USO DA IEC61499 EM CONTROLE E MONITORAMENTO DISTRIBUÍDO AMBIENTES CLOUD/FOG/EDGE COMPUTING

Uriel Pacheco de Souza, André Bittencourt Leal, Herbert Albérico de Sá Leitão,
Yuri Kaszubowski Lopes, Roberto Silvio Ubertino Rosso Junior

INTRODUÇÃO

A Indústria 4.0 tem impulsionado a necessidade de descentralizar o controle e o processamento de dados nas fábricas. A computação de borda (*Edge Computing*) distribui o processamento primário para mais perto das fontes de dados, reduzindo latência, riscos de segurança e consumo de banda, requisitos importantes para Internet das Coisas (IoT) e para operações industriais em tempo real [1, 2]. Para estruturar o *software* de controle nesses ambientes distribuídos, a norma IEC 61499 introduz um modelo baseado em Function Blocks (FB) orientados a eventos [3]. Esse novo modelo supera as limitações da norma anterior (IEC 61131-3), que dependia de uma varredura cíclica centralizada com execução em um *loop* sequencial contínuo. Além disso, a nova abordagem permite a reconfiguração dinâmica do sistema sem a necessidade de parar a planta por completo [4]. A transição prática para sistemas orientados a eventos, no entanto, apresenta entraves de implementação e arquitetura [5], e a integração de FB com o protocolo OPC UA, padrão de interoperabilidade na automação industrial, é apontada como uma prática robusta para execução distribuída na Internet Industrial [6]. Este trabalho teve como objetivo desenvolver e validar um sistema de controle e monitoramento distribuído para manufatura, cobrindo três etapas: modelagem de uma esteira transportadora em ambiente virtual, migração da lógica de controle para o CLP físico WAGO PFC200 e integração com o barramento interno do CLP (K-BUS) para leitura e acionamento de sinais reais.

DESENVOLVIMENTO

A etapa inicial consistiu na modelagem e simulação da planta em ambiente 3D no *software* Factory I/O, representando uma esteira transportadora com uma caixa e um sensor de presença na extremidade. Utilizando o Eclipse 4diac, ambiente de desenvolvimento IDE e infraestrutura de execução FORTE, a lógica de controle foi desenvolvida baseada em FB da norma IEC 61499. A comunicação entre a lógica e a planta virtual foi estabelecida via protocolo OPC UA, instanciando os blocos Publish_1 (para comando do motor da esteira) e Subscribe_1 (para leitura do sensor).

Na segunda etapa, o processamento lógico por meio do FORTE migrou do ambiente virtual para a borda, passando a ser executado fisicamente no *hardware* do CLP Wago PFC200 (módulo principal 750-8212), como demonstra a Figura 1. Para embarcar o controle, realizou-se a compilação cruzada (*cross-compile*) no sistema Ubuntu via CMake e *toolchain* oficial da fabricante. Para garantir a compatibilidade do sistema com o *firmware* do CLP (v04.02.13-24), foi necessário realizar o *downgrade* das ferramentas do 4diac e da biblioteca open62541, transferindo os binários executáveis resultantes via rede pelo protocolo SSH.

Por fim, o executável FORTE foi recompilado para suportar o K-BUS, barramento interno proprietário da Wago responsável pela comunicação com os periféricos, utilizando bibliotecas extraídas via container Docker e a aplicação do bloco WAGO_MASTER combinado com validações empíricas no 4diac. No ensaio físico final, um botão mecânico na *protoboard*, alimentado por uma fonte de 24V, enviava o sinal ao módulo de entrada digital (750-1405), lido pelo bloco de entrada discreta (IX). A lógica interna processou o evento e disparou o comando para o bloco de saída discreta (QX), endereçado ao módulo de saída digital (750-1504), fechando o circuito e acendendo o LED embutido (Figura 2).

RESULTADOS

Os resultados demonstram a viabilidade de migrar de um sistema centralizado para uma arquitetura distribuída de borda, validada primeiro em ambiente virtual e depois fisicamente no controlador. A validação da lógica na etapa virtual confirmou a importância da simulação de sistemas ciberfísicos em IEC 61499 antes da aplicação física [7]. A comunicação inicial entre o 4diac IDE e o Factory I/O usava o protocolo Modbus, mas a migração para OPC UA se mostrou necessária por oferecer maior estabilidade, independência de plataforma e escalabilidade [3, 6], permanecendo como padrão definitivo mesmo após a migração para o CLP físico.

Durante a integração via OPC UA, os blocos padronizados Publish_1 e Subscribe_1 foram instanciados diretamente na raiz do servidor, exigindo expansão manual do limite de leitura do cliente, obstáculo que ilustra bem as dificuldades de integração de *software* discutidas em revisões recentes sobre CLPs orientados a eventos [5]. Ao avançar para o *hardware*, a heterogeneidade e a restrição computacional do CLP se tornaram pontos críticos, refletindo problemáticas já discutidas na literatura sobre computação de borda em dispositivos com poucos recursos [1, 8]. A necessidade de *downgrade* de ferramentas e de extrair bibliotecas via Docker para suportar o K-BUS, resultado da documentação fragmentada do conjunto 4diac, reforça as dificuldades atuais de padronização em integrações ciberfísicas complexas [5, 9]. Como a documentação elétrica do CLP não estava clara e a integração com o 4diac consiste em uma tarefa mais complexa, a plataforma CODESYS foi utilizada inicialmente para diagnosticar as conexões elétricas, devido a sua simplicidade.

Superadas essas etapas, o ensaio físico final funcionou de forma autônoma: o sinal do botão foi processado corretamente pela lógica distribuída, acionando o LED em tempo real. Esse resultado confirma a maturidade do modelo orientado a eventos da IEC 61499 [3, 9] e mostra que a execução da lógica diretamente no CLP de borda, sem processamento em nuvem, reduz significativamente latências de rede e garante segurança nas decisões de controle local, de forma consistente com trabalhos que avaliam a distribuição de blocos de função via OPC UA em ambientes reais e obtêm latências de rede inferiores a 15 ms [6], além de propostas de execução determinística e priorizada de eventos para aplicações de borda em IEC 61499 [10].

CONSIDERAÇÕES FINAIS

O trabalho cumpriu o objetivo de desenvolver e validar uma arquitetura de controle e monitoramento distribuído para manufatura sob o paradigma de *Edge Computing*. Embora a implementação física tenha se limitado a um único controlador como prova de conceito, isso não prejudicou a validação do modelo orientado a eventos. Pelo contrário, a transição bem-sucedida da simulação virtual para a operação no Wago PFC200 comprovou, em escala reduzida, os ganhos de autonomia e responsividade característicos das arquiteturas descentralizadas na borda. Apesar dos desafios de integração de infraestruturas de *software* multiplataforma, como o *cross-compile* de bibliotecas legadas e a configuração do FORTE para suportar o K-BUS, o sistema demonstrou flexibilidade para contornar essas barreiras, e o ensaio final confirmou a integração entre sensoriamento, processamento lógico e atuação física. Como trabalhos futuros, recomenda-se expandir o sistema com mais módulos de entrada e saída e controladores para testar dinâmicas de controle distribuído mais complexas; investigar mecanismos de reconfiguração dinâmica em tempo de execução e blocos de função seguros para mitigar riscos ciberfísicos; além de avaliar a incorporação de modelos leves de Inteligência Artificial nos dispositivos de controle para otimização preditiva.

Palavras-chave: automação; edge computing; iec 61499; opc ua; sistemas ciberfísicos.

ANEXOS

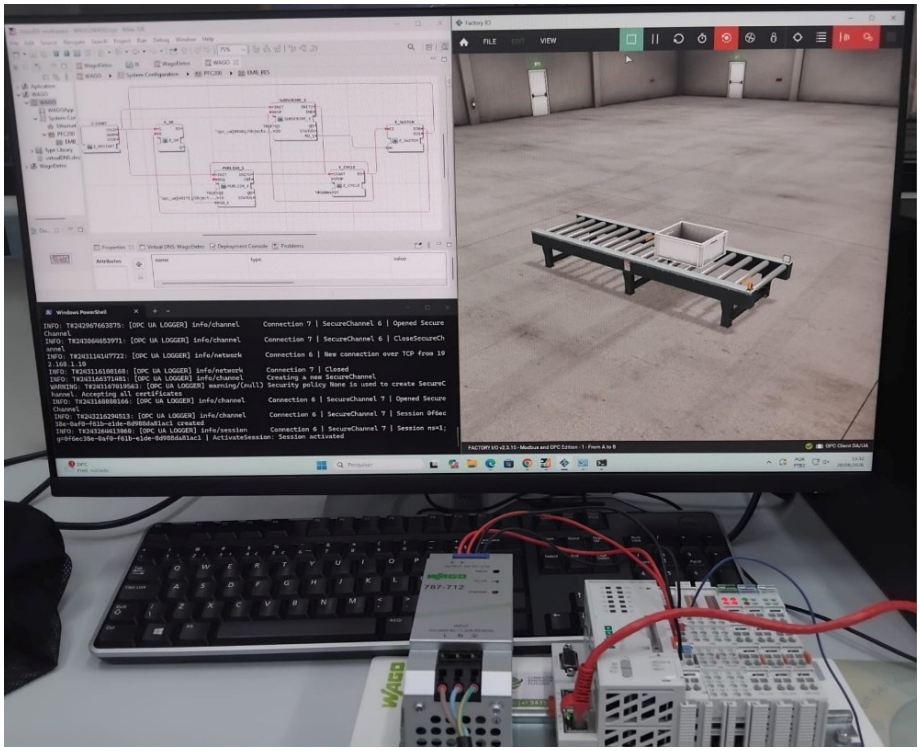


Figura 1. *Planta no Factory I/O (à direita), modelagem no 4diac IDE (superior esquerdo) e 4diac FORTE no CLP via terminal SSH (inferior esquerdo), com comunicação via OPC UA.*

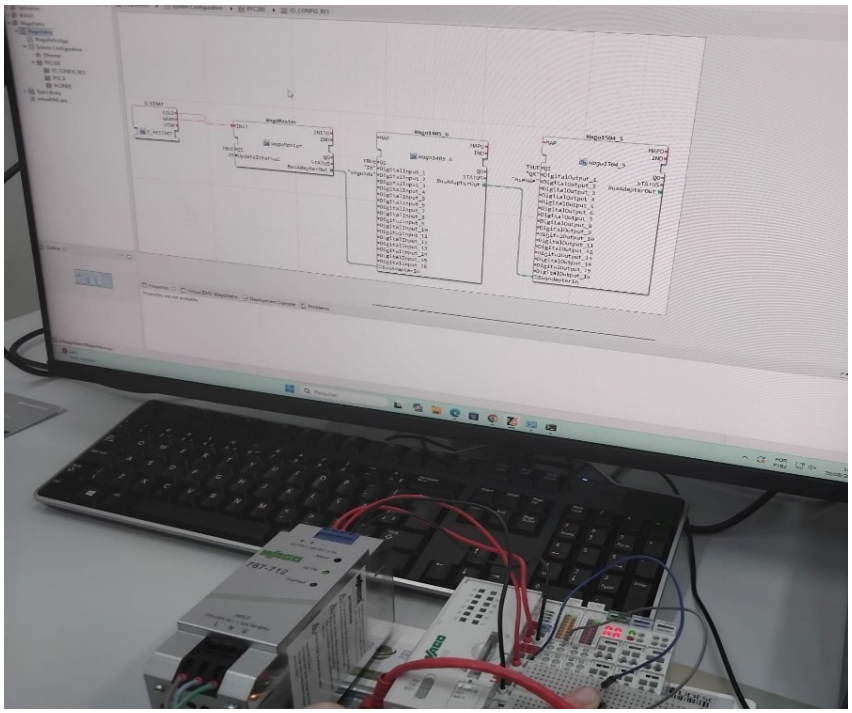


Figura 2. *Blocos de função WAGO no 4diac IDE, botão sendo apertado na protoboard e led acendendo nos módulos de entrada digital (laranja) e saída digital (vermelho).*

REFERÊNCIAS BIBLIOGRÁFICAS

[1] MOR, N. Edge Computing. **acmqueue - Research for Practice**, 2018.
[2] HUA, H. et al. Edge Computing with Artificial Intelligence: A Machine Learning Perspective. **ACM Computing Surveys**, v. 55, n. 9, 2023.
[3] ZOITL, A.; LEWIS, R. **Modelling Control Systems Using IEC 61499**. 2. ed. London: The Institution of Engineering and Technology (IET), 2014.
[4] PINTO, L. I.; LEAL, A. B.; ROSSO JR, R. S. U. Safe Dynamic Reconfiguration Through Supervisory Control in IEC 61499 Compliant Systems. In: **IEEE**, 2017.
[5] FOLETTI, M. et al. Implementation and System Architecture Challenges for Event-Based Programmable Logic Controllers - A Literature Review. **Procedia CIRP**, v. 130, p. 1529-1536, 2024.
[6] RECH, R. P. et al. Distributed System with OPC UA and Industrial Internet for IEC 61499 Function Blocks. In: 2023 15th IEEE International Conference on Industry Applications (INDUSCON). **Anais [...]**. [S. l.: s. n.], 2023. p. 1468-1473.
[7] PEARCE, H. A. et al. Simulation of Cyber-Physical Systems using IEC61499. In: **MEMOCODE 17 - 15th ACM-IEEE International Conference on Formal Methods and Models for System Design**, 2017.
[8] CAPROLU, M. et al. Edge Computing Perspectives: Architectures, Technologies, and Open Security Issues. In: **2019 IEEE International Conference on Edge Computing (EDGE)**, 2019.
[9] PINTO, L. I. et al. ICARU-FB: An IEC 61499 Compliant Multiplatform Software Infrastructure. **IEEE Transactions on Industrial Informatics**, v. 12, n. 3, p. 1074-1083, 2016.
[10] GAO, S. et al. Prioritized Deterministic Real-Time Execution Semantics for Industrial Edge Applications Based on IEC 61499 Event Types. **IEEE Transactions on Industrial Informatics**, v. 21, n. 11, p. 8550-8561, 2025.