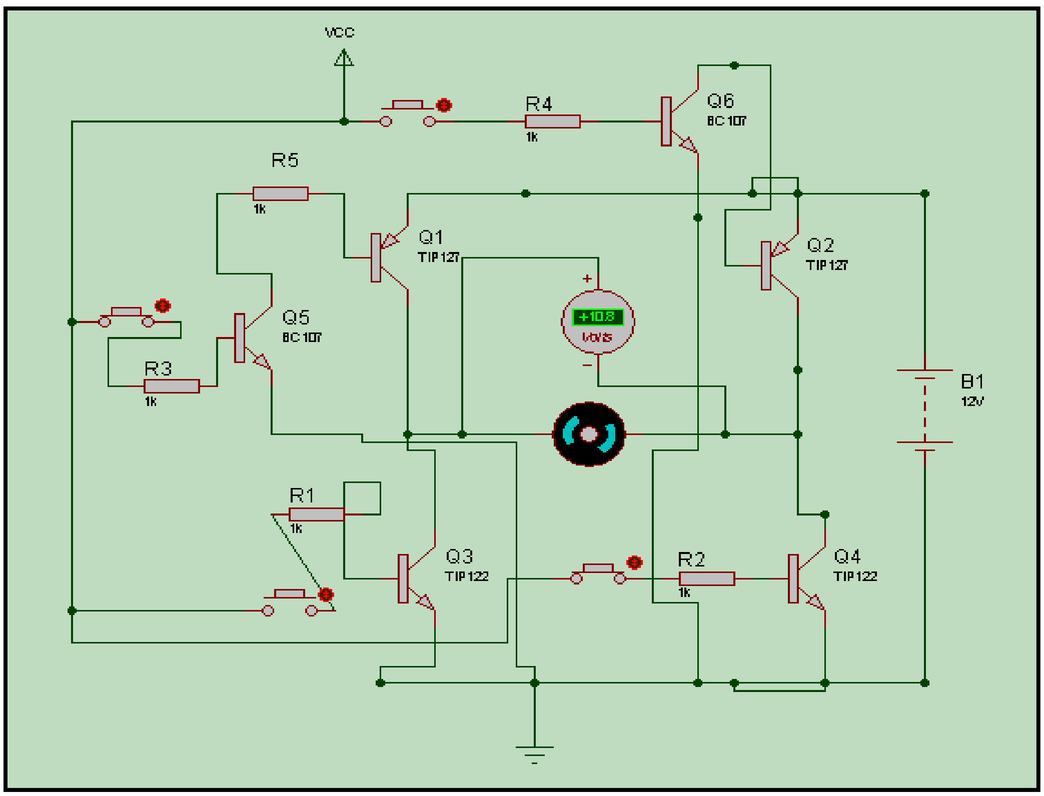


Figure 2.8: Simulated H-bridge Circuit

After successfully operation of simulink model, the final PCB is fabricated which is shown below in figure 2.11 and in Appendix I.

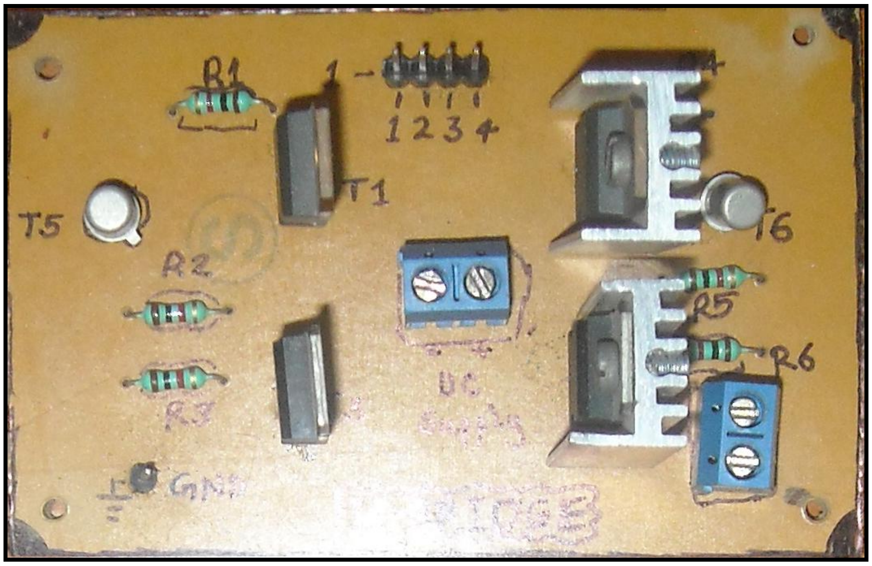


Figure 2.9: H-bridge Printed circuit board

CHAPTER 3

Control Strategies

Control strategies are necessary for any system to perform accurately. Some of these are given below.

3.1 PID Controller

A Proportional-Integral-Derivative (PID) controller is a basic feedback control loop algorithm widely used in industrial, chemical, and automation process control systems. A PID controller minimizes the error between a desired set point and the measured process variable by calculating the error value. The corrective action then apply to adjust the process rapidly to keep minimum error [22].

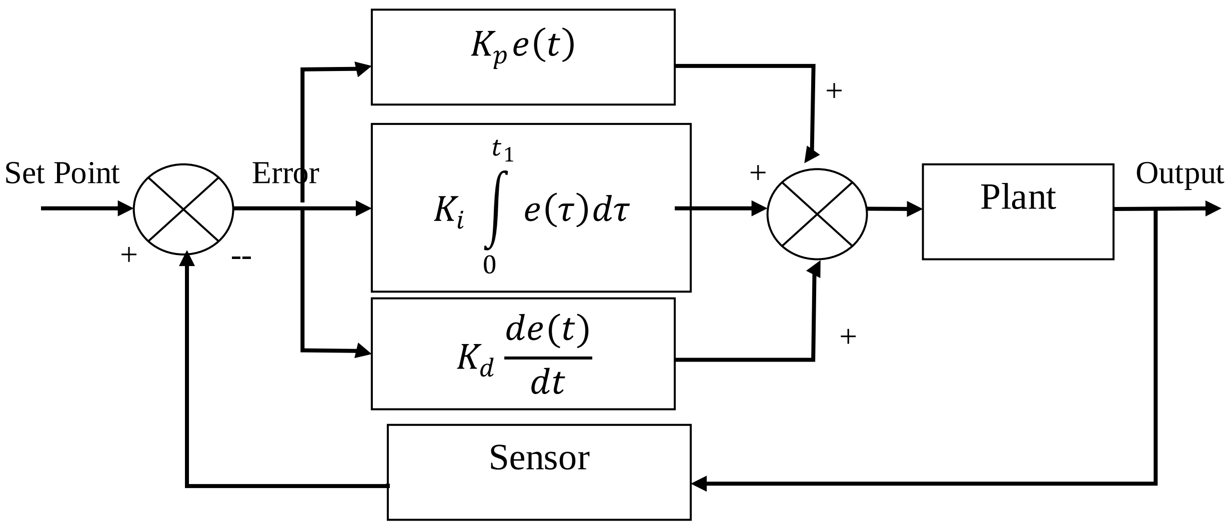


Figure 3.1: Basic Block Diagram of a conventional PID controller

For the PID controller represented in figure 3.1,

Output of PID controller is represented by equation 1 [22],

$$u(t) = K_p e(t) + K_i \int_0^{t_1} e(\tau)d\tau + K_d \frac{de(t)}{dt} \quad \ldots\ldots\ldots\ldots\ldots \quad (1)$$

Problem in structure of a controller arises during designing a control system, defining structure and controller parameters and tuning them.

3.1.1 Proportional

The proportional term is given by equation [2]

$$P_{out} = K_p e(t) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

If K_p is high then the system becomes unstable. If it is too low then control action may not be effective for responding to disturbances.

3.1.2 Integral

The integral term is given by equation [3],

$$I_{out} = K_i \int_0^t e(\tau)d\tau \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (3)$$

The integral term eliminates the steady-state error by the movement of the process towards set point. Nut it may cause overshoot so an additional component is required.

3.1.3 Derivative

The derivative term is given by equation [4],

$$D_{out} = K_d \frac{de(t)}{dt} \quad \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots.. (4)$$

The derivative term reduces the overshoot and improves the stability og the process. But it slows the transient response of the controller.

Stability

If the tuning parameters of PID controller are not chosen properly, the process may become unstable and the instability is caused by excess gain. PID controller algorithm is given by equation [5],

$$u(t) = K[e(t) + \frac{1}{T_i} \int_0^{t_1} e(\tau)d\tau + T_d \frac{de(t)}{dt} \quad \dots\dots\dots\dots\dots\dots (5)$$

3.2 Closed-Loop Controller Tuning Methods

There are various methods for tuning a PID controller:

1. Manual Tuning
2. Ziegler-Nichols Method
3. Tyreus-Luyben Method

3.2.1 Manual Tuning

The manual tuning is accurate and provides stability to the process but time consuming. In the manual tuning, to obtained the exact tuning value the user must have to adjust the individual parameters i.e proportional, integral and derivative terms using numbers of trails and errors.

3.2.2 Ziegler-Nichols Method

The most widely used method for tuning PID controller was introduced by John G. Ziegler and B. Nichols in 1942[22] [1]. The Zeigler-Nichols closed loop method involves the following steps for tuning.

- Initially K_i and K_d gains are made set to zero. Increase the proportional (P) gain until we get the continuous oscillation in the response. At this point, the maximum value of P gain is attained and on further increase in gain value makes the system unstable.
- The P gain value at which the closed loop system starts oscillating is now known as ultimate gain (K_u) and peak-to-peak period is called as ultimate period (P_u).
- Using K_u and P_u value, the P, PI, and PID controller can be easily tuned using the table 3.1 below.

Control Type	K_p	K_i	K_d
P	0.50 K_u	-	-
PI	0.45 K_u	$P_u/1.2$	-
PID	0.60 K_u	$P_u/2$	$P_u/8$

Table 3.1 Ziegler Nichol parameter

3.2.3 Tyreus-Luyben Method

Tyreus-Luyben have proposed the tuning parameter approach that results in less oscillatory and less sensitive to changes in the process. This tuning method is somehow similar to the above approach as discussed in section 3.2.2. The tuning method is developed by Tyreus-Luyben [22]. In his approach, both K_i and K_d gain are initially made to zero. The Proportional gain is keep on increasing till we starting get the oscillation in the response i.e ultimate gain K_u and oscillation period P_u. These parameters are then used to tune the controller using the table 3.2 below.

Control Type	K_p	K_i	K_d
PI	$K_u/3.2$	$2.2P_u$	-
PID	$K_u/3.2$	$2.2P_u$	$P_u/6.3$

Table 3.2 Tyreus-Luyben Parameter

3.3 Characteristics of PID Controller

Table 3.3 shows the comparison between Rise Time, Overshoot, Settling Time, Steady-State error for different tuning parameters of PID controller. Effects of each controller K_p, K_d and K_i on a closed-loop system are summarized in the table shown below. By varying one of these three variables made other two variables to suffer. The table can only be used as a reference to evaluate K_p, K_d and K_i.

Parameters	Over-shoot	Rise-Time	Settling-Time	Steady-State Error
K_p	Increases	Decreases	Small-Change	Decreases
K_i	Increases	Decreases	Increases	Eliminated
K_d	Decreases	Small Change	Decreases	Small-Change

Table 3.3: Effect of K_p, K_d and K_i on PID Controllers

A proportional controller can reduce the Rise Time and Steady-State error but cannot remove it. An integral eliminates the Steady-State error, but makes the transient response sluggish. A derivative controller improves the stability of the system, reduces the Overshoot and improves the transient response.

3.4 PID Palette in LabVIEW

The figure 3.2 shows the functions palette of PID in LabVIEW. The following PID palette VIs is obtained from the Control and Simulation Toolkit [24].

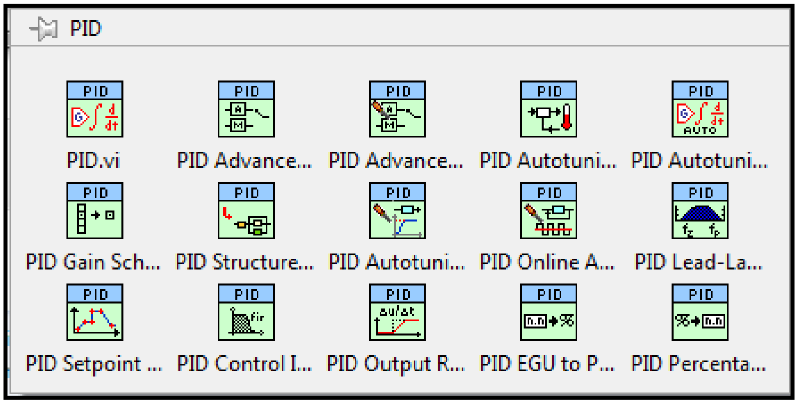


Figure 3.2: LabVIEW PID palette

CHAPTER 4

Programming in LabVIEW

4.1 Introduction to LabVIEW

NI LabVIEW (Laboratory Virtual Instruments Engineering Workbench), the productive software components of the Virtual Instrumentation architecture, is the graphical development platform for test, design and control applications. LabVIEW is a highly productive graphical programming environment that combines easy to use graphical developments icons usually known as VIs with the flexible programming tool [25]. LabVIEW platform provide a user-interface graphical development environment which is simple to program.

With this type of prototype, it can be used in applications of controls, automations, real-time systems, and other general applications. The core programming is written in a tradition language such as C/C++ and Visual Basic which makes this software excels in data measurement and generation, control, and data analysis or treading [26]. Basically, in text-based programming languages there are syntax and compliers. As a user it is difficult to learn so many syntax to write program. Whereas, LabVIEW based graphical-programming provides more flexibility in structured data flow diagrams rather than text-based programming language and has a relatively high speed complier [27].

In LabVIEW, there are two windows pane one is for user-interface commonly known as front panel consists indicators, control and graphs. The block diagram contains source code of graphical programming. Beside front panel a front panel objects appear as a terminals on the block diagram which then interconnected to form a code program. LabVIEW majorly used in data acquisition and automation industries, it is capable of reading direct measurement such as current, voltage, pressure, force, torque, and speed and many more parameters.

4.2 Introduction to Virtual Instrumentation

The vision of virtual instrumentation revolutionized the way scientist and engineers with greater flexibility, better development time and lesser costs. LabVIEW codes are known as VI (Virtual Instrumentation). Virtual instruments are specific to hardware measurement and generations, tends to their modular connectivity to hardware approaches. There are three main components in the LabVIEW virtual instrumentation as discussed below:

1. Front panel – contains indicators, control, and graphs acts as a customize window.
2. Block diagram –runs on the backend of the front panel. It contains the front panel objects, structures, loops, and VI source code.
3. Icon and connector pane – it provide the input or output terminus to the VI. Upto 25 terminals could be assigned to a connector pane.

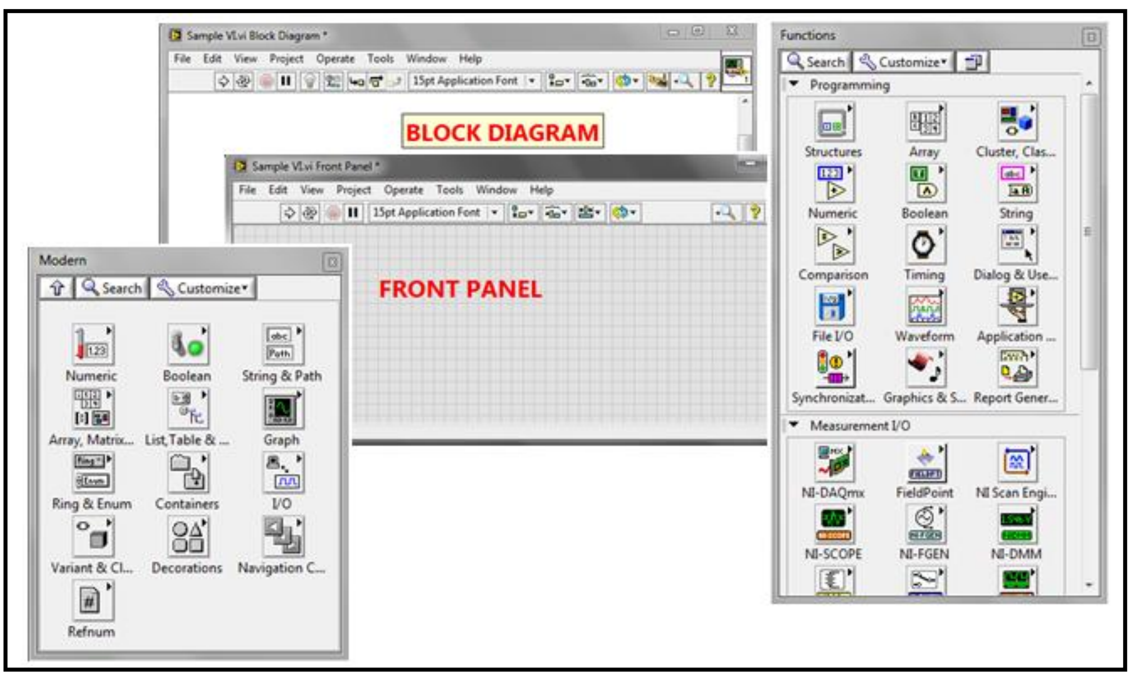


Figure 4.1: Development environment of LabVIEW

In LabVIEW control palettes only access to the front panel and contains indicators, graphs, controls, and other VI used to customize the front panel.

The function palette is only accessible to the block diagram. This function pane contain all function related to arithmetic, Boolean, timing, signals, and many toolbox such as data acquisition, control and simulation, etc.

4.2.1 Building the front panel

The front panel is the user interface of a VI. Generally, the front panel is design first, then a block diagram is design to perform tasks on the inputs and outputs created on the front panel.

The front panel is build with controls and indicators, which are the interactive input and output terminals of VI, respectively. Controls are knobs, push buttons, dials, and other input devices. Indicators are graphs, LEDs, and other displays. Controls simulate instrument input devices and supply data to block diagram of the VI. Indicators simulate instruments output devices and display data the block diagram acquires and generates.

4.2.2 Building the Block Diagram

Once the front panel is build, a code added using graphical representation of functions to control the front panel objects. The block diagram contains this graphical source code, which could be written in languages such as C/C++, VB, and java.

4.2.3 Local Variable and Global Variable

Local and Global variables share the data between the application, where wired connection could not possible.

LabVIEW contains two types of variable, to share a data between them. One is local variable and other is global variable. As name sounds, local variable share the data between the two or more local variables one the same VI. The global variables read and write the information among multiple VI running parallel or in network-published shared variables.

4.3 Data Formats in NI-DAQmx

Data format deals with the types of data that is read or written. Here, analog channel data formats are discussed [26].

4.3.1 Waveform

The waveform data format includes the channel name, timing, and unit information with the actual 64-bit scaled floating-point data.

For inputs tasks, the additional information can be used for a variety of purposes. For example, graph can be updated to show the timing information and include labels with the channel names. Because there is overhead associated with including this additional information, NI-DAQmx allows you to configure the information you want to include.

For output tasks, the timing information is the primary field that is useful. A library that generators a waveform can include timing information that sets up the timing for the output task.

When reading data, the waveform data includes the time when the first sample in the waveform was acquired, and the amount of time that elapsed between each sample.

4.3.2 64-Bit floating-Point Numbers

The 64-bit floating-point number format allows reading or writing scaled data with no additional information. This format is used to work with scaled data that requires higher performance than the waveform format provides. This format is also used because it is a better match for the libraries planned to use.

4.3.3 Unsigned and Signed Integers

The unsigned and signed integer format reads or writes data in the native format of the device. This format is used for maximum performance. The trade-off is that the application has to understand how to interpret and manipulate data that is not in engineering units.

4.4 Various Blocks Used in LabVIEW Programming

4.4.1 DAQmx Create Virtual Channel

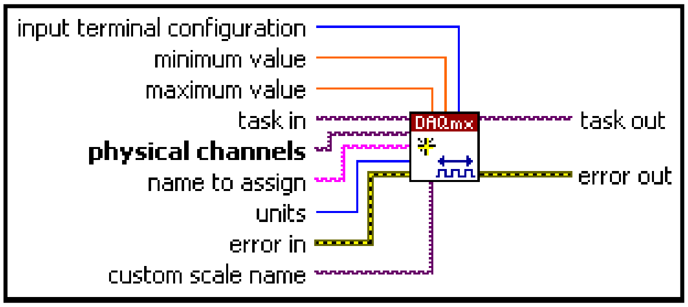


Figure 4.2 : Block diagram of DAQmx Create

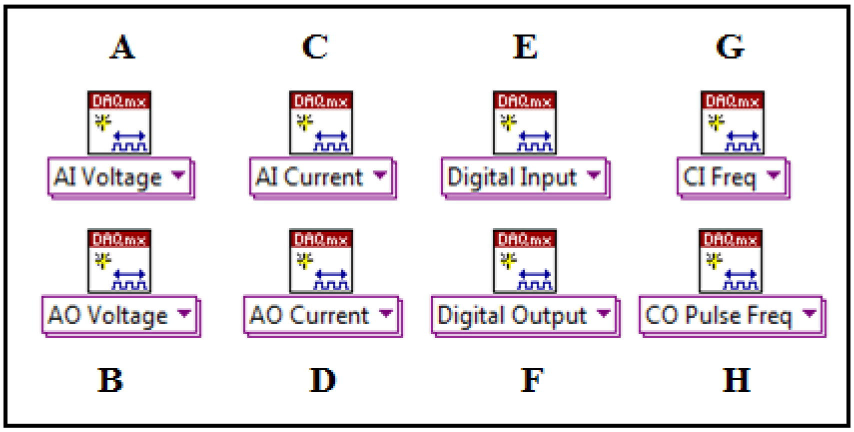


Figure 4.3: Different modes in DAQmx Create

In above figure 4.3, A and B are analog in and analog out voltage VI, and C and D are analog in and analog current VI. E and F are digital in and digital out type, and whereas G and H counter types VI. A virtual physical channel required an input task that can be an analog out, analog in, digital in, digital out, and counter. In Addition to this, analog in, analog out, and counter will use a certain minimum and maximum range value of inputs in configuration of channel. Optimization is carried out by using these minimum and maximum expected values of the signals for measurement and generation.

4.4.2 DAQmx Timing

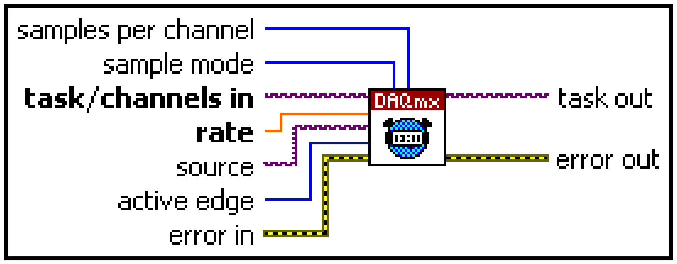


Figure 4.4: Block Diagram of DAQmx Timing

The DAQmx Timing function configurations palette is shown in figure 4.4. DAQmx Timing function is used to provide the timing for secure hardware data acquisition operation. This function tells the program about its operations i.e whether it is continuous or finite sampling type, the sampling rate or Samples/Sec to acquire or generate for finite interval operation can be configured from this function, and can also create a buffer when needed.

For hardware-timed operations that requires sample rate i.e, analog in, analog out, and counter, the DAQmx Timing function sets the sampling time, which can be an external or internal, and control its rate. The DAQmx timing clock controls the sampling rate at which data are acquired or generated. The acquisition and generation start every single pulse for each physical channel that includes in the given task.

4.4.3 DAQmx Start Task

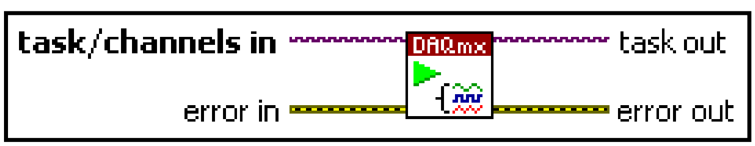


Figure 4.5: Block Diagram of DAQmx Start Task

This block can be used only once in a single acquisition operation. Figure 4.5 shows the block for DAQmx Start task function. The input and output can be seen in the above figure. This VI is an optional in some data acquiring application, for better performance and result one should use this VI. In hardware-timed generation application DAQmx Start task is preferred. In writing applications the VI should known when the task will starts.

4.4.4 DAQmx Read

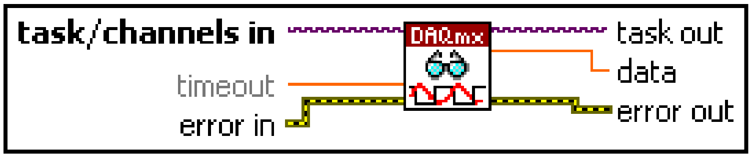


Figure 4.6: Block Diagram of DAQmx Read

Figure 4.6 shows the block for the DAQmx Read function. The DAQmx Read function reads samples from the specified virtual physical channel from the data acquisition task to be performed. The data can be of different type i.e analog, digital, and counter. The type of data, number of sample to be read, numbers of virtual physical channels, can be easily configured. Different forms of the DAQmx Read VI are shown below in figure 4.6.

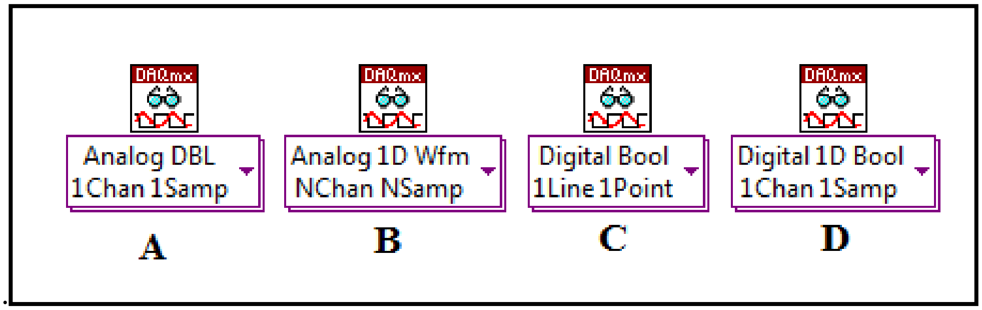


Figure 4.7: Different modes of DAQmx Read

Instance A and B are analog read function VI. In reading operation A can read one channel one sample in one timing cycle and B can read a data in the form of waveform from N-channel N samples. Whereas, instances C and D are of digital type read function VI. Instance C has a tendency to read a digital signal from the single port, and D can read a digital signal from the number of digital ports.

4.4.5 DAQmx Write

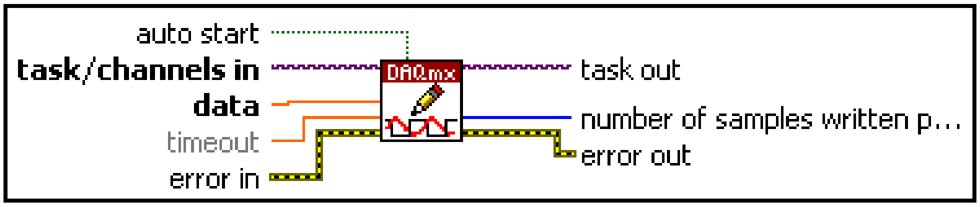


Figure 4.8: Block Diagram of DAQmx Write

The DAQmx Write block can be seen in figure 4.8. The DAQmx Write function acquires the samples from the specified virtual channel in the given task. The different forms of DAQmx Write function have different type of data generation i.e analog or digital. The data type, number of samples, and the number of physical channels can be easily configured. The different forms of DAQmx Write function are given below.

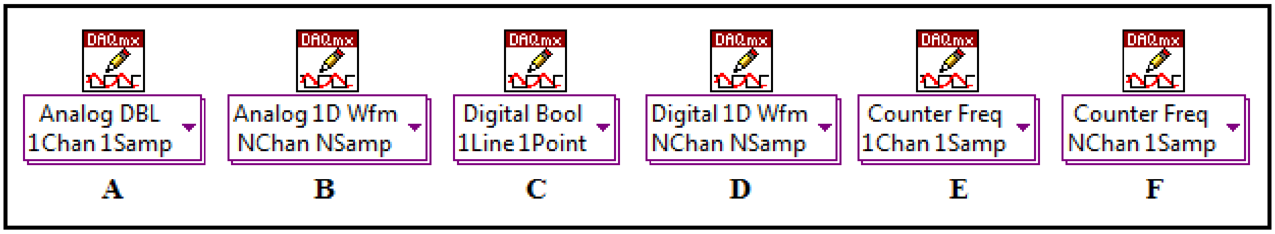


Figure 4.9: Various Modes of DAQmx Write

These blocks write the data to the virtual physical channel as assigned in the task. Different forms of DAQmx Write have different functions as discussed below. The block A and B can generate analog signal, with 1-channel 1 sample and N-channel N sample respectively. Block C and D can generates a digital signal and E and F is a counter.

4.4.6 DAQmx Clear Task

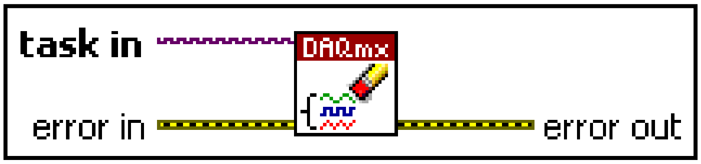


Figure 4.10: Various Modes of DAQmx Clear Task

The DAQmx Clear task function shown in figure 4.10 select only once at the end of task, to empty the buffer memory. Once the data is passed through DAQmx Clear function it has been deleted forever and cannot be recovered. DAQmx Stop function is different from this block, DAQmx stop the task but not empty the buffer.

CHAPTER 5

Data Acquisition

This chapter includes the basics of the data acquisition (DAQ), the importance of data acquisition in real world, how DAQ devices communicate with the computer, and how the data acquisition hardware acquires data. The DAQ device, drivers, application software and specification of DAQ device used in the experiment, are also discussed in later part of this chapter.

5.1 Introduction to Data Acquisition

Data acquisition is the process of measuring and electrical or physical phenomenon such as voltage, current, temperature, pressure and sound. These physical signals are acquired, get digitized and send to the computer. Both modular hardware and driver software needed for PC-based data acquisition. A simple block diagram shown in figure 5.1, working condition of DAQ devices connected with computer.

Data acquisition involves acquiring signals, analysis, storage and presentation on to a computer. Today, you can find many manufactured of DAQ devices along with their suitable connectivity, such as PCI, PCI Express, USB, Wireless, RS-232 instruments, and Ethernet. NI DAQ hardware devices are perfectly suitable for LabVIEW platform [28], because they compatible and easily get configured using NI MAX configuration tool.

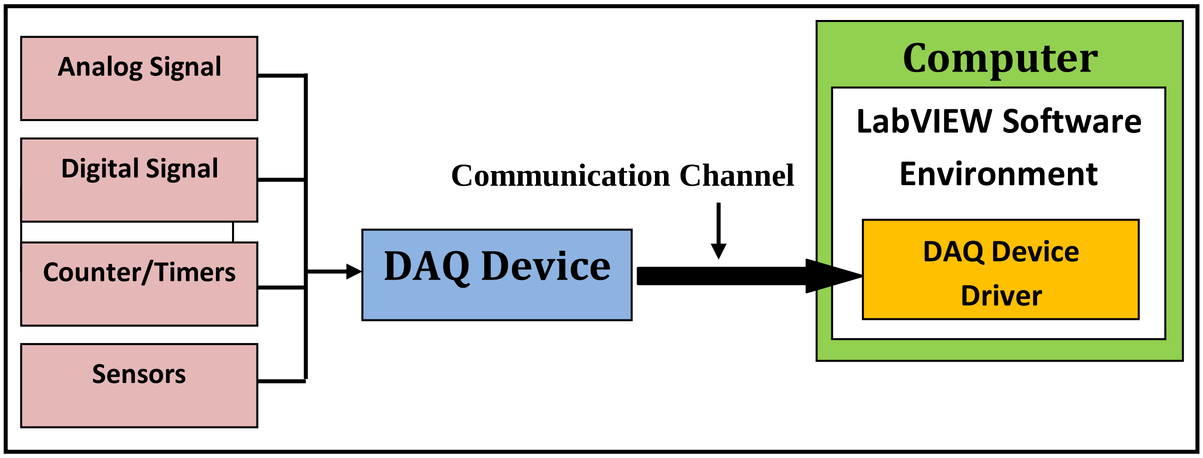


Figure 5.1: Block diagram of DAQ working

5.2 Signals/Sensors

A sensor or transducer is a device that measures the physical parameters and converts into a measureable electrical signal [29].

Transducers convert one form of energy into measurable energy form, however, different signal should be measure in different ways. Therefore, signals are divided into two categories: analog and digital. The table below shown some commonly used transducer corresponding to their physical phenomenon.

Physical Phenomenon	Transducers
Temperature	RTD, Thermocouple, Thermistor
Light	Photo Sensor
Sound	Microphone
Force and Pressure	Strain-gauge, Piezo-electric crystal
Acceleration	Accelerometer

Table 5.1: Some commonly used Transducer

5.3 Computer Communicate to DAQ Devices

The physical quantity can be measured through sensor. Now, this physical signal must be converted to standard electrical signal i.e analog or digital before delivered to the DAQ device. DAQ devices is only single component of entire system, direct connection of DAQ devices cannot be possible. The signal generated by transducer should be in limit of DAQ devices, otherwise it may get damaged. For this reason, in some cases signal conditioning is connected before the DAQ device, for signal in desired limit [28].

DAQ devices can be connected in two configuration is shown Figure 5.2. In case A, the DAQ card is placed inside the computer on PCI or PCI express slot. This type of configuration offer parallel communication which can acquire and generate data at a much faster rate. In case B, the DAQ device is connected externally, in this type of arrangements USB, Ethernet, and serial communication is possible. The given arrangements are best suited for control, automation, and distance learning.

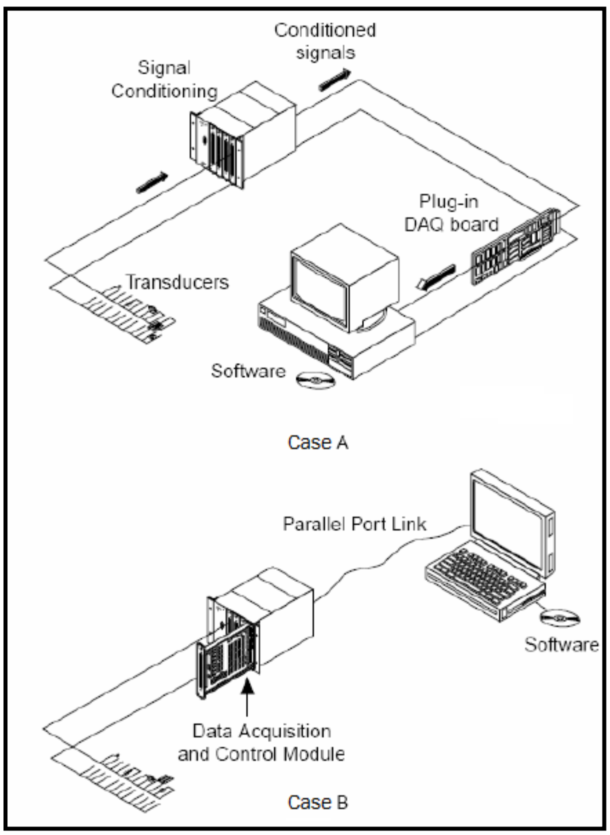


Figure 5.2: Different configuration of DAQ Device

5.4 Role of Software

Without software to control, drive a hardware and data acquisition does not work properly. Software transforms the raw or physical data recorded by data acquisition in a form which can understand by humans. Driver software is a communication layer between the computer and any hardware.

Driver software is written by a programmer having core-knowledge register-level programming. Driver software are unique for a particular device. In this thesis, NI DAQmx has been used as the driver software to communicate with the hardware.

5.5 Application Software

Application software is a clusters of different function in which you build a custom application in development environment that meets specific criteria, or can also be designed with present functionality. For this purpose, the user is not necessary to have knowledge of core instrument programming. Application software adds analysis, functionality and presentation capabilities to driver software. LabVIEW web server has been used as the application software.

5.6 Hardware Specifications

The NI USB 6009 DAQ is a 34-pin board which required power USB cable to communicate with the PC shown in figure 5.3. USB 6009 DAQ card is compatible with Labview platform and capable of read/write analog and digital signal at rate of 48K Samples/Sec [30].

Analog Input	Analog Input Channel	Differential	4
		Single-ended	8
	Input Resolution	Differential	14 bits
		Single-ended	13bits
	Max Sample Rate	48KS/s	
	Range	Differential	$\pm 5, \pm 10$
		Single-ended	± 10V
	Working Voltage	± 10V	
	Input Impedance	144KΩ	
	Over Voltage Protection	±35 V	

Analog Output	Channel	2	
	Resolution	12 bits	
	Max. Update Rate	150Hz	
	Range	0 to +5V	
	Output Impedance	50Ω	
Digital Input/Output	Digital Input/Output lines	P<0-7>	8 lines
		P<0-3>	4 lines
	Compatible	TTL,CMOS	
	Input Voltage	Low	-0.3V
		High	2V
	Output Voltage	+5V	
Bus Interface	USB 2.0	12Mb/s	

Table 5.2: USB 6009 DAQ Specification Table

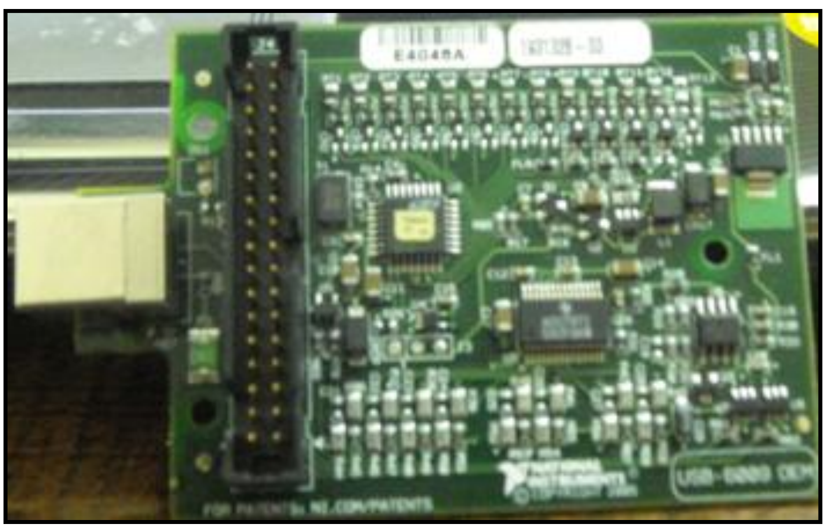


Figure 5.3: USB DAQ 6009

CHAPTER 6

Implementation of Web Server

6.1 Introduction

The Internet plays a significant role in real time industrial manufacturing, scheduling, monitoring and management. An extensive research work has led to the development of new technologies that uses the internet for supervision and control of the industrial processes. Internet-based control systems addresses the challenges that need to be overcome before the internet can be beneficially used not only for monitoring of but also remote control industrial plants [31].

In the last decade, the most successful network developed has been the internet that has proved a powerful tool for distributed collaborative work. The emerging internet technologies offer unprecedented interconnection capability. Internet-based control systems are characterized as globally remote monitoring by the internet [32]. In the recent years, internet-based control systems have gained considerable attention in science and engineering, since they provide a new and convenient unified framework for system control application.

Modern day process plants, construction sited, agricultural industry, petroleum, power distribution network, wireless sensors network, refinery industry and every other industry where data is of prime importance use wireless data acquisition, data processing and data logging equipments. Acquiring data from the field with the help of different sensors are always challenging. Essentially, Internet-based control systems have been developed by means of extending discrete control systems. The use of the Internet as a communication medium provides cost-effective, flexible and easy-to-access distributed control systems that are not limited to any geographical region.

6.2 Internet Based Remote Labs

An Internet based remote laboratory is meant to be a computer-based controlled laboratory that can controlled through some distance over an internet communication medium [33]. For this accessed, LabVIEW platform provide such communication with a remote panel control and monitored over the internet communication channel within a web browser. In the Internet based remote labs, the remote panel of server side is connected to the experiment through a proper connectivity and with the host computer a connected to the internet. The client can be any computer connected to the internet running a simple browser. Once connected. The client will see the same front panel as the local host and also have the same program functionality.

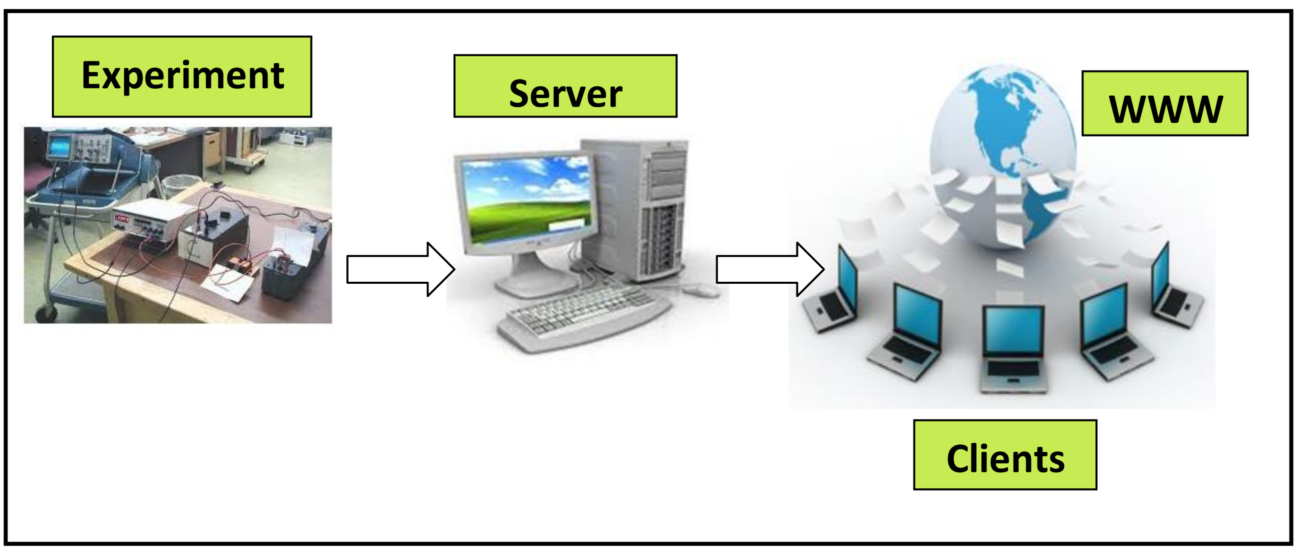


Figure 6.1: Schematics of Internet Based Remote Laboratory

6.3 Development of Internet Based Control System

The Internet is widely used for efficient and reliable dissemination of digital information. Thus, it is a natural choice for data communication tasks arising in remote monitoring and control of processes [34]. In this work, we use the internet as a channel to monitor and control speed of DC motor control process. This methodology eliminated the need for the user to interact with the motor process physically. Internet-based graphics tool, enable developed of interactive graphical user interface (GUI) for process monitoring and control [35].

6.4 Web Service

A Web service is software system designed to support interoperable machine-to-machine interaction over a network [36]. Web services provide a standard means of interoperating between different software applications, running on a variety of platforms and frameworks. Using web-based protocols, target remote panel initiates the Web services. A remote server receives a request from the client, the request then analyze by the client and with response it retrieve back to the remote server, which is then processes and published the application. Web browsing, chatting, E-mails, and others are dependent on this communication environment for daily activities.

All the components of a Web service are explained as follows.

- Server – Enable the appropriate execution between the host and client computer by sending the appropriate request.
- Client – An application request has been send to the server and waits to respond. The client can analyze the request after receiving it back.
- Network – This is the layer over which data is being transmitted between the host and client, the network such as Ethernet and IEEE 802.11.
- Standard Protocols – Data transmission between the server and client over the physical networks using Web based protocols such as HTTP can be used.

6.5 Client/Server Architecture

Client/server architecture can be considered as a network environment that exchanges information between a server machine and a client machine where server has some resources that can be shared by different clients. Architecture of Client/Server based on internet is shown in figure 6.2. In order to develop the program on a server computer, LabVIEW Web server is used. To access the DC motor speed control system through an internet, LabVIEW Web Server program is used [37]. When all obtained programs are appropriately operated on a server, a user friendly power program is obtained. On the computer of clients, only internet connection and internet explorer are enough to monitor the speed of motor. So, they do not need any additional programs.

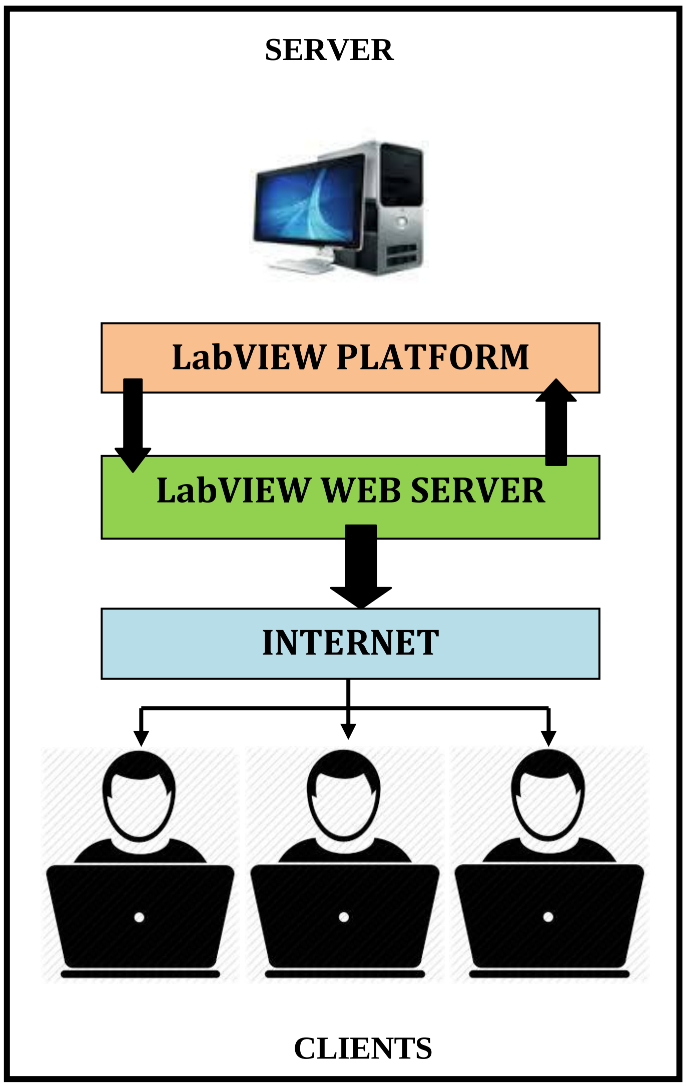


Figure 6.2: Architecture of Client/Server based on Internet

6.6 Web Server Configuration

The Web Server application using LabVIEW makes it an easy task for the user. This services offers a communication between the server front-panel helps to controls, receive, and send or receive data through Web Server [38]. To create such VIs for a Web Services application are to be followed.

6.6.1 Building a LabVIEW Application Web Server

I. To configure LabVIEW Web Server, it needs a plug-in designed by Microsoft named Silver-light which helps in publishing front panel on web.

II. First enable the remote panel server form the web server in options. The Web application service is enabled by using Configure Web Application Server as shown in figure 6.3 below. It is then launch the URL in web browser and direct it to configure panel.

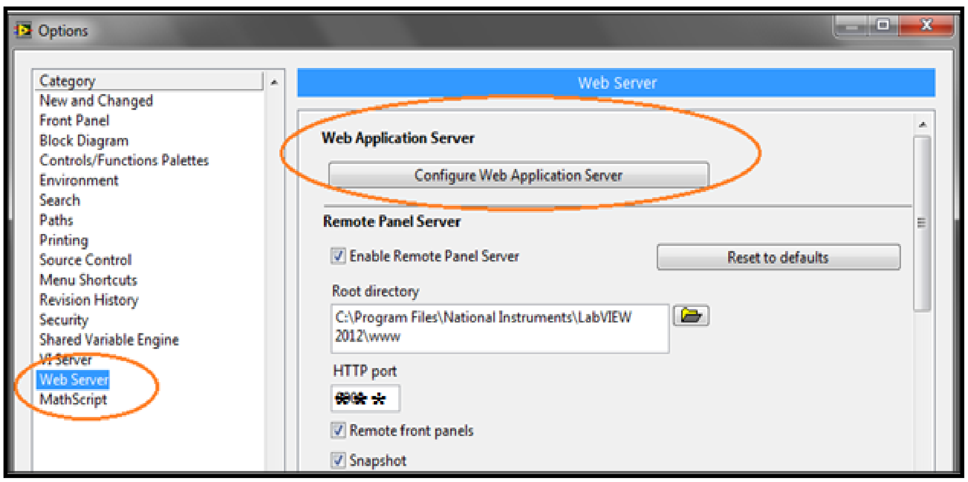


Figure 6.3: Configuration of Web Application Server

III. The URL is a password protected by NI. One can easily Login to get excess to the NI Web Server network setting by entering user ID and left password column empty.

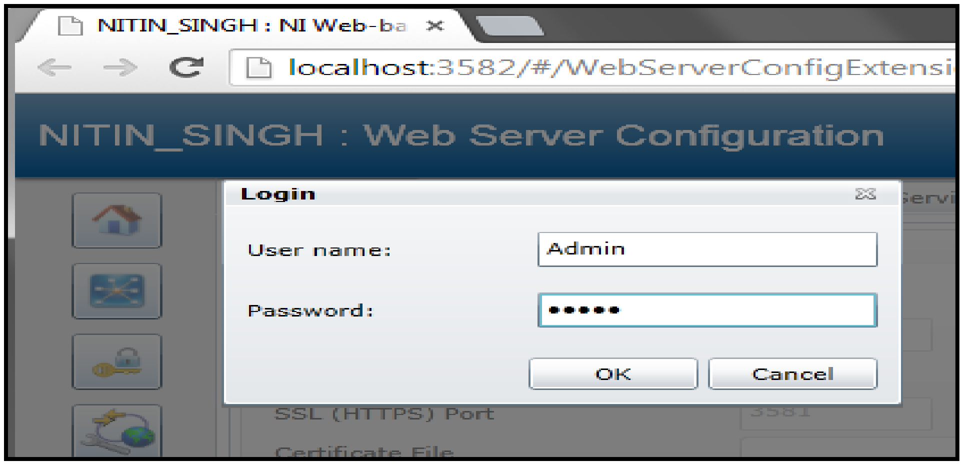


Figure 6.4: Screenshot of password protection

IV. The NI-based Web Server configure URL is shown in figure 6.5. From the system server marks the HTTP enabled button and one can also make secure web server by enabling SSL. In the given figure enable the application web server and enter the HTTP port number as convenient.

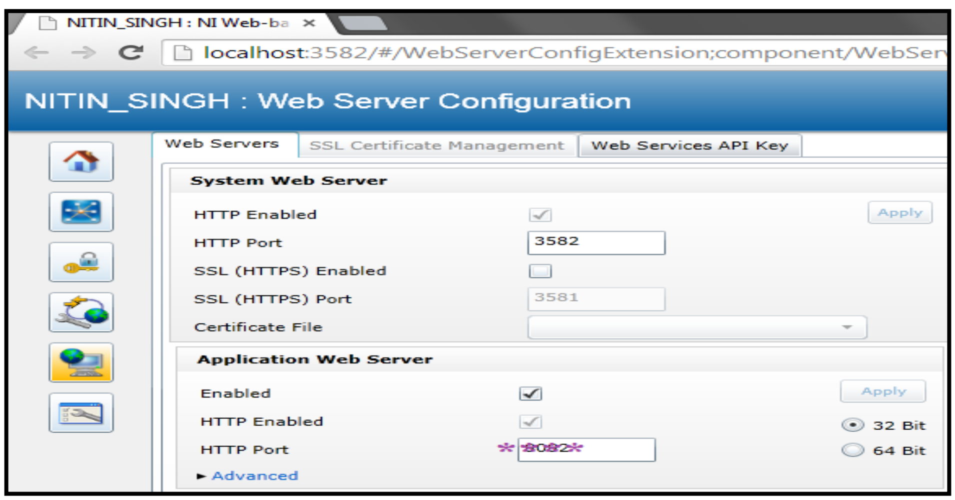


Figure 6.5: Screenshot of Web Server configure in Web Browser

V. Now, Open a default web browser installed on PC. Note: must have a latest version IE 7, Chrome or Mozilla.

VI. In the selected web browser, the URL must be in given format/syntax:

- To run application web server on a host computer, put the URL in web browser: [*http://local_host:8000*](http://local_host:8000) , where (*8000*: default port address).

- To run application web server on a target computer, put the URL in web browser: *http://(X.X.X.X):8000* , where (*X.X.X.X host IP address)*

6.6.2 Enable Remote Panel

Once the LabVIEW web server configured, now we need to publish the VI running on server side system. This can be achieved by using Web Publishing Tool in LabVIEW platform, which we can see in Tools dialog box form the menu. The web publishing tool window is very well created to load VI directly to the LabVIEW buffer memory [39].

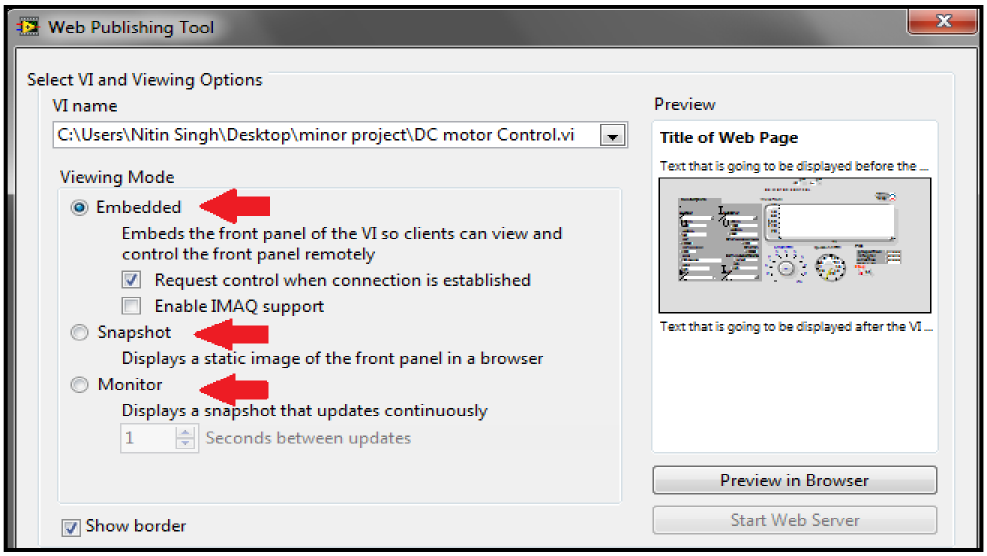


Figure 6.6: Screenshot of Web Publishing Tool

In web publishing tool we can choose VI, which will be a remote laboratory as we can see on Figure 6.6. Other options to choose is viewing mode, it is possible to elect one of the three modes:

- Embedded – Embeds the front panel of the VI, so clients can view and control the front panel remotely.
- Snapshot – Displays a static image of the front panel in a browser test.
- Monitor – Displays a snapshot that updates continuously in the given time interval.

To next step is need to click on button Next.

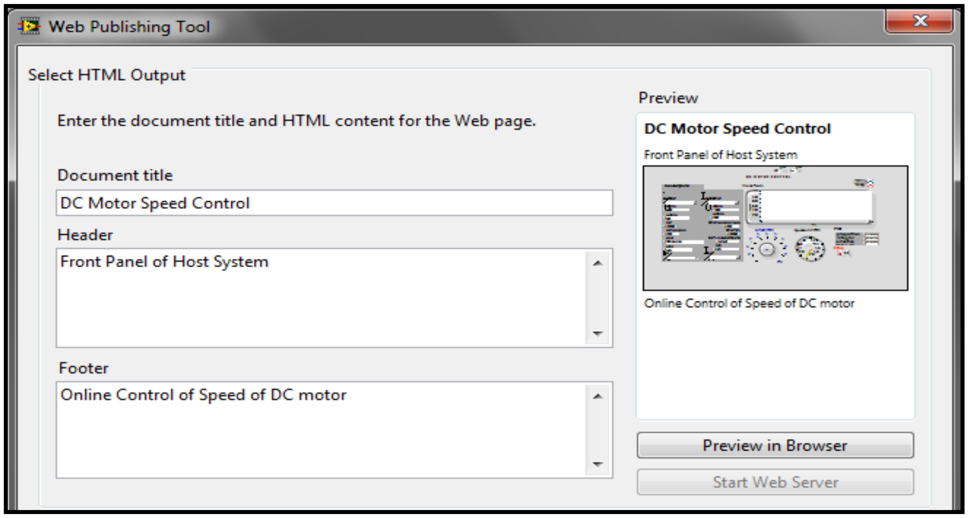


Figure 6.7: Customize panel of web publishing tool

In figure 6.7 we can see a next window of web publishing tool, there is a preview of web page. Document Title, Header, and Footer are all text editing space, which we can use to makeover the Web page publishing on the web browser.

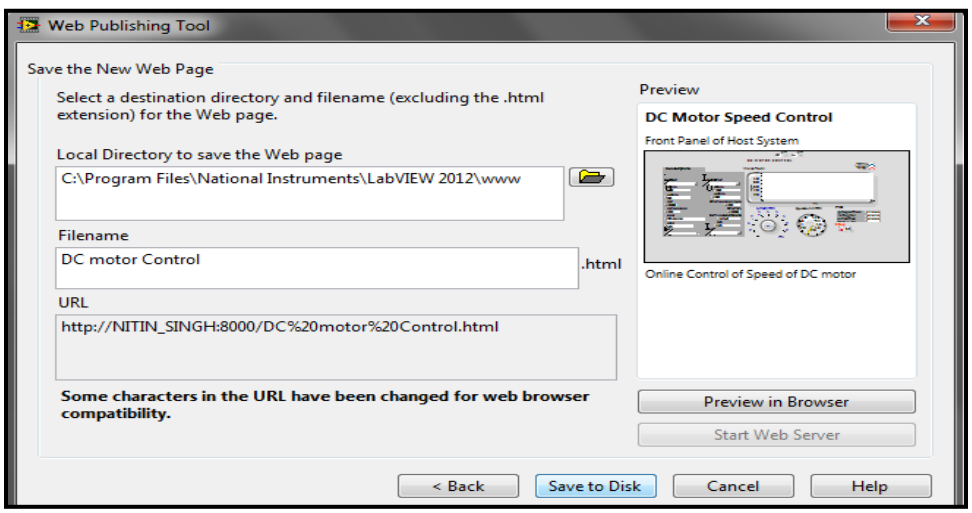


Figure 6.8: Screenshot of URL window pane

The above window pane in figure 6.8 is shown on clicking the next button. In this window we can set a local directory to save the web page for future reference in HTML format [14], the file name (optional) which we want publish on the web, and the URL can be used by the client to access the VI from the remote distance over a web server [41].

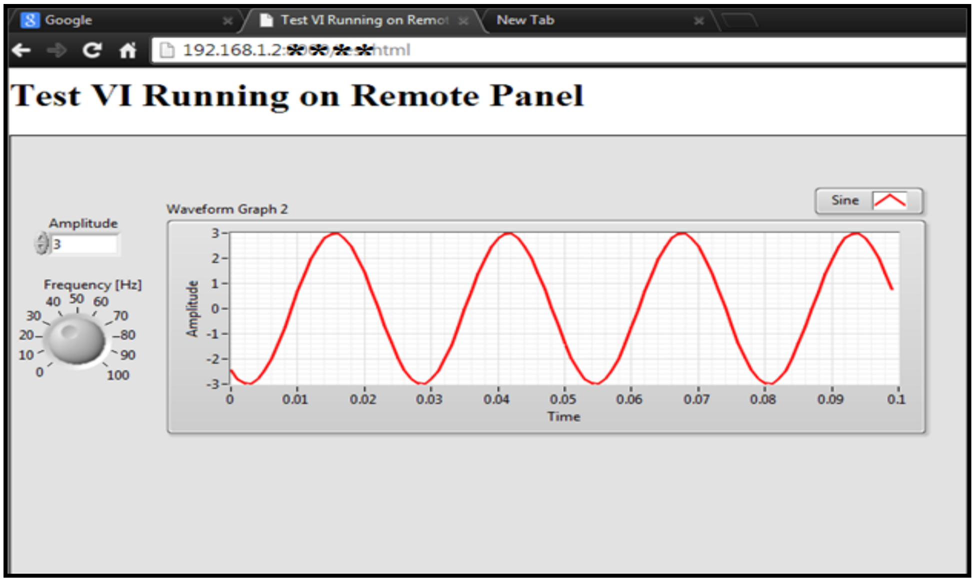


Figure 6.9: Front Panel of Server side running on Client system

With all these setting above makes a perfect running VI on the client system. The above figure 6.9 is just a Test VI running on client to check whether it is working or not, the main experiment will shown in later chapter Result. The number clients can able to view this image running in a single time and even one the client can control the VI the from the distance location. Now the implementation of web server is ready to perform a motive task.

6.7 Client -Side Operation

The client computer should also configured and need software to get access to web server.

6.7.1 Essential Software

To operate a LabVIEW VI from remote distance, the client must install the free LabVIEW software named “LabVIEW Run-time engine” from NI official Website. With this software installed on the client computer, we need not to install the LabVIEW software package to access the program in the web browser. After installing we just need to enter the URL of the remote panel.

6.7.2 Client Priority Control

Once the connection established between the remote panel system and client system, client have an option to control and monitors. Client can be one or than one at same instance of time, so we need a client priority control to manages the client database or prioritize the client according to their status. In monitoring the remote panel experiment system, more than one client link to the server by deploying a small request to the program. But controlling is not a similar task, server must made a priority table according to request before handing an authority to the client. When client gets the access control, the message pop-up (Waiting for control: Either the server is locked or another client has a control). If some other client wants control the experiment it must sends a request to the server and waited for its reply. When the first client logged off the server, then a next client have a remote access to control the system.

6.7.3 Disable Client Control

When then client want to leave server system or move to some other URL, can be achieved by selecting the Release control button. The control will be handed over to next client in the client priority database.

6.7.4 Application Security

The application security is must for any network communication medium. No one want that their data can be seen by some other user. So, NI LabVIEW Remote Panels has its own powerful security tool. The security is provided at both Application layer and Network layer. In figure 6.10 the client can have a setting as shown below:

- Allows Viewing and Controlling
- Allows Viewing
- Deny Access

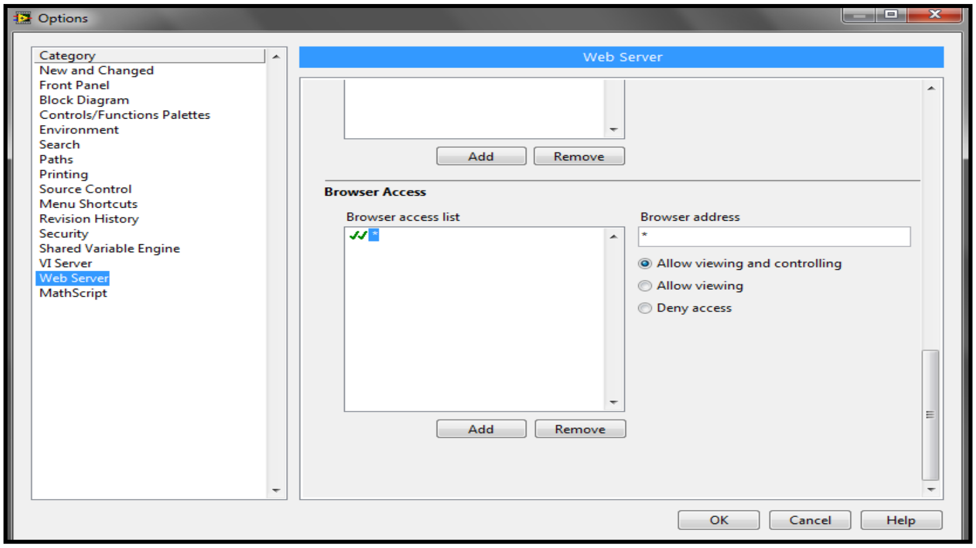


Figure 6.10: Option window for SSL Security

The above given options can be applied form the remote panel system to enhance more secure network and more secure data transfer. SSS security shown in figure 6.11 is applied at the network layer which will provide an authentic path for data flow between the remote computer and the clients.

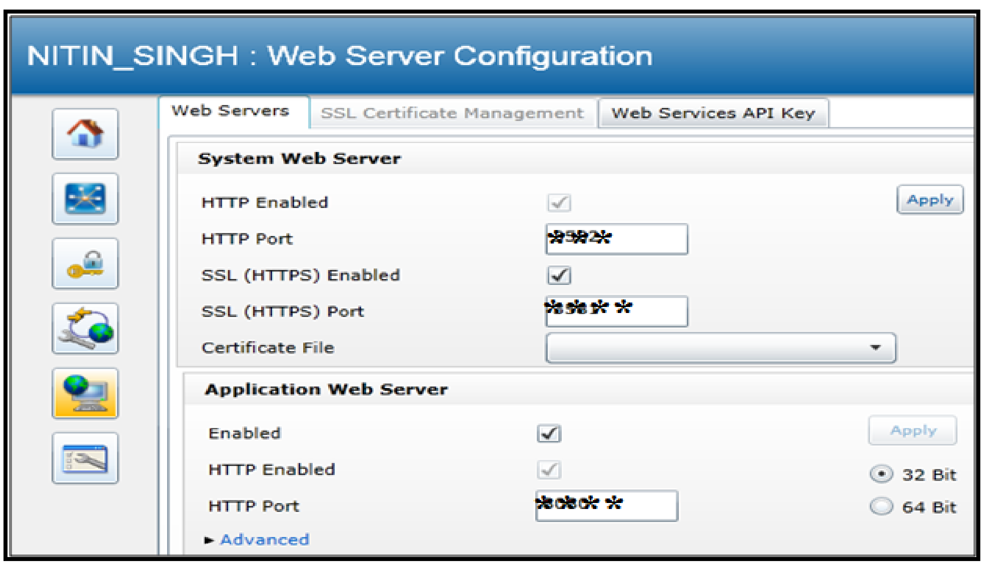


Figure 6.11: Screenshot of Web Server Security

CHAPTER 7

Experimental Results and Discussion

The driving circuit and H-bridge PSB has been fabricated in earlier chapter 2. The driving circuit is basically a PWM generator that converts analog voltage signal 0-5 V of data acquisition into a PWM signal. Now, the generated PWM is fed to H-bridge circuit, which controls the DC motor. An open loop control of the DC motor is obtained. Using the sensor feedback, closed loop control was carried out with using different controller's i.e P, PI, PID, and experimental results are reported. Finally, web-based control using build-in in web server in LabVIEW also provided in this chapter.

7.1 PWM Generation

PWM is generated by using IC 3524 obtained of frequency 525 Hz. The figure below shows the PWM generated at different values of input voltage signal generated by the DAQ 6009. It is shown clearly, as the voltage output of the data acquisition card varies the duty cycle of the PWM wave.

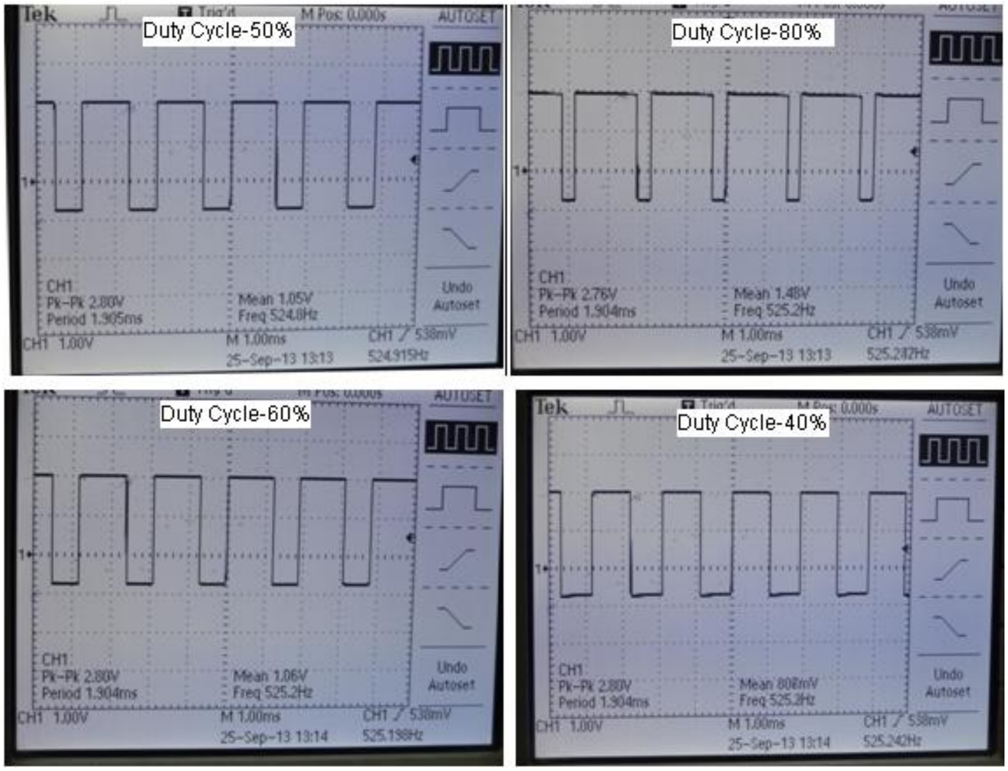


Figure 7.1: PWM with different Duty cycles

7.2 Speed and Voltage Calibration Table

The speed (RPM) and control signal (Volts) has been calibrated by using Laser type Digital tachometer. The readings are tabulated below in table 7.1 and corresponding graph in figure 7.2 has also indicates the control signal is varying almost linearly with speed, which is requirement of control system. The scaling obtained from this table and must be added in further programming.

S.No	Control Signal (Volts)	Input Voltage (Volts)	Speed(RPM)
1	1	1.7	692
2	2	3.35	1600
3	3	4.6	2286
4	4	5.8	2945
5	5	6.7	3440

Table 7.1: Voltage and Speed Calibration table

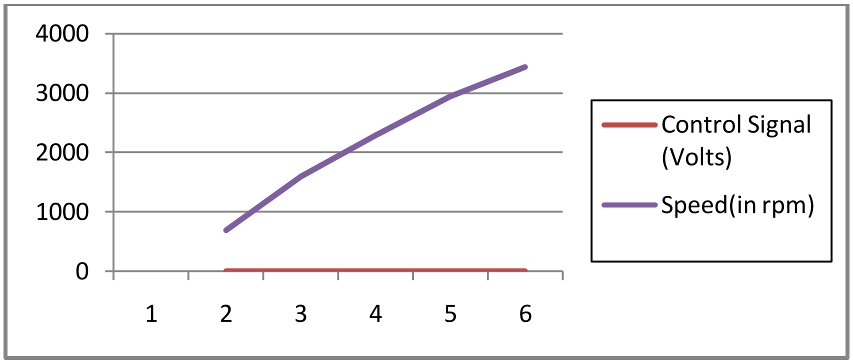


Figure 7.2: Graph between Control signal and Speed

7.3 Open Loop Control

The open loop control Simulink model is designed in LabVIEW environment using DAQmx driver as shown in figure 7.3. The simulink includes physical channel configuration, samples read and write, and clear buffer memory. The open loop control of the DC motor is carried out by varying the control signal (RPM) output from the simulink model.

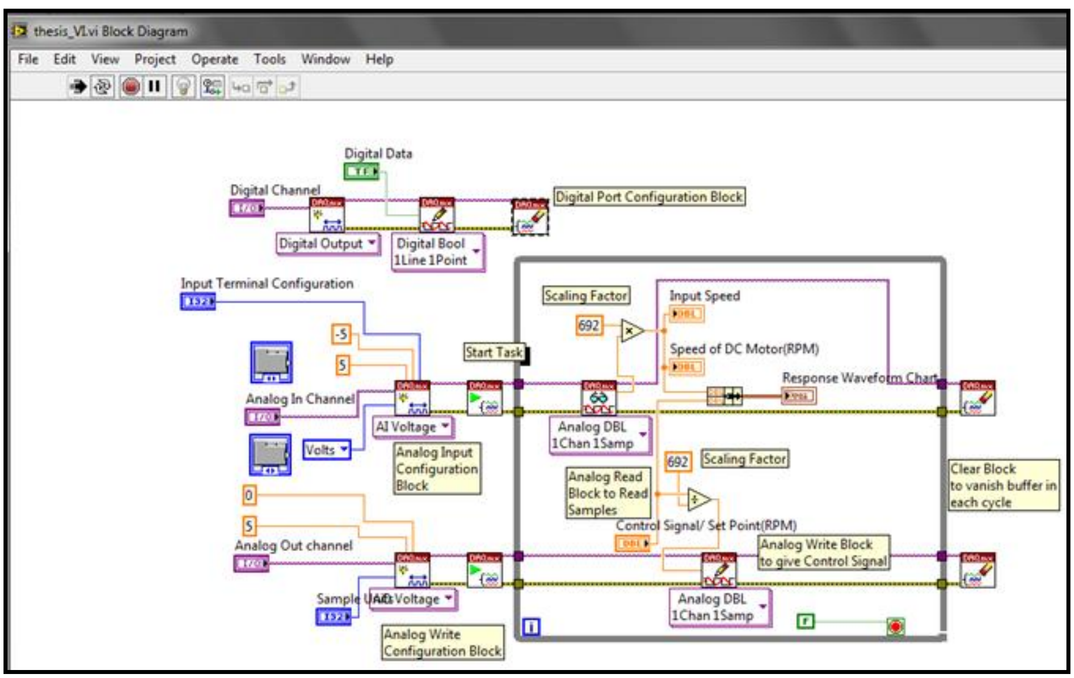


Figure 7.3: Block diagram of Open loop control

The front panel running on host computer of an open loop control can be controlled by the client computer is shown below in figure 7.4.

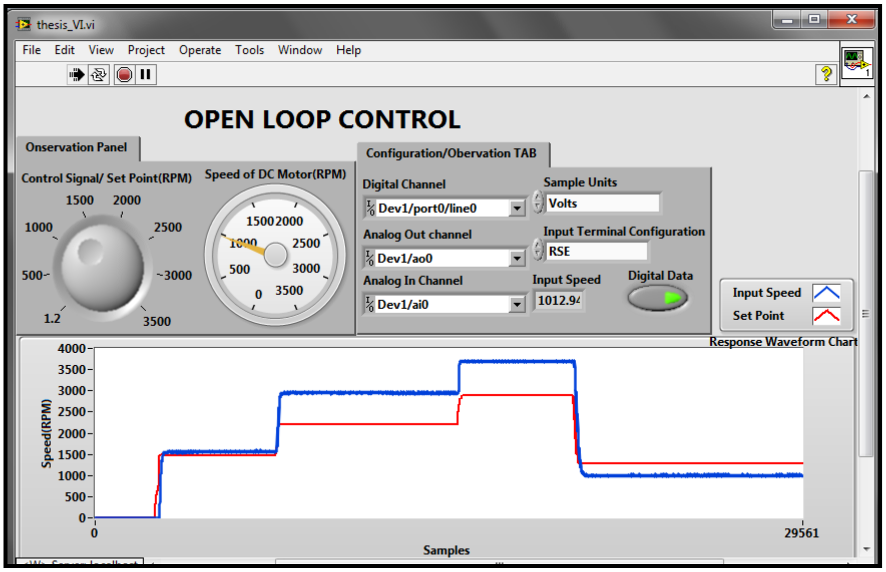


Figure 7.4: Front panel on host computer

The web-based motoring and control though the internet has been achieved. The screenshot of HTML web pages publish in web browser is shown in figure 7.5 below.

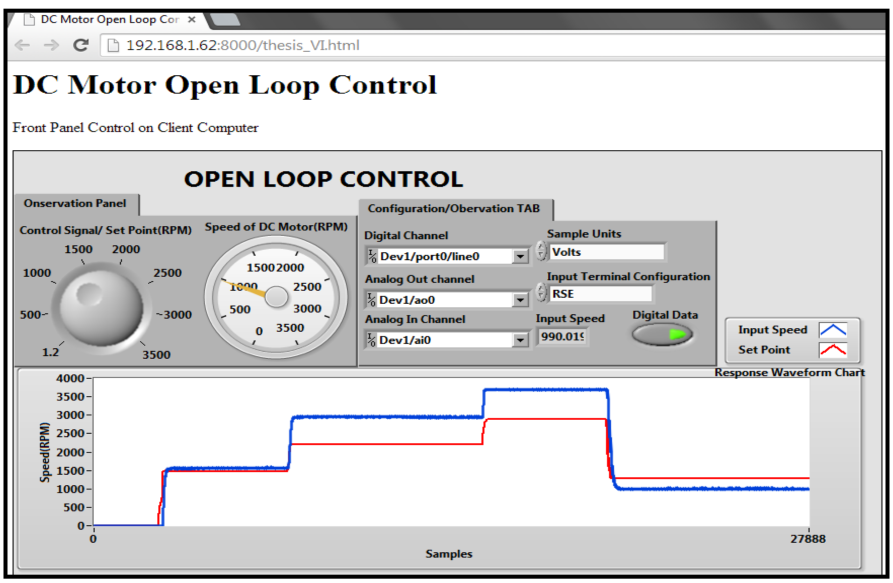


Figure 7.5: Front panel on Client computer

7.4 Closed Loop Control

The closed loop control of the motor under study is carried out from simulink with hardware in loop. Motor speed under various speed references have been studied by changing the reference or set point. The closed loop control is implemented using P, PI, and PID controller. The block diagram of simulink is shown in figure 7.6.

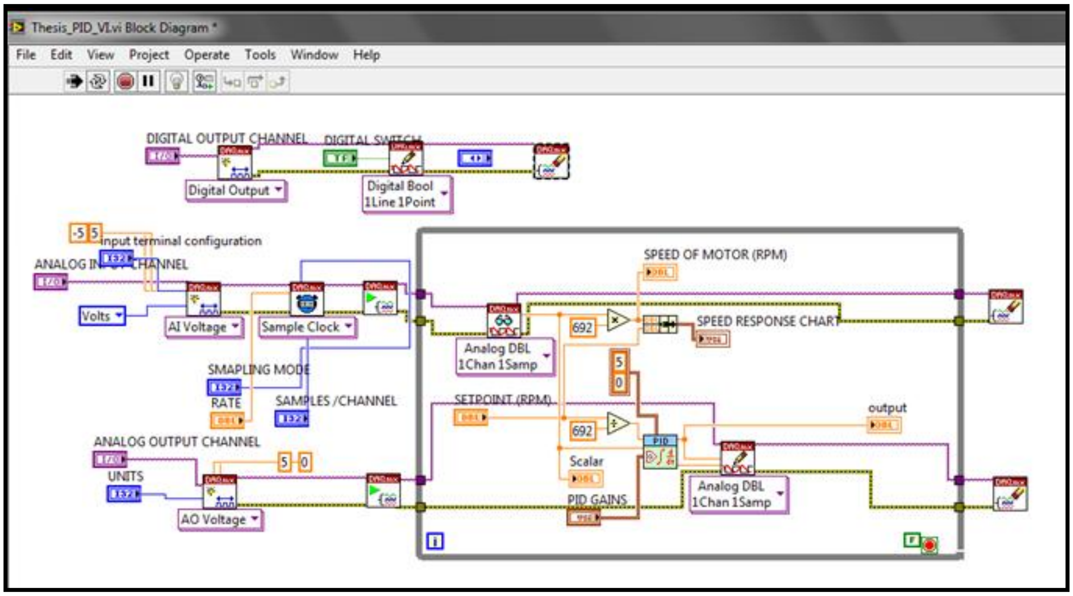
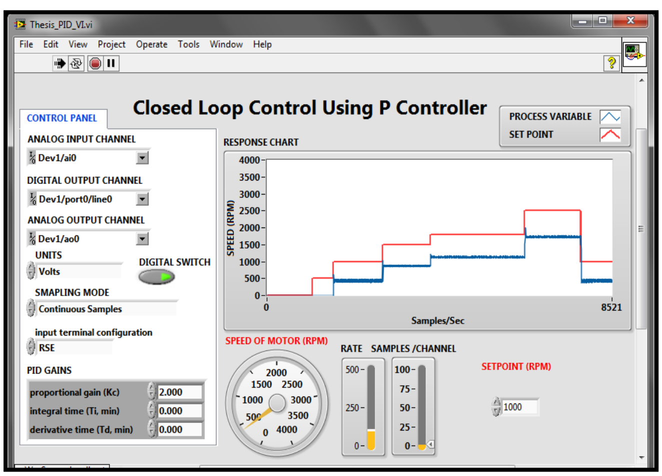
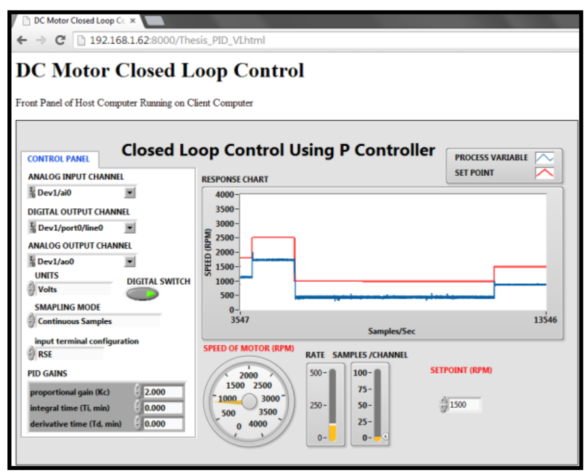


Figure 7.6: Block diagram of closed loop control

The front panel of all P, PI, and PID controller response without load and PI, and PID controller with load are shown in figure 7.7(A)(C)(E)(G)(I)(K), and their corresponding HTML web pages for control and monitoring remotely are shown in figure 7.7(B)(D)(F)(H)(J)(L).

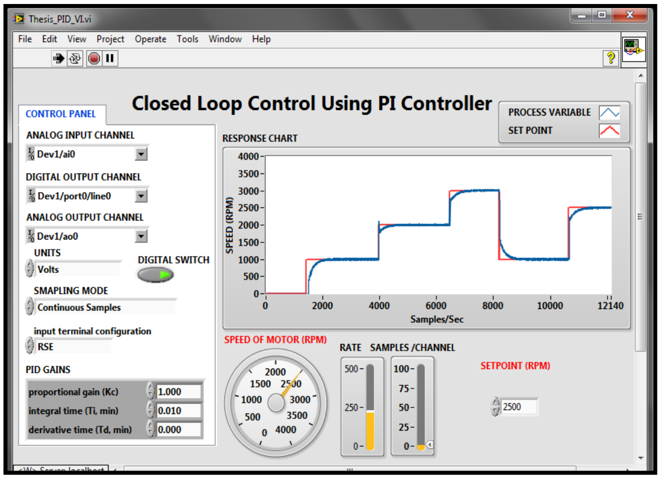


(A)

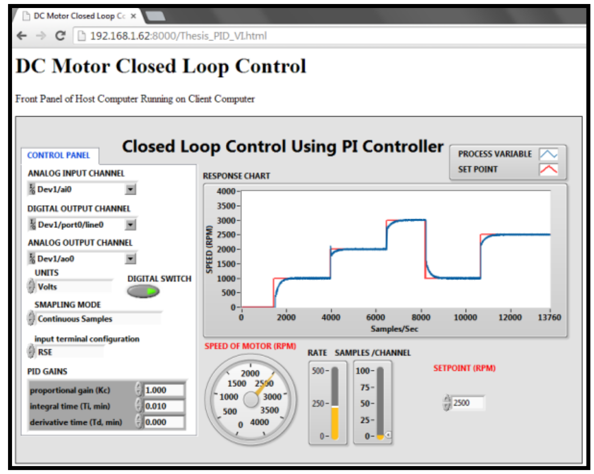


(B)

Figure 7.7: (A) DC motor control using P controller and (B) P controller HTML page

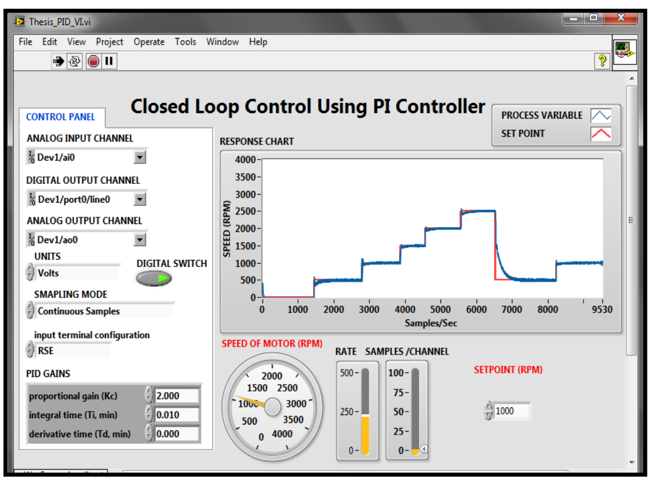


(C)

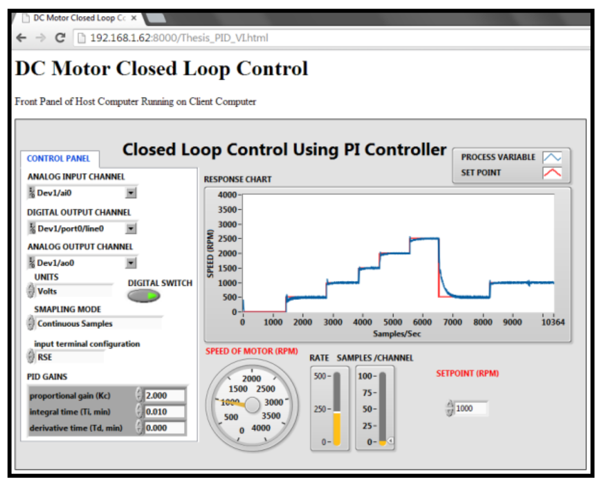


(D)

Figure 7.7: (C) DC motor control using PI controller and (D) PI controller HTML page

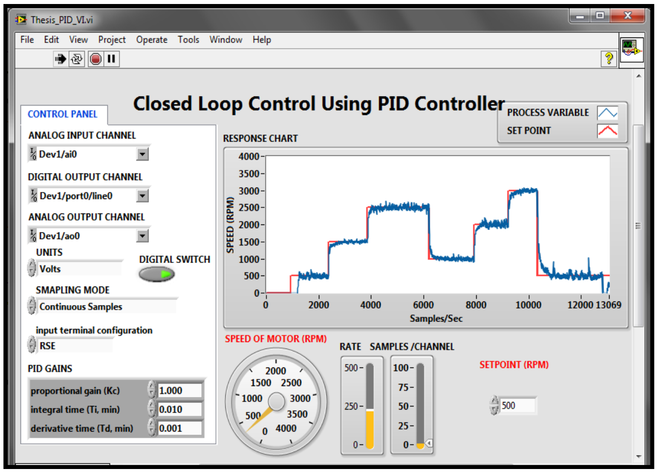


(E)

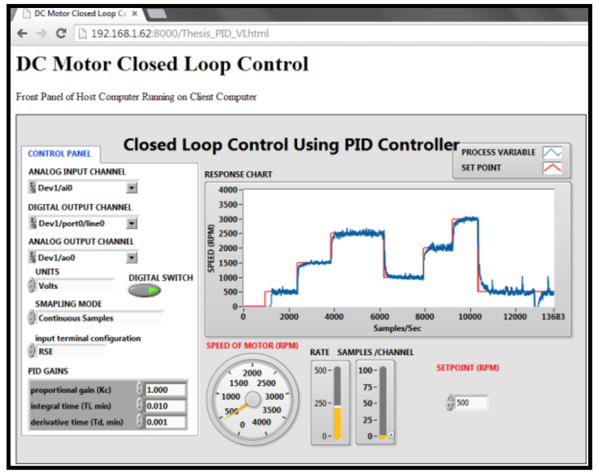


(F)

Figure 7.7: (E) DC motor control using PI controller and (F) PI controller HTML page

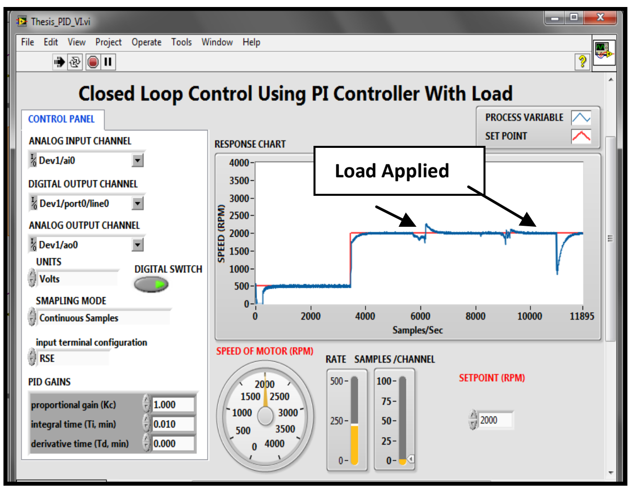


(G)

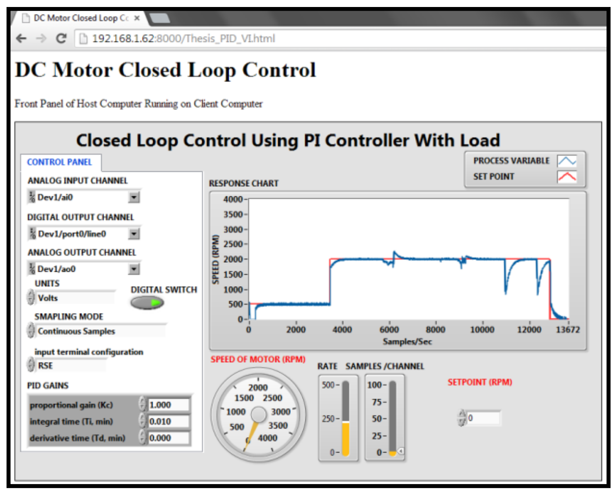


(H)

Figure 7.7: (G) DC motor control using PID controller and (H) PID controller HTML page

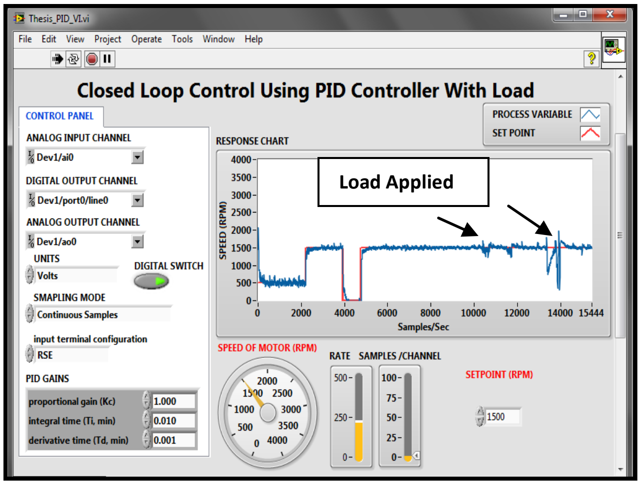


(I)

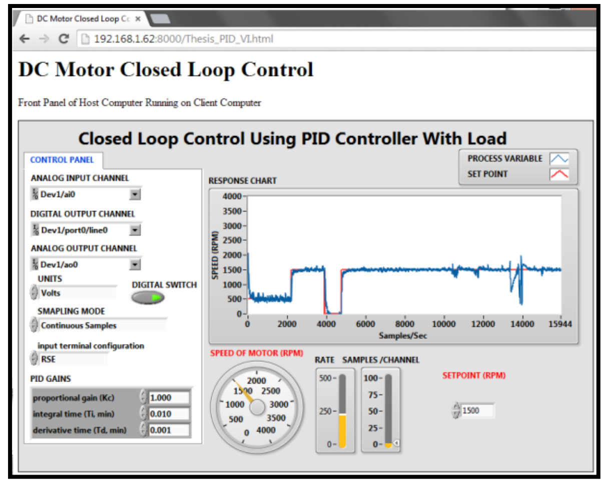


(J)

Figure 7.7: (I) DC motor control with Load using PI controller and (J) PI controller HTML page



(K)



(L)

Figure 7.7: (K) DC motor control with Load using PID controller and (L) PID controller HTML page

Table 7.1 shows the comparison between the results under varying values of K_p, K_i, and K_d.

S.No	Kp	Ki	Kd	Overshoot (in %)	Settling time (in sec)	Offset (in %)
1	2	0	0	-	-	-
2	1	0.01	0	-	12	0
3	2	0.01	0	3	8	1
4	1	0.01	0.001	8	3	1
5 (with_load)	**1**	**0.01**	**0**	**9.3**	**4**	**0**
6 (with_load)	**1**	**0.01**	**0..001**	**16**	**2**	**2**

Table 7.1: Parameters Comparison table

It can be concluded from the above table that proportional controller has an offset, which is eliminated using integral controller, but with a overshoot and large settling time. With the introduction of derivative gain, the system tends to become fast with a lower value of settling time. The above results show figure 7.7 (I) (K), introducing of a load could not affect the system much. After few second it regain its reference values.

CONCLUSION

In this Thesis, the driver circuit is fabricated and a voltage to PWM generator is implemented. The frequency of the PWM generated was 525 Hz and we can get a frequency upto 100khz by making some changes in electronic circuitry. The control signal of 0-5Volts generated at the output of the DAQ device will change the duty cycle of the PWM from 0-100%, which is given to the input of the H-bridge circuit to drive motor in either direction.

The hardware is implemented and works efficiently. In open loop control, it is found that the system is stable but it is not able to track the reference trajectory. The closed loop was carried out with different control strategies i.e. P, PI, and PID controller and s satisfactory results were obtained.

The remote distance monitoring and control system is designed with different controller strategies by using Build-in Web Server and Simulation tool in LabVIEW. The Web Server has been implemented in LabVIEW to view the DC motor system through remote places. The front panel of the host computer can be accessible by the client system, all the client are connected through main server to control and monitored the experiment remotely over an internet. The HTML page is created by using Web Publishing tool in LabVIEW. Now, this HTML page can be accessed by web browser in a wide area. With a condition, only one client has a permission to gain access to control the process, not all. But, one or more client can monitored the process at a distance through internet. Web based process control system is successfully created by LabVIEW software and achieved by controlling speed of DC motor remotely just through internet connection.

REFERENCE

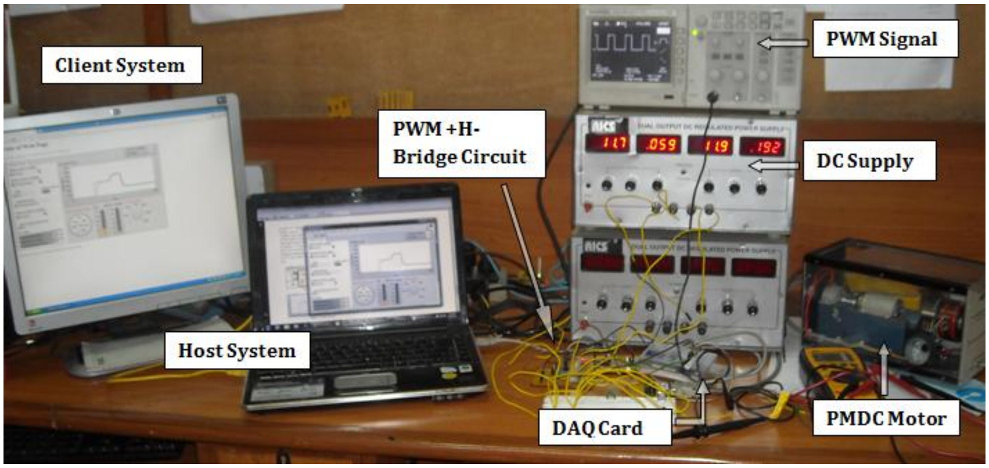
1. J. G. Ziegler and N. B. Nichols, “Optimum Setting for Automatic Controllers”, Transactions of American Society of Mechanical Engineers, Nov. 1942.

2. F. Faizan, F. Farid, M. Rehan, S. Mughal, M.T. Qadri, "Implementation of discrete PID on Inverted pendulum," Education Technology and Computer (ICETC), Proceedings of 2nd International Conference on , pp. 22-24, June 2010
3. Sufendi, B. R. Trilaksono, S. H. Nasution, and E. B. Purwanto, “Design and Implementation of Hardware- in-loop-simulation for UAV using PID control method”, Third International Conference on ICICI-BME, pp. 124-130, Nov.2013.
4. M. H. A. Jalil, M. H. Marzaki, N. Kasuan, and M. N. Taib, “ Implementation of Anti Windup scheme on PID controller for Regulating Temperature of Glycerine Bleaching process”, Third International Conference on System and Engineerinig and Technology (ICSET), pp. 113-117, Aug.2013
5. Kim Jungsoo, M. M. Sabry, D. Atienza, and K. Vaidyanathan, “ Global fan Speed Control considering non-ideal Temperature measurements in enterprise servers”, Design, Automation and Test in Europe Conference and Exhibition, pp. 1-6, March 2014
6. Jose Sanchez, Sebastian Dormido, Rafael Pastor, and Fernado Morilla, “ A Java/Matlab-based Environment for Remote Control System Laboratories: Illustrated with an Inverted Pendulum”, IEEE Transactions on Education, vol. 47, no. 3, August2004
7. S. Bogosyan, M. Gokasan, A. Turan, and R. W. Wies, “Development of Remotely Accesible Matlab/Simulink Based Electrical Drive Experiments”, IEEE Transactions, pp-2984-2989, July 2007
8. H. Wang, J. Yao, and H. Cheng, “Development of a Remote Process Control Laboratory using Configuration Software”, International Conference on Electrical and Control Engineering, pp. 815-818, Oct. 2010.

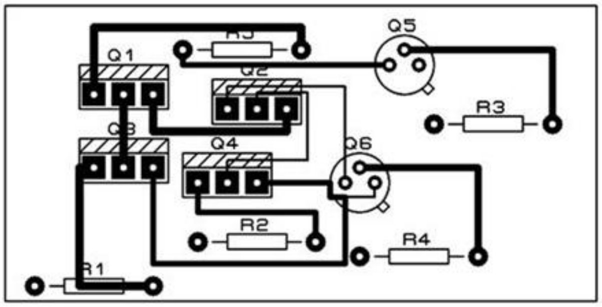
9. Subhransu Padhee and Yaduvir Singh, “Data Logging and Supervisory Control of Process using LabVIEW”, Proceeding of the IEEE Students Tech. Symp., pp. 329-334, Jan. 2011.
10. C. J. Hong, L. K. Luong, and S. Y. Chark, “Building Automation Through Web Interface”, IEEE conference on Sustainable Utilization and Development in Engineering and Technology, pp. 299-304, Oct. 2012.
11. I. Titov, and E. Titov, “ Labicom.net-Putting your laboratory online in less than Five Minutes with Web Pager Tool”, IEEE Transaction, pp. 180-183, 2013
12. Module 9 DC machines, Version 2 EE, IIT Kharagpur [Online].
13. Allan R. Hambley, Eletrical Engineering Principle and Applications, Chapter 16.
14. Giorgio Rizzoni, Principle and Applications of Electrical Engineering, Chapter 17.
15. Tachno Instruments PMDC Motor User Manual.
16. IC 3524 Datasheet [Online]. http://www.ti.com/lit/ds/symlink/lm2524d.pdf
17. TIP 122 and TIP127 Datasheet [Online]. http://www.onsemi.com/pub_link/Collateral/TIP120-D.PDF
18. IC CD 4066 Datasheet [Online]. http://www.ti.com/lit/ds/symlink/cd4066b.pdf .
19. http://www.ni.com/labview/toolkit
20. http://www.ni.com
21. http://www.ni.com/labview
22. Wells L. K., Travis J., labview for everyone, Graphical Programming made more easy. Prentice Hall, 1997
23. http://www.ni.com/dataacquisition
24. Mechanical Measurements and Instrumentation and Control by A. K. Sawhney.
25. DAQ USB 6009 User Manual From www.ni.com
26. Gomes, L., and J. G. Zubia, Advances on Remote Laboratories and E-learning experiences, University of Deusto, Bilbao, 2007.
27. Zafer Aydogmus, Omur Aydogmus, "A Web-Based Remote Access Laboratory Using SCADA, IEEE Trans. Edu., vol. 52, issue. 1, pp. 126-132, Feb 2009
28. Overstreet, J. W., and Tzes, A, “An Internet-Based Real-Time Control Engineering Laboratory”, IEEE Control Systems Magzine, vol. 19, no. 5, pp. 19-34, 1999.

29. Coito, F., P. Almeida, and L. B. Palma, "SMCRVI – A LabVIEW/Matlab based Tool for Remote Monitoring and Control", Tenth IEEE International Conference on Emerging Technologies and Factory Automation, Set 19-22, Italy, 2005.
30. Labview Web UI Builder Overview [Online]. Available : ni.com/white-paper/11602/en.
31. Remote Panel in labview – Distributed Application Development [Online]. Available: ni.com/white-paper/4791/en.
32. Eren, H., W. J. Nichlos, and I. Wongso, "Towards an Internet-Based Virtual-Wire Environment with Virtual Instrumentation", IEEE Instrumentation and Measurement Technology Conference, pp. 817-820, May 2001.
33. M. Shaheen, K. A. Loparo, and M. R. Buchner, "Remote Laboratory Experimentation", Proceeding of the American Control Conference, pp. 1326-1329, June 1998.
34. National Instruments "Remote Panels in LabVIEW Distributed Application Development", http://discoverlab.com/References/WP2238.pdf, 2010
35. M. Shaheen, K. A. Loparo, and M. R. Buchner, "Remote Laboratory Experimentation", Proceeding of the American Control Conference, pp. 1326-1329, June 1998.

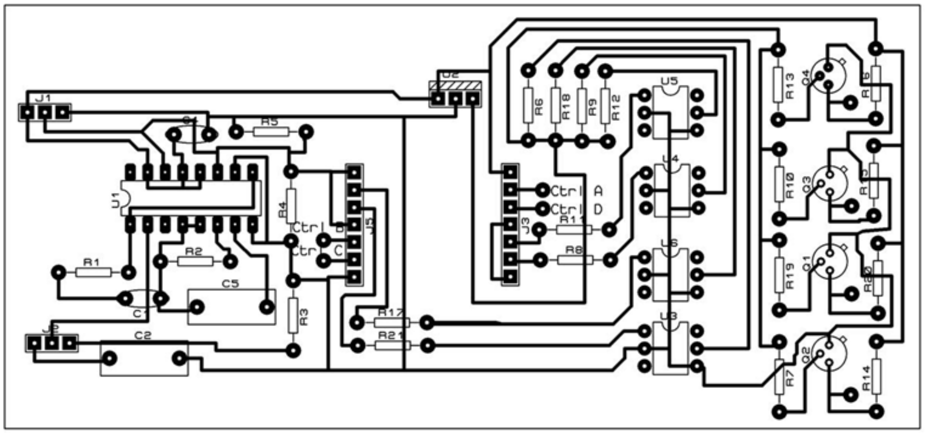
APPENDIX I



Experimental Setup in Laboratory



H-Bridge Printed Circuit Board


PWM Generator and Digital Control Printed Circuit Board