

# Controle de Temperatura Pid Desenvolvido em Labview®

**Renan Pereira da Costa**  
rpcosta86@gmail.com  
UBM

**Jeovane Viana de Queiroz**  
didjobm@hotmail.com  
UBM

**Leonardo de Carvalho Vidal**  
leonardo.carvalho.vidal@hotmail.com  
UBM, AEDB

**Resumo:**São muitos os processos em que é necessário o controle de temperatura. Em uma indústria siderúrgica por exemplo, a temperatura do aço é tratada como uma das mais importantes variáveis de processo, pois afeta diretamente a qualidade final do produto desenvolvido. Existem hoje no mercado inúmeros tipos de controladores para este fim. Este projeto visa estudar uma abordagem diferente da implementação do controle de temperatura através do controle PID (Proporcional-Integral-Derivativo) disponibilizado pelo software de programação LabVIEW®, que possui grandes recursos para análise de dados e resultados. Foi utilizado um controlador lógico programável apenas como módulo de aquisição de dados. Foi elaborado um protótipo em escala reduzida para fins de comprovação das teorias de controle, todo o projeto foi desenvolvido com os recursos disponíveis no Laboratório de Controle e Automação da Universidade de Barra Mansa, localizada no campus Cicuta em Barra Mansa – RJ. O estudo deste projeto provou ser possível implementar um controle eficiente de temperatura com relativa facilidade.

**Palavras Chave:** Controle - Temperatura - PID - LabVIEW - OPC

## 1. INTRODUÇÃO

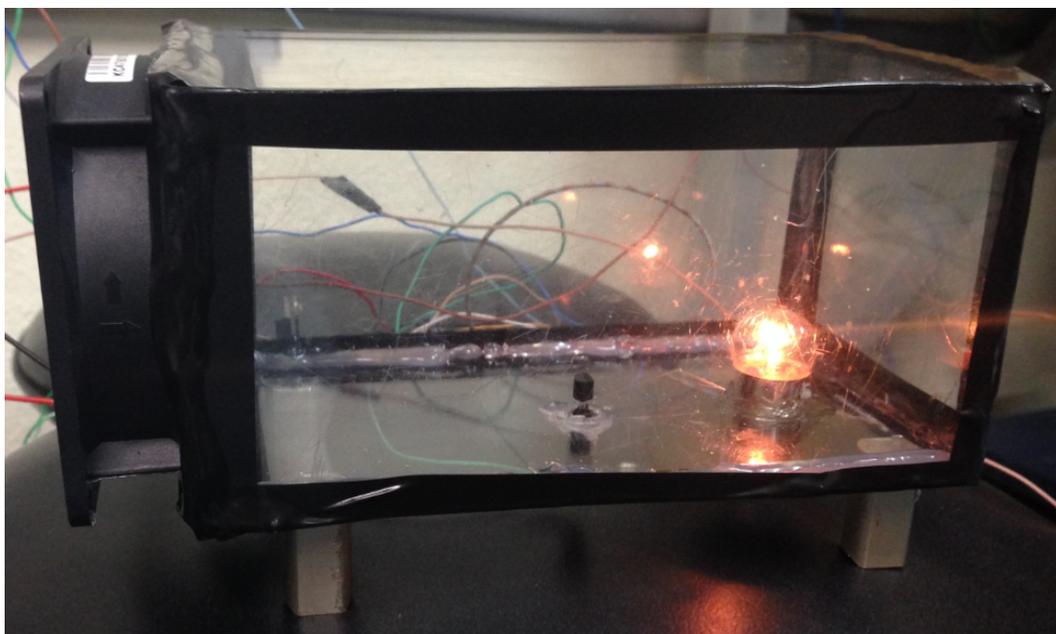
O controle de processo surgiu com a necessidade de se obter melhores desempenhos de equipamentos e sistemas industriais. Consiste na técnica de manter as variáveis de um processo em valores pré-determinados, conhecidos como *setpoint's*, a partir de algoritmos relacionados às variáveis que são fornecidas pelos sensores do processo em um controle de malha fechada. Um dos *softwares* que contém amplas ferramentas para técnicas de controle é o LabVIEW®.

Este projeto tem a finalidade de oferecer uma opção de controle de temperatura em um ambiente fechado implementando uma opção na área de automação de processos industriais, pois utiliza um controlador proporcional, integral e derivativo (PID) desenvolvido na plataforma gráfica LabVIEW® e CLP Allan Bradley® como interface de aquisição de dados, para se obter um controle de temperatura efetivo através de um sistema lógico desenvolvido em um computador e realimentação (*Feedback*).

## 2. DESENVOLVIMENTO

### 2.1. PROTÓTIPO

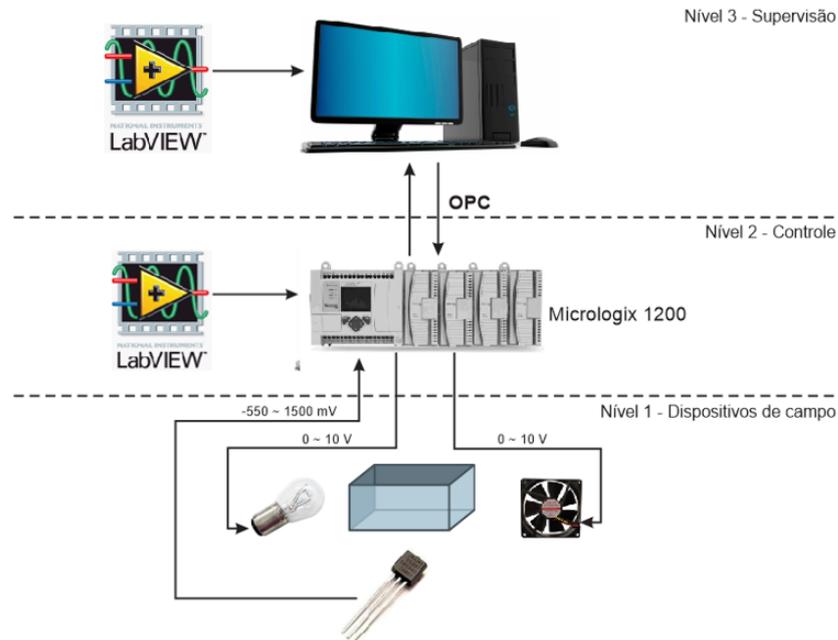
Foi desenvolvido um protótipo em escala reduzida a fim de representar em laboratório o controle de temperatura PID, conforme mostrado na Figura 1. O protótipo foi montado em uma caixa de acrílico para simular um ambiente fechado, minimizando as interferências externas e perturbações ao sistema, para um melhor desempenho da malha de controle. Foram utilizados como atuadores uma lâmpada incandescente como fonte de calor e um ventilador (*cooler*) como fonte de resfriamento. Como elemento sensor que faz o *feedback* do sistema de controle, foi utilizado o sensor de temperatura LM35. O controle foi implementado digitalmente pelo software de instrumentação LabVIEW® e para a interface entre sensores, atuadores e o controle foi utilizado o CLP Allen Bradley® Micrologix® 1200.



**Figura 1:** Protótipo do sistema em escala reduzida. **Fonte:** Própria

## 2.2. ARQUITETURA DO SISTEMA

Conforme critérios estabelecidos pela pirâmide de automação, onde é definida uma hierarquia para os dispositivos de automação, a arquitetura do sistema encontra-se disposta conforme mostrado na Figura 2.



**Figura 2:** Hierarquia do sistema de controle. **Fonte:** Própria

Para a implementação do protótipo, somente foram necessários os três primeiros níveis da pirâmide de automação.

- Nível 1 (Dispositivos de campo) – É o nível onde são feitas as ligações físicas da rede. Neste nível encontram-se os sensores e atuadores do protótipo, que neste caso são: O sensor de temperatura LM35DZ, lâmpada incandescente e o ventilador. O objetivo deste nível é transmitir informações do campo para a camada de controle e vice versa.
- Nível 2 (Controle) – O segundo nível compreende os equipamentos que fazem o controle do nível 1. Este nível possui controladores lógico programáveis (CLP), sistema digital de controle distribuído (SDCD) e relés. O objetivo deste nível é de fato controlar os dispositivos de campo encontrados no nível 1, lendo os sinais dos sensores e enviando sinais de referência para os atuadores.
- Nível 3 (Supervisão) – Neste nível é permitido a supervisão e otimização do processo. Neste nível se encontra o computador e o *software LabVIEW®*. O objetivo deste nível é receber todas as informações inerentes ao processo, analisar, simular e processar estas informações para enviar à camada de controle. É neste nível onde é feita a interface entre homem e máquina (IHM).

### 2.3.3. AQUISIÇÃO DE DADOS

A aquisição de dados do protótipo foi feita através do CLP *Allen-Bradley® Micrologix® 1200*, que possui um módulo de entradas e saídas (*I/O*) analógicas, modelo 1762-IF20F2, este módulo permite ler o valor da temperatura enviado pelo termopar, na entrada analógica I:3.0 através de um sinal de corrente de 4-20 mA. Desta mesma forma é possível

manipular as saídas analógicas O:3.0 e O:3.1, onde estão conectados lâmpada e ventilador, respectivamente, através de um sinal de tensão de 0-10V. Estas entradas e saídas são lidas e escritas pelo LabVIEW® através do *software* RS Linx pelo padrão OPC, utilizando-se de pontos flutuantes disponibilizados pelo *software* RS Logix 500, que farão interface direta com as entradas e saídas do CLP.

### 2.3.3.1. CIRCUITO AMPLIFICADOR DE POTÊNCIA

Como o CLP não fornecia corrente suficiente para alimentar a lâmpada e o cooler, foi necessário a implementação de dois amplificadores de potência, um para cada atuador. Foi necessário também a montagem de um circuito RC (Resistivo-Capacitivo) para fazer a filtragem e casamento de impedância entre a saída do sensor LM35DZ e a entrada do módulo de entradas e saídas analógicas do CLP.

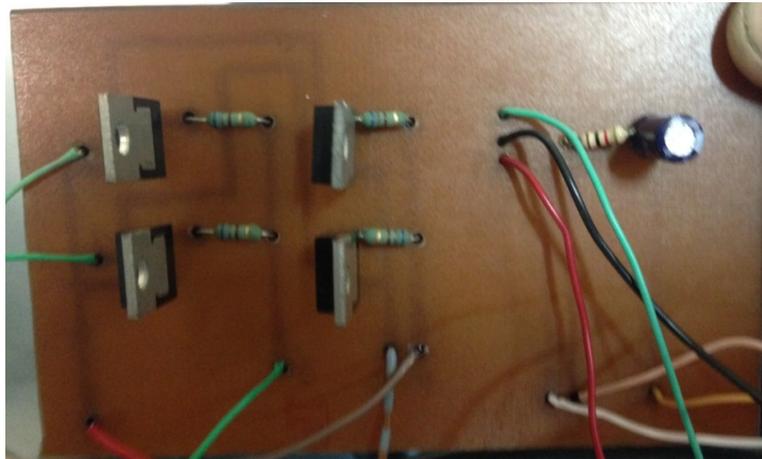


Figura 3: Circuito amplificador de potência. Fonte: Própria

Pode-se observar o diagrama eletrônico dos amplificadores de potência na Figura 4.

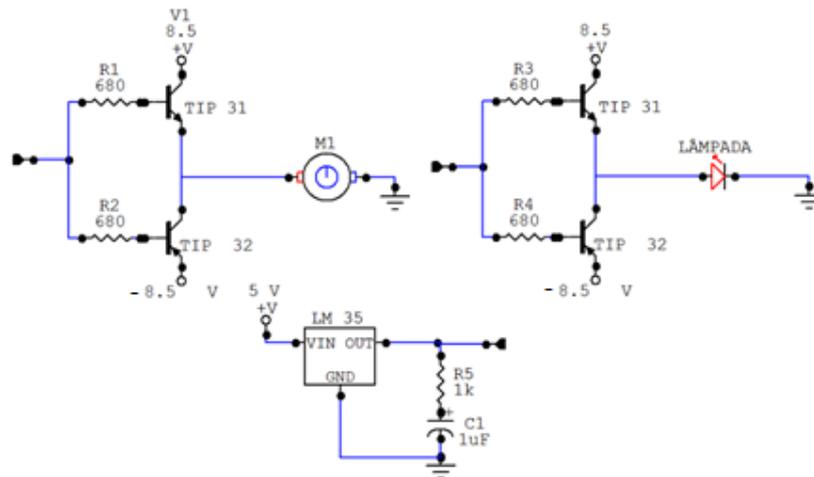


Figure 4: Diagrama eletrônico dos circuitos amplificadores de potência. Fonte: Própria

### 2.3.3.2. PROGRAMAÇÃO LADDER

Como o intuito do CLP Allen-Bradley® Micrologix® 1200 era apenas o de aquisição de dados, não foi necessária uma programação LADDER complexa para o seu funcionamento, conforme podem observar na Figura 5.

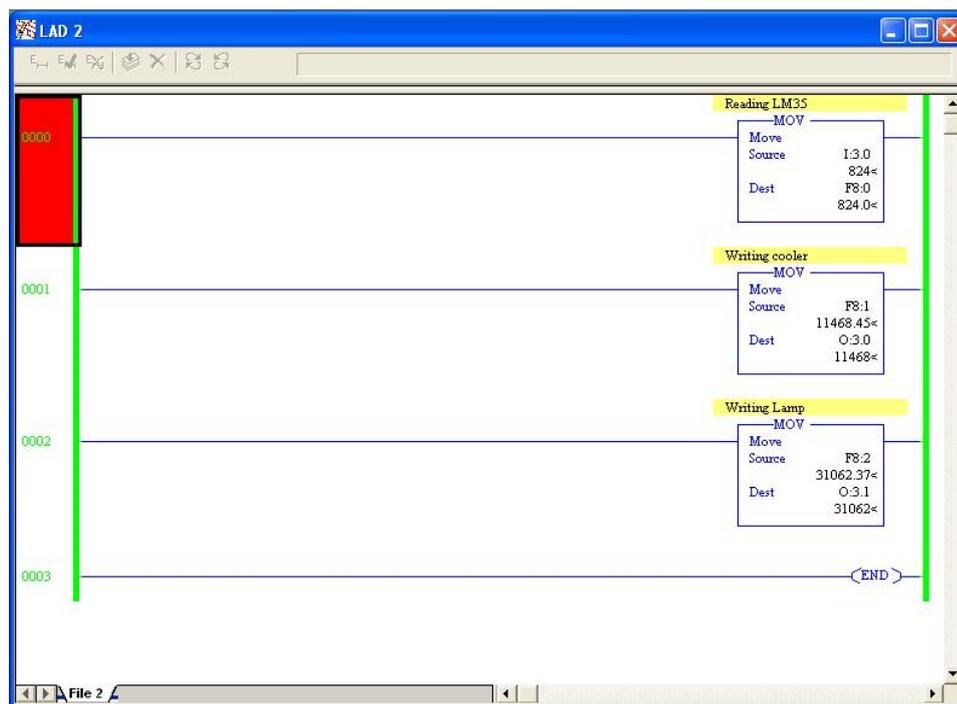


Figure 5: Programação LADDER. Fonte: Rockwell Software RSLogix® 500

Conforme mostrado na Figura 5, pode-se observar três blocos MOV, que fazem a movimentação do valor de uma variável para outra variável, alimentados diretamente, sem a necessidade de nenhuma entrada. A única função desta programação é movimentar as variáveis processadas pelo LabVIEW®, via OPC, para suas portas de saída analógica, que se encontram no cartão 1762-IF2OF2, bem como movimentar a leitura fornecida pela porta analógica de entrada, deste mesmo cartão analógico, via OPC, para o LabVIEW®.

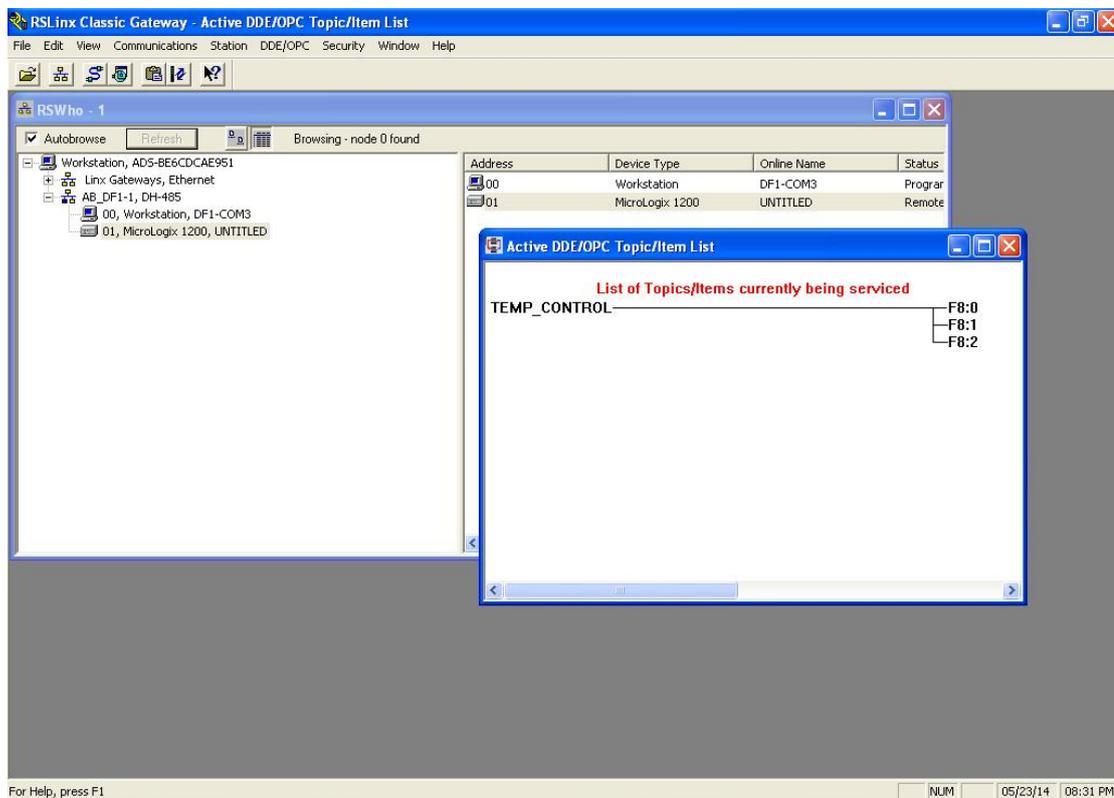
A linha 0000 nos mostra a leitura da porta analógica I:3.0, que corresponde a primeira entrada do cartão analógico 1762-IF2OF2 onde está conectado a saída do LM35DZ, e a movimentação desta leitura para a variável de tipo ponto flutuante (*Float*) F8:0. Na linha 0001 e 0002 é possível observar a leitura das variáveis *float* F8:1 e F:82 e sua movimentação para as saídas analógicas O:3.0 e O:3.1 que correspondem aos atuadores, cooler e lâmpada, respectivamente.

#### 2.3.4. COMUNICAÇÃO OPC

OLE for Process Control, onde OLE é uma sigla para “*Object Linking and Embedding*”, ou simplesmente OPC, pode ser definida como a tecnologia que permite diferentes tipos de aplicações escritas em diferentes tipos de linguagens de programação executados em diferentes plataformas, compartilhem informações entre chão de fábrica, planta e softwares administrativos, como por exemplo, o Microsoft Excel.

Isso se faz necessário pois em ambiente industrial e administrativo existem diversos tipos de *hardwares* e *softwares* diferentes que precisam “conversar” entre si para que o fluxo de dados, análise, tomada de decisão e informações administrativas estejam disponíveis para os diferentes tipos de profissionais que uma grande indústria contém.

A comunicação feita via OPC, entre o LabVIEW® e o *software* RSLogix® 500, que manipula o CLP *Allen Bradley* Micrologix® 1200, foi feita pelo *software*, também da Rockwell, RSLinx® Classic.



**Figure 6:** Active DDE/OPC. **Fonte:** Rockwell Software RSLinx® Classic

Conforme mostrado na Figura 6, pode-se observar, na janela “RSWho – 1” a comunicação ativa entre o CLP Micrologix® 1200 e a estação de trabalho, computador, onde se encontram os *softwares* utilizados neste trabalho, aberta pela comunicação serial DH-485, nomeada AB\_DF1-1.

Ainda observando a Figura 44, na janela “Active DDE/OPC Topic/Item List”, pode-se notar a disponibilização, via OPC, do grupo “TEMP\_CONTROL” criado e das variáveis *float* utilizadas para a comunicação entre *LabVIEW*® e *RSLogix*®: F8:0, F8:1 e F8:2.

### 2.3.5. PROGRAMAÇÃO LABVIEW

O nome *LabVIEW*® é um acrônimo para *Laboratory Virtual Instrument Engineering Workbench* ou, numa tradução direta, Bancada de Engenharia de Instrumentação Virtual e é uma linguagem de programação gráfica originária da *National Instruments*. A primeira versão do *LabVIEW*® surgiu em 1986 para o Macintosh, mas com o crescimento da instrumentação e automação industrial e comercial a *National Instruments* abrangeu o *LabVIEW*® para outros sistemas operacionais como *Windows*, *Linux* e *Solaris*.

A automação e realizações de medições são os principais campos de aplicação do *LabVIEW*®, que oferece inúmeras vantagens para a manipulação e aquisição de dados por sua programação que é feita de acordo com o modelo de fluxo de dados.

O programa desenvolvido em *LabVIEW*® que faz o controle PID da temperatura digitalmente está representado na Figura 7.

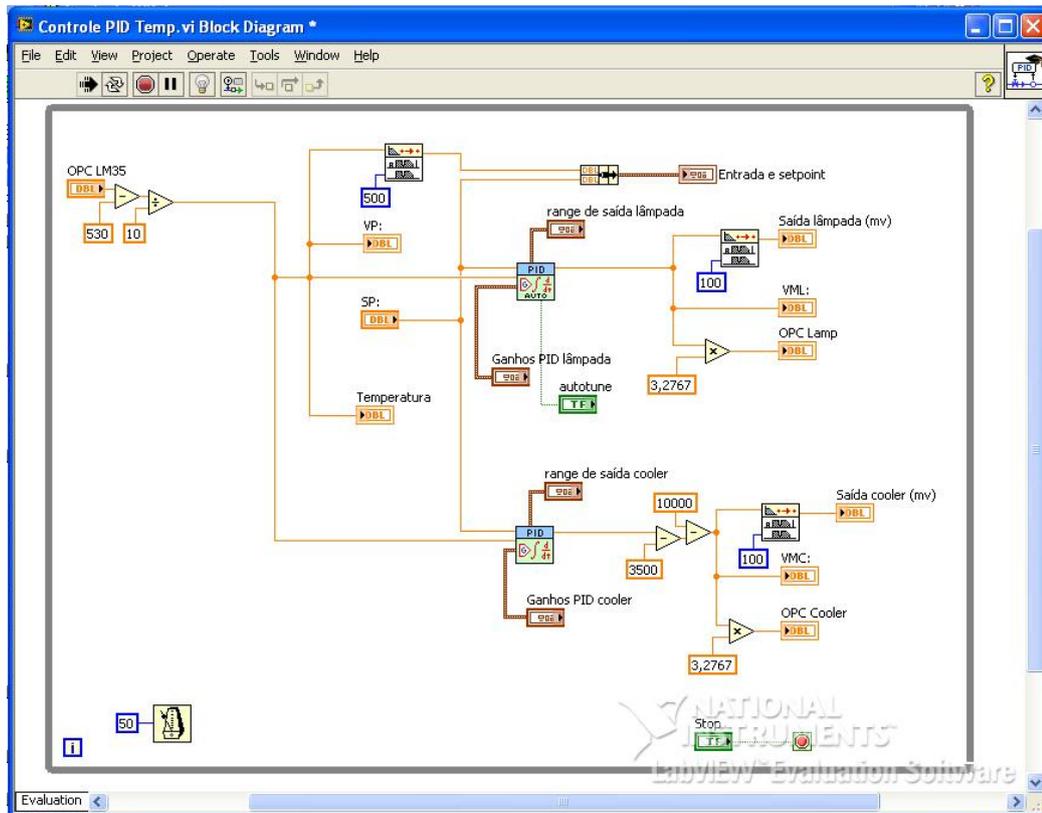


Figure 7: Programação LabVIEW® do controle PID. Fonte: LabVIEW® 2011

Como podem observar, foram utilizados dois sub-IVs que fazem o controle PID digital, previamente programados e disponíveis na biblioteca do *LabVIEW*®. Nesse sub-IV, chamado de *NI\_PID*, é necessário fornecer algumas informações do processo, como valor do *setpoint*, variável de processo, os ganhos do controlador PID, dados por  $K_p$ ,  $T_i$  e  $T_d$ , e o *range* da saída que será gerada pelo controlador.

A programação em *LabVIEW*® pode ser dividida em blocos, para melhor entendimento. O primeiro bloco, representado pela Figura 8, é o bloco que alimenta o programa, o controle numérico “OPC LM35” faz a interface direta com o *RSLinx*® OPC Server, e recebe o valor da variável *float* “F8:0” do *RSLogix*®. É possível notar ainda neste bloco os comandos de subtração e divisão que fazem a conversão do valor em mV para o valor em graus celcius, conforme especificação do LM35DZ.

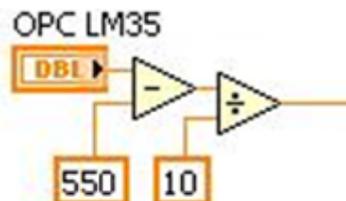


Figure 8: Bloco 1 de programação LabVIEW®. Fonte: LabVIEW®

Para o funcionamento do programa, são necessárias duas informações, o valor medido da temperatura e o *setpoint*, que será fornecido pelo usuário no controle numérico “SP:”. O controle numérico “SP:” pode ser observado na Figura 7.

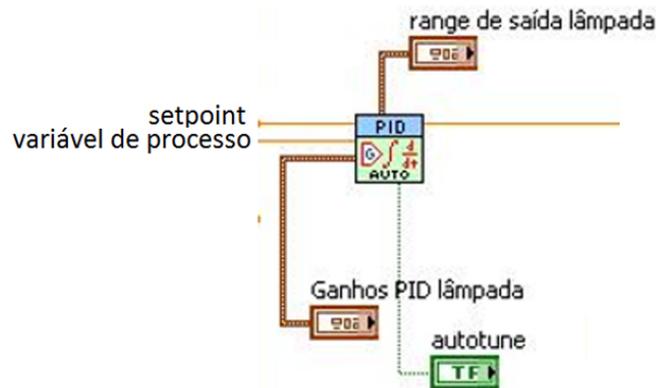


Figure 9: Parâmetros de alimentação sub-IV PID. Fonte: LabVIEW® 2011

A partir destas duas informações é possível plotar o gráfico no indicador “Entrada e *setpoint*” e alimentar os sub-IV’s PID. Para o funcionamento do sub-IV PID é necessário alimentá-lo com uma série de parâmetros. Na Figura 9 pode-se observar estes parâmetros.

É possível observar, na Figura 47, que mostra o sub-IV PID da lâmpada, as quatro entradas de alimentação do sub-IV necessárias para o funcionamento do protótipo e o controle booleano “autotune”, onde é feita a opção pelo funcionamento do *autotuning* fornecido pelo sub-IV PID. Os parâmetros de alimentação do sub-IV PID são: *setpoint*, variável de processo, range de saída e ganhos PID.

O *setpoint* é um controle numérico manipulado pelo usuário, a variável de processo é o valor da temperatura lido no LM35DZ, o range de saída especifica o range de trabalho do controlador PID e os ganhos PID são os valores de Kp, Ti e Td.

A Figura 10, mostra a alimentação do indicador gráfico “Entrada e *setpoint*”. É possível observar o filtro de média atuando na variável de processo para amenizar o ruído apresentado pelo módulo de entrada analógica do CLP.

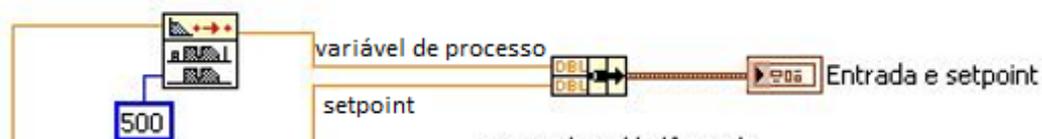


Figure 10: Alimentação do gráfico "Entrada e setpoint". Fonte: LabVIEW® 2011

A Figura 11 mostra a saída do sub-IV PID, que alimenta alguns indicadores. O indicador “Saída lâmpada (mv)” é o gráfico em que mostra a ação da atuador, neste caso a lâmpada e também foi necessário a aplicação do filtro de média para amenizar os efeitos do ruído. O indicador “VML:” é o *sidebar* que mostra a variável manipulada. Pode-se observar que ambos os indicadores são alimentados pela mesma variável e mostram o mesmo valor, mas de forma diferente, para uma melhor análise.

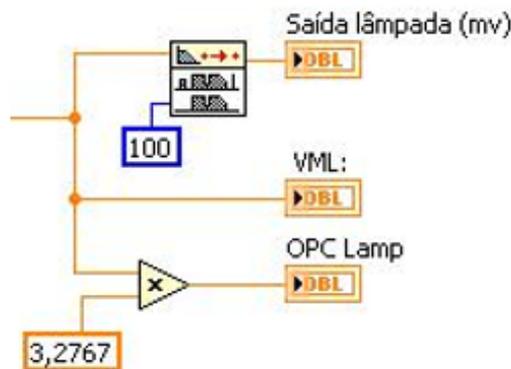


Figure 11: Saída do controlador PID. Fonte: LabVIEW® 2011

Ainda na Figura 11, é possível observar o indicador “OPC Lamp”, este indicador faz a interface direta com o RSLinx® OPC Server e este valor será enviado à uma das saídas analógicas do CLP, neste caso, a saída O:3.1. Para este indicador foi necessário a multiplicação da variável manipulada pela constante “3,2767”, isso se deve pelo fato de que a saída analógica do CLP trabalha com um range de 0 a 32.767, onde o 32.767 corresponde a 10Vcc. Como o range de saída utilizado para este protótipo está operando em mV, esta multiplicação foi adotada para a utilização dos 10Vcc fornecidos pela saída analógica do CLP.

Toda a programação foi feita dentro de um *loop while*, para que o funcionamento da malha seja constante. O tempo entre interações deste *loop* é dado pela função “Wait Until Next”, que faz com que o loop espere um determinado tempo até a próxima interação. O funcionamento da função de tempo pode ser observado na Figura 12. O tempo entre interações adotado para o funcionamento do protótipo foi de 50ms.

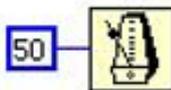


Figure 12: Aplicação da função Wait Until Next. Fonte: LabVIEW® 2011

A programação feita no diagrama de blocos gera automaticamente uma parte gráfica no painel frontal, onde estão dispostos todos os controles e indicadores apresentados pelo diagrama de blocos. O painel frontal funciona como um sistema supervisor, onde é possível observar a temperatura no interior da câmara, alterar o valor do *setpoint*, gerar gráficos e analisar o comportamento do controle, fazendo então o papel de interface entre homem e máquina (IHM).

Na Figura 13 é possível observar, na IHM criada para o protótipo, a temperatura no interior da câmara, o *setpoint*, a tensão enviada ao cooler e a lâmpada, os ganhos dos controladores, o *range* de saída dos controladores e a variação das variáveis manipuladas.

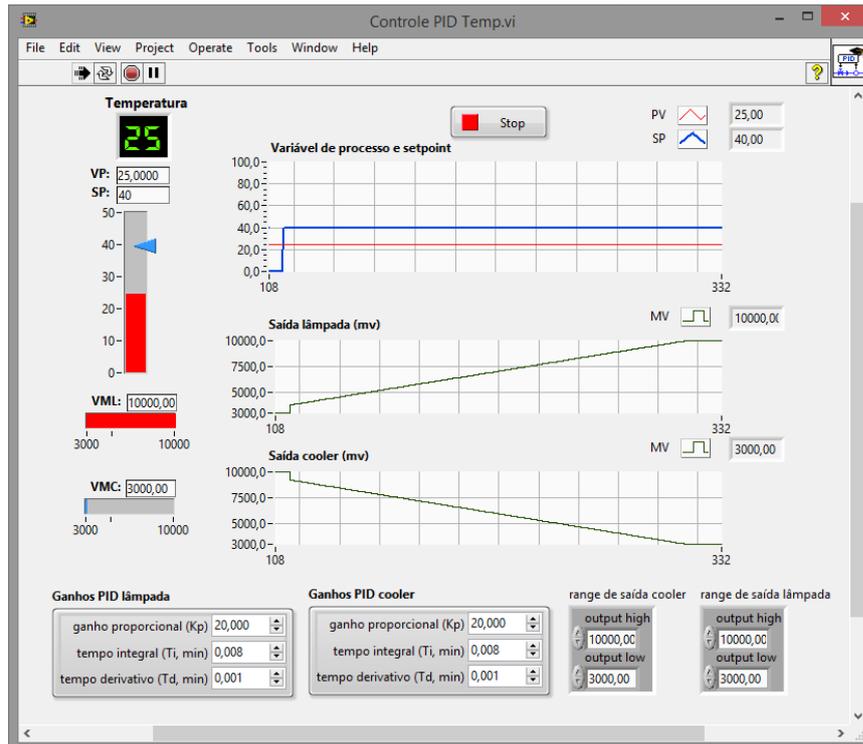


Figure 13: IHM criada para controle e análise das variáveis do processo. Fonte: LabVIEW® 2011

### 3. DESEMPENHO

Os ranges de saída adotados para os atuadores foi de 3.000mV a 10.000mV, isso se deve pelo fato da lâmpada não oferecer um mínimo de dissipação de calor para tensão abaixo de 3Vcc e o cooler não partir com a tensão também abaixo 3Vcc por não ter torque suficiente para vencer a inércia do motor.

O sensor LM35DZ não pôde ser utilizado com precisão, pois o cartão de entrada analógica do CLP utilizado tem a resolução de 8mV, como o LM35DZ fornece uma variação de 10mV/°C, a temperatura oscilava cerca de 1°C, gerando um atraso na ação dos atuadores e adicionando um alto ruído na curva da variável de processo e na variável manipulada. O ruído pode ser observado na Figura 14.

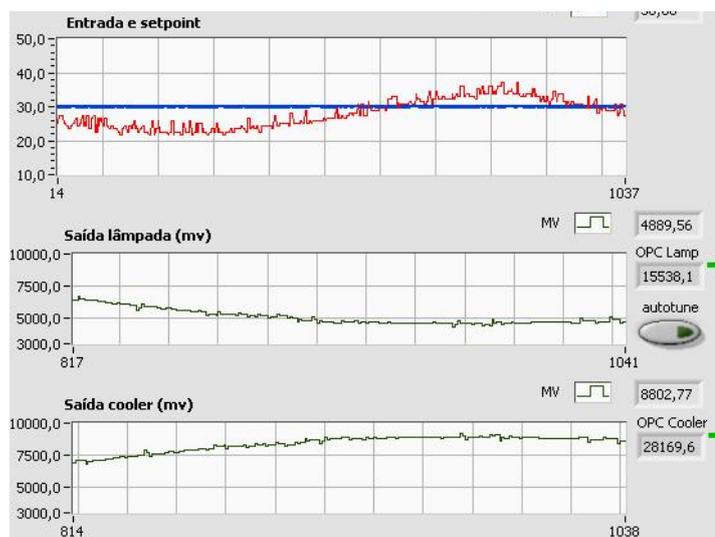


Figure 14: Ruído gerado pela resolução de leitura do cartão de entrada analógica. Fonte: LabVIEW® 2011

Para amenizar os efeitos do ruído e melhorar a análise do controle, no gráfico de variável de processo e *set point* foi adicionado um filtro de média à programação em LabVIEW®. É possível observar a ação do filtro de média na Figura 15.

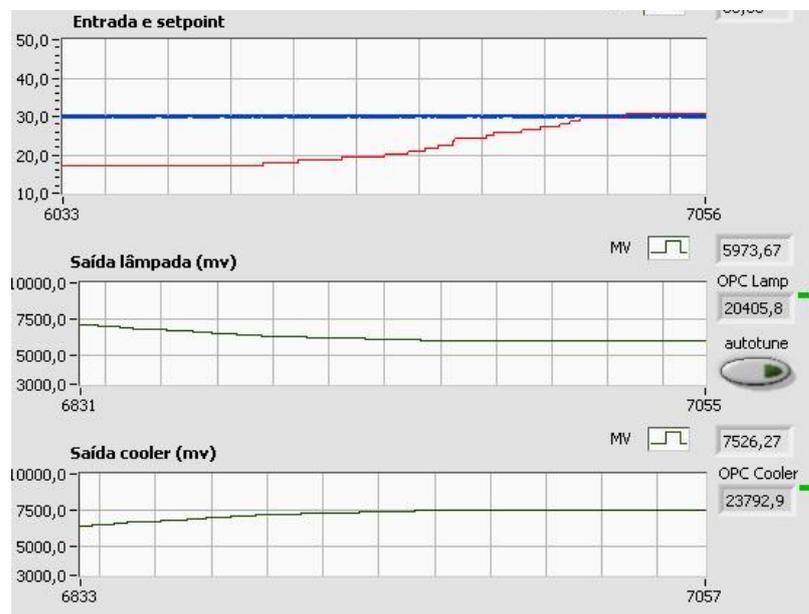


Figure 15: Suavização do ruído aplicando filtro de média. Fonte: LabVIEW® 2011

Utilizando os valores de ganho proporcional ( $K_p$ ), tempo derivativo ( $T_d$ ) e tempo integral ( $T_i$ ) padrões do sub-IV PID, que são 20, 0,001 e 0,008, respectivamente, o controle se manteve estável em regime estacionário, porém com um erro considerado relativamente alto. Com estes valores os atuadores entravam em controle liga/desliga, quando a variável de processo ultrapassava o valor de *setpoint* o controle desligava a lâmpada e ligava o cooler, quando a variável de processo era menor que o *setpoint* o cooler era desligado e a lâmpada ligada, sempre nos seus valores máximos de potência e mantendo o mesmo ciclo, sempre com um erro de  $\pm 2^\circ\text{C}$ .

A fim de se obter novos ganhos para o sistema e melhorar o controle, foi utilizado a função *Autotune* do sub-IV PID, que gera valores para  $K_p$ ,  $T_i$  e  $T_d$  automaticamente de acordo com o desvio do sistema. Os valores obtidos com a função *Autotune* foram:  $K_p = 405,365368$ ;  $T_i = 0,01575$  e  $T_d = 0,003937$ . Conforme mostrado na Figura 16.

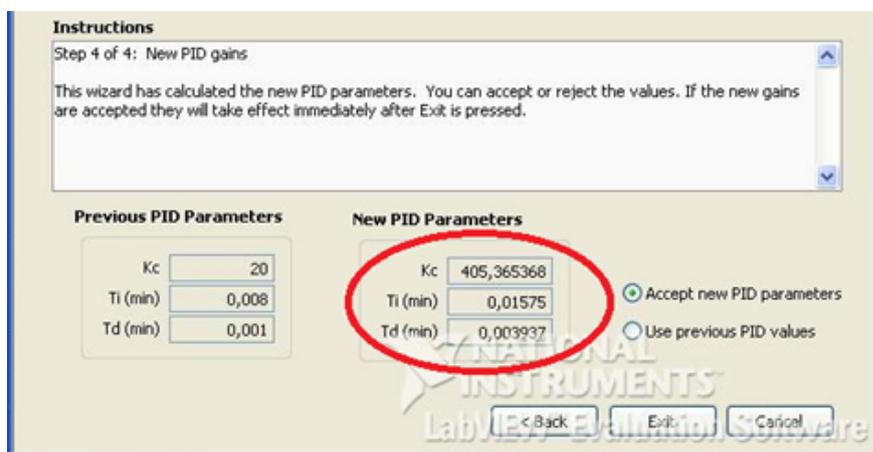


Figure 16: Função Autotune do sub-IV PID. Fonte: LabVIEW® 2011

Utilizando os valores fornecidos pelo *autotune* no controle, foi possível observar que o sistema obteve uma resposta mais rápida dos atuadores, mas com um alto *overshoot* e um aumento considerável no tempo de acomodação, beirando a instabilidade do controlador. Conforme mostrado na Figura 17. Dessa forma pode-se concluir que o efeito da função *autotuning* disponibilizado pelo sub-IV PID, não foi considerado satisfatório neste controle e por isso foi descartado.

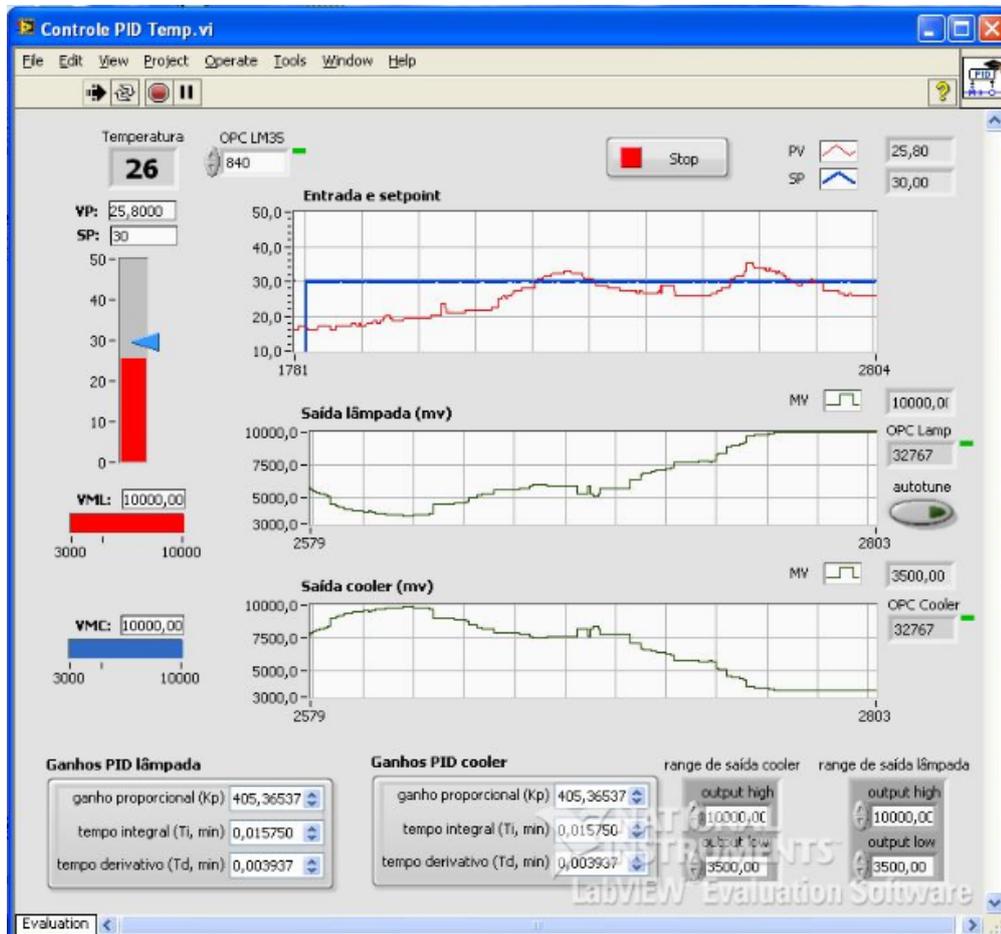


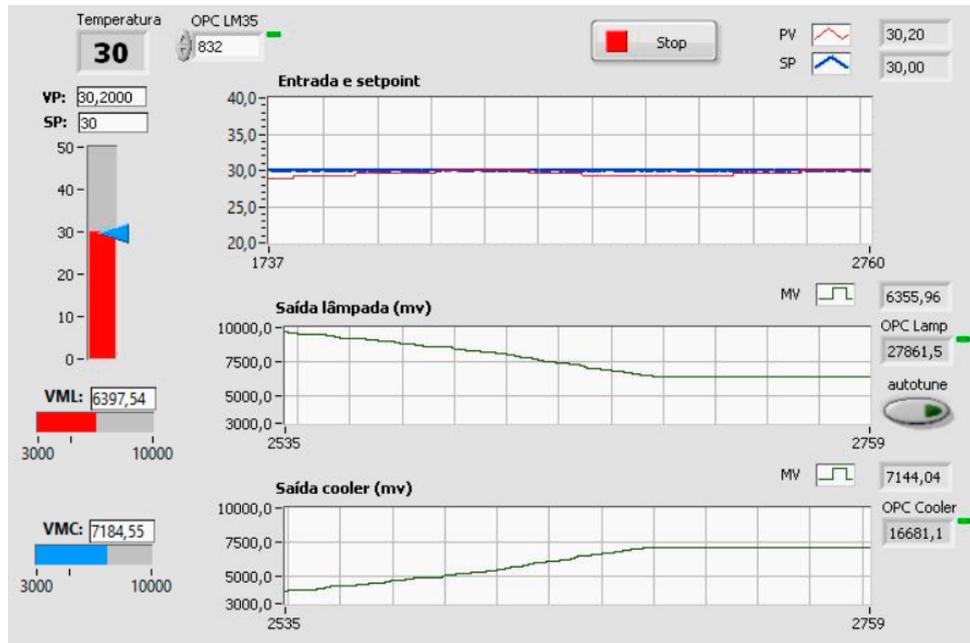
Figure 17: Resposta do sistema para valores fornecidos pelo autotune. Fonte: LabVIEW® 2011

Como os valores dos ganhos fornecidos pelo *autotune* levaram a resposta do sistema ao limite da estabilidade, eles foram descartados. Os valores de  $K_p$ ,  $T_i$  e  $T_d$  padrão, fornecidos pelo sub-VI PID do LabVIEW®, obtiveram um desempenho próximo do que foi estabelecido previamente como razoável, dessa forma, era necessário apenas aumentar a velocidade de resposta do sistema e diminuir o erro em regime estacionário. Empiricamente foram aumentados o ganho proporcional  $K_p$ , que teoricamente deveria aumentar o tempo de resposta do sistema, e o tempo integral  $T_i$ , que tem a função de diminuir o desvio em regime estacionário.

Desta forma, os valores encontrados dos parâmetros PID foram:  $K_p = 100$ ;  $T_i = 0,01$  e  $T_d = 0,001$ . Estes foram os parâmetros obtidos para o PID da lâmpada. Como o cooler consegue esfriar o ambiente mais rápido do que a lâmpada consegue esquentar, foi necessário diminuir o ganho proporcional em relação ao ganho da lâmpada, pois sua ação é naturalmente mais rápida. Sendo assim, os parâmetros do PID do cooler foram:  $K_p = 70$ ;  $T_i = 0,01$  e  $T_d = 0,001$ .

Com os novos ganhos dos controladores PID, o erro do sistema baixou para cerca de 1°C em regime estacionário, que é considerado aceitável, levando em consideração a resolução

de leitura do sensor de temperatura. O desvio em regime estacionário pode ser observado na Figura 18.



**Figure 18:** Desvio em regime estacionário. **Fonte:** LabVIEW® 2011

#### 4. CONCLUSÃO

Neste trabalho foi apresentado o estudo e aplicação do controle de temperatura utilizando controlador PID desenvolvido em plataforma LabVIEW®.

O LabVIEW® se mostrou como uma ferramenta prática e eficiente, onde é possível implementar inúmeras técnicas de controle com simplicidade. O painel frontal do LabVIEW® fornece uma imensa gama de opções para análises de comportamento do sistema.

O controle da temperatura no protótipo se mostrou satisfatório, porém como os atuadores e módulo de aquisição de dados utilizados neste trabalho não proporcionaram a precisão necessária para um melhor controle, pode-se concluir que o desempenho deverá ser melhor com a utilização de atuadores de maior precisão.

O protótipo foi desenvolvido em escala reduzida, e pôde comprovar as teorias de sistemas de controle, este mesmo protótipo pode ser utilizado em proporções reais, como a utilização em indústrias, escritórios, ambientes residenciais ou qualquer outro processo que necessite de um controle de temperatura. Será necessário apenas um estudo de viabilidade para cada caso, pois quanto maior a precisão necessária maior deverá ser o investimento aplicado.

A comunicação via OPC se mostrou de fácil implementação e eficiente, pois fez as transições entre plataformas em tempo real, não atrasando as respostas dos atuadores.

Este trabalho poderá ser utilizado como mais uma ferramenta didática para os estudantes de engenharia devido ao fato de que os componentes e ferramentas utilizados para sua elaboração, agregada a fácil disposição e visualização dos resultados obtidos estimulam e fornecem de maneira prática as teorias de sistemas de controle.

Como oportunidade de melhoria para este trabalho observamos que pode ser elaborado o modelamento da malha de controle deste sistema, afim de ser obter os melhores valores para os parâmetros do controlador PID.

## 6. REFERÊNCIAS

**MARIANO, F.;** **CÉSAR, O.** Escola Técnica aberta do Brasil - Controle Automático de Processos.

**FERNANDO, LUÍS;** **FELIPE, JOSÉ;** Sistemas de Controle Digital Pontífica - Universidade Católica do Rio Grande do Sul Departamento de Engenharia Elétrica.

**NISE, N. S.** Engenharia de sistemas de controle, Rio de Janeiro, LTC, 2002 **OGATA, K.** Engenharia do controle moderno, 3ª Ed, Rio de Janeiro: LTC, 1998

**NATIONAL INSTRUMENTS CORPORATION;** Manual de treinamento do LabVIEW® Básico I, Curso do software - versão 6.0; Edição de outubro de 2000.

**AMADEU, S.** Artigo Sobre LabVIEW®; Cerne Tecnologia;

**BELL, M.** Alpret Control Specialists; OLE for Process Control; Electricity + Control; June 1998.

**FONSECA, M.** Comunicação OPC – Uma abordagem prática; VI Seminário de Automação de Processos, Associação Brasileira de Metalurgia e Materiais, 2002 – Vitória – ES, Brasil.

**VILAS, R.** PADRÃO OPC: Uma Alternativa de Substituição dos Drivers Proprietários para Acessar Dados de CLPs; Universidade FUMEC; Belo Horizonte 2004.

**TORRICO, C.** Controle Clássico Projeto de um controlador PID; Universidade do Estado de Santa Catarina.