

Spring 2016

PLC Water Pump Control

Kevin Logsdon
kal62@zips.uakron.edu

Please take a moment to share how this work helps you [through this survey](#). Your feedback will be important as we plan further development of our repository.

Follow this and additional works at: http://ideaexchange.uakron.edu/honors_research_projects



Part of the [Controls and Control Theory Commons](#)

Recommended Citation

Logsdon, Kevin, "PLC Water Pump Control" (2016). *Honors Research Projects*. 386.
http://ideaexchange.uakron.edu/honors_research_projects/386

This Honors Research Project is brought to you for free and open access by The Dr. Gary B. and Pamela S. Williams Honors College at IdeaExchange@UAkron, the institutional repository of The University of Akron in Akron, Ohio, USA. It has been accepted for inclusion in Honors Research Projects by an authorized administrator of IdeaExchange@UAkron. For more information, please contact mjon@uakron.edu, uapress@uakron.edu.

PLC Water Pump Control

Kevin Logsdon

Department of Electronic Engineering Technologies

Honors Research Project

Keywords

PLC (Programmable Logic Controller) - Programmable computer to control industrial processes

Variable Speed Drive - Supplies power to the water pump

Proximity Sensor - Determines distance to an object in front of it

PID Controller - Internal to the PLC, PID controller is able to determine a proper output according to the user defined parameters

Setpoint - User desired value that the system tries to achieve

Process Variable - Input variable for the PID controller

Control Variable - Output variable from the PID controller

Oscillation - When the response of a signal has begun to oscillate

RSLogix 5000 - Software used to program the PLC using ladder logic

Closed Loop System - A system that has an automated response to changes

Abstract

This project set out to use a Programmable Logic Controller (PLC) to control a water pump to maintain the water level in a tank at a desired setpoint. A proximity sensor is mounted above the upper reservoir to detect the distance to the water level from the sensor, which will be communicated back to an Allen-Bradley CompactLogix PLC. To control the water level, an AC variable speed drive, that can receive instructions from the PLC, is connected to a water pump that will ramp up the RPM as the water level drops below the setpoint to bring it back to the set normal level, and will slow down the rpm as the water level gets closer to the setpoint. The PLC was programmed using software called RSLogix 5000, produced by the Allen-Bradley company, so that the proximity sensor can trigger the pump to start and begin pumping water into the main tank as it is drained. The RSLogix design would be limited to only this tank and pump, however, all of the PLC equipment could be repurposed to be used in a wide variety of applications.

Introduction

The purpose of the work done here was to gain further knowledge on the usage and applications of PLCs and PID controllers. The basic components of the system are the main water tank, an Allen-Bradley CompactLogix(1769-L30ERM) PLC, a Pepper & Fuchs proximity sensor (UJ 750-30GK-13) to detect the level of the water in the tank, an Optidrive single phase variable speed drive (ODE2-11005-1H012-01), and the water pump that is part of the water tank assembly.

The goal when designing the project was to use a PID to control the water level of the tank, while the drain is held open, so that the water level would remain at a determined value. Proportional-Integral-Derivative (PID) control is used extensively in the design of industrial control systems, and was vital to this project. The systems in which PID control is used are called closed loop systems. A closed loop system is any system that has a process variable that can affect the output, but can also automatically adjust the output so that it reacts to changes in the process variable. For example, a closed loop system that should be familiar would be a cruise control system on a vehicle. When the desired speed is set, the process variable becomes the current speed, and the output being the throttle, if the current speed is less than the set speed the throttle will open to accelerate the vehicle, alternatively, if the current speed goes above the set speed the throttle will close and let the car slow back down, until the set speed is reached again, and then the throttle will open enough to maintain the set speed. In these systems, the PID controller is what actually calculates what the output should be. Going back to what PID actually stands for, the controller works by finding the proportional, integral and derivative responses of the system, and then summing them together to determine the output.

Types of response

The proportional response is the ratio of the output to the error signal, which is the difference between the set point and the process variable. Increasing the coefficient for the proportional gain will increase the speed that the system responds, however, it will also create oscillation within the response.

Integral response uses the same error signal, but instead this part sums the error term over time. The integral response will increase as the system runs, until the error becomes zero.

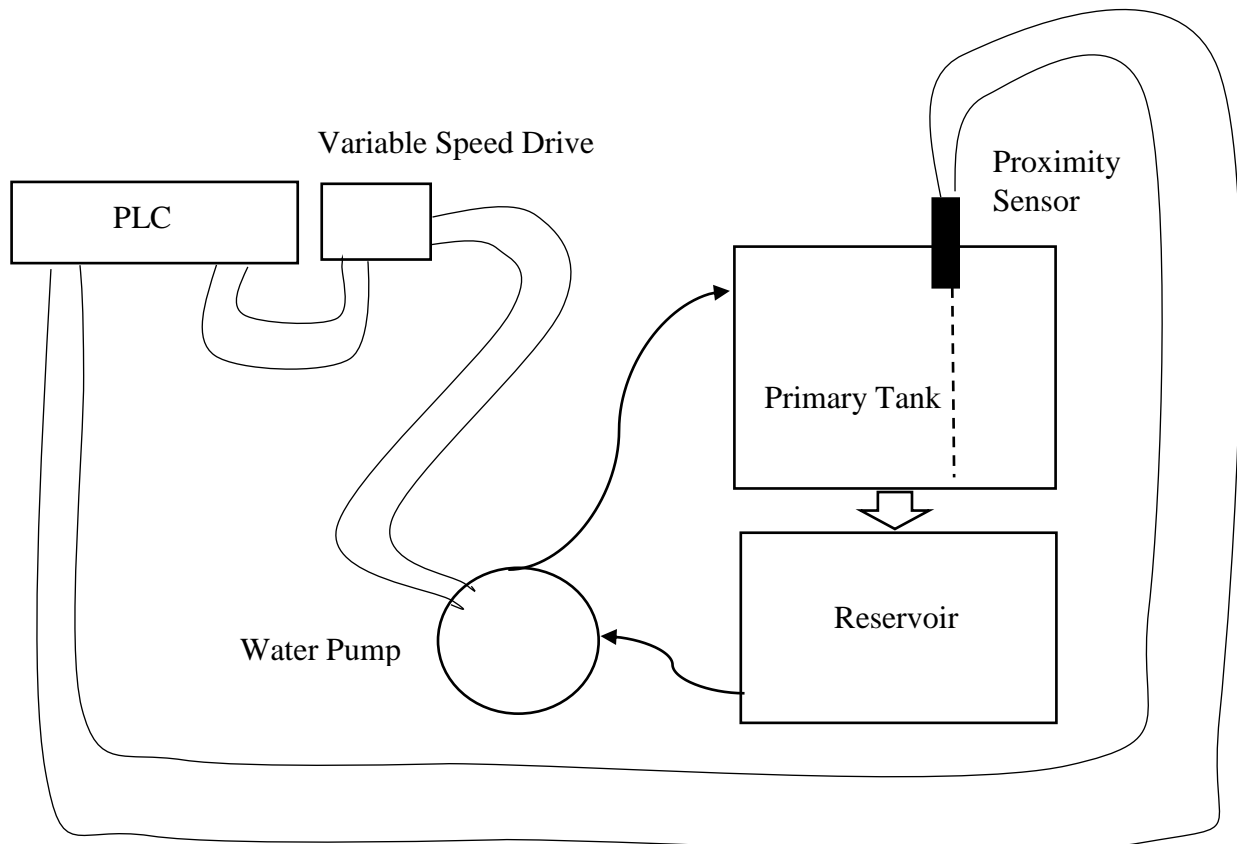
Increasing the coefficient of the integral gain will affect the steady-state error, which is the error between the final output and the set point. Steady state error is shown as:

$$\text{Steady state error} = \lim_{t \rightarrow \infty} (r - y(t))$$

The variables used in the equation being as follows, r is the desired output of the system, and $y(t)$ is the output of the system at a given time. While tuning the integral response, consideration should be taken to get the steady state error as low as possible.

The derivative response is affected by the rate of change of the process variable. This means that increasing the coefficient of the derivative response will dampen the oscillations of the output, and cause the output to resist fast changes.

Design



This project is constructed around a pre-manufactured hydraulics table. Above the upper tank is where the proximity sensor is mounted so that it can point down into the open top of the tank so that it can detect the level of the water as it flows in. Just below the upper tank there is a reservoir which is where the upper tank drains water into. The table also has a water pump built into the assembly, which is driven by the variable speed drive so that the water flow can be adjusted by the PLC itself.

Proximity Sensor

The proximity sensor used in this project is a Pepper & Fuchs UJ 750-30GK-13, which is an ultrasonic sensor. These operate by emitting ultrasonic sound waves, which are reflected back after hitting an object. While operating, the sensor will switch between sending and receiving modes, that way, once there is return of the ultrasonic sound waves the sensor will receive them, and calculate the distance based on the time it took to be reflected back. In this project the sensor will emit the waves towards the bottom of the tank, and reflect off of the water to determine the level.

To program the PLC so that it would operate as needed, a way to translate the proximity sensor's raw data in a range of 0-32000 bits, to an easier to use scale: centimeters. To do this, the proximity sensor was connected to its power supply, of 24v DC, and placed on a table with a meter stick next to it. With this setup, two data points were taken, one at 20cm away which had a raw value of approximately 6400, the second data point was at 74cm, and had one of

approximately 29000. With these two data points, to determine the equation to change the raw data into centimeters, some simple algebra was incorporated.

$$\frac{29000-6400}{74-20} = \frac{22600}{54} = 418.518$$

$$y = 418.518 * X(cm) + b$$

$$29000 = 418.518 * (74cm) + b$$

$$-1970.33 = b$$

$$\frac{y-(-1970.33)}{418.518} = X(cm)$$

$$Distance(cm) = \frac{TankProx + 1970.33}{418.518}$$

The final equation is used to display an easy to recognize value, through the usage of a “Compute Block” within the RSLogix5000 software, while also being used in the PID configuration. However, for the equation to be used in the PID configuration the format of the values had to be changed. As it is in this equation, a value of 35cm for example, is closer to the sensor, and therefore a higher level in the tank, than a value of 65cm. When configuring the PID the equation must be changed so that the PID can accept the parameters of the operating range, to do this all that had to be done was create an additional compute block with the equation changed:

$$Distance(cm) = 100 - \frac{TankProx + 1970.33}{418.518}$$

With this equation, a value of 65cm will be closer to the sensor rather than a value of 35cm, which is nearer to the bottom. With these values, they will fit the parameters for the PID configuration, so that a high and low limit could be set on the process variable.

Variable Speed Drive

The AC variable speed drive used in this project is an Optidrive ODE2-11005-1H012-01, which is a ½ HP, 7Amp drive. This drive is connected to the preinstalled single phase water pump from the hydraulics table, going through a disconnect switch to the 120VAC input, from the drive's AC Voltage output. To operate as intended the drive's parameters had to be set, within the programming of the drive, to match those of the water pump. In particular, the AC voltage output had to be set at 120V, the maximum RPM of the water pump was input at 1200 RPM, and the maximum current output had to be set to 7 Amps, and the analog input was set to use the 0-20mA scale. To enable the output for the drive, terminals 1 and 2 were connected together with a jumper wire, terminal 1 is a 24v user output meant to be used with a switch inline to terminal 2, which is the digital input used to enable the output when there is a High logic level of 8v to 30v, or disable the output if there is Low logic level of 0v to 4v, however, since the disconnect that is used is located between the variable drive output and the water pump input, it is not necessary to have one.

Programmable Logic Controller

The PLC used in this project is an Allen-Bradley CompactLogix 1769-L30ERM, however, the PLC itself is not enough to construct the project, a few additional input and output cards must be used. The cards that were needed for this project to operate as intended were an analog input card (1769-IF4/B), as well as an analog output card (1769-OF4/A). The analog

input card has two sets of contacts, one being inputs for voltage devices, and the other being current inputs for current devices, despite having 8 contacts, there can only be 4 total inputs into the card. The proximity sensor, that operates on a 4-20mA scale, is connected to V/I In 0 (-) and I in 0 (+), so that the correct input is given to the PLC. Within the programming of the PLC the card itself had to be configured so that it could accept the input, to do this input 0 on the card was enabled and set to a 4-20mA scale. The tag, within RSLogix5000, for the proximity sensor was named "TankProx" and set to a data scale of 0-32000 bits.

The analog output card is set up similarly, however this card has 10 contacts, 2 for Analog Common, "I Out 0 (+)" through "I Out 3 (+)," and an equivalent for voltage with contacts 0 through 3, on this card there can also only be 4 outputs in total. The output from the PLC is set up to be output on a 0-20mA scale, from within the configuration of the card in RSLogix5000. This output is connected, on the card, to "I Out 0 (+)" and to one of the "ANLG Com" terminals, and is connected to the variable speed drive in terminals 6 and 7, respectfully, which is the analog input for the drive, used to give the variable drive its input so that it can send it's proportional output to the water pump. For both of these cards, the selection for the voltage or current contacts is done within the configuration for the I/O, each channel can be configured independently, meaning channel 0 can be set to accept a current input, while channel 1 could be configured for a voltage input, and so on.

While going unused, the PLC also had more cards that could be used for other systems. There is a DC input card (1769-IQ16/A), which has 16 possible inputs, and 2 "DC COM" terminals, these can be used for any digital input into the PLC. Another card installed is a relay card (1769-OW16/A), which has 16 relays that can be used. The last additional card is a DC output card (1769-OB16/B) which has 16 DC outputs.

PID Control

The main component of this project would be the PID control for the output into the AC variable speed drive, within RSLogix 5000, which uses the input from the proximity sensor and the output from the AC variable speed drive to power the water pump. The process variable that is used for the PID is the output of the equation:

$$Distance(cm) = 100 - \frac{TankProx + 1970.33}{418.518}$$

This equation gives the PID the proper parameters needed for the process variable to be configured. For the project a setpoint was chosen of 65.0, keeping in mind that with this equation, the setpoint of 65.0 is equivalent to 35cm of distance from the proximity sensor. This means that the PID will operate with the goal to get the process variable to match the setpoint at 65.0, and remain at that level. The process variable was configured to have an operating range of 70.0 at the unscaled max, and 35.0 at the minimum, with this range the setpoint could have been set to any of the values between them.

The control variable, which has the tag “Motor_Ctrl,” is scaled to the range of -32,000 to +32,000 bits, representing 0% and 100% output, respectively, and is configured to output in the 0-20mA range, within the analog output card configuration. This is what is used as the input for the AC variable speed drive, which will scale it to drive the water pump. In the configuration of the PID, high and low limits on the control variable can be set, and are set to 100% for the high limit, and 18% for the low limit. The minimum output being set to 18% is done so that the water pump doesn't fully spins down, even if the process variable is equivalent to the setpoint, this greatly reduces the time it takes for water to start being pumped into the top reservoir, as without this if the drain is opened, the PID is able to react, unfortunately the water pump itself isn't able

to respond quickly enough to the input change and an unacceptable amount of water is lost before it begins to refill, causing the system to take a long period of time to raise the water level back to the setpoint.

The tuning of the proportional, integral and derivative components for this PID were done through the method of trial and error, as the configuration tool for the PID control makes it easy to adjust values quickly and while the system is operating. Therefore, to find the desired coefficients, the integral and derivative components were set to 1.0, the proportional value was then increased until the system responded quickly enough, after which the water level was decreased and the integral value was changed until it was able to reach the setpoint quickly and maintain the level without too severe of oscillation, and finally the derivative value was changed to decrease the severity of the oscillation into acceptable levels. The proportional gain is set to a value of 5.0, the integral gain is set to 3.0, and the derivative component is set to 2.0, with these settings, when the drain is propped open, it is able to maintain the water level of the given setpoint.

Results

The goal of this project was to create a closed loop system that could maintain set water level in a hydraulics table. After constructing the system, with a fair share of troubleshooting, it was able to perform as desired. When the drain is closed and the water level at the setpoint, the AC variable speed drive will output, to the water pump, its minimum output of 18%, and once the drain is opened and the water level drops below the setpoint the variable drive will quickly ramp up the output, based on its input from the PID, and increase the amount of water being

pumped, so that it can resist the change in the process variable. When changing the setpoint, the project will still work correctly, and increases the water level or allows it to fall lower, before ramping up the water pump, depending on what value is used for the setpoint.

Discussion

The completion of this project has taught me a lot in regard to the design and operation of PLCs and closed loop systems. This knowledge could be applied to a wide variety of career opportunities, as PLC and PID control are widely used in many forms for manufacturing processes. PLCs are used in virtually all automated manufacturing processes, for example PLCs are used to control automotive manufacturing machinery, such as large presses. An example of PID control is their usage in large furnaces to control the temperature, to ensure the temperature remains where it should be for the process. In the future, the knowledge gained from this project could help influence a potential employer, who may find it valuable. After working on this project I have learned much about how PLCs can be used in closed loop systems to perform a wide variety of tasks. If the proper input or output cards are installed, a PLC can be used to control any combination of systems. In addition to the knowledge gained about PLC, I have also learned a great deal about PID control in closed loop systems. From the work on this project I gained a better understanding of the concepts involving PID control, such as effective ways to tune a PID, and the types of responses that make up the PID controller.