



Escola Politécnica da Universidade de São Paulo
PTC3418 – Laboratório de Automação

LAB3

Controle de Processo Industrial

Nota Técnica

Implementação de Controladores PI / PID no OpenPLC

Sumário

1	Resumo executivo	3
2	Recursos comuns a controladores PID	3
2.1	Formulação ISA.....	3
2.2	Ação direta/reversa.....	3
2.3	Operação automática e manual	4
2.4	Anti-windup.....	5
3	O bloco funcional PID do OpenPLC	6
4	Implementação alternativa para o controlador PI	7
5	Referências.....	9

1 Resumo executivo

Este documento apresenta alguns detalhes de implementação de controladores PI / PID no OpenPLC: parametrização, ação direta/reversa, auto/manual, anti-windup e temporização.

2 Recursos comuns a controladores PID

2.1 Formulação ISA

A formulação ISA (de International Society for Automation) é uma equação padronizada para implementações de controladores PID. No domínio de Laplace, em sua forma mais simples é dada por¹

$$\frac{U(s)}{E(s)} = KP \left[1 + \frac{1}{TRs} + TDs \right],$$

onde $U(s)$ é a saída do controlador (variável manipulada), $E(s)$ é a entrada do controlador (sinal de erro), KP é o ganho do controlador (componente proporcional), TR é o tempo de repetição (componente integral) e TD é o tempo derivativo (componente derivativa).

$E(s)$, a entrada do controlador é usualmente o sinal de erro entre a referência (*setpoint*) e a saída da planta (variável controlada ou variável de processo) e é dado por

$$E(s) = R(s) - Y(s) \quad \text{para ação direta}$$

ou

$$E(s) = Y(s) - R(s) \quad \text{para ação reversa,}$$

onde $R(s)$ é a referência e $Y(s)$ é a saída da planta.

2.2 Ação direta/reversa

Com relação à ação do controlador, há dois tipos comuns de ações: direta e reversa.

Quando uma variação positiva na saída do controlador ocasiona uma variação positiva na saída da planta, diz-se que o controlador apresenta ação direta.

Quando uma variação positiva na saída do controlador ocasiona uma variação negativa na saída da planta, diz-se que o controlador apresenta ação reversa.

Por exemplo, no tanque da Figura 2-1, um controlador de nível que atue na válvula de entrada tem ação direta (um aumento na abertura da válvula tende a aumentar o nível), ao passo que um controlador de nível que atue na válvula de saída tem ação reversa (um aumento na abertura da válvula tende a diminuir o nível).

¹ Por simplicidade, a formulação ISA apresentada aqui tem uma função de transferência imprópria. Implementações reais utilizam aproximações próprias para o termo derivativo.

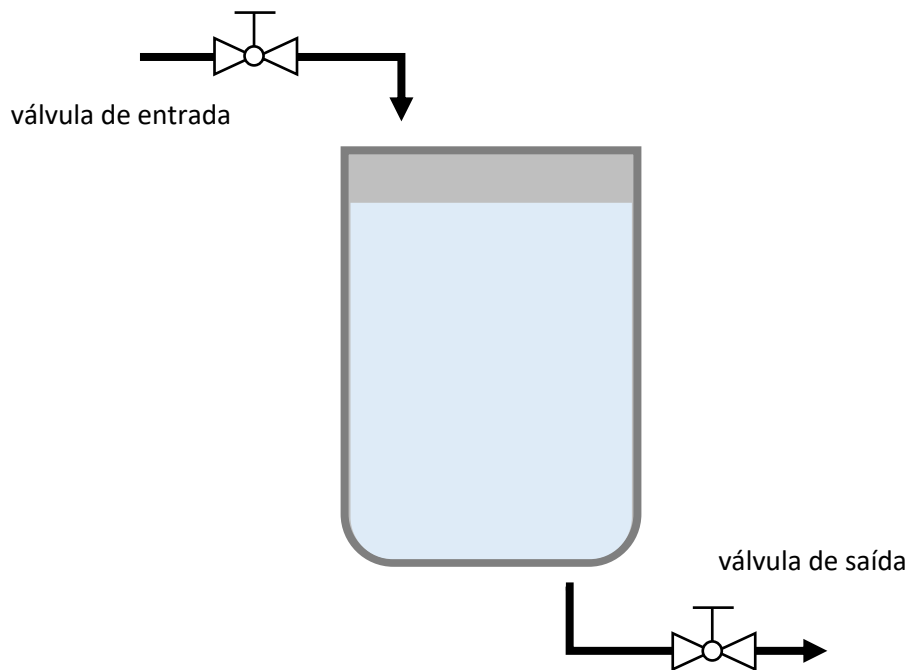


Figura 2-1 Exemplo de ações direta e reversa

2.3 Operação automática e manual

Na operação automática, a saída do controlador é calculada dinamicamente a partir do sinal de erro, segundo a lei de controle do PID. Em operação manual, o operador da planta especifica explicitamente o valor da saída do controlador.

É usual que os controladores possuam algum tipo de recurso para evitar que haja mudanças abruptas na saída do controlador quando se muda de modo de operação (comumente referido como *bumpless transfer*).

Operação manual é desejável em diversas situações, por exemplo:

- a) Na inicialização de um controlador. Quando um controlador PID se inicia, a condição inicial do integrador tem um valor arbitrário, usualmente nulo, o que pode produzir instantaneamente uma saída inadequada, a depender do ponto de operação da planta nesse instante. Um modo usual de se iniciar um controlador é operá-lo manualmente até que a saída do controlador esteja num valor adequado ao estado da planta e então ativar o modo automático.
- b) Operações não usuais. Partida ou parada de processos, esvaziamento de tanques, mudanças no processo, etc.
- c) Comportamento inadequado do controlador, especialmente por sintonia inadequada.

2.4 Anti-windup

O fenômeno de *windup* ocorre quando a saída de um controlador que tenha uma componente integral satura e o sinal de erro é não nulo. Como nessa situação o controlador não é capaz de produzir uma ação de controle que anule o sinal de erro, o valor absoluto da componente integral aumenta continuamente, ocasionando perda de controle. Caso o valor seja muito alto, mesmo que a situação que ocasionou o *windup* seja sanada, um tempo muito longo pode ocorrer até que esse valor retorne aos níveis normais.

Toda implementação de controlador PID deve ter algum tipo de recurso de anti-*windup*. Há vários esquemas para isso, sendo que o mais comum é congelar o estado do integrador quando ocorre saturação do controlador.

3 O bloco funcional PID do OpenPLC

O bloco PID da biblioteca padrão do OpenPLC é apresentado na Figura 3-1.

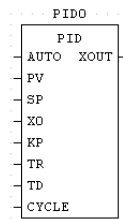
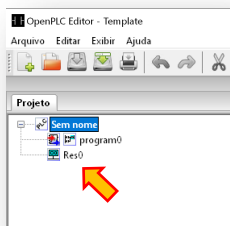


Figura 3-1 Bloco funcional PID

Mais informações sobre esse bloco podem ser encontradas em [1,2]. Os parâmetros de entrada são os seguintes:

- AUTO [1]: operação em modo automático. XOUT é calculado segundo a lei de controle.
[0]: operação em modo manual. $XOUT = XO$ (XO pode ser especificado pelo usuário).
- PV (Process Variable). A saída da planta.
- SP (Setpoint). Referência.
- XO $XOUT = XO$ quando o controlador opera em modo manual.
- KP Ganho do controlador (componente proporcional, intensidade geral do controle).
- TR Tempo de repetição (componente integral).
- TD Tempo derivativo (componente derivativa).
- CYCLE Período de amostragem para a implementação discreta do controlador. Esse valor deve ser o mesmo especificado no ciclo do programa em que ele está inserido. Para verificar, dê um duplo clique no objeto Res0



e verifique na tarefa correspondente o período especificado (valor padrão: 20ms ou T#20ms na formatação de variáveis do tipo TIME do OpenPLC). O parâmetro CYCLE deve receber o mesmo valor.

A saída do bloco, XOUT corresponde à saída do controlador, calculada segundo a formulação ISA.

Observações importantes:

- a) O sinal de erro neste bloco é calculado por $(PV - SP)$, ou seja, o bloco tem ação reversa. Para implementar ação direta, use valores negativos de ganho.
- b) Caso a carga computacional fique excessiva, diminua o período de amostragem. Não é necessário modificar KP, TR ou TD quando se altera o período.
- c) O bloco não tem recurso de anti-*windup*.

4 Implementação alternativa para o controlador PI

Dadas as limitações do bloco PID, especialmente a ausência de anti-*windup*, sugere-se a implementação da Figura 4-1 para implementar um controlador PI com ação direta e anti-*windup*.

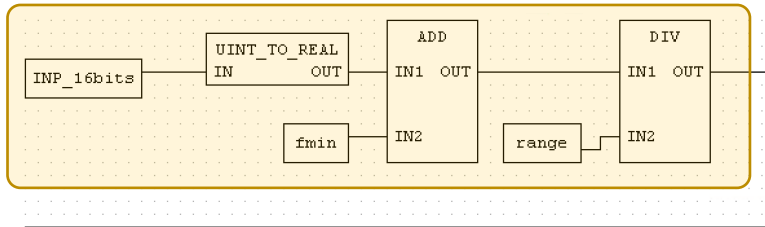
A implementação faz uso do bloco INTEGRAL da biblioteca do OpenPLC, que usa argumentos similares aos do bloco PID, particularmente é necessário especificar o argumento CYCLE.

O esquema de anti-*windup* limita os valores da saída (e.g. entre 0 e 100%). Quando essa limitação está ativa, ou seja, quando o valor limitado é diferente do valor originalmente calculado, a entrada do integrador é substituída por um valor nulo, efetivamente congelando o estado do integrador, porém deixando as outras componentes do controlador ativas.

O diagrama da figura não implementa operação automática/manual, mas isso pode ser feito com relativa facilidade, caso desejado, utilizando uma lógica adequada e os argumentos R1 e X0 do bloco INTEGRAL. Mais detalhes sobre este bloco podem ser vistos em [2].

Note que tanto a entrada como a saída do controlador utilizam o tipo REAL do OpenPLC (número de ponto flutuante de 32 bits).

Conversão 16 bits → real (entrada)



Conversão real → 16 bits (saída)

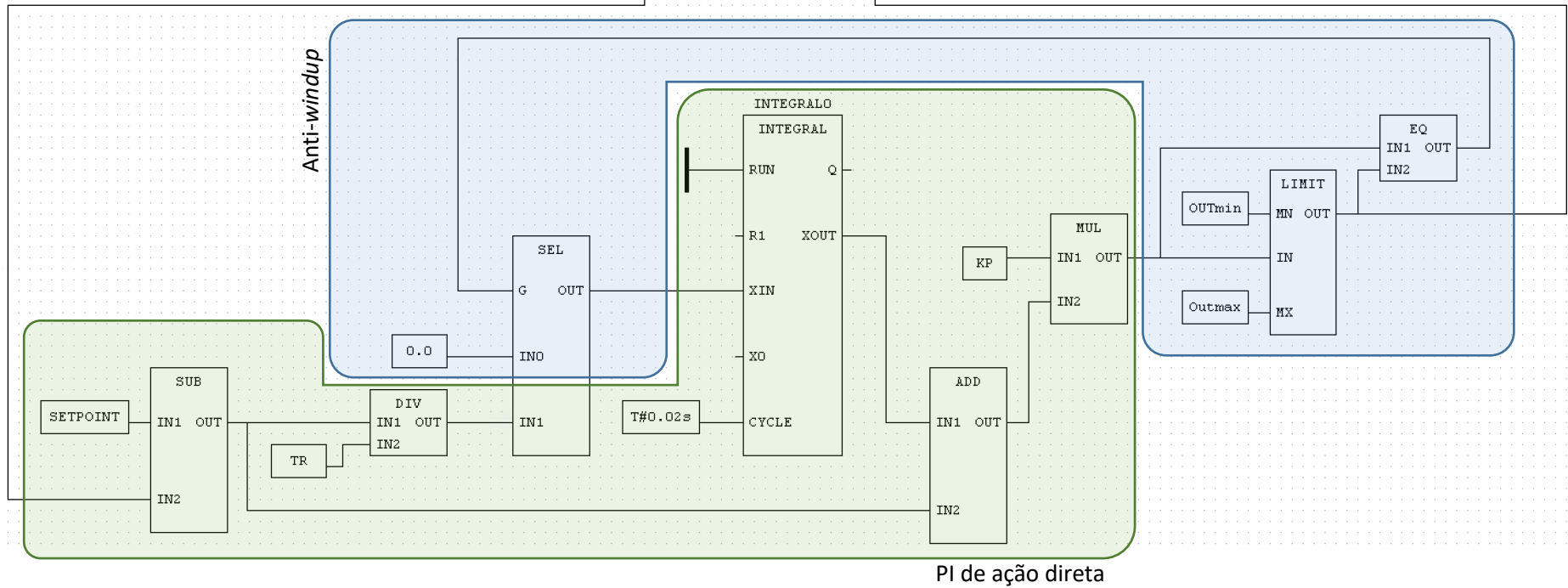
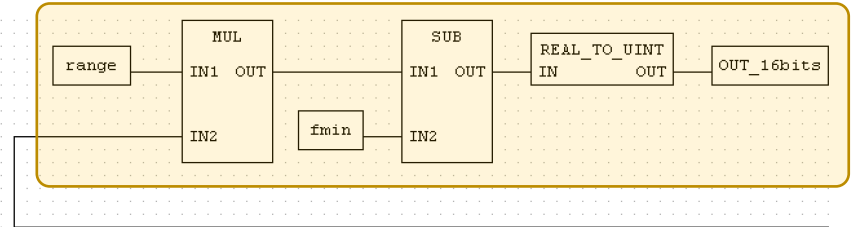


Figura 4-1 Controlador PI de ação direta com anti-windup

5 Referências

- [1] How to set up the PID block on OpenPLC Editor?
<https://openplc.discussion.community/post/how-to-set-up-the-pid-block-on-openplc-editor-9322249> (acessado em junho/2021).
- [2] PID in openplc
<https://openplc.discussion.community/post/pid-in-openplc-10427381> (acessado em junho/2021).

V2021a (RPM)