

Controle de semáforo urbano utilizando programação ladder em ambiente de simulação: uma abordagem didática para o ensino de automação industrial

Vitor Amadeu Souza¹; 0009-0002-1857-6799

1 – UniFOA, Centro Universitário de Volta Redonda, Volta Redonda, RJ.
vitor.amadeu@foa.org.br

Resumo: Este trabalho apresenta o desenvolvimento e implementação de um sistema de controle de semáforo urbano utilizando a programação Ladder no software EcoStruxure Machine Expert Basic da Schneider Electric. O projeto foi desenvolvido com foco na aplicação didática para cursos de Engenharia Elétrica, demonstrando a importância da simulação no processo de aprendizagem antes da implementação prática em laboratório. O sistema proposto controla um semáforo de três cores (verde, amarelo e vermelho) através de uma botoeira de pedestres, implementando temporização controlada por meio de blocos temporizadores TON. A metodologia empregada envolveu o desenvolvimento do código Ladder, simulação computacional e análise dos resultados obtidos. Os resultados demonstraram eficácia na implementação do controle sequencial, com transições adequadas entre os estados do semáforo, validando a abordagem pedagógica proposta. O trabalho contribui para o ensino de automação industrial ao fornecer uma aplicação prática e didática dos conceitos de Controladores Lógicos Programáveis (CLPs), demonstrando a relevância da simulação no processo de ensino-aprendizagem em engenharia.

Palavras-chave: Controlador Lógico Programável. Programação Ladder. Semáforo. Simulação. Automação Industrial. Ensino de Engenharia.

INTRODUÇÃO

A automação industrial tem se tornado cada vez mais relevante no contexto tecnológico contemporâneo, sendo os Controladores Lógicos Programáveis (CLPs) elementos fundamentais neste cenário (Bolton, 2015). Os CLPs representam uma tecnologia consolidada para o controle de processos industriais, oferecendo flexibilidade, confiabilidade e facilidade de programação através de linguagens padronizadas pela norma IEC 61131-3 (Lewis, 1998). Entre estas linguagens, a programação Ladder destaca-se pela sua simplicidade conceitual e similaridade com diagramas de circuitos elétricos convencionais, facilitando a transição de profissionais da área elétrica para a automação (Parr, 1999).

O controle de semáforos urbanos representa uma aplicação clássica e didaticamente rica para o ensino de automação, pois envolve conceitos fundamentais como controle sequencial, temporização, interfaces de entrada e saída, e lógica de intertravamento (Kilian, 2006). Sistemas de controle de tráfego urbano têm evoluído significativamente, incorporando tecnologias avançadas como sensores, comunicação sem fio e algoritmos de otimização (Papageorgiou *et al.*, 2003).

O software EcoStruxure Machine Expert Basic, desenvolvido pela Schneider Electric, representa uma ferramenta moderna e profissional para programação de CLPs, oferecendo ambiente de simulação integrado que permite validação completa dos projetos antes da implementação física (Schneider Electric, 2020). Esta característica é particularmente valiosa no contexto educacional, pois permite aos estudantes experimentar com diferentes cenários e configurações sem necessidade de hardware físico imediato. A crescente complexidade dos sistemas de automação industrial demanda profissionais com sólida formação teórica e prática em CLPs (Groover, 2015).

O presente trabalho tem como objetivo principal demonstrar a implementação de um sistema de controle de semáforo urbano utilizando programação Ladder no ambiente de simulação do EcoStruxure Machine Expert Basic. Especificamente, pretende-se: desenvolver um programa Ladder para controle sequencial de semáforo com acionamento por botoeira; validar o funcionamento através de simulação computacional; analisar os resultados obtidos

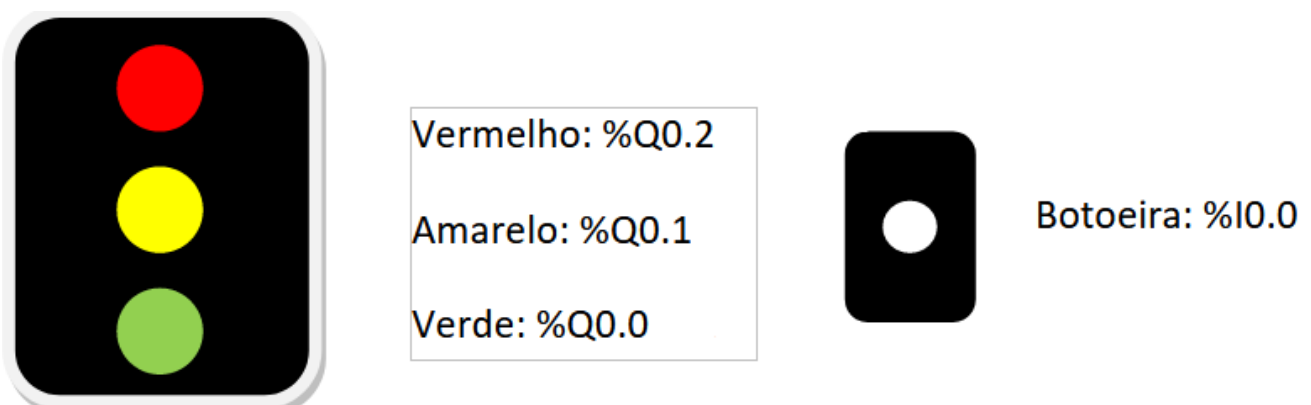
quanto à eficácia do controle implementado; e demonstrar a aplicabilidade da abordagem como ferramenta didática para o ensino de automação industrial.

MÉTODOS

A primeira etapa consistiu na definição dos requisitos funcionais do sistema de controle do semáforo. O sistema deveria atender às seguintes especificações: operação padrão com luz verde permanentemente acesa; acionamento do ciclo semaforico através de botoeira de pedestres conectada à entrada digital %I0.0; sequência operacional consistindo em transição verde-amarelo-vermelho-verde; temporização de 3 segundos para a luz amarela e 10 segundos para a luz vermelha; retorno automático ao estado inicial (verde) após completar o ciclo; e prevenção de múltiplos acionamentos durante a execução do ciclo.

O mapeamento de entradas e saídas foi definido conforme apresentado na Figura 1. A botoeira de pedestres foi conectada à entrada digital %I0.0, configurada como normalmente aberta com lógica positiva. As saídas digitais foram mapeadas da seguinte forma: luz vermelha conectada à saída %Q0.2, luz amarela à saída %Q0.1, e luz verde à saída %Q0.0. Esta configuração seguiu as práticas recomendadas para sistemas de semáforos urbanos, garantindo identificação clara e manutenção facilitada (Siemens, 2022).

Figura 1 - Pinagem usada no projeto



Fonte: O autor

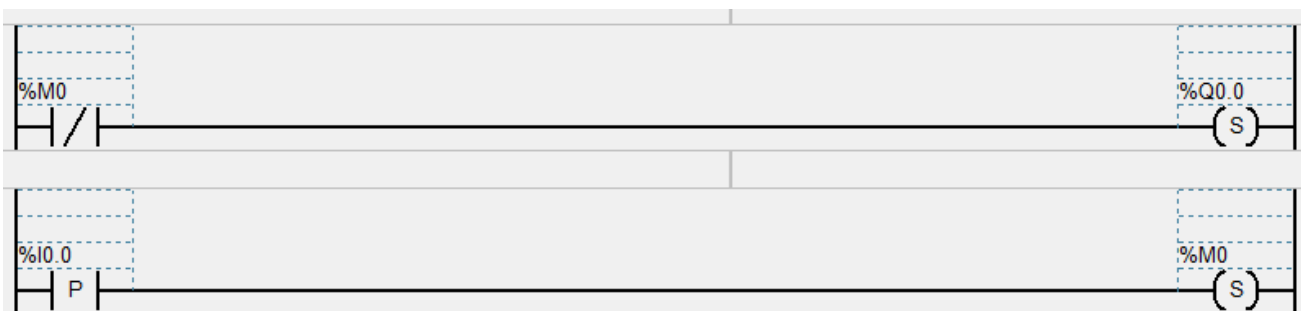
A implementação da lógica de controle foi realizada utilizando a linguagem Ladder, empregando blocos temporizadores do tipo TON (Timer On-Delay) para controle das temporizações. A arquitetura do programa foi estruturada em cinco rungs principais, cada

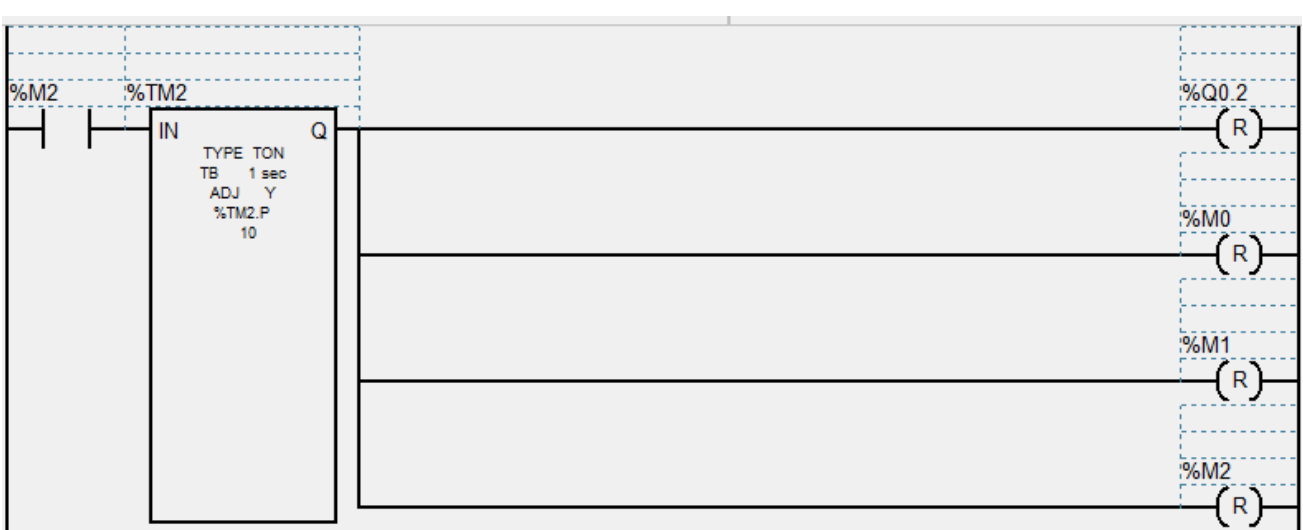
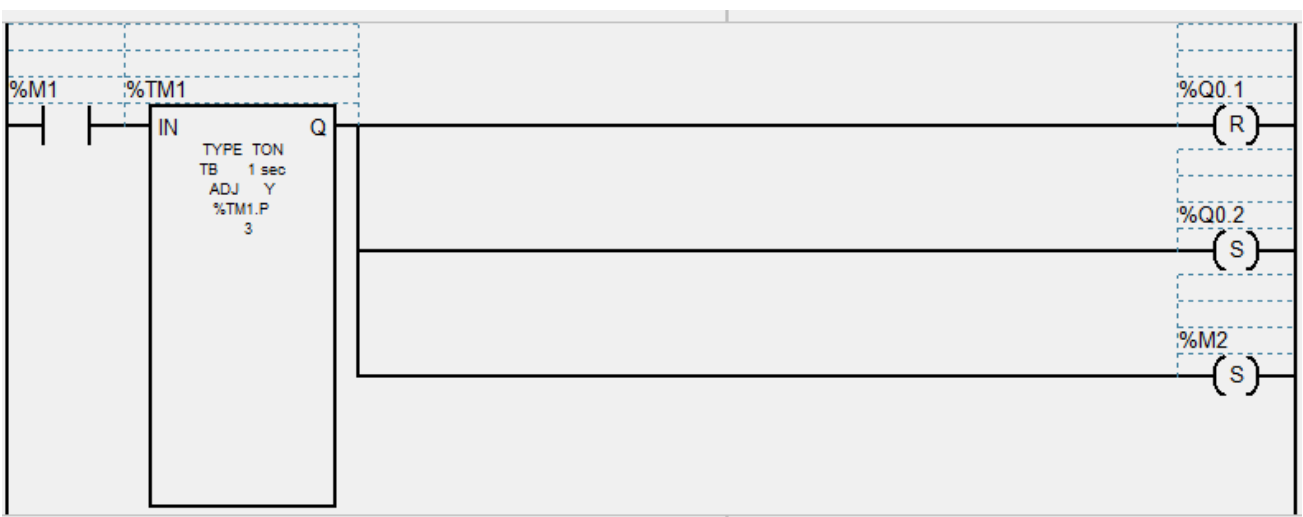
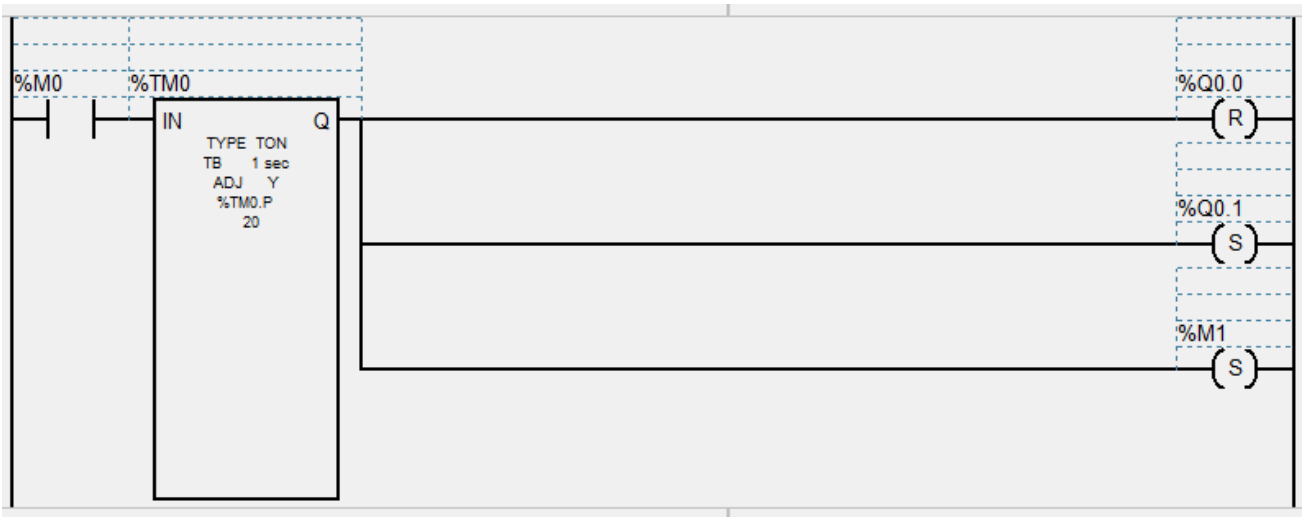


uma responsável por uma função específica do sistema. A primeira rung implementou a lógica de acionamento inicial, utilizando a entrada %I0.0 (botoeira) em série com um contato normalmente fechado da memória %M0, garantindo que o ciclo só seja iniciado quando o sistema estiver em repouso. Esta abordagem previne acionamentos múltiplos durante a execução do ciclo, conforme recomendado por Webb & Reis (2014). A segunda rung controlou o estado padrão da luz verde, implementando lógica que mantém a saída %Q0.0 energizada quando o sistema não está executando nenhum ciclo temporizado. A condição implementada utiliza contatos normalmente fechados dos temporizadores e memórias auxiliares, garantindo que a luz verde permaneça acesa exceto durante as transições programadas.

As rungs subsequentes implementaram o controle sequencial das luzes amarela e vermelha, utilizando temporizadores TON configurados com os tempos especificados. O temporizador %TM0, responsável pelo controle da luz amarela, foi configurado com preset de 3 segundos (3000ms), enquanto o temporizador %TM2, controlando a luz vermelha, recebeu preset de 10 segundos (10000ms). Esta configuração atende às normas técnicas para semáforos urbanos estabelecidas pelo CONTRAN (2007). A Figura 2 apresenta esta implementação.

Figura 2 - Implementação em Ladder





Fonte: O autor.

O ambiente de simulação permite monitoramento em tempo real de todas as variáveis do programa, incluindo estados das entradas, saídas, temporizadores e memórias auxiliares.

RESULTADOS E DISCUSSÃO

O sistema apresentou comportamento correto no estado inicial, mantendo a luz verde (saída %Q0.0) permanentemente energizada quando nenhum ciclo está em execução. Esta condição foi implementada por meio da lógica apresentada na segunda imagem do projeto, onde a combinação de contatos normalmente fechados dos temporizadores %TM0, %TM1 e %TM2 assegura que a luz verde permaneça acesa na ausência de qualquer temporização ativa. Este resultado está em conformidade com as práticas recomendadas para sistemas de semáforos, nos quais o estado padrão deve favorecer o fluxo principal de tráfego (Koonce *et al.*, 2008).

A lógica desenvolvida na primeira rung, conforme ilustrado nas imagens do projeto, utiliza a entrada %I0.0 em série com um contato normalmente fechado da memória %M0, garantindo que novos acionamentos sejam ignorados durante a execução do ciclo. Esta implementação previne efetivamente condições de corrida e garante a execução completa de cada ciclo iniciado, sendo esta uma característica essencial para a segurança em sistemas de controle de tráfego (Traffic control systems handbook, 2005).

A temporização de 3 segundos para a luz amarela foi realizada utilizando o temporizador %TM0, conforme ilustrado na segunda imagem. A transição da luz verde para a amarela ocorreu de forma instantânea com o acionamento da botoeira, demonstrando resposta adequada do sistema. A lógica implementada garante que apenas a luz amarela permaneça acesa durante esse período, atendendo às normas de segurança que proíbem múltiplas luzes acesas simultaneamente (NEMA, 2020).

O temporizador %TM2, responsável pelo controle da luz vermelha, apresentou funcionamento preciso com temporização de 10 segundos. Esta sequência automática elimina intervalos sem sinalização luminosa, mantendo o controle contínuo do tráfego. Durante o período da luz vermelha, observou-se que o sistema permaneceu em estado estável, ignorando novos acionamentos da botoeira. Esta característica é fundamental para



garantir tempo adequado para travessia de pedestres, conforme estabelecido nas diretrizes de engenharia de tráfego (Highway capacity manual, 2010).

Após o término da temporização de 10 segundos da luz vermelha, o sistema retorna imediatamente ao estado de luz verde acesa, sem necessidade de intervenção externa. Esta funcionalidade foi implementada por meio da lógica de reset automático dos temporizadores e memórias auxiliares. A análise temporal revelou que o ciclo completo (verde → amarelo → vermelho → verde) tem duração total de 13 segundos, sendo 3 segundos para a luz amarela e 10 segundos para a luz vermelha. Este tempo total está dentro dos parâmetros recomendados para semáforos de pedestres em áreas urbanas (Traffic engineering handbook, 2009).

CONCLUSÕES

A utilização do ambiente de simulação provou ser fundamental para validação do projeto antes da implementação física, permitindo identificação e correção de possíveis falhas de lógica sem riscos ou custos associados ao hardware. Esta abordagem demonstra a relevância da simulação como ferramenta pedagógica no ensino de automação industrial, proporcionando experiência prática valiosa aos estudantes.

A metodologia empregada mostrou-se adequada para projetos educacionais, proporcionando experiência sistemática no desenvolvimento de sistemas de automação. A combinação de análise de requisitos, desenvolvimento estruturado e validação por simulação oferece base sólida para projetos mais complexos.

Como trabalhos futuros, sugere-se implementação de funcionalidades adicionais como detecção de falhas, modo manual de operação, interface homem-máquina (IHM) para monitoramento e configuração, integração com sensores de presença para otimização automática dos tempos, e implementação física em bancada laboratorial para validação prática completa.

REFERÊNCIAS

BOLTON, W. Programmable Logic Controllers. 6. ed. Amsterdam: Elsevier, 2015.

CONTRAN, Conselho Nacional de Trânsito. Resolução nº 236, de 11 de maio de 2007. Estabelece volume I do Manual Brasileiro de Sinalização de Trânsito. Diário Oficial da União, Brasília, DF, 15 maio 2007.

GROOVER, M. P. Automation, Production Systems, and Computer-Integrated Manufacturing. 4. ed. Upper Saddle River: Prentice Hall, 2015.

Highway Capacity Manual 2010. Washington, DC: Transportation Research Board, 2010.

KILIAN, C. T. Modern Control Technology: Components and Systems. 3. ed. Clifton Park: Thomson Delmar Learning, 2006.

KOONCE, P.; RODEGERDTS, L.; LEE, K.; QUAYLE, S.; BEAIRD, S.; BRAUD, C.; BONNESON, J.; TARNOFF, P.; URBANIK, T. Traffic Signal Timing Manual. Washington, DC: Federal Highway Administration, 2008.

LEWIS, R. W. Programming Industrial Control Systems Using IEC 1131-3. 2. ed. London: Institution of Electrical Engineers, 1998.

NEMA (National Electrical Manufacturers Association). TS 1-1989 (R2020): Traffic control systems standards. Rosslyn, VA: NEMA, 2020. Disponível em: <https://www.nema.org/standards/view/Traffic-Control-Systems>. Acesso em: 1 ago. 2025.

PAPAGEORGIU, M.; DIAKAKI, C.; DINOPOULOU, V.; KOTSIALOS, A.; WANG, Y. Review of road traffic control strategies. Proceedings of the IEEE, v. 91, n. 12, p. 2043-2067, 2003.

PARR, A. Programmable Controllers: An Engineer's Guide. 2. ed. Oxford: Newnes, 1999.

SCHNEIDER ELECTRIC. EcoStruxure Machine Expert Basic Programming Guide. Rueil-Malmaison: Schneider Electric, 2020.

SIEMENS. SIMATIC S7-1200 Programmable Controller System Manual. Nuremberg: Siemens AG, 2022.

Traffic Control Systems Handbook. Washington, DC: Federal Highway Administration, 2005.

Traffic Engineering Handbook. 6. ed. Washington, DC: Institute of Transportation Engineers, 2009.

WEBB, J. W.; REIS, R. A. Programmable Logic Controllers: Principles and Applications. 5. ed. Upper Saddle River: Prentice Hall, 2014.