Design and Implementation of PID Controller in Programmable Logic Controller for DC Motor Position Control of the Conveyor System

MUHAMMAD SHARFI NAJIB, MOHD SHAWAL JADIN, RAJA M. TAUFIKA RAJA ISMAIL, MOHD RUSLLIM MOHAMED
Faculty of Electrical & Electronics Engineering,
Universiti Malaysia Pahang,
Karung Berkunci 12, 25000 Kuantan, Pahang,
MALAYSIA

Abstract: **-** Direct Current (DC) motor position control using Programmable Logic Controller (PLC) is one of the applications which are widely used in automation industries. The aim of this project is to implement a Proportional Integral Derivative (PID) controller in a DC motor control system of The Conveyor System. The project is consists of two stages, classical controller and PID controller. The classical controller and PID controller were implemented in PLC and the results of both methods were compared. The system response of both controllers were compared and analyzed. Throughout the analysis, the PID control system illustrated better performance as compared to the classical control system controller in DC motor position control.

Key-Words: - PID controller, conveyor system, PLC, DC Motor, automation, position control

1 Introduction

Any controller design for any system commonly needs some knowledge about the system before it will be developed. This involves a mathematical description of the relation among inputs to the process, its variables and its output, that is called the model of the system. The model can be represented as a set of transfer functions, which is usually called mathematical modeling. Modeling for the complex systems can be a very difficult task. For example, in a complex system such as a multiple inputs and multiple outputs system, the inaccurate models will cause the systems is unstable or has a bad system performance.

Proportional Integral Derivative (PID) controllers are widely used in industrial practice over 60 years ago. The invention of PID control is in 1910 (largely owing to Elmer Sperry's ship autopilot) and the straightforward Ziegler-Nichols (Z-N) tuning rule in 1942 [1]. Today, PID is used in more than 90% of practical control systems, ranging from consumer electronics such as cameras to industrial processes such as chemical processes. The PID controller helps get our output (velocity, temperature, position) where we want it, in a short time, with minimal overshoot, and with little error [2]. It also the most adopted controllers in the industry due to the good cost and given benefits to the industry [3]. Many nonlinear processes can be controlled using the well-known and industrially proven PID controller [4]. A considerable direct performance increase (financial gain) is demanded when replacing a conventional control system with an advanced one [4]. The maintenance costs of an inadequate conventional control solution may be less obvious. The tricky part of controller design is to figure out just how much of a corrective effort the controller should apply to the process in each case. Some situation requires tighter control of the process variable than On-Off control can provide. Proportional control provides better control because its output operate linearly anywhere between fully on and fully off [5]. As its name implies, its output changes proportionally to the input error signal. Proportional controller simply multiplies the error by a constant to compute its next output. In 1930s the control engineers discovered that the error could be eliminated altogether by automatically resetting the set point to an artificially high value [3][6]. The PID controllers function is to maintain the output at a level that there is no difference (error) between the process variable and the setpoint in as fast response as possible

The PID controller based on an input-output inversion procedure which is allow to determine a closed form expression of the command signal to be applied to the closed loop system. To obtain the best possible PID control for a particular application, the gain setting for each mode must initially be made. These setting are different for each system. While the system is actually running, tuning the adjustment are often be made to the gain setting to attain optimal performance. Gain adjustments can be performed by trial and error or automatically by auto tune controller [7].

Although the general PID concepts and the effects of adjusting the K_p, K_i and K_d constants are the same, when using a programmable logic controller to perform a PID control function, there are some minor differences in the way PID is adjusted. The PID control unit of a PLC performs all the necessary PID calculation on the iterative basis [8]. The PID calculation are not done continuously, but are triggered by a timing function.

2 Problem Formulation

DC Motor are easily damage without contribution of control methodology in it system. Frequently, the desired performance characteristics of control systems are specified in terms of the transient response. The transient response of a practical control system usually exhibits damped oscillation before reaching steady state. As for DC motor, having a high overshoot is an undesired condition since the starting current is very high. The position of single phase DC Motor of a conveyor system has been controlled by using PID controller. The position control of DC motor is very difficult when it is done by using traditional control techniques, as it requires a very complex mathematical model. By using PID controller, we can eliminate the need for the mathematical modeling and allows easy realization as a solution. Implementation of PID into PLC offers the easier way to troubleshoot a system as compared to the system using microprocessor, microcontroller or other controllers. With this advantage, the men in charge do not have to troubleshoot the system from a scratch when there is system problems happen.

3 Problem Solution

3.1 Research Methodology

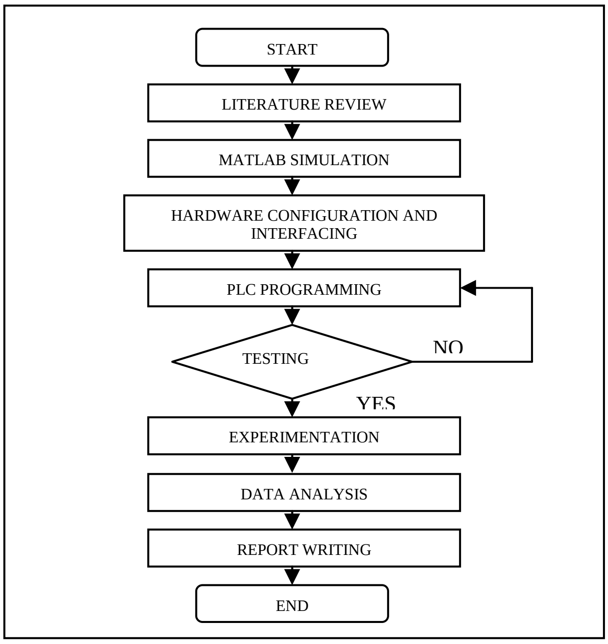

Figure 1: Research Methodology

3.2 Block Diagram of Conveyor System

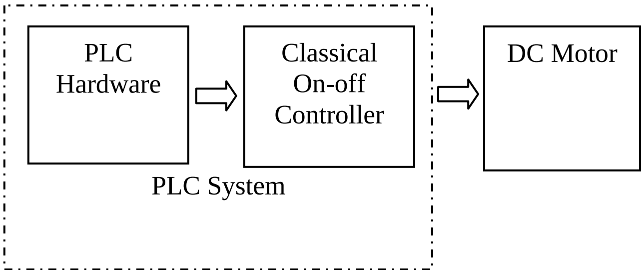

Figure 2: Implementation of On-off controller in PLC System

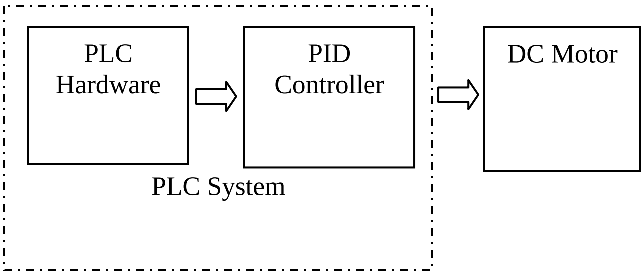

Figure 3: Implementation of On-off controller in PLC System

3.2.1 PID Controller Equation

The output of the system, $u(t)$ is the sum of three correcting terms, i.e.

$$u(t) = P_{\text{out}} + I_{\text{out}} + D_{\text{out}}$$

where P_{out}, I_{out} and D_{out} are the contributions to the output from the PID controller from each of the three terms, which defined as

$$P_{\text{out}} = K_p e(t)$$

$$I_{\text{out}} = K_i \int_0^t e(t)dt \quad \text{and}$$

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

where:
t = Time of the PLC clock
$u(t)$ = Output
$e(t)$ = Error = Setpoint – Process variable
K_p = Proportional Controller Mode Gain
K_i = Integral Controller Mode Gain
K_d = Derivative Controller Mode Gain

Here, P_{out} acts as the accelerator, which accelerates the target position. Besides, I_{out} and D_{out} are terms which reducing error to minimal error position (zero-error-position) and reducing oscillation due to load variation respectively.

For convenience, and K_d are expressed in term of K_p, i.e. $K_i = K_p / T_i$ and $K_d = K_p T_d$ where T_i is the integral time of the controller and T_d is the rate time.

Hence, the mathematical representation of the system is

$$u(t) = K_p e(t) + \frac{K_p}{T_i}\int_0^t e(t)dt + K_p T_d \frac{de(t)}{dt}$$

or in the s-domain representation

$$\frac{U(s)}{E(S)} = K_p\left(1 + T_d s + \frac{1}{T_i s}\right)$$

3.3 Hardware Configuration

Implementations of Classical Controller and PID controller in PLC are the same in hardware configuration and installation but different in software part using ladder programming. Instructions sets used in Classical Controller is too complex and very lengthy to program as compared to PID controller. The advantages of implementing of PID controller requires no complex instruction set for programming and having simpler calculation and algorithm as compared to Classical Controller. In hardware configuration of the PLC system, the physical connection of the motor with the solid state relay is shown in Figure 4. A pulse I/O board is used to generate PWM output to the solid state relay to meet high speed processing operation.

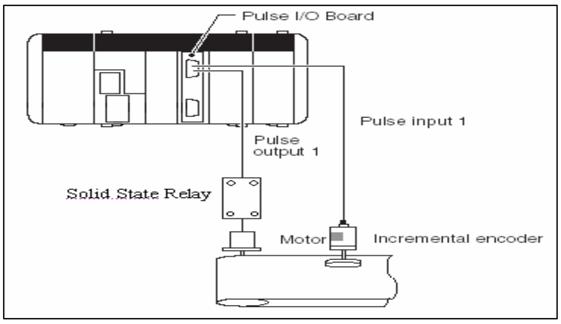

Figure 4: Physical Connection of the Hardware

3.4 Hardware Connection

In automation industries, a DC motor was considered to be a variable speed motor and an AC motor was considered as a constant speed motor. Although the future trend is towards the AC drives, DC drives are currently used in many industries because of the torque speed characteristics of DC motor that can be varied over a wide range while retaining high efficiency compared to AC motors. In robotic and actuations applications, a wide variety of speed or position control is required. While, in many manipulators, DC motor are used to follow a predetermined speed or position track under variable loads.

In Figure 4 the system configuration of the motor position controller using either the PID control input is shown. There PLC analogue card is inserted into the PLC system so that the analog interface can be connected to the DC motor. This system consist of analog interface, is an interfacing which converts the encoder data into PID control inputs. Based on the encoder reading, the PID will control the motor position plant and sending compensated voltage output that control the DC motor position. The PID controller

3.5 Results

In Table 1, it is shown that the implementation of programmable logic PLC with PID controller will produce a better control system performance of a DC Motor position with less error as compared to On-off classical Controller. It shows that complex calculation of classical controller algorithm does require improvement of mathematical modeling PID only needs human experience from decision making method.

Table 1: Comparison of Position Error Percentage

Type of Controller	Position Error Percentage (%)
On-off Controller	5.45
PID controller	3.25

3.5.1 P-Controller Mode

In Figure 5 shows the step response if only the P (Proportional) controller is applied in the PLC in Figure 3.3. A proportional controller allows tighter control of the process variable because its output can take any value between fully on and fully off, depending on the magnitude of the error signal. With proportional band, the controller output is proportional to the error or a change in measurement and offset (deviation from set-point) is present. This steady state error is the difference between the attained value of the controller and the required value. In this PLC, the gain for P is 110 and the set point is 3. As we can see, there is overshoot at 3.8 and it over the target value which is 3. In the end the system doesn't settle out any quicker than it would have with lower gain, but there is more overshoot. If we kept increasing the gain we would eventually reach a point where the system just oscillated around the target and never settled out-the system would be unstable.

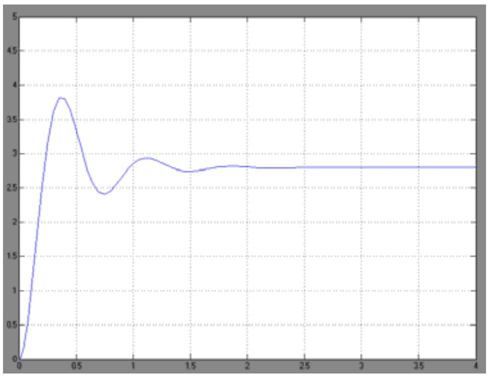

Figure 5: Response in MATLAB Proportional mode

3.5.2 PI-Controller Mode

In Figure 6, it shows that when the P (Proportional) and I (Integral) gain is applied in the PLC. To eliminate the offset error, the controller needs to change its output until the process variable error is zero. Integral action gives the controller a large gain at low frequencies that results in eliminating offset and "beating down" load disturbances. It also can reduce the final error in a system. Summing even a small error over time produces a drive signal large enough to move the system toward a smaller error. In this system, the gain for Integral is 20. In this graph we can see that the steady state error still occur but it is now at the set point input which is 3. However the overshoot became greater than before which is at 4.3. The rise time is also faster than just using the proportional controller.

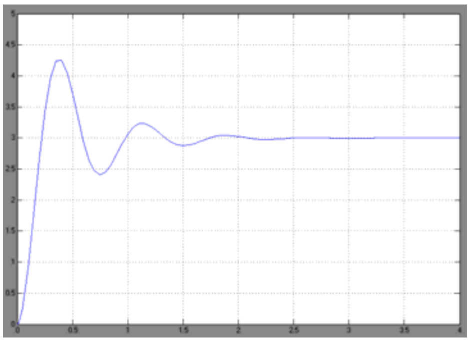

Figure 6: Response in MATLAB Proportional+Integral mode

3.5.3 PID-Controller Mode

In Figure 7 show the step response if the entire element in PID controller is applied in the system. As the proportional gain is increased, the controller responds faster. If the proportional gain is too high, the controller may become unstable and oscillate. The integral gain acts as a stabilizer. It provides power even if the error is zero. The Derivative controller counteracts the KP and KI terms when the output changes quickly. This helps reduce overshoot and ringing and it has no effect on final error. In this PLC, the gain for Proportional is 100, Integral is 20 and the gain for Derivative is 15. The step response for this system became smooth and the response is better than if we use P, PI, and PD. There is no overshoot; the rise time is faster and less error. The settling time is exactly at the set point which is 3.

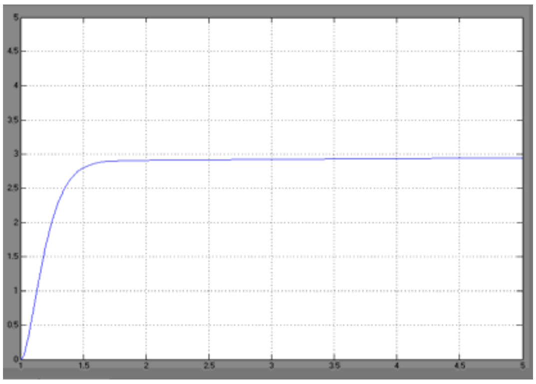

Figure 7: Response in MATLAB Proportional+Derivative+Integal Mode

4 Conclusion

The design and implementation of PID based on PLC has been presented in this project. The development of the PID was done after series detail study of PID theories and OMRON CQM1H PLC manual. Through the study of this project, it has proved that PID can be implemented into PLC with certain optimization, with the proper hardware installations and knowledge about PLC programming.

.

References:

[1] J. Swder,G.Wszoek,W.Carvalho, Programmable controller design forElectro-pneumatic Systems, Journal of Material Processing Technology 164- 1655(2005) 14659-1465

[3] Aurelio Piazzi and Antonio Visioli, A Noncausal Approach for PID control, Journalof Process Control, 4 March 2006.

[2] Terry Bartelt, Industrial Control Electronics: Devices, Systems and Applications, Second Edition, Delmar, 2002

[4] S. Gerks¡ic¡ et al. / Control Engineering Practice 14 (2006) 935–948, Advanced Control Algorithms Embedded in a Programmable Logic Controller, 22 July 2005.

[5] Paul H. Lewis and Chang Yang (1997). Basic Control Systems Engineering: Prentice Hall

[6] Carl Knopse, Guest Editor, PID Control, IEEE Control System Magazine, February 2006

[7] Carl Knopse, Guest Editor, PID Control, IEEE Control System Magazine, February2006

[8] John R. Hackworth and Frederick D. Hackworth, Jr. Programmable Logic Controllers, Programming Methods and applications, Pearson Prentice Hall, 2004.