

UTILIZAÇÃO DO MATLAB® NO ENSINO DA TECNOLOGIA OPC APLICADA A CONTROLE DE PROCESSOS

CARLOS R. M. DUARTE, LUIZ C. FIGUEIREDO, MARCELO V. CORRÊA

Laboratório de Controle e Instrumentação, Departamento de Engenharia Elétrica, Centro Universitário do Leste de Minas Gerais – Unileste MG. Av. Tancredo Neves 3500, 35170-056 Bairro Universitário Coronel Fabriciano, MG, BRASIL

E-mails: duarte_eng@yahoo.com.br, figueiredo@unilestemg.br, mcorrea@unilestemg.br

Abstract— This paper presents the use of the Matlab® 7.0 software in the teaching of the OPC technology applied in process control, more specifically, in a flow loop feedback control. It highlights the importance of learning technologies used in the industry for the future engineers' formation, approaching theory and practice.

Keywords— Matlab®, OPC, Process Control.

Resumo— Este artigo apresenta a utilização do *software* Matlab® 7.0 no ensino da tecnologia OPC aplicada a controle de processos, mais precisamente no controle de uma malha de vazão. Destaca a importância da aprendizagem das tecnologias utilizadas na indústria para a formação dos futuros engenheiros, aproximando teoria e prática.

Palavras-chave— Matlab®, OPC, Controle de Processos.

1 Introdução

O aprimoramento das técnicas de controle associado à evolução dos sistemas de Automação Industrial, torna complexa a determinação dos equipamentos e do tipo de controle que melhor se ajusta às necessidades de uma dada empresa. Acompanhar essa evolução requer da empresa um corpo técnico continuamente atualizado (Silveira et al., 1998).

Pereira e Lima (1998) ressaltam a importância do treinamento recebido pelo engenheiro durante a graduação. Citam que sua prática profissional é fortemente influenciada por este treinamento apesar das mudanças nos métodos provocado pelo avanço tecnológico.

A fim de fechar esta lacuna, os cursos de engenharia devem aproximar visual e operacionalmente o experimento laboratorial aos processos reais. Seria conveniente dispor de experimentos que utilizem as mesmas grandezas, dispositivos e *software* encontrados na prática industrial (Jr. e Bazanella, 2002).

Considerando o ensino de controle e sistemas de automação uma questão chave é reduzir a distância entre os cursos teóricos clássicos e a prática industrial. Portanto, é importante permitir aos alunos operar com dispositivos, sistemas e técnicas que abordem o mais próximo possível o cenário industrial (Zeilmann et al., 2003)

Neste artigo, apresenta-se um sistema didático com essas características que vêm sendo utilizado com sucesso para o ensino da tecnologia OPC (OLE - *Object Linking and Embedding - for a Process Control*) em controle de processos, mais precisamen-

te em um controle de vazão da planta didática Smar III, localizada no laboratório de Controle e Instrumentação do Curso de Graduação em Engenharia Elétrica do Centro Universitário do Leste de Minas Gerais – Unileste MG.

O artigo está organizado da seguinte forma: na seção 2, são apresentados os conceitos e propriedades referentes à tecnologia OPC. Na seção 3, são apresentadas a planta didática Smar e a malha de vazão 1, na seção 4, a interface de controle utilizada no ensino da tecnologia OPC desenvolvida no *software* MATLAB®. Na seção 5, alguns comentários conclusivos.

2 Fundamentação Teórica

2.1 A tecnologia OLE/COM

A tecnologia OLE 1.0 foi desenvolvida pela *Microsoft* em meados de 1990, para suprir a necessidade de se integrar diferentes aplicações dentro da plataforma *Windows*, de forma a solucionar os problemas de desempenho e confiabilidade do até então utilizado padrão DDE (*Dynamic Data Exchange*) (Fonseca, 2002). Nessa época, introduziram-se dois conceitos: *Linking* – Cria vínculos ou referências aos objetos, armazenando no documento principal apenas os dados realmente necessários para exibir, imprimir, etc. *Embedding* – Incorpora os dados dos objetos ao documento principal. Neste contexto, surgiram os conceitos de objeto vinculado e de objeto incorporado: Objeto Vinculado – São informações (objetos) criadas em um arquivo (arquivo origem) e inseridas

em outro arquivo (arquivo destino). Embora o objeto vinculado não se torne parte do arquivo de destino, existe um vínculo, uma conexão entre os dois arquivos de forma que o objeto vinculado no arquivo de destino seja automaticamente atualizado quando o arquivo de origem é atualizado. Objeto Incorporado – São informações inseridas em um arquivo de destino. Ao ser incorporado, o objeto se torna parte do arquivo. Ao clicar duas vezes no objeto incorporado, ele é aberto no programa de origem em que foi criado. Qualquer alteração feita no objeto incorporado se refletirá no arquivo de destino.

Outro conceito importante na tecnologia OLE é o conceito de Cliente Servidor:

- Cliente – Aplicação que solicita os dados.
- Servidor – Aplicação que disponibiliza os dados.

No OLE 2.0, houve muitos aperfeiçoamentos dentre os quais o mais importante é a Automação OLE, pois permite que uma aplicação seja controlada por outra aplicação.

A tecnologia OLE é montada sobre a tecnologia COM (*Component Object Model*) que define um modo padronizado para um módulo-cliente e um módulo-servidor se comunicarem por meio de uma interface específica. Módulo indica um aplicativo ou uma biblioteca (uma DLL – *Dynamic Link Libraries*). Os dois módulos podem ser executados no mesmo computador ou em máquinas diferentes conectadas através de uma rede. Muitas interfaces são possíveis, dependendo do papel do cliente e do servidor, e podem-se incluir novas interfaces para propósitos específicos.

2.2 A tecnologia OPC

Os controles de processos industriais têm se tornando cada vez mais complexos devido à exigência de qualidade dos produtos, rapidez na entrega e concorrência de mercado. Tal complexidade produz grandes

quantidades de dados a serem gerenciados pelos três níveis de controle de processos: dispositivos de campo, sistemas de controle e *softwares* para gerenciamento e negócios (Figura 1). A integração desses níveis utilizando efetivamente a informação disponível em cada um deles é de fundamental importância para a indústria.

Tradicionalmente, quando uma aplicação necessita acessar informações de um equipamento de controle ou de um dispositivo de chão de fábrica, uma interface customizada ou um *driver* de comunicação necessários, para a maioria dos casos, devem ser desenvolvidos. Muitas destas aplicações não conseguem acessar as informações devido à inconsistência entre fabricantes de *drivers* e *hardwares*.

Na busca de uma solução para esse problema, foi desenvolvida a tecnologia OPC, que é uma tecnologia para conectar aplicações *Windows* e equipamentos de controle de processos. O OPC é um protocolo de comunicação aberto que permite um método consistente de acesso aos dados de inúmeros equipamentos dos mais diversos fabricantes. O método é o mesmo, independente da origem dos dados, o que vem oferecer ao usuário final uma maior liberdade na escolha dos equipamentos independentemente da disponibilidade de *drivers* de comunicação proprietários.

O OPC é construído usando tecnologia *Microsoft OLE/COM*, mas a especificação OPC foi desenvolvida por uma fundação aberta, a *OPC Foundation*, para atender as necessidades gerais da indústria e não as necessidades específicas de alguns fabricantes de *hardware* e *software* (OPC Foundation, 1998). A especificação ainda prevê a evolução das funcionalidades ao longo do tempo e por isso, os componentes OPC podem se manter no topo das necessidades emergentes da indústria.

Os componentes OPC se classificam em duas categorias: Clientes OPC e Servidores OPC. Um Cliente OPC é tipicamente um usuário dos dados tais como uma Interface de Operação ou um sistema su-

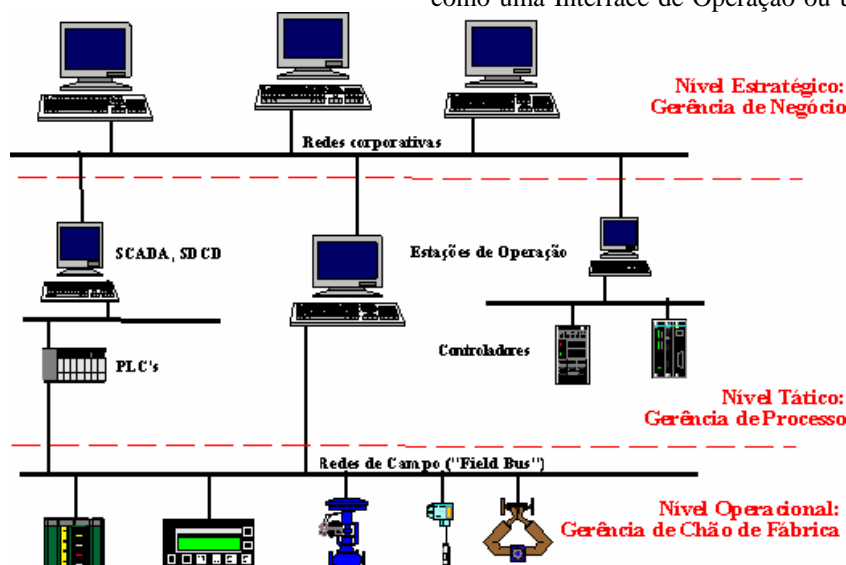


Figura 1. Níveis de Controle de Processos.

pervisório (SCADA). Um Servidor OPC é uma fonte de dados que coleta ou gera dados a partir de um processo, disponibilizando-os aos Clientes OPC.

O Cliente OPC interage com o Servidor OPC usando uma interface bem definida. Qualquer Cliente OPC pode se comunicar com qualquer Servidor OPC, independentemente do tipo de dispositivo e do fabricante, conforme está esquematicamente representado na Figura 2. Essa comunicação é válida somente para o OPC-DA (*Data Access*), uma vez que existem diferentes tecnologias OPC.

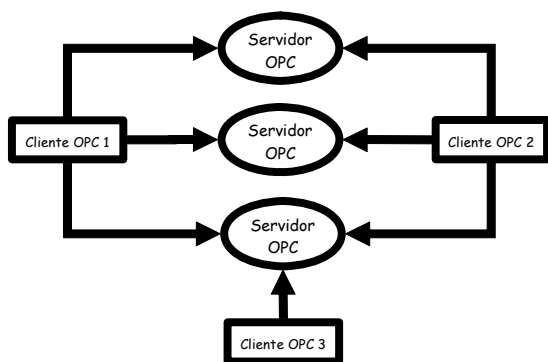


Figura 2. Relação entre Clientes e Servidores.

2.3 Arquitetura OPC

A arquitetura OPC pressupõe três objetos básicos: servidor, grupo e item, apresentados na Figura 3. Do ponto de vista do cliente, um servidor é essencialmente uma estrutura de armazenagem para grupos que, por sua vez, têm como função básica o armazenamento de itens. Esses itens, elementos mais simples na especificação, representam conexões a pontos de entrada ou saída. Assim, o item OPC não é um valor, mas apenas um meio de acesso a um valor. Desta forma, uma única variável de entrada ou saída pode ser representada por itens diferentes, com propriedades distintas e compartilhada por mais de um cliente. É tarefa dos grupos reunir o conjunto de itens que interessam a um determinado cliente, assumindo o papel principal na interação cliente-servidor.

Os grupos também são responsáveis por satisfazer pedidos de leitura e escrita, bem como por enviar atualizações para seus clientes, periodicamente ou por exceção. Essas transações de atualização podem ser ativadas ou desativadas no grupo ou nos itens individuais.

Os grupos presentes em um servidor OPC são normalmente definidos pelos clientes, e somente o cliente criador do grupo pode acessá-lo; tal tipo de grupo é dito privado. Em alguns casos, porém, pode ser interessante que o servidor ofereça grupos passíveis de serem compartilhados por vários clientes.

Quando essa capacidade é desejada, implementa-se a funcionalidade opcional dos grupos públicos.

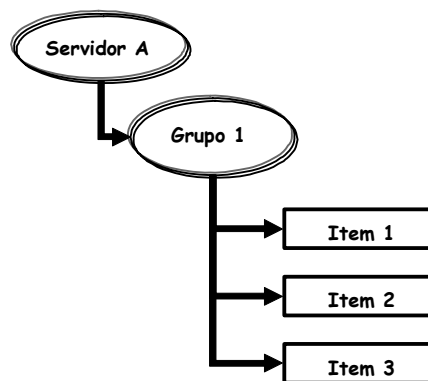


Figura 3. Arquitetura OPC.

Do ponto de vista do cliente, a função básica do servidor é prover uma infra-estrutura de suporte aos grupos. Além disso, cabe também a ele gerenciar aspectos relacionados à conexão com uma fonte de dados, tais como parâmetros de comunicação ou taxa máxima de amostragem. Outra responsabilidade do servidor é implementar uma estrutura de endereçamento capaz de associar itens com variáveis reais. Mais detalhes podem ser encontrados em (Souza et al, 1998).

O item é uma estrutura a qual estão associadas três propriedades (Fonseca, 2002):

- *Value*: último valor armazenado pelo servidor no cache de memória do item e que é atualizado sempre que o servidor faz uma leitura no dispositivo.
- *Quality*: informação de estado que define a qualidade do dado que pode ser: *Good* – dado válido; *Bad* – perda do *link* de comunicação com o dispositivo de campo; e *Uncertain* – no caso de existir o *link* e o dispositivo de campo estiver fora de comunicação.
- *Time Stamp*: tempo de amostragem.

3 Planta Didática Smar III

A Planta Didática Smar III, apresentada na Figura 4, tem por objetivo demonstrar didaticamente a operação das diversas malhas de controle utilizando os mesmos equipamentos e ferramentas de configuração, em *software*, desenvolvidos para aplicação em controle industrial (Smar, 2003).

A planta utiliza a tecnologia *Foundation Fieldbus*, barramento industrial de comunicação responsável por conectar os dispositivos de campo tais como sensores, atuadores, indicadores e controladores, que permitem o controle das malhas de temperatura, vazão e nível existentes, auxiliando no aprendizado de Instrumentação Industrial e Sistemas de Controle.

A planta está conectada a um microcomputador com 256 MB de RAM, processador AMD Sempron

(tm) 2400+ 1,66 GHz e sistema operacional *Windows XP*.

No microcomputador estão instalados os servidores OPC da rede *Fieldbus* ('Smar.DfiOleServer.0') e do CLP LC700 ('Smar.LC700v7Server.2'), responsável pelo intertravamento dos comandos e acionamentos elétricos das bombas 1 e 2. Também se encontra no microcomputador o *software* de supervisão *Process View*[®] (Smar, 2003), que permite monitorar e atuar no sistema em funcionamento.



Figura 4. Planta Didática Smar III.

3.1 Malha de Vazão 1

A Planta Didática apresenta duas malhas de vazão: malha 1 e a malha 2. A malha de vazão 1 controla a vazão de água bombeada pela bomba nº 1 do tanque principal ao tanque nº 1. Este por sua vez quando está cheio retorna a água para o tanque principal. A medição de vazão se dá por meio de um transmissor indicador de vazão (FIT-31). O controle é executado pela interface de controle no microcomputador e por uma válvula tipo globo (FY-31). Nessa malha tem-se instalado também um transmissor de nível no tanque nº 1 (LIT-31). A Figura 5 apresenta os itens acima citados com exceção do tanque principal.

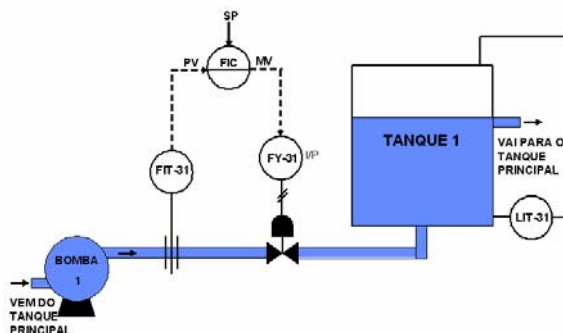


Figura 5. Malha de Vazão 1.

4 Interface de Controle

4.1 MATLAB[®] 7.0

O MATLAB[®] é um programa interativo para cálculos científicos e de engenharia, muito utilizado no meio

acadêmico por alunos e professores. A família de programas MATLAB[®] inclui o programa principal e uma variedade de *toolboxes*, uma coleção de arquivos especiais chamados *M-files* que estendem a funcionalidade do programa principal. Juntos, o programa principal e o *Control System Toolbox* propiciam a capacidade para projetar e analisar sistemas de controle (Dorf e Bishop, 2001).

O MATLAB[®] 7.0 apresenta novos *toolboxes*, destacando-se entre eles o OPC *toolbox*, que é uma coleção de funções que estendem a capacidade numérica do MATLAB[®], capaz de estabelecer comunicação com os Servidores OPC, assumindo o papel de Cliente OPC. É válido ressaltar que o MATLAB[®] e seus *toolboxes* são produtos comerciais da *Mathworks* (www.mathworks.com).

4.2 Interface de Controle

A interface de controle foi desenvolvida no ambiente de programação do MATLAB[®] 7.0 – GUIDE (*Graphic User Interface Design Environment*) que permite criar objetos de controle de interface com o usuário em janelas gráficas, permitindo o controle da planta sem a utilização do sistema de supervisão instalado, o *ProcessView*[®].

A interface de controle realiza o controle de vazão da malha 1 da Planta Didática utilizando a tecnologia OPC e consiste em uma janela que apresenta os seguintes campos, conforme mostra a Figura 6:

- Gráfico de tendências com legenda que mostra em tempo real as seguintes variáveis: saída do processo (PV), valor desejado (SP), e sinal de controle (MV).
- Botões liga e desliga bomba que permitem ligar e desligar o processo.
- Controle manual ou automático. Em manual o controle se dá em malha aberta, onde é possível estabelecer a vazão entre 0% e 100%. Em automático o controle se dá em malha fechada com controladores P, PI ou PID de acordo com a equação que se segue:

$$MV(t) = Kp \left(e(t) + \frac{1}{Ti} \int_0^t e(\tau) d\tau + Td \frac{d}{dt} e(t) \right) \quad (1)$$

sendo que $e(t)$ representa o erro entre o valor desejado e a saída do processo e Kp , Ti e Td são os parâmetros do controlador PID, denominados respectivamente de Ganho Proporcional, Tempo de Integral e Tempo Derivativo.

- Tipo de controlador P, PI ou PID que permite ao usuário escolher e comparar as diferentes estratégias de controle.
- Alteração dos parâmetros do PID, Kp , Ti e Td .
- Alteração do valor desejado de vazão (SP) e visualização da saída do processo (PV), sinal de

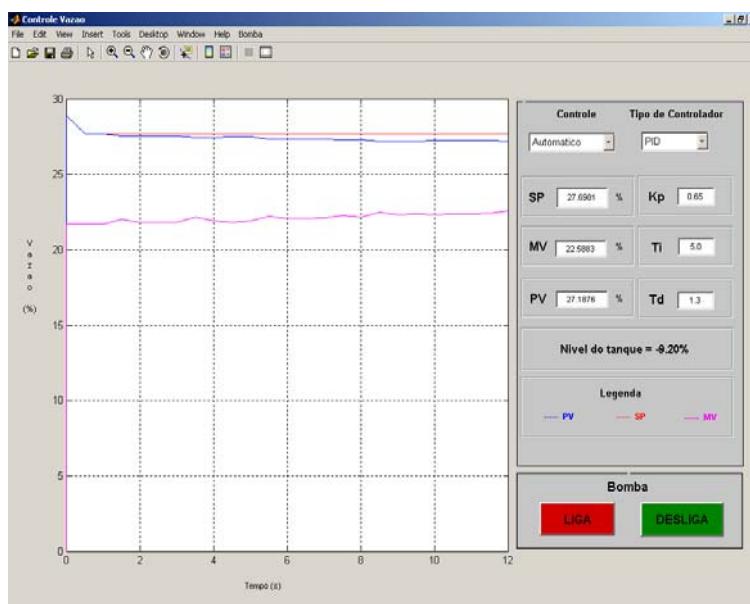


Figura 6. Interface de Controle.

controle (MV) e o valor de nível medido no tanque nº 1.

4.3 Algoritmo de Controle

Em conjunto com a interface gráfica, utilizou-se um algoritmo de controle PID elementar. Para permitir o acesso aos valores medidos pelos transmissores FIT-31 e LIT-31 e a atuação da válvula FY-31, utilizaram-se as funções do MATLAB® 7.0 que permitem conectá-lo aos servidores OPC da Planta Didática descritos na seção 3.

O primeiro passo para se conectar é entender como o OPC toolbox é implementado. Este está dividido em três objetos básicos dispostos em uma hierarquia específica, que serão descritos a seguir:

- *OPC Data Access Client Objects (OPC-DA)* representa um Cliente OPC específico conectado a um único Servidor, que tem a função de armazenar os grupos e administrar a conexão e a comunicação com este. O OPC-DA apresenta função semelhante ao servidor na arquitetura OPC descrita no item 2.3 no que diz respeito à administração dos grupos.
- *Data Access Group objects (dagroup)* representam os grupos contidos no OPC-DA e que armazenam um ou mais itens. Têm a função de administrar como os itens devem ser lidos, o armazenamento de históricos, quando estes existirem, e a criação ou retirada de itens. Os grupos existem somente dentro de um OPC-DA
- *Data Access Item Objects (daitem)* representam os itens, também conhecidos como variáveis, que são lidos ou escritos pelo servidor nos dispositivos de campo. Os itens existem somente dentro de um *dagroup*.

O segundo passo é gerar um OPC-DA por meio da função 'opcda' informando o local onde está instalado o servidor e nome do servidor ao qual se

deseja conectar. Depois de implementado, conectá-lo ao servidor pela função 'connect'.

Um *dagroup* pode ser acrescentado em um OPC-DA a partir da função 'addgroup'. O mesmo pode ser feito em relação ao *daitem* em um *dagroup* por meio da função 'additem'. Para a função 'additem' devem ser informados os nomes das variáveis que se desejam obter dados (leitura) ou enviar dados (escrita). A Tabela 1 relaciona cada instrumento ou dispositivo ao seu respectivo item e servidor OPC.

Tabela 1. Servidores, Itens e Dispositivos.

Servidor 'Smar.DfiOleServer.0'	
Dispositivo	Item
FIT-31	FIT-31_AH1.OUT.VALUE
LIT-31	LIT-31_AH1.OUT.VALUE
FY-31	FY-31_AO1.OUT.VALUE
Servidor 'Smar.LC700v7Server.2'	
Dispositivo	Item
Bomba nº 1	PD3.A.000.01 (Liga) PD3.A.000.02 (Desliga)

A leitura de dados em uma variável é efetuada a partir da utilização da função 'read' e a escrita de dados pela função 'write'. Após o término das atividades que utilizam as funções do OPC toolbox, deve-se desconectar o OPC-DA dos servidores por meio da função 'disconnect'.

A Figura 7 apresenta um exemplo de programa que utiliza as funções descritas anteriormente.

```

% Programa que estabelece conexão com o
% servidor 'Smar.DfiOleServer.0' por meio do
% cliente 'da'.
% Gera um grupo,'grp', com duas variáveis e
% por meio deste realiza a leitura da
% variável 'FIT-31_A01.OUT.VALUE' e a
% escrita de dados na variável
% 'FY-31_A01.OUT.VALUE'.

% Geração de um Cliente (OPCDA)
da=opcda('localhost', 'Smar.DfiOleServer.0');
% Estabelece conexão entre Cliente e Servidor
connect(da);
% Geração de um grupo de variáveis (dagroup)
grp=addgroup(da);
% Adição de variáveis ao grupo (daitem)
item1=additem(grp, 'FIT-31_A01.OUT.VALUE');
item2=additem(grp, 'FY-31_A01.OUT.VALUE');
% Leitura da variável 'FIT-31_A01.OUT.VALUE'
variavel_auxiliar=read(item1);
valor_de_vazao=variavel_auxiliar.Value;
% Escreve o valor 50 na variável
% 'FY-31_A01.OUT.VALUE'
write(item2,50);
% Desconecta o cliente do servidor
disconnect(da);
% Apaga o cliente
delete(da);
% Limpeza das variáveis relacionadas ao
% cliente
clear item1 item2 grp;

```

Figura 7. Programa exemplo que utiliza as funções do OPC *Tool-box*.

Atualmente, a maioria dos instrumentos de campo e controladores industriais já são fornecidos com os respectivos Servidores OPC, o que torna relevante o ensino de tal tecnologia no meio acadêmico.

5 Conclusões

Foi apresentada a utilização do software MATLAB® como cliente OPC voltada para o ensino da tecnologia OPC em controle de processos. Verificou-se por meio de um exemplo, que o desenvolvimento de uma aplicação de controle em uma malha de vazão real (Planta didática Smar III) é um bom caminho para o ensino da tecnologia OPC. A execução do algoritmo de controle PID também poderia ser realizada pelos dispositivos *Foundation Fieldbus* da Smar e por meio do cliente OPC (MATLAB®) ao enviar apenas os parâmetros K_p , T_i e T_d para o dispositivo controlador e monitorar os valores associados aos sensores e transmissores.

A demonstração prática da simplicidade de projetos utilizando a tecnologia OPC visa encorajar o aluno no desenvolvimento de sistemas mais complexos para análise, síntese, projeto e implementação de malhas de controle. Desta forma o aluno é levado a trabalhar com sistemas e problemas encontrados na indústria, ao invés de uma simulação, enriquecendo sua bagagem acadêmica.

A interface desenvolvida neste trabalho pode ser executada em qualquer planta industrial ou sistema que disponha de Servidores OPC conectados aos instrumentos de campo, bastando alterar apenas os nomes do Servidor OPC e dos itens do grupo.

Agradecimentos

Os autores agradecem ao CNPq e à FAPEMIG pelo apoio financeiro.

Referências Bibliográficas

- Dorf, R. C., Bishop, R. H.(2001). *Sistemas de Controle Modernos*. 8 ed. Rio de Janeiro: Livros Técnicos e Científicos, 659p.
- Fonseca, M. O. (2002). *Comunicação OPC – Uma abordagem Prática*, VI Seminário de Automação de Processos, Associação Brasileira de Metalurgia e Materiais, Vitória – ES, Brasil.
- Jr., J. M. G. S., Bazanella, A. S. (2002). *Um Sistema Didático para o Ensino de Ajuste de Controladores PID*, XIV Congresso Brasileiro de Automática, Natal, pp. 1403–1408.
- OPC Foundation (1998). *OPC Overview*, disponível em <http://www.opcfoundation.org/> (10 mar. 2005)
- Pereira, L. F. A., Lima, J. C. M. (1998). *Reestruturação do Ensino de Automação e Controle através da Implantação de Laboratórios*, XII Congresso Brasileiro de Automática, Vol. II, Uberlândia, pp. 637–642.
- Silveira, M. R., Loures, E. R. e Arruda, L. V. (1998). *A Utilização de Computadores no Ensino de Controle: Até que ponto é vantajoso?*, XII Congresso Brasileiro de Automática, Vol. II, Uberlândia, pp. 643–647.
- Smar (2003). *Manual de Operação Planta Didática III*.
- Souza, L. C. A., Filho, C. S. e Pena, R. T. (1998). *Padrão de Acesso a Dados OPC e sua Implementação em um Driver OPC-MODBUS*. In: V Simpósio Regional de Instrumentação/ II Congresso Mineiro de Automação, ISA / GRINST – IBP, Belo Horizonte, p. 157-164.
- Zeilmann, R. P., Silva, J. M. G., Bazanella, A. S. and Pereira, C. E. (2003). *Web-based Control Experiment on a Foundation Fieldbus Pilot Plant*. IFAC International Conference on Fieldbus Systems and their Applications, Aveiro, Portugal, p. 325-330.