PID Control Parameters

In this article you will learn the **Level simulator behaviour based on Proportional, Integral and Derivative parameters** that we gonna play around.

I try to make this document as simple as possible to make PID Control easy to understand

Explaining PID Control to a kid!

"Imagine you have a 🤖 robot friend that helps you bake 🍪 cookies. The robot has a special tool called a PID controller.

The robot's job is to make sure the oven temperature stays just right so that the cookies turn out perfectly. 🌡️🍪👌 The PID controller has three parts: P, I, and D. 🔄

P stands for Proportional

It means that the robot looks at the oven temperature and checks if it's too 🔥 hot or too ❄️ cold compared to the right temperature for baking cookies.

- If it's too hot, the robot knows it needs to make it cooler.
- If it's too cold, the robot knows it needs to make it hotter.

  The robot tells the oven to adjust the temperature just a little bit to get closer to the right temperature. ⚖️🔧

I stands for Integral

The robot also looks at how long the oven temperature has been different from the right temperature.

If it's been a long time, the robot knows it needs to do a little more to fix it.

It adds up all the times the temperature has been wrong and makes a bigger adjustment to get it back on track. ⏰🔁🔧

D stands for Derivative

The robot pays attention to how fast the oven temperature is changing. If it's changing really quickly, the robot knows it needs to be careful and not make big adjustments.

It helps the robot to be smoother and not overreact. 📈🔧

So, the PID controller helps the 🤖 robot to keep the oven temperature just right by looking at the temperature, how long it's been wrong, and how fast it's changing.

The robot uses this information to make small or big adjustments to the oven until the cookies bake perfectly. Yummy cookies for you to enjoy! 😋🍪"

Now let's understand the effect of PID parameters on our Level simulator

Pre-conditions

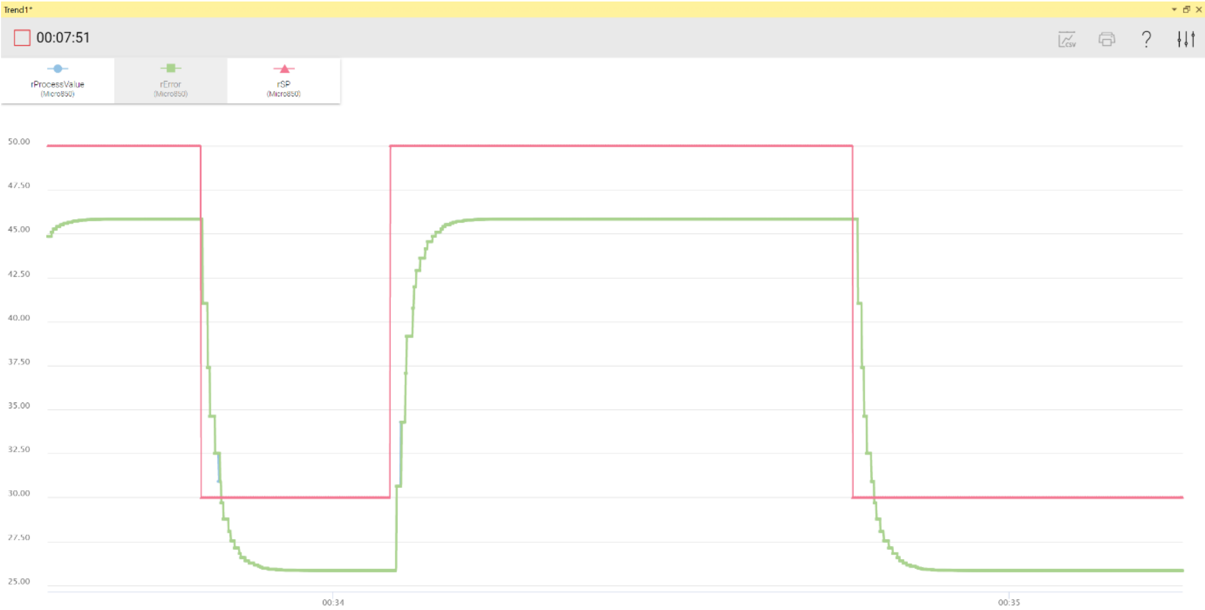
We are using the following parameters for the Level Simulator which defines the reaction time of the filling and draining valves

level_sim_1.rDraining_coefficient		2.0
level_sim_1.rFilling_coefficient		2.0

If your co-efficient values are different then with the same gains as mentioned in the below cases, your response will be different. So you need to alter the gain. One example with Co-efficient 0.5 for Filling and heating in the end of this section.

Case 1: P Control

Gain, Kc = 12.0
Draining = 50.0

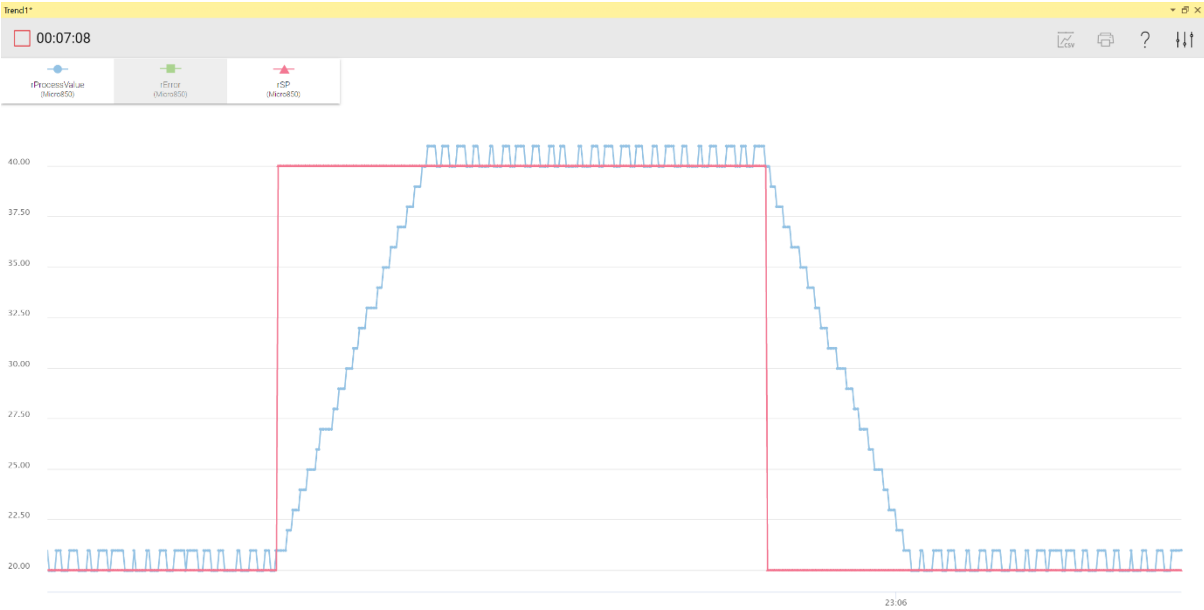


Steady state error

When you use only P Control, it cannot reduce the error if there exist external disturbance (draining) and you will always get steady state error.

Case 2: PI Control

Gain, Kc = 12.0, Gain Ti = 0.0001, Gain Td = 0.0, Gain.FC= 0.0
Draining = 50.0
Error = 0 ~ 2.0



Oscillation

We notice that the integral action is trying to reduce the error but not 100%. These oscilaltions means, we have to increase the integral action

Case 3: PI Control with Overshoot

Gain, Kc = 12.0, Gain Ti = 4.0, Gain Td = 0.0, Gain.FC= 0.0
Draining = 50.0
Error = 0.0005

No Oscillation but overshoot

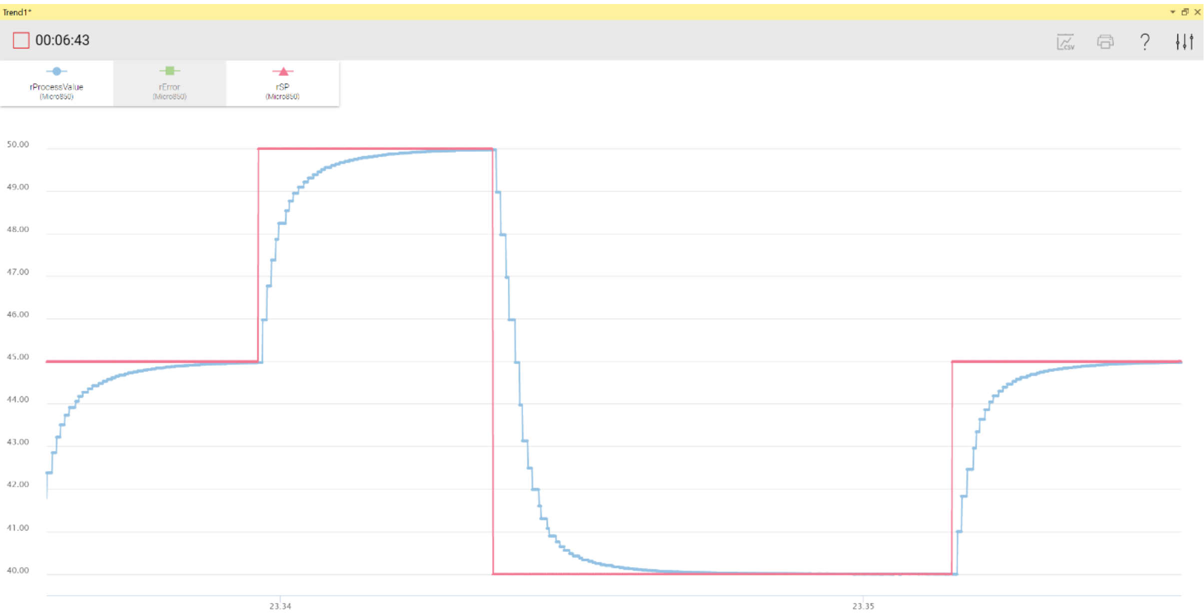
We notice that the oscillation has been removed and the error is close to zero but still there is an overshoot that we can remove by giving derivative action. So let's increase the derivative action

Case 4: PID Control

Gain, Kc = 12.0, Gain Ti = 4.0, Gain Td = 3.0, Gain.FC= 20.0

Draining = 50.0

Error = 0.0005



Optimized response

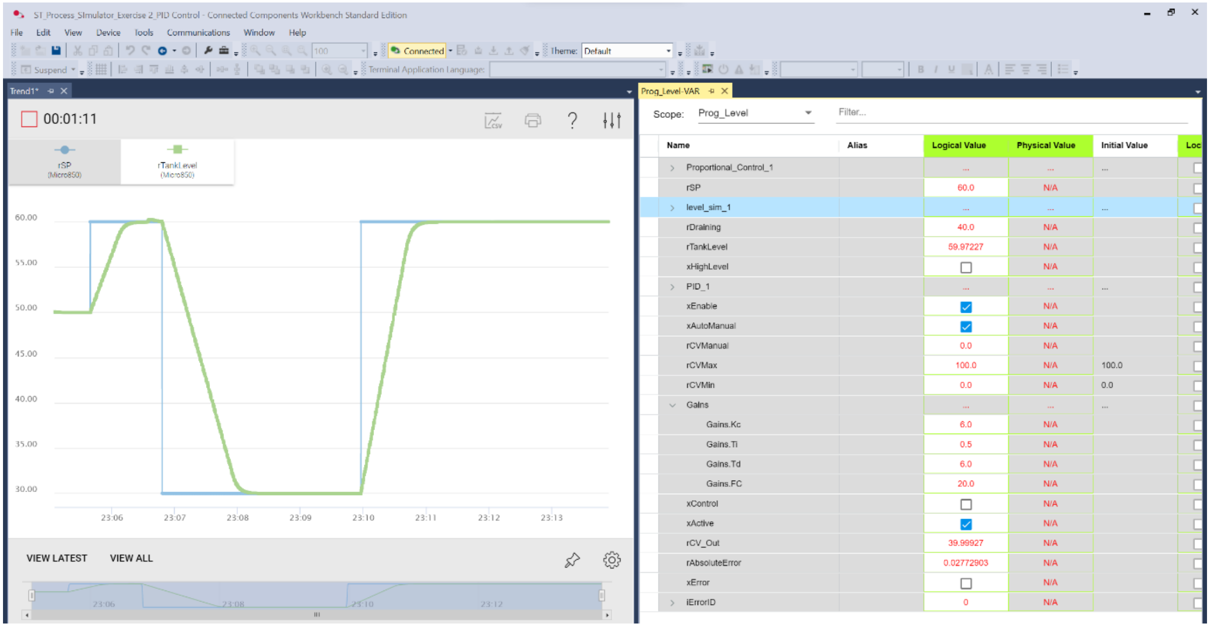
Finally the overshoot is removed using Derivative action and FC which smoothes the response of the PID Control.

Pre-conditions

Using Filling and draining co-efficient as 0.5

level_sim_1.rFilling_coefficient		0.5
level_sim_1.rDraining_coefficient		0.5

Reducing the co-efficient from 2.0 to 0.5 will reduce the filling and draining action in the process simulator which means it will take longer for filling operaiton to reach the set-point. Hence the PID Gain parameter will change. I have tuned the Process simulator with co.effient 0.5 as below



We hope you enjoyed the lesson. Feel free to play around with co-efficient and PID gain to get your optimize response. Feel free to share your PID curve on LinkedIn page.