

## APLICAÇÃO DE CONTROLADOR LÓGICO PROGRAMÁVEL E SISTEMA DE SUPERVISÃO NO CONTROLE DE SEMÁFOROS

**RADEL, Felipe Vesenick<sup>1</sup>; SCHMITZ, Gabriel Figueiredo<sup>1</sup>; LOPES, Danien Leyes<sup>2</sup>; PAIM, Guilherme Pereira<sup>2</sup>; ESPOSITO, Marcelo<sup>3</sup>**

<sup>1</sup>Graduação em Engenharia de Controle e Automação; <sup>2</sup>Graduação em Engenharia Eletrônica;

<sup>3</sup>Universidade Federal de Pelotas, Centro de Engenharias. marcelo.esposito@ufpel.edu.br

### 1 INTRODUÇÃO

Ruas e avenidas são o meio físico de circulação dos veículos de uma cidade. Normalmente, num cruzamento entre duas ou mais vias, existem veículos cujos movimentos não podem ser realizados simultaneamente, pois são conflitantes entre si. Portanto, é necessário estabelecer alguma norma de controle de direito de passagem, a fim de se aumentar as condições de fluidez do cruzamento e reduzir os riscos de acidentes, tanto entre veículos como veículos-pedestres.

O semáforo é um dispositivo de controle de tráfego que, através de indicações luminosas transmitidas para motoristas e pedestres, alterna o direito de passagem de veículos e/ou pedestres em interseções de duas ou mais vias. Já o controlador de tráfego é um equipamento que comanda o semáforo através do envio de pulsos elétricos para comutação das luzes dos focos. Nos controladores de tempo fixo o tempo de ciclo é constante e a duração, e os instantes de mudança dos estágios são fixos em relação ao ciclo. Assim por exemplo, controlar uma interseção isolada em tempo fixo significa sinalizar o cruzamento, dando sempre o mesmo tempo de sinalização verde, amarelo e vermelho a cada corrente de tráfego que por ali passe, independentemente da variação do volume de veículos que chegam nas proximidades da interseção (DENATRAN, 1984). Embora o sistema apresentado neste trabalho se enquadre neste tipo de controlador ele pode ser ampliado para satisfazer os requisitos de controladores por demanda de tráfego, providos de detectores de veículos e lógica de decisão.

Sistema ou *software* supervisorio é uma ferramenta de desenvolvimento de aplicativos que se presta a fazer a comunicação entre um computador e uma rede de automação e que traz ferramentas padronizadas para a construção de interfaces entre o operador e o processo. É parte fundamental de praticamente qualquer sistema de automação de processos e permite realizar as seguintes funções: monitorar variáveis do processo em tempo real (temperatura, pressão, nível, etc.), diagnosticar falhas ou condições indevidas através de alarmes e eventos, ajustar parâmetros do processo (*set-points*, ganhos de controladores, alarmes, etc) e enviar comandos para o sistema.

É comum definir o *software* de supervisão como SCADA (proveniente do termo em inglês, *Supervisory Control And Data Acquisition*). Rigorosamente, um sistema SCADA é composto não somente pelo sistema supervisorio, mas também pela instrumentação, Controladores Lógicos Programáveis (CLPs) e toda a infraestrutura de comunicação utilizada no sistema de automação e controle. Tipicamente, o supervisorio faz a interface com os elementos de controle.

Por outro lado, com o conceito de *software* supervisorio é possível concluir que sistemas baseados em linguagens convencionais (Visual Basic, Delphi, etc.), feitos sob encomenda para comunicação com uma máquina ou um processo

simples, não são *softwares* supervisórios, pois alterações significativas nestes sistemas geram um novo programa. Sistemas feitos para coletar dados de uma família de controladores, não são supervisórios, pois não permitem interagir com outras famílias. Da mesma forma, *softwares* de configuração de Interfaces Homem Máquina (IHM), localizadas no chão de fábrica, para que o operador tenha acesso local às informações do processo, também não são supervisórios, pois normalmente permitem configurações de telas de uma determinada família de IHMs, limitando-se a existência de um *hardware* específico.

Os *softwares* de comunicação dispõem de duas maneiras para se comunicar com equipamentos de diferentes fabricantes. A primeira delas é o emprego de *drivers* de comunicação para a implementação do protocolo suportado pelo equipamento com o qual se pretende comunicar. Outro modo é o uso do OPC (*OLE for Process Control - Object Linking and Embedding*) que estabelece as regras para que sejam desenvolvidos sistemas com interfaces padrão para comunicação dos dispositivos de campo (CLPs, sensores, atuadores, etc.) com sistemas de monitoramento, supervisão e gerenciamento (SCADA, IHM, etc.) (Pasetti, 2005).

A norma IEC 61131-3, estabelecida pelo *International Electrotechnical Commission* divide um sistema automatizado em duas partes distintas: parte operativa, correspondente ao processo físico a automatizar, que opera sobre a matéria prima e o produto, constituída pelos atuadores, tais como válvulas, motores, lâmpadas, etc; e a parte de comando, que se caracteriza por receber as informações vindas do operador e/ou do processo a ser controlado e por emitir informações ao sistema controlado, coordenando assim as ações da parte operativa. Dentre as tecnologias de comando podem-se citar comandos pneumáticos, hidráulicos, através de painéis de relés e CLPs (Rosário, 2005).

Os CLPs foram desenvolvidos com o intuito inicial de substituírem os painéis de relés no controle discreto, porém, atualmente, os controladores são bem mais complexos, pois as plantas industriais normalmente precisam manipular não somente funções lógicas binárias, como por exemplo, tipo E e OU, mas também malhas analógicas (Franchi, 2009).

O presente trabalho utiliza o conceito de um semáforo, visto como um objeto que pode sofrer dois tipos de operação: aberto ou fechado. Os objetivos foram: desenvolver um programa em linguagem Ladder, que por meio das entradas e saídas digitais do CLP fosse equivalente ao funcionamento de um semáforo e criar um sistema supervisório para este caso. Para tanto, utilizando-se a ferramenta Eclipse SCADA, que possibilitou o acompanhamento em tempo real e a animação de telas, além de um CLP da marca WEG. A unidade experimental foi montada utilizando-se LEDs (Diodo emissor de luz) ligados às saídas do CLP.

## 2 MATERIAL E MÉTODOS

Todos os testes de validação do *software* em Ladder e do supervisório foram conduzidos utilizando-se um controlador programável modelo TPW 03 (30HR-A) da marca WEG, um microcomputador marca HP com sistema operacional Windows 32 bits, um conversor de sinal USB/RS232, uma matriz de contato (*protoboard*), seis LEDs de cores verde, amarelo e vermelho, seis resistores de 2,2 kohm e uma *DIP switch* para o acionamento das entradas digitais em modo manual. Para o desenvolvimento e implementação do sistema supervisório foi utilizada a ferramenta Eclipse SCADA em modo de Demonstração.

O driver *Modbus* para comunicação entre o CLP e o supervisor foi gentilmente fornecido pela Elipse, enquanto que os dados de endereçamento foram obtidos a partir do manual do TPW 03 (WEG, 2012).

### 3 RESULTADOS E DISCUSSÃO

A Fig. 1 mostra o diagrama em Ladder utilizado na programação do CLP. Nota-se que além de contatos NA (normalmente aberto), NF (normalmente fechado), memórias internas (M0) e bobinas (saídas), também foram utilizados sete temporizadores com retardo para ligar (TON – *Timer On Delay*). Esta configuração depende dos recursos de programação disponibilizados pelo fabricante do equipamento e da habilidade do programador. Por exemplo, se estivessem disponíveis funções de comparação e *special memories* (Siemens) para este modelo de CLP o mesmo programa seria criado com apenas um temporizador.

A Fig. 2 mostra a interface principal do controle supervisorio. Embora não seja possível identificar pela figura, há uma animação em tempo real: o semáforo sinaliza os três estados, vermelho, amarelo e verde e o automóvel se move respeitando o sinal de trânsito. A Fig. 3 mostra a implementação experimental do sistema. Percebe-se a correspondência entre os estados LEDs/Supervisorio e identificam-se as chaves X000 e X001 de acionamento manual. Apenas duas vias da chave *DIP switch* estão ligadas nas entradas digitais do CLP.

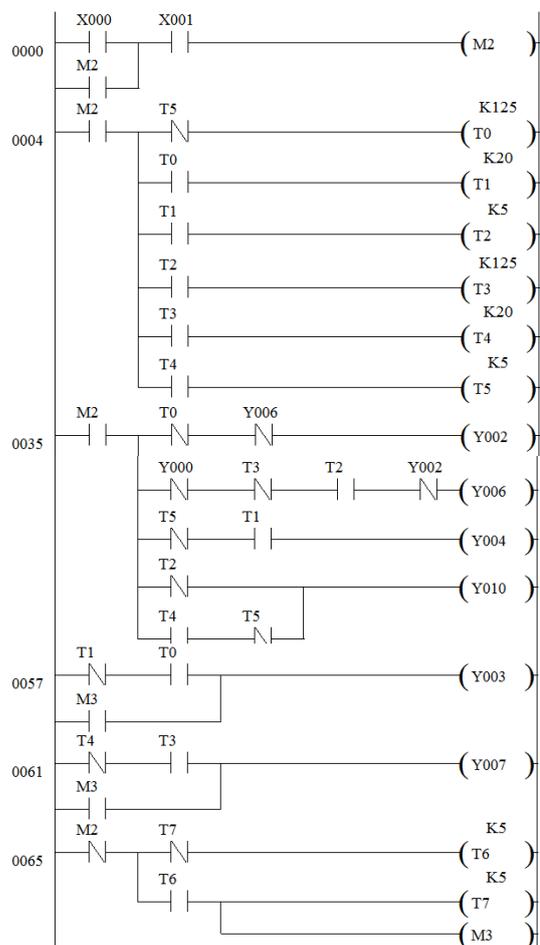


Figura 1 – Programação do CLP em Linguagem Ladder.



Figura 2 – Interface principal do supervisorio.

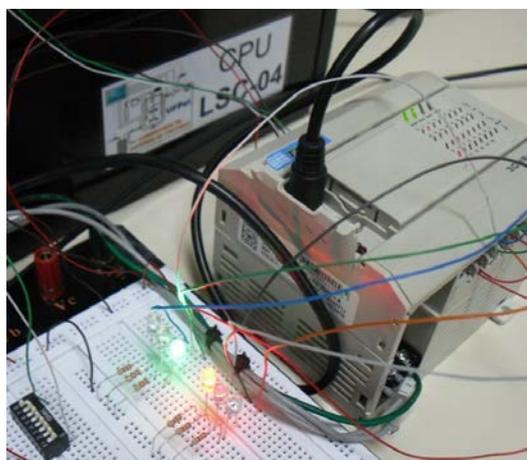


Figura 3 – CLP e semáforo implementado utilizando LEDs.

O grupo também desenvolve trabalhos utilizando a ferramenta E3 (Elipse, 2012) que possui arquitetura Cliente-Servidor.

#### 4 CONCLUSÃO

A aproximação entre a academia e a indústria nem sempre ocorre de forma natural e espontânea, por este motivo é que a utilização de equipamentos e *softwares* tipicamente encontrados nos mais variados tipos de plantas industriais é de extrema importância, dentro deste contexto o presente trabalho foi desenvolvido. A evolução foi medida pela satisfação dos autores ao constatar o salto de conhecimento adquirido e pela importância das ferramentas por eles dominadas.

Considerando experiências anteriores do grupo, foi possível verificar que para a criação de aplicativos de supervisão e controle não basta que estes sejam acessíveis, amigáveis e de baixo custo. A ferramenta ideal para a automação deve eliminar a necessidade de soluções demoradas e caras, deve também garantir competitividade, eficiência, flexibilidade e qualidade ao processo. Uma vez em funcionamento a integração entre Elipse SCADA e CLP mostrou-se robusta e de fácil entendimento. Não foram identificadas situações de perda ou falha de comunicação entre o supervisor e o processo, o que é comum em sistemas dedicados, desenvolvidos para casos específicos, onde o operador deve ser treinado antes de operar o sistema de supervisão e controle.

#### 5 REFERÊNCIAS

DENATRAN, Departamento Nacional de Trânsito. **Manual de Semáforos**, Brasília, 1984.

ELIPSE, **Instalador do Elipse SCADA**. Disponível em: <<http://www.elipse.com.br/>>. Acesso em: 08 de julho de 2012.

FRANCHI, Claiton Moro; CAMARGO, Valter Luís Arlindo de. **Controladores Lógicos Programáveis – Sistemas Discretos**. São Paulo: Editora Érica, 2009.

PASETTI, Giovani. **Instrumentação, controle e supervisão de uma coluna de destilação piloto utilizando tecnologia foundation fieldbus**. 2005. Dissertação (Mestre em Engenharia Elétrica, Área de Concentração em Controle, Automação e Sistemas), Universidade Federal de Santa Catarina, Setembro/2005.

ROSÁRIO, João Maurício. **Princípios de mecatrônica**. São Paulo: Editora Prentice Hall, 2005.

**Templates de objetos gráficos para programação interna da parte visual**. Disponível em: <<http://aventura.jogosloucos.com.br/jogos-de-motor-madness.html>>. Acesso em: 08 de julho de 2012.

WEG, **TPW-03 - Controlador programável – Manuais técnicos**. Disponível em: <<http://www.weg.net/br/Produtos-e-Servicos/Geral/Central-de-Downloads>>. Acesso em: 08 de julho de 2012.