



# LabView based PI Controller for a Level Control System

Jagdish B. Mandhare  
Bharati Vidyapeeth College of Engineering,  
Sector 7, C.B.D., Belpada,  
Navi Mumbai 400614.

Sharad P. Jadhav  
Ramrao Adik Institute of Technology,  
Dr. D.Y. Patil Road, Sector 7, Phase-I., Nerul,,  
Navi Mumbai – 400 706.

## ABSTRACT

The primary aim of this paper is to get advantage of computer based system [10]. The Hardware PID is replaced with soft PID that has equal controlling capabilities as that of instrument. Virtual Instrumentation PI controller is implemented using software called LabView developed by National Instruments. We have used a Data Acquisition board (DAQ) for interfacing with the hardware. This DAQ card is product of the same company National Instruments [7] [2]. The process is a Multi-loop Trainer set up mounted with a tank whose level has to be controlled using a feedback control loop. The flow of project execution is: The measured inputs to the designed PI will be provided by Level transmitter. The designed PI will be generating the necessary controlling electronic signal. This signal will be acquired by DAQ card. The DAQ card transfers it to the I / P converter which will convert the electrical pulses 4-20mA into pneumatic signal 3-15psig to actuate the control valve. This virtual PID can replace hard wired PID. By virtual PID implementation we are getting freedom of reconfiguration and flexibility of control strategy.

## Keywords

LabView, PID, DAQ, Hardware / Software, process variable, control signal, etc.

## 1. INTRODUCTION

PID controller is a three action most widely used controller is process control. Here "P" represents proportional control action, "I" represents integral control and "D" represents derivative control action. It has tuning constants which brings the process value as close to the the desired operating point i.e. set point. Setting the parameters of PID is called as tuning of PID controller, which controls the respective control actions. In most conditions, the requirement is that the controller should act in such a manner that the process value is as close to the set point as possible. The control engineer uses the PID algorithms to achieve this. Proportional action: It simply amplifies the error based upon the gain. P mode generates offset. Integral action: The integral term magnifies the effect of long-term steady-state errors, applying ever-increasing effort until they reduce to zero. [7]

Derivative action: The derivative part is concerned with the rate-of-change of the error with time: If the measured variable takes longer duration to approach the setpoint then the derivative action would speed up the controller action so that the process variable will rapidly reach the set point. Derivative action makes a control system behave much more intelligently. High value of Derivative constant would make controller action oscillatory.

To obtain soft PID controller is possible by programming the PID algorithm using some high level language such as C,

Fortran, Use of high level language allow us to use floating-point math. [9]. LabView is a very good high level language with an additional advantage of builtin graphical user interface (GUI). It is quite user-friendly software and flexible to give better system performance. To provide improvements is an easy task in Labview based soft PID. The system performance can be observed by the graphical representations of set point, process value, control signal, etc. The transient specifications such as overshoot, settling time, offset error, response time can be obtained from the graphs and PID algorithm can accordingly change the proportional gain, reset time and derivative time in order to get desired performance [9] [1].

Organisation of paper :

Section 2 includes description on process hardware setup and overall system block diagram. Section 3 has description about LabView based PID VI, Hardware implementation, signal conditioning and programming in LabView. Section 4 includes description about tuning of PID to get optimal controller output. Section 5 shows the results. Section 6 describes the conclusion made from implementation of work.

## 2. PROCESS SETUP

The process is a cascade control setup shown in figure 1.



Fig 1: Process Hardware setup



The major components of the cascade control trainer are described as follows:

#### Supply tank

Supply tank is used for storage water. It is situated below the main setup assembly. The main setup assembly covers the supply tank.

#### Level tank

Transparent level tank is supported on the support plate of the setup assembly. Graduated scale is provided on the tank to indicate tank water level in %. The level tank has an outlet at bottom, which opens in to the supply tank. Outflow of the tank can be varied with the help of valve provided at the outlet.

#### Level sensor

Capacitance type level sensor transmitter is mounted on one of the level tanks. The level of water is sensed and transmitted to interfacing unit for displaying the level and communicate to computer for control.

#### Control valve

A pneumatic control valve is supported on the support plate controls the flow of water to the level tank.

#### E/P Converter

The current to pressure converter (E/P converter) receives signal from computer through interfacing unit. It converts supply pressure in to controlled output pressure in proportion to the signal received for pneumatic valve actuation.

#### Other units

An air regulator, pressure gauges, pump for water circulation, rotameter pneumatic piping and water piping are the other units in the set up.

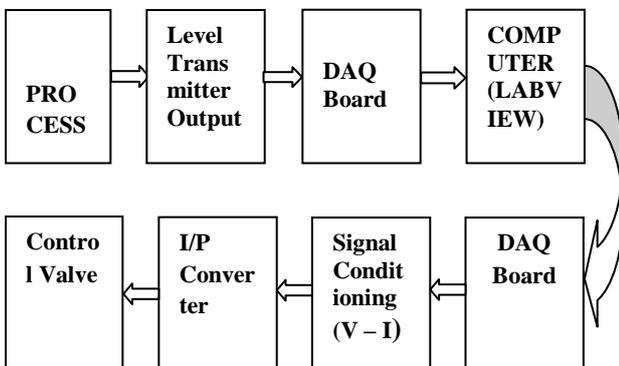


Fig 2: Process Block Diagram

### 2.1 Process Block Diagram

Figure 2 shows the process block diagram of overall system. Level is measured by level transmitter, outputs of which is 4-20 mA signal. This transmitter signal is sent to LabView Based PID Controller through DAQ. LabView based PID Controller is used which generates the control signal. The control signal is a voltage signal, which is converted to current using V-I convertor circuit. The output of V-I is given to I/P convertor which manipulates the inlet flow control valve [11].

### 3. LABVIEW BASED PI CONTROLLER

Here for controlling level of process setup PID controller is used. The existing hardware based PID is bypassed and LabView based PID is designed[12]. The PID algorithm

features control output range limiting with integrator anti-windup and bumpless controller output for PID gain changes [4]

### 3.1 PID VI (Virtual PID)

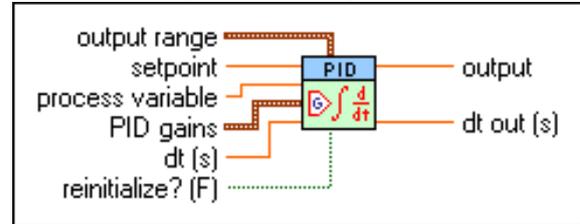


Fig 3: PID VI Tool in LabView

Inputs of a virtual PID block [5] [6] :

**output range** specifies the range to which to coerce the control output.

**setpoint** specifies the setpoint value, or desired value, of the process variable being controlled.

**process variable** specifies the measured value of the process variable being controlled. This value is equal to the feedback value of the feedback control loop.

**PID gains** specifies the proportional gain, integral time, and derivative time parameters of the controller.

**proportional gain (Kc)** specifies the proportional gain of the controller. The default is 1. In the equation that defines the PID controller,  $K_C$  represents the proportional gain.

**integral time (Ti, min)** specifies the integral time in minutes. The default is 0.01.

**derivative time (Td, min)** specifies the derivative time in minutes. The default is 0.

**dt (s)** specifies the loop-cycle time, or interval in seconds, at which this VI is called.

**reinitialize?** specifies whether to reinitialize the internal parameters, such as the integrated error, of the controller. Set **reinitialize?** to TRUE if your application must stop and restart the control loop without restarting the entire application. The default is FALSE.

Outputs of a virtual PID block :

**output** returns the control output of the PID algorithm that is applied to the controlled process. If this VI receives an invalid input, **output** returns NaN.

**dt out (s)** returns the actual time interval in seconds. **dt out (s)** returns either the value of **dt (s)** or the computed interval if you set **dt (s)** to -1.

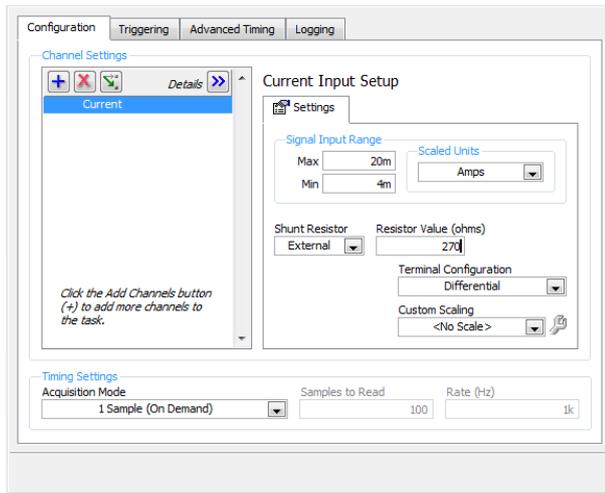
### 3.2 HARDWARE IMPLEMENTATION

Acquire and generate signals using DAQ Board No. BNC – 2120 with a PCI Card No. PCI – 6014

#### 3.2.1 Configure DAQ Assistant to acquire signal and Input port :

To acquire a signal from Transmitter (4-20 mA current), connected a 270 ohm resistor in series with the transmitter signal. Connected the transmitter output to Analog Input port AI-04. As shown in Figure 4, Configured the DAQ to acquire the signal by selecting acquire signals and select current. Then selected the range of measurement, i.e. 4-20 mA. Also enter

the value of resistor connected in series at transmitter output

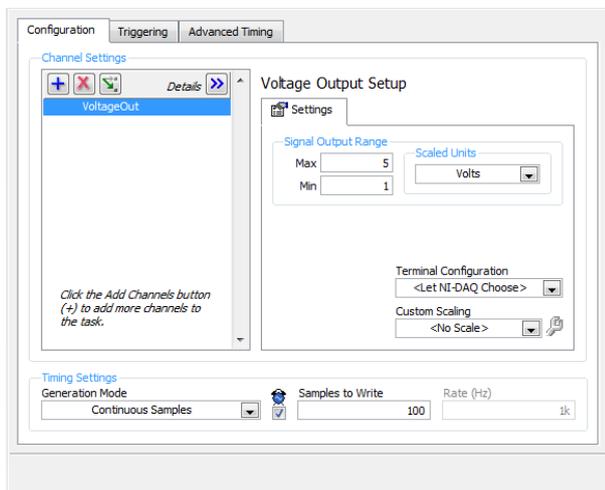


**Fig 4: Input port configuration**

Configured the selected port to receive the analog signal select input range, shunt resistor value, terminal configuration, acquisition mode, etc.

### 3.2.2 Configure DAQ Assistant to generate signal and Output port :

Configuration settings are shown in Figure. 5. To configure the port, generation mode is selected continuous, output range 1 – 5 volts, terminal configuration let NI DAQ choose.



**Fig 5: Output port configuration**

### 3.3 Signal Conditioning

The LabView can generate only analog voltage output signal. So the control signal generated by LabView PID VI is given to DAQ Assistant which is configured to provide output voltage signal of 1 – 5 volts to analog output port AO-2. This signal is given to V-I convertor, shown in figure 6 to get 4-20 mA current output.

#### Design of V – I Convertor with grounded load.

terminals.

$$I_L = \frac{V_{in}}{R}$$

**Output Current Range :**

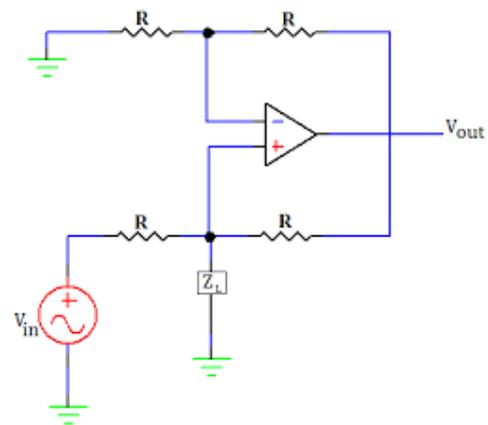
$$I_L = 4mA - 20mA$$

**Input Voltage Range :**

$$V_{in} = 1V - 5V$$

$$R = 250\Omega$$

The output of V-I Convertor is given to I/P Convertor which converts 4 – 20mA to 3 – 15 psig Air pressure. This will provide Air signal to Control Valve.



**Fig 6: V – I Convertor with grounded load**

### 3.4 Programming PID VI in Labview

Figure 7 shows Block Diagram VI. For programming Block diagram VI in LabView a While Loop is selected and transmitter signal acquired by DAQ Asst was converted from 4-20mA to 1-5 Volts and given to PID VI input as process variable. Numeric Indicator Tank tool is used which displays real time level in the tank. Set point is given through Numeric Control of Vertical pointer slide tool. PID Input / Output range is set to 1-5 Volts. D(t) is set to 1. PID gains selected by trial and error method and given as input to PID VI. The output of PID controller is given to DAQ Asst. 2 to generate control output at output port. The PID Controller output is also viewed on numeric indicator tool simultaneously. All three real time values of controller output, process variable and set point are displayed on waveform chart. Front Panel VI is shown in Figure 8 [13].

### 4. TUNING OF PID

The outlet valve of the tank, i.e. load is adjusted to some set value so that tank does not overflow and the bottom tank pump is started. The overhead tank, whose level is to be controlled, starts filling at a constant rate. The process variable is displayed on the front panel which shows the real time water level in the tank. The PID values are taken based on experience [14].

Thus, the controller is used in P-I mode. The proportional gain  $K_c$  is adjusted to -3. The negative value of  $K_c$  is taken because



the final control valve is Air-to-close i.e. normally open. Therefore the PID controller output should increase to close the valve. So reverse action of controller is required. This reverse action of controller is achieved by negation of proportional term and keeping integral and derivative terms as it is. The derivative time constant is kept zero as it causes fluctuation in control output.

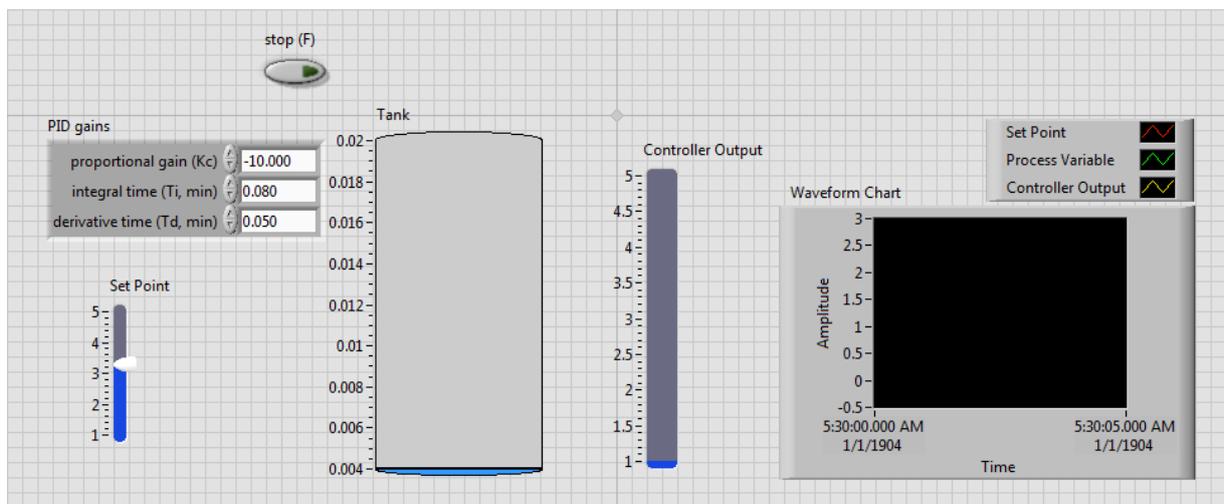
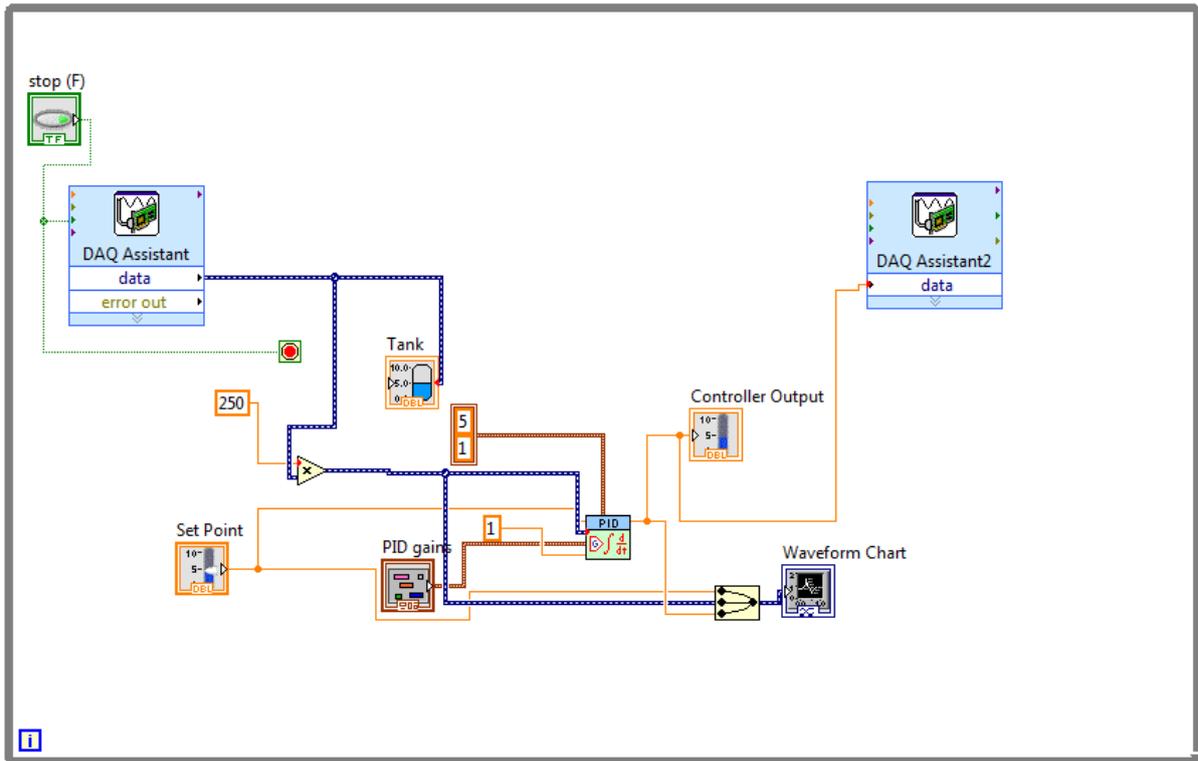


Fig 8: Front Panel VI



## 5. RESULTS

After successful implementation and tuning, the following charts show the setpoint, process variable and controller output plotted with respect to time.

Case I :

Figure 9 shows the previous value of setpoint was 3 which is changed to 4. As the setpoint increases the controller output falls as it is a reverse acting mode. Thus the control valve has more opening and the level begins to rise. The process variable traces the setpoint in about 2 mins and keeps following it accurately.

Case II :

Figure 10 shows the change of setpoint from 4 to 3. As the setpoint is reduced the controller output increases instantly and the control valve is closed for some instance and flow is reduced. The process variable traces the setpoint in less than 1 min and maintains close control.

Case III :

Figure 11 shows same reduction in setpoint from 3 to 2. Here the process variable comes close to setpoint in less than 30 secs.

Case IV :

Figure 12 shows the major increase in setpoint from 2 to 4. The process variable traces the set point in about 2 mins and remains at set point value.

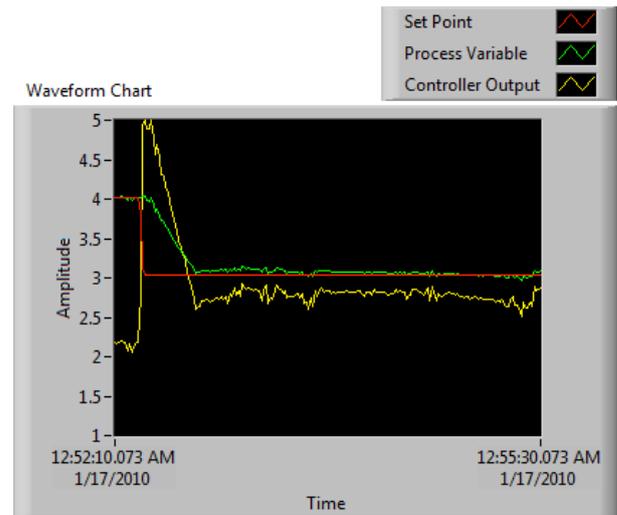


Fig 10: Set point = 3

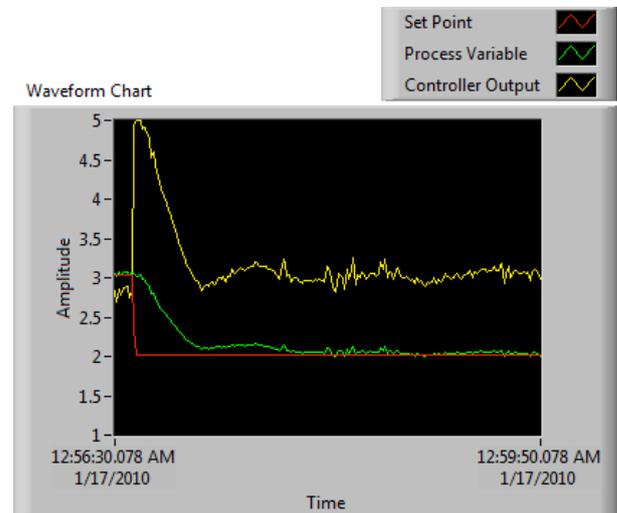


Fig 11: Set point = 2

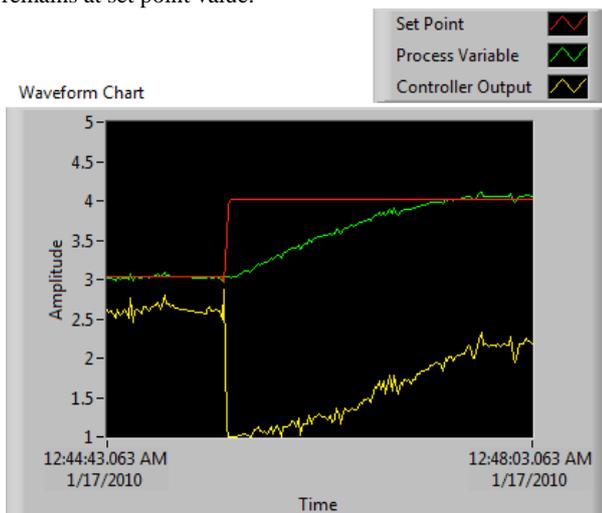


Fig 9: Set point = 4

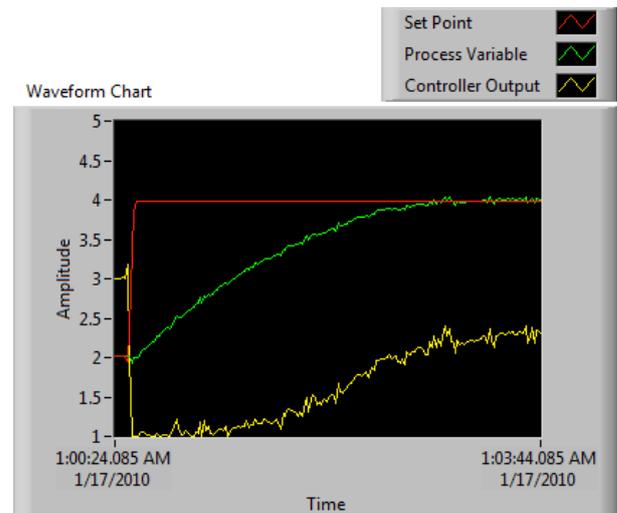


Fig 12: Set point = 4



## 6. CONCLUSIONS

By using LabView controls, indicators and block diagram functions a virtual PID is designed and tested [9]. This paper replaces the conventional PID instrument with a virtual controller. It saves time of manufacturing instrument, since the PID logic can be designed on LabView in a very short period. This PID controller is flexible and easy to use. Practically this application needs no maintenance and very easy to upgrade. [8] The experimental investigations in NI LabView show that the new system developed would be highly flexible and easy in controlling the level. In real time environment for better performance of process plant it is difficult and expensive to conduct experiments directly. These results will be useful to do the required modifications in process industries for efficient control [3]. Also the flow-level cascade control for which the soft PID is implemented is a commonly used control requirement. The test results can help study the cascade control, its control using Labview based PID can be used in many practical industrial applications. The graphical user interface of LabView enhances the use of the software. Future scope can be further study of the process, obtain process model and using the model for fine tuning of the controller to get better system performance.

## 7. REFERENCES

- [1] Mohammad A. K. Alia, Tariq M. Younes, Shebel A. Alsabbah, A Design of a PID Self-Tuning Controller Using LabVIEW Journal of Software Engineering and Applications, 2011, 4, 161-171.
- [2] P. Thepsatom<sup>1</sup>, A. Numsomran<sup>2</sup>, DC Motor Speed Control using Fuzzy Logic based on Lab VIEW, SICE-ICASE International Joint Conference 2006 Oct. 18-21, 2006.
- [3] K. V. L. P srinivas<sup>1</sup>, P Durga Prasada Rao<sup>2</sup>, “Modelling and simulation of complex, Control systems using labview,” International Journal of Control Theory and Computer Modelling, IJCTCM, VR Siddhartha Engineering College, Vijayawada, India.
- [4] J. M. Jacob, “Labview graphical programming for instrumentation,” Industrial Control Electronics. Prentice-Hall International, Inc., New Jersey, U.S.A, 1989.
- [5] B. E. Paton, “Labview graphical programming for instrumentation,” Programming for Instrumentation. Prentice Hall PTP, New Jersey, U.S.A, 1999.
- [6] “Labview basics introduction,” National Instruments, Course Manual, U.S.A, 2002.
- [7] P. P. P. Harsh Kaji, Shruti Annigeri, “Designing pid controller using labview for controlling fluid level of vessel,” International Journal of Engineering Research and Applications (IJERA), vol. 3, pp. 1329–1333, March-April 2013.rol Electronics.
- [8] R. A. . S. Mohan, “Complete industrial solution for automation in temperature and humidity monitoring using labview,” VIT University, Vellore, India, p. 5.
- [9] M. K. A. Z. MOHAMMAD A.K. ALIA, “A closed-loop temperature control system by utilizing a labview custome-design pid controller,” Al-Balqa Applied University Amman, Jordon, p. 4.
- [10] James R. Matey, “General Principles for Applications of Computers in Experimental Physics,” Computers in Experimental Physics, Vol. 7, No. 1, pp. 12-17, Jan/Feb 1993.
- [11] Robert Hennessey, “LabVIEW-Based Automatic Control Systems Laboratory Using Local and Remote Experimentation Approaches”, Electrical and Computer Engineering Department, University of Texas at El Paso
- [12] “PID Contol Toolkit Manual,” National Instruments, 2006. <http://www.ni.com>.
- [13] G. W. Johnson, “LabView Graphical Programming,” MCGarw-Hill, Inc., 2006, New York.
- [14] “Improving PID Controller Performance,” Development zone National Instruments USA.