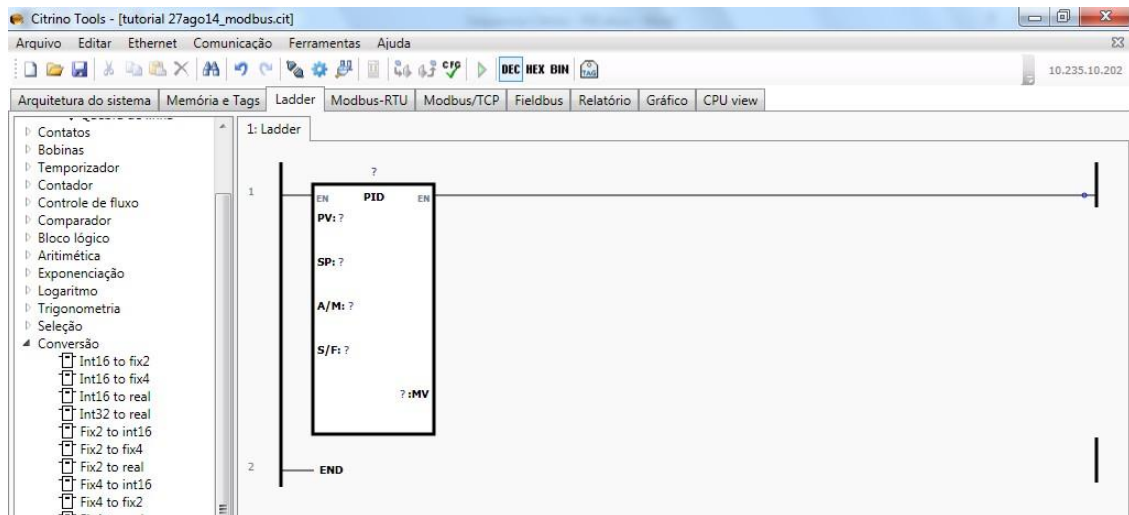


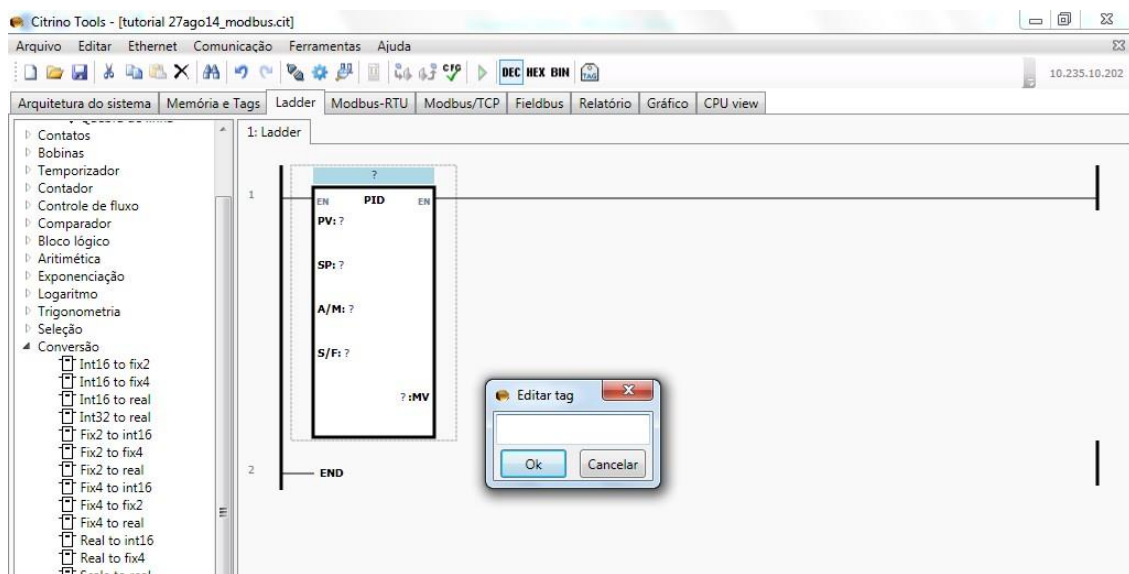
Tutorial experimento 1: Configuração de controlador PID no CLP Citrino

Ganhos Kp, Ki e Kd do controlador

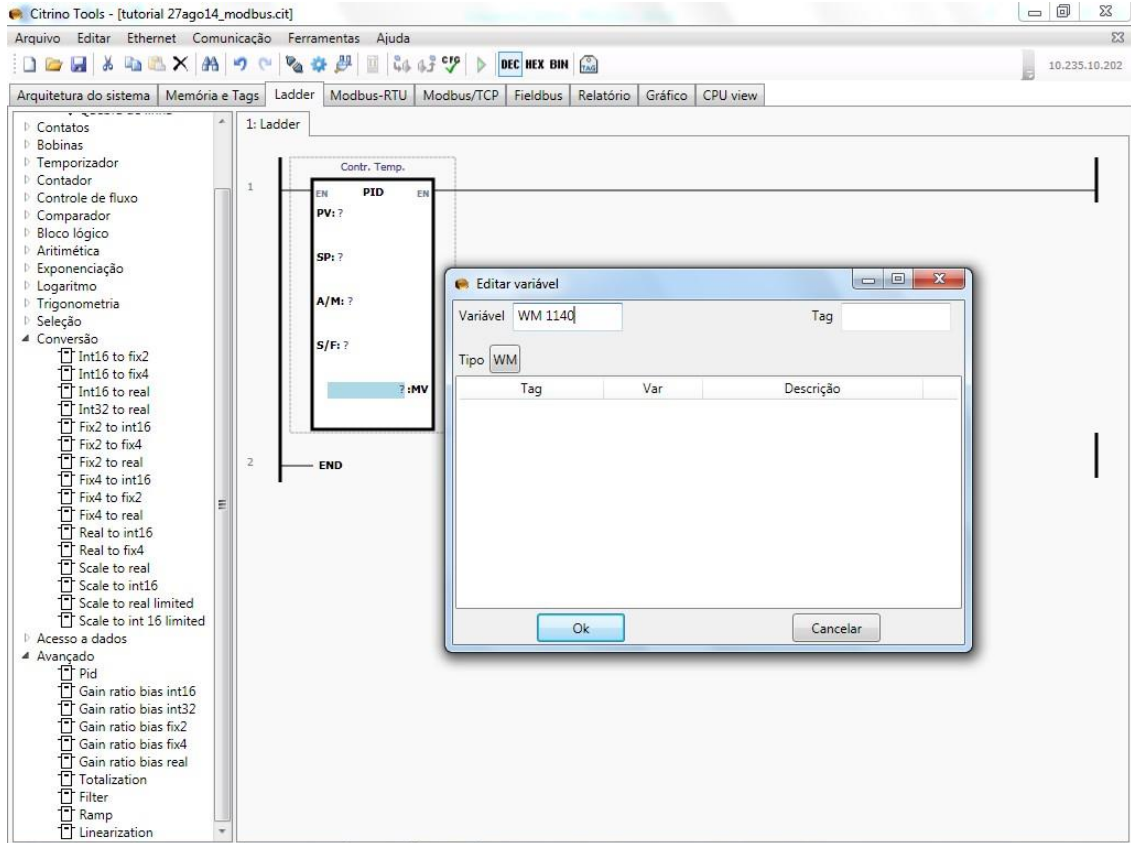
Siga as instruções adiante para inserir um bloco PID na lógica ladder. Dentro da pasta “Avançado”, localize o bloco PID e insira-o na lógica.



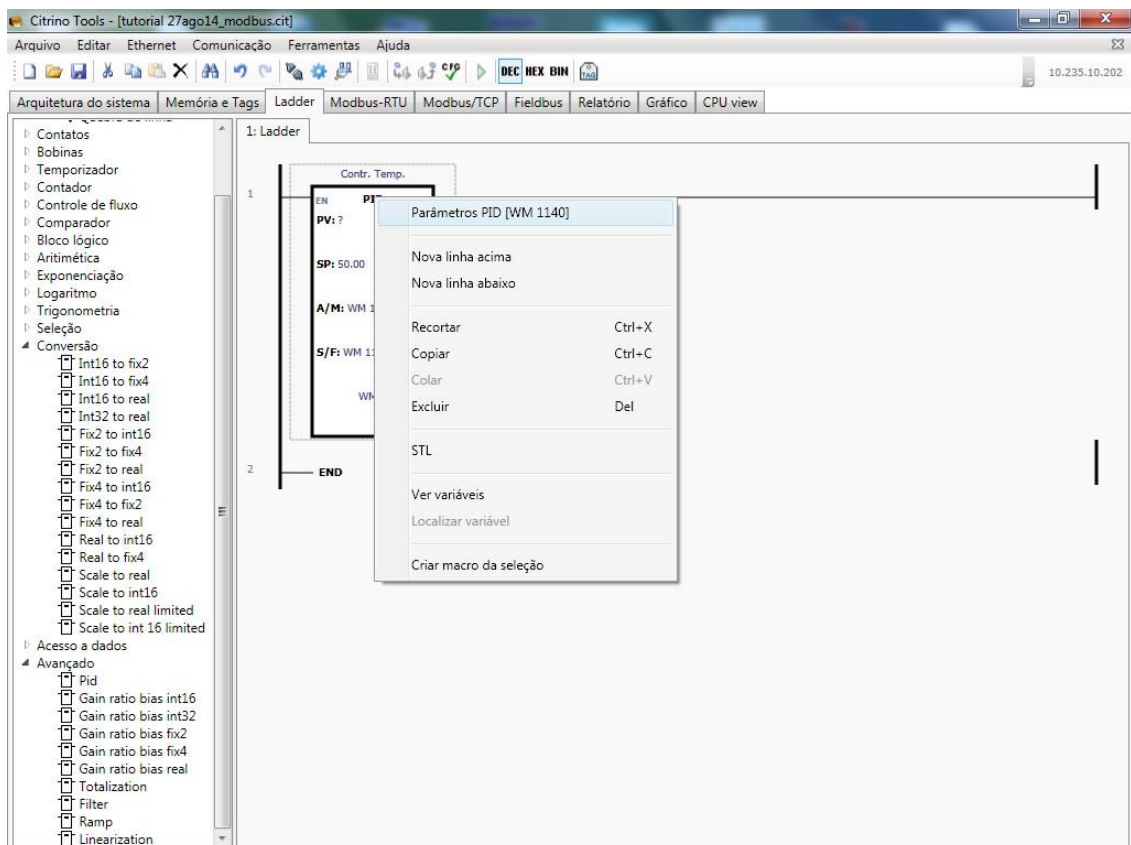
Clique com o botão direito no título do bloco defina um TAG para este controlador.



Clique com o botão direito no “?” relacionado ao MV e defina um endereço para alocar os parâmetros do bloco PID a partir deste endereço.

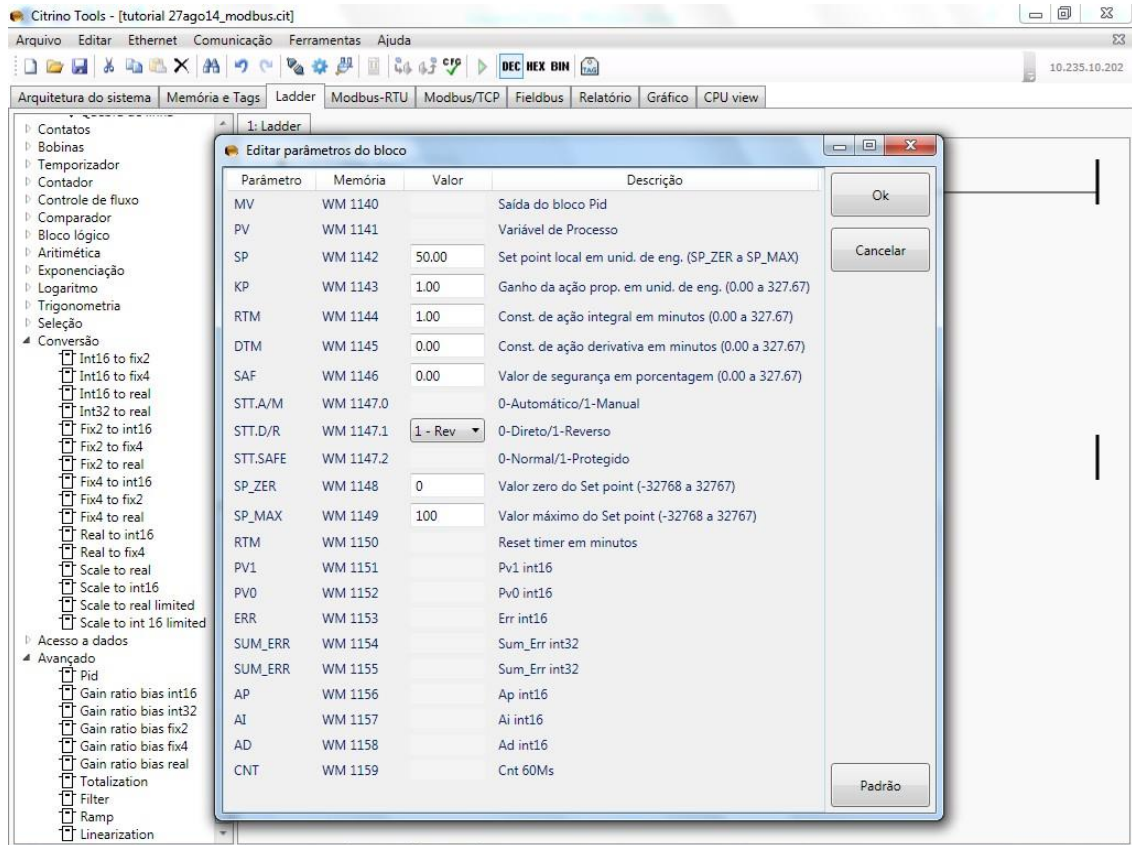


Clique com o botão direito sobre o bloco PID e selecione a opção Parâmetros PID.



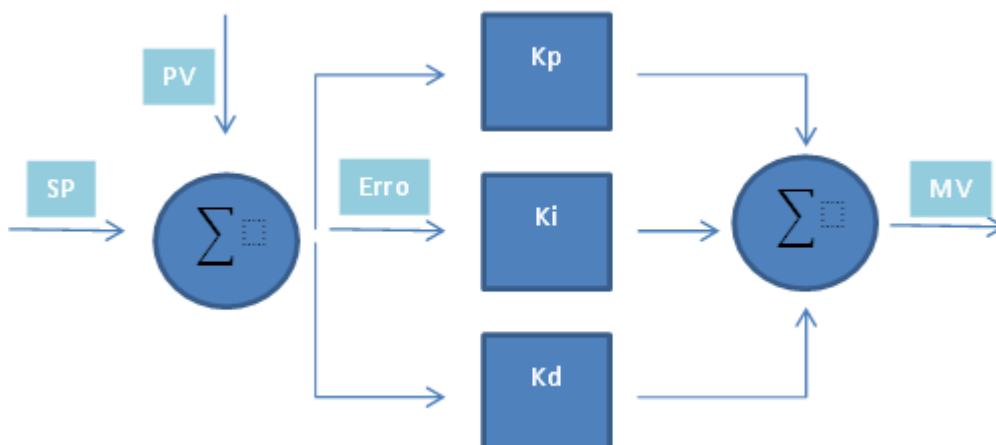
Verifique na janela apresentada a seguir que o conjunto de parâmetros internos do bloco PID foi alocado em 20 posições de memória WM (de 16bits) a partir do endereço WM definido para a MV.

Obs: Deixar sempre habilitado em modo “auto”, safe em “off” na parte externa do bloco PID, e ação de controle (STT.D/R ou WM 1147.1) em modo reverso.



Entre os parâmetros, estão aqueles que definem a sintonia do controlador: termos proporcional (P), integral (I) e derivativo (D), além do Set Point (SP).

Na prática o controlador P.I.D. do Citrino aplica em paralelo os termos P (K_p), I (K_i) e D (K_d) ao sinal de erro, semelhante ao diagrama de blocos a seguir:



As variáveis Kp, Ki e Kd na figura são definidas na memória do Citrino com inteiras de 16bits (WM) denominadas KP, RTM e DTM respectivamente, pois o bloco PID só pode ser posicionado na região de variáveis WM por definição. Como todas elas são de 16 bits (inteiras), os seus valores são limitados pelo software da seguinte maneira:

Termo proporcional KP = -32768 a 32767

Termo integrador RTM = 0 a 32767

Termo derivativo DTM = 0 a 32767

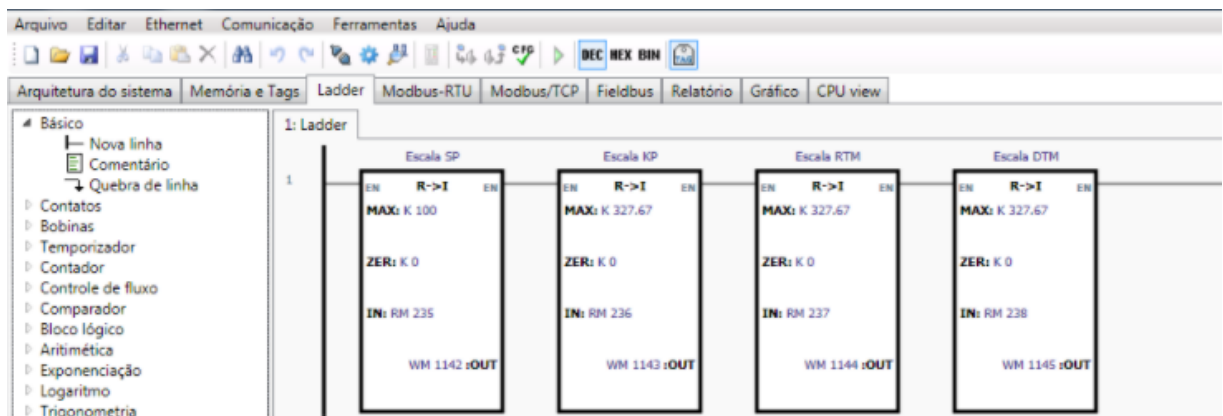
Porém, sabemos que para se fazer boas sintonias, é necessário que estes valores tenham pequenas granularidades e isto só é possível com pequenos acréscimos nestas constantes, como por exemplo alterar um KP de 1.00 para 1.01. Desta maneira, é preciso atentar para o seguinte aspecto de operação do controlador PID: o bloco PID “enxerga” estas variáveis como ponto fixo de duas casas decimais em minutos, ficando da seguinte maneira:

KP = 0 a 327.67

RTM = 0 a 327.67 (minutos)

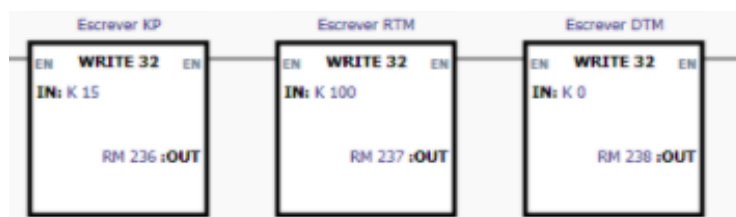
DTM = 0 a 327.67 (minutos)

Desta forma, caso se necessite escrever os valores diretamente em número de ponto flutuante, deve-se realizar uma conversão de tipos entre Real e Inteiro. Isto pode ser realizado como no código a seguir:



Utilize os blocos “Real to Int16” dentro da opção “Conversão” para converter, por exemplo os números reais em RM236, 237 e 238 nos números inteiros em WM1143, 1144, 1145 respectivamente, WMs estes coincidentes com os WMs dos termos de sintonia do PID.

Para escrever os valores de Kp, Ki e Kd, inserimos blocos “write 32”. Em “IN” se deve inserir a entrada do ganho no parâmetro a ser modificado, utilizando o tipo de variável “KR” e na saída “OUT” o mesmo tipo e valor de memória onde irá converter e limitar a faixa de cada parâmetro (como descrito anteriormente), da seguinte maneira:



Set Point e PV

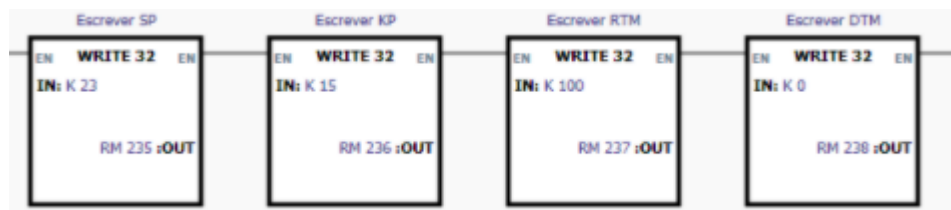
A entrada da variável de processo é medida pelo transmissor volumétrico. Essa entrada é a entrada analógica AI 1, mas é possível substituir por um nome conveniente (o exemplo abaixo mostra como “mede vazao”).

Quanto ao Setpoint, podemos defini-lo em porcentagem do range da tabela do roteiro de vazão (onde 0% corresponde a uma vazão de 315 L/h e 100% a uma vazão de 1045 L/h). Portanto, adicionamos mais um bloco de conversão de escala R -> I, onde ZER é 0, valor mínimo, e MAX é 100.

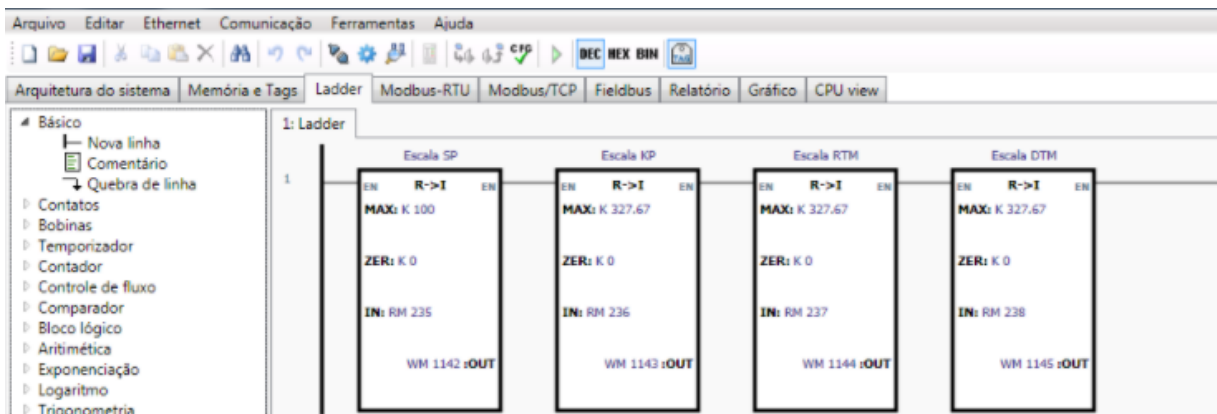
Para definir o SP, defina um valor entre 0 a 100. Isso pode ser feito diretamente no bloco PID, como a imagem abaixo (no caso abaixo, o SP corresponde a 23.0%, que pela tabela, resulta em SP de 515 Litros/hora aproximadamente):



Ou por um bloco “write 32” como foi feito para os ganhos do controlador:

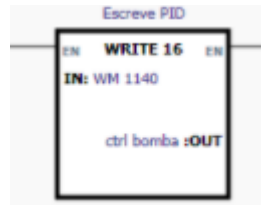


Não esqueça de colocar a escala de 0 a 100, como mostra a figura abaixo (idêntica a figura dos ganhos do controlador):

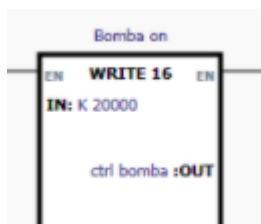


Variável de saída do processo:

A saída AO 1 é o comando para controle da bomba (abaixo, ele está como “ctrl bomba”). Ela é uma saída analógica que escreve um valor de 0 a 32000 na saída do CLP. Esse valor você provavelmente definiu na variável WM 1140. Utilize um bloco “write 16” e escreva esta variável de memória na saída da bomba, como na figura abaixo.



No caso do modo manual, o valor para a saída do CLP (que então vai para a bomba 1) pode ser escrito diretamente, através de um bloco “write 16”.



Observação: Na lógica, só a saída pelo bloco PID OU pelo modo manual podem estar ativas. Nunca as 2 ao mesmo tempo, isso resulta em conflito no valor dado para AO 1.

Observação 2: Para ligar a bomba (tanto no modo automático quanto no modo manual), ative o bit DO 2, que ativa o inversor da bomba 1.

Muito importante:

Sempre que for armazenar uma configuração no CLP, utilizar o menu Comunicação -> Armazenar configuração e dados da memória.

Não utilizar o botão de atalho, pois por ele não se armazenam os dados da memória.

Veja imagem adiante:

