



Universidade Federal de Ouro Preto  
Escola de Minas  
Engenharia de Controle e Automação



Matheus Amarante Carvalho

## **Modelagem e Implementação de Sistema de Automação para Processo Cervejeiro em Batelada Baseado na ANSI/ISA-88 com Integração OpenPLC–ScadaBR**

Ouro Preto, 2026

Matheus Amarante Carvalho

# **Modelagem e Implementação de Sistema de Automação para Processo Cervejeiro em Batelada Baseado na ANSI/ISA-88 com Integração OpenPLC–ScadaBR**

Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Engenharia de Controle e Automação.

Orientador: Profa. Dra. Karla Boaventura Pimenta Palmieri

Ouro Preto, 2026



MINISTÉRIO DA EDUCAÇÃO  
UNIVERSIDADE FEDERAL DE OURO PRETO  
REITORIA  
ESCOLA DE MINAS  
DEPARTAMENTO DE ENGENHARIA CONTROLE E  
AUTOMACAO



**FOLHA DE APROVAÇÃO**

**Matheus Amarante Carvalho**

**Modelagem e implementação de sistemas de automação para processo cervejeiro em Batelada  
baseado na ANSI/ISA - 88 com integração OpenPLC - ScadaBR**

Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharel em Engenharia de Controle e Automação

Aprovada em 02 de março de 2026

**Membros da banca**

Profa. Dra. Karla Boaventura Pimenta Palmieri - Orientadora - Universidade Federal de Ouro Preto

Profa. Dra. Luciana Gomes Castanheira - Universidade Federal de Ouro Preto

Prof. Dr. Danny Augusto Vieira Tonidandel - Universidade Federal de Ouro Preto

Karla Boaventura Pimenta Palmieri, orientadora do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 02/03/2026



Documento assinado eletronicamente por **Karla Boaventura Pimenta Palmieri, PROFESSOR DE MAGISTERIO SUPERIOR**, em 02/03/2026, às 15:09, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site [http://sei.ufop.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **1067847** e o código CRC **C255D22D**.

# Resumo

Este trabalho desenvolveu e avaliou a automação de um processo cervejeiro em batelada com base na ANSI/ISA-88, estruturando o sequenciamento por máquina de estados e implementando a lógica de controle no OpenPLC (IEC 61131-3) e a supervisão no ScadaBR, com comunicação via Modbus TCP. O sistema contemplou monitoramento em tempo real, registro histórico da temperatura e alarmes, apresentando execução determinística das etapas, coerência entre estados e acionamentos, além de sincronismo confiável entre controlador e supervisor. Conclui-se que a arquitetura proposta é tecnicamente viável e alinhada a práticas industriais, evidenciando o potencial de soluções *open source* para automação de processos em batelada.

**Palavras-chave:** automação industrial; processo em batelada; ISA-88; OpenPLC; supervisão SCADA.

# Abstract

This work developed and evaluated the automation of a batch brewing process based on the ANSI/ISA-88 standard, structuring the sequence through a state-machine approach and implementing the control logic in OpenPLC (IEC 61131-3) and the supervisory layer in ScadaBR, communicating via Modbus TCP. The system included real-time monitoring, historical temperature logging, and alarms, achieving deterministic sequencing, consistent actuator triggering across logical states, and reliable synchronization between the controller and the SCADA system. The results indicate that the proposed architecture is technically feasible and aligned with industrial practices, highlighting the potential of open-source solutions for batch process automation.

**Keywords:** industrial automation; batch process; ISA-88; OpenPLC; SCADA supervision.

# Lista de figuras

|                                                                                                                       |    |
|-----------------------------------------------------------------------------------------------------------------------|----|
| Figura 1 – Funcionamento e esquema básico de um CLP . . . . .                                                         | 12 |
| Figura 2 – Fluxograma simplificado do processo cervejeiro . . . . .                                                   | 20 |
| Figura 3 – Arquitetura hierárquica do sistema desenvolvido. . . . .                                                   | 24 |
| Figura 4 – Lógica de inicialização e transição do estado 0 para o estado 1. . . . .                                   | 27 |
| Figura 5 – Transição temporizada utilizando temporizador TON. . . . .                                                 | 28 |
| Figura 6 – Transição condicionada ao atingimento de 62°C. . . . .                                                     | 29 |
| Figura 7 – Manutenção temporizada do patamar térmico. . . . .                                                         | 29 |
| Figura 8 – Incremento discreto da variável TEMPERATURA quando o aquecedor<br>está ativo. . . . .                      | 30 |
| Figura 9 – Decremento discreto da variável TEMPERATURA quando o sistema<br>de resfriamento está ativo. . . . .        | 31 |
| Figura 10 – Configuração do ( <i>data source</i> ) Modbus TCP no ScadaBR para comu-<br>nicação com o OpenPLC. . . . . | 33 |
| Figura 11 – Tela supervisória desenvolvida no ScadaBR durante execução da batelada. . . . .                           | 35 |
| Figura 12 – Curva térmica registrada durante execução completa do ciclo produtivo. . . . .                            | 35 |
| Figura 13 – Correlação entre acionamento do aquecedor e evolução da temperatura. . . . .                              | 36 |
| Figura 14 – Registro histórico de eventos configurados no supervisório. . . . .                                       | 37 |

# Lista de tabelas

|                                                                                            |    |
|--------------------------------------------------------------------------------------------|----|
| Tabela 1 – Correspondência entre estados e fases do processo . . . . .                     | 26 |
| Tabela 2 – Mapeamento das variáveis do OpenPLC para supervisão via Modbus<br>TCP . . . . . | 32 |
| Tabela 3 – Duração observada das principais etapas em escala reduzida . . . . .            | 36 |

# Lista de abreviaturas e siglas

|       |                                                  |
|-------|--------------------------------------------------|
| ANSI  | <i>American National Standards Institute</i>     |
| CLP   | Controlador Lógico Programável                   |
| CPU   | <i>Central Processing Unit</i>                   |
| DNP3  | <i>Distributed Network Protocol 3</i>            |
| FBD   | <i>Function Block Diagram</i>                    |
| GE    | <i>Greater or Equal</i> (maior ou igual)         |
| IEC   | <i>International Electrotechnical Commission</i> |
| IHM   | Interface Homem-Máquina                          |
| IoT   | <i>Internet of Things</i>                        |
| ISA   | <i>International Society of Automation</i>       |
| MQTT  | <i>Message Queuing Telemetry Transport</i>       |
| MTU   | <i>Master Terminal Unit</i>                      |
| OPC   | <i>Open Platform Communications</i>              |
| PID   | Controlador Proporcional–Integral–Derivativo     |
| RTU   | <i>Remote Terminal Unit</i>                      |
| SCADA | <i>Supervisory Control and Data Acquisition</i>  |
| SFC   | <i>Sequential Function Chart</i>                 |
| ST    | <i>Structured Text</i>                           |
| TCP   | <i>Transmission Control Protocol</i>             |

# Sumário

|            |                                                                      |           |
|------------|----------------------------------------------------------------------|-----------|
| <b>1</b>   | <b>INTRODUÇÃO</b>                                                    | <b>9</b>  |
| <b>1.1</b> | <b>Objetivos</b>                                                     | <b>9</b>  |
| <b>1.2</b> | <b>Metodologia</b>                                                   | <b>10</b> |
| <b>1.3</b> | <b>Estrutura do Trabalho</b>                                         | <b>10</b> |
| <b>2</b>   | <b>REVISÃO DE LITERATURA</b>                                         | <b>11</b> |
| <b>2.1</b> | <b>Automação e Controle em Batelada</b>                              | <b>11</b> |
| <b>2.2</b> | <b>Controlador Lógico Programável (CLP)</b>                          | <b>11</b> |
| 2.2.1      | Arquitetura do CLP                                                   | 11        |
| 2.2.2      | Ciclo de Varredura <i>Scan Cycle</i>                                 | 12        |
| 2.2.3      | Confiabilidade e Aplicabilidade Industrial                           | 12        |
| 2.2.4      | Aplicações do CLP em Processos Industriais                           | 13        |
| 2.2.5      | Linguagens de Programação e Norma IEC 61131-3                        | 13        |
| 2.2.6      | OpenPLC como Plataforma Open Source                                  | 14        |
| <b>2.3</b> | <b>Sistemas Supervisórios (SCADA)</b>                                | <b>14</b> |
| 2.3.1      | Conceito de Supervisão e Aquisição de Dados                          | 14        |
| 2.3.2      | Arquitetura e Componentes de um Sistema SCADA                        | 15        |
| 2.3.3      | Função do Operador no Processo                                       | 15        |
| 2.3.4      | Integração entre CLP e SCADA                                         | 15        |
| 2.3.5      | Interfaces Homem-Máquina e Gestão de Alarmes                         | 16        |
| 2.3.6      | Aplicação do ScadaBR como Plataforma Open Source                     | 16        |
| <b>2.4</b> | <b>Instrumentação para Processos em Batelada</b>                     | <b>17</b> |
| 2.4.1      | Sensores e Variáveis de Processo                                     | 17        |
| 2.4.2      | Atuadores e Elementos Finais de Controle                             | 17        |
| 2.4.3      | Tratamento de Sinais e Confiabilidade                                | 18        |
| <b>2.5</b> | <b>Processos em Batelada na Automação Industrial</b>                 | <b>18</b> |
| 2.5.1      | Evolução Histórica dos Processos em Batelada                         | 18        |
| 2.5.2      | Conceito e Características dos Processos em Batelada                 | 18        |
| 2.5.3      | Estrutura Funcional do Processo Cervejeiro como Processo em Batelada | 19        |
| 2.5.4      | Sequenciamento por Etapas na Automação                               | 20        |
| 2.5.5      | Controle Baseado em Tempo versus Controle Baseado em Eventos         | 21        |
| 2.5.6      | Intertravamentos e Segurança Operacional                             | 21        |
| 2.5.7      | Complicações Típicas em Processos em Batelada                        | 21        |
| 2.5.7.1    | Falhas de Instrumentação                                             | 22        |
| 2.5.7.2    | Desvios Dinâmicos e Inércia Térmica                                  | 22        |

|            |                                                                                                 |           |
|------------|-------------------------------------------------------------------------------------------------|-----------|
| 2.5.7.3    | Problemas de Sequenciamento e Erro Operacional . . . . .                                        | 22        |
| 2.5.7.4    | Impacto na Qualidade e Repetibilidade . . . . .                                                 | 22        |
| <b>2.6</b> | <b>Síntese e Relação com o Trabalho Proposto . . . . .</b>                                      | <b>22</b> |
| <b>3</b>   | <b>MODELAGEM, IMPLEMENTAÇÃO E SUPERVISÃO DO PRO-<br/>CESSO CERVEJEIRO EM BATELADA . . . . .</b> | <b>24</b> |
| <b>3.1</b> | <b>Arquitetura Geral do Sistema . . . . .</b>                                                   | <b>24</b> |
| <b>3.2</b> | <b>Modelagem Procedural segundo a ANSI/ISA-88 . . . . .</b>                                     | <b>26</b> |
| <b>3.3</b> | <b>Implementação da Lógica de Controle em Ladder . . . . .</b>                                  | <b>27</b> |
| 3.3.1      | Inicialização e Transição de Estados . . . . .                                                  | 27        |
| 3.3.2      | Transições Temporizadas . . . . .                                                               | 27        |
| 3.3.3      | Transições Baseadas em Temperatura . . . . .                                                    | 28        |
| 3.3.4      | Manutenção de Patamares Térmicos . . . . .                                                      | 29        |
| 3.3.5      | Modelo Discreto de Simulação Térmica . . . . .                                                  | 29        |
| <b>3.4</b> | <b>Comunicação Modbus TCP entre OpenPLC e ScadaBR . . . . .</b>                                 | <b>31</b> |
| <b>3.5</b> | <b>Supervisão e Monitoramento . . . . .</b>                                                     | <b>33</b> |
| <b>4</b>   | <b>RESULTADOS . . . . .</b>                                                                     | <b>34</b> |
| <b>4.1</b> | <b>Validação do Sequenciamento da Máquina de Estados . . . . .</b>                              | <b>34</b> |
| <b>4.2</b> | <b>Análise Quantitativa da Evolução Térmica . . . . .</b>                                       | <b>35</b> |
| <b>4.3</b> | <b>Correlação entre Acionamento e Evolução Térmica . . . . .</b>                                | <b>35</b> |
| <b>4.4</b> | <b>Análise Temporal das Etapas . . . . .</b>                                                    | <b>36</b> |
| <b>4.5</b> | <b>Validação da Comunicação Modbus TCP . . . . .</b>                                            | <b>36</b> |
| <b>4.6</b> | <b>Avaliação do Sistema de Alarmes . . . . .</b>                                                | <b>36</b> |
| <b>5</b>   | <b>CONSIDERAÇÕES FINAIS . . . . .</b>                                                           | <b>38</b> |
|            | <b>Referências . . . . .</b>                                                                    | <b>39</b> |

# 1 Introdução

A automação industrial tem evoluído de forma significativa nas últimas décadas, impulsionada pelo avanço da eletrônica, da instrumentação e das tecnologias de controle digital. A substituição de sistemas eletromecânicos por controladores programáveis permitiu maior flexibilidade, confiabilidade e integração entre diferentes camadas do processo produtivo. Nesse contexto, destacam-se os processos organizados em batelada, amplamente utilizados nas indústrias alimentícia, farmacêutica e química, caracterizados por execução sequencial orientada por receita e comportamento predominantemente transitório.

Diferentemente de processos contínuos, nos quais o objetivo principal é manter variáveis em regime permanente, os processos em batelada exigem organização estruturada por estados e transições condicionais. Essa característica demanda integração entre controle regulatório e controle lógico sequencial, tornando necessária a utilização de arquiteturas capazes de coordenar múltiplas etapas de forma segura e repetível.

A norma ANSI/ISA-88 estabelece diretrizes para modelagem e organização de processos em batelada, propondo separação entre modelo físico e modelo procedural, além de padronizar terminologias e estruturas hierárquicas. Sua aplicação contribui para modularização do controle, reutilização de código e maior clareza na implementação de receitas produtivas.

Paralelamente, observa-se crescente interesse na utilização de plataformas *open source* em automação industrial, especialmente em ambientes acadêmicos e experimentais. Ferramentas como OpenPLC e ScadaBR permitem implementação de arquiteturas compatíveis com padrões industriais, incluindo suporte à norma IEC 61131-3 e protocolos de comunicação amplamente difundidos, como Modbus TCP, sem dependência de soluções proprietárias.

Nesse cenário, surge a seguinte questão de pesquisa: é tecnicamente viável implementar um sistema estruturado de automação para um processo cervejeiro em batelada, baseado na ANSI/ISA-88, utilizando exclusivamente plataformas *open source*, mantendo coerência arquitetural com práticas industriais?

## 1.1 Objetivos

### Objetivo Geral

Desenvolver e avaliar a implementação de um sistema de automação aplicado a um processo cervejeiro em batelada, estruturado segundo a norma ANSI/ISA-88 e

implementado por meio das plataformas OpenPLC e ScadaBR.

## Objetivos Específicos

- Modelar o processo cervejeiro conforme princípios da ANSI/ISA-88;
- Implementar a lógica sequencial em linguagem compatível com a IEC 61131-3 no OpenPLC;
- Configurar comunicação Modbus TCP entre controlador e sistema supervisorio;
- Desenvolver interface gráfica no ScadaBR para monitoramento do processo;
- Implementar registro histórico e sistema de alarmes;
- Avaliar o comportamento sequencial e a evolução térmica do sistema simulado.

## 1.2 Metodologia

A metodologia adotada envolveu modelagem procedural do processo cervejeiro, implementação da lógica sequencial no OpenPLC e integração com o ScadaBR via protocolo Modbus TCP. Foram realizadas simulações completas da batelada, com registro histórico das variáveis e análise quantitativa da evolução térmica. A validação concentrou-se na verificação do sequenciamento, sincronismo entre controle e supervisão e funcionamento dos mecanismos de alarme.

## 1.3 Estrutura do Trabalho

O Capítulo 2 apresenta a revisão de literatura, abordando fundamentos de automação industrial, controle de processos, Controladores Lógicos Programáveis, sistemas SCADA, instrumentação e processos em batelada. O Capítulo 3 descreve a modelagem, implementação e supervisão do processo cervejeiro em batelada. O Capítulo 4 apresenta os resultados obtidos e sua análise. Por fim, o Capítulo 5 apresenta as considerações finais e sugestões para trabalhos futuros.

## 2 Revisão de Literatura

Este capítulo apresenta a fundamentação teórica que sustenta o desenvolvimento do presente trabalho. São discutidos os principais conceitos relacionados à automação industrial, controle de processos e aplicação de Controladores Lógicos Programáveis (CLPs), estabelecendo o diálogo entre a literatura especializada e a proposta desenvolvida.

### 2.1 Automação e Controle em Batelada

A automação industrial moderna baseia-se na integração coordenada entre sensores, atuadores e sistemas de controle para operar processos de forma segura e repetível. Em processos em batelada, caracterizados pela execução sequencial de etapas definidas em receita com comportamento predominantemente transitório, o controle regulatório atua em conjunto com o controle lógico sequencial (FRANCHI, 2011; FOGLIATTO; NIANG, 2008). O Controlador Lógico Programável (CLP) atua como o núcleo dessa arquitetura, substituindo antigos painéis de relés eletromecânicos por lógicas de software flexíveis (BOLTON, 2015). Sua capacidade de executar ciclos de varredura determinísticos e implementar intertravamentos torna o CLP ideal para coordenar as transições de estados e temporizações inerentes aos processos industriais em batelada.

### 2.2 Controlador Lógico Programável (CLP)

#### 2.2.1 Arquitetura do CLP

De acordo com (BOLTON, 2015), o CLP apresenta arquitetura modular composta por Unidade Central de Processamento (CPU), módulos de entrada e saída, memória, fonte de alimentação e interfaces de comunicação.

A Figura 1 apresenta o esquema básico de arquitetura de um CLP, destacando a CPU, a memória, o barramento interno e os módulos de entrada e saída responsáveis pela interface com transdutores e atuadores.

A CPU executa o programa de controle, processa as entradas e atualiza as saídas. Os módulos de entrada realizam o condicionamento dos sinais provenientes de sensores industriais, enquanto os módulos de saída comandam atuadores como válvulas e motores.

A modularidade permite expansão do sistema conforme as necessidades do processo, característica que contribui para sua ampla aplicabilidade industrial.

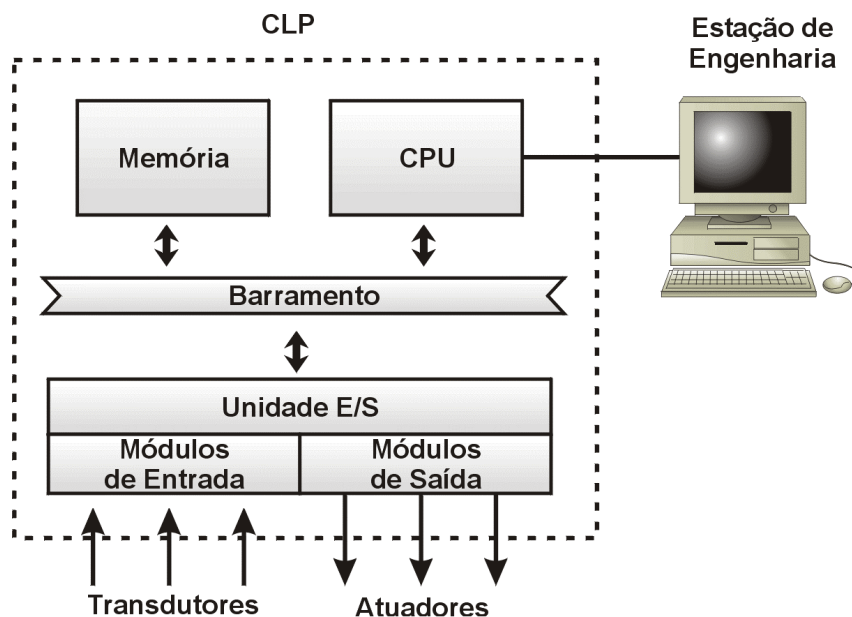


Figura 1 – Funcionamento e esquema básico de um CLP

Fonte: (DESMARAIS, 2008).

### 2.2.2 Ciclo de Varredura *Scan Cycle*

O funcionamento do CLP baseia-se no ciclo de varredura, processo cíclico e determinístico no qual o controlador executa repetidamente três etapas principais: leitura das entradas, execução do programa e atualização das saídas (BOLTON, 2015).

Durante a leitura, os estados físicos das entradas são copiados para uma imagem interna de memória. O programa é então executado utilizando essa imagem como referência. Ao final do processamento, os resultados são transferidos para a imagem de saída, atualizando fisicamente os dispositivos de campo.

A previsibilidade temporal do ciclo é essencial para aplicações industriais que exigem comportamento determinístico, especialmente em processos sequenciais e em batelada.

### 2.2.3 Confiabilidade e Aplicabilidade Industrial

Os CLPs foram concebidos para operar em ambientes industriais severos, apresentando imunidade a ruídos elétricos, resistência mecânica e estabilidade operacional. Incorporam mecanismos internos de segurança, como *watchdog timers*, responsáveis por monitorar o funcionamento da CPU e reiniciar o sistema em caso de falha (BOLTON, 2015).

Em processos em batelada, o CLP desempenha papel central na organização da lógica de transição entre estados, garantindo que cada fase seja executada apenas quando as condições previamente estabelecidas forem atendidas. Essa característica o torna particularmente adequado para sistemas que exigem controle sequencial estruturado.

### 2.2.4 Aplicações do CLP em Processos Industriais

A aplicabilidade do CLP em processos industriais é evidenciada em estudos de diferentes segmentos.

([SOUSA; TAIRA; PARK, 2019](#)) apresentam a integração de um sistema de controle aplicado a um reator químico em batelada, no qual o CLP executa a lógica de intertravamento, sequenciamento e controle das operações. O estudo demonstra que, mesmo inserido em arquiteturas associadas à Indústria 4.0, o CLP permanece como elemento central da camada de controle, responsável pela execução determinística do processo.

De forma complementar, ([DESMARAIS, 2008](#)) descreve a automação de uma Unidade de Destilação em Batelada na indústria do petróleo. O trabalho evidencia a implementação da lógica de controle em linguagem padrão de CLP e sua integração com sistema supervisor. A estruturação por etapas operacionais, com uso de lógica Ladder para controle de válvulas e seleção de modos operacionais, exemplifica aplicação clássica do CLP em processos organizados por estados.

A análise desses estudos demonstra que o CLP constitui a base operacional do sistema de automação, assegurando execução determinística da lógica, implementação de intertravamentos, sequenciamento estruturado e integração com supervisão ([SOUSA; TAIRA; PARK, 2019](#); [DESMARAIS, 2008](#)).

### 2.2.5 Linguagens de Programação e Norma IEC 61131-3

A norma IEC 61131-3 estabelece a padronização das linguagens de programação para CLPs, promovendo portabilidade e organização dos programas de controle industrial.

Segundo ([GUIMARÃES, 2005](#)), a norma consolida cinco linguagens principais: *Ladder Diagram*, *Function Block Diagram*, *Instruction List*, *Structured Text* e *Sequential Function Chart*.

O *Ladder Diagram* é amplamente utilizado em lógicas discretas e intertravamentos. O *Function Block Diagram* favorece organização modular de funções. O *Structured Text* (SFC) permite maior flexibilidade algorítmica. O *Sequential Function Chart* é direcionado à representação de processos estruturados por etapas, sendo particularmente adequado a aplicações em batelada.

O SFC é composto por passos, transições e ações. Quando a condição associada a uma transição é satisfeita, ocorre a progressão para o próximo estado. Essa estrutura facilita a modelagem de processos organizados por fases sucessivas ([GUIMARÃES, 2005](#)).

## 2.2.6 OpenPLC como Plataforma Open Source

Com a evolução da Indústria 4.0, observa-se crescente demanda por soluções de automação que conciliem robustez industrial e flexibilidade de desenvolvimento.

Santana *et al.* (2024) descrevem a implementação de um CLP *open source* com suporte às linguagens da IEC 61131-3 por meio do software OpenPLC. O ambiente é composto por *Editor* e *Runtime*, sendo o código convertido para linguagem C pelo compilador MatIEC, permitindo execução em diferentes plataformas.

O modelo de execução utiliza *Image Tables*, mecanismo semelhante ao ciclo de operação de CLPs comerciais, no qual entradas são copiadas para memória interna antes da execução do programa (SANTANA *et al.*, 2024).

O sistema oferece suporte a protocolos industriais como Modbus e MQTT, ampliando a capacidade de integração com supervisão e aplicações baseadas em nuvem.

No presente trabalho, a adoção do OpenPLC justifica-se por sua compatibilidade com a IEC 61131-3, estrutura semelhante à de CLPs comerciais e viabilidade para aplicações didáticas e experimentais em ambiente *open source*.

## 2.3 Sistemas Supervisórios (SCADA)

### 2.3.1 Conceito de Supervisão e Aquisição de Dados

Os sistemas SCADA (*Supervisory Control and Data Acquisition*) constituem plataformas destinadas à supervisão e aquisição de dados de processos industriais distribuídos. Diferentemente dos controladores locais, cuja função principal é executar o controle direto sobre variáveis físicas, os sistemas SCADA operam em nível superior da arquitetura de automação, realizando monitoramento, registro de dados e interface com o operador.

De acordo com (BOYER, 2004), um sistema SCADA permite o monitoramento remoto de instalações industriais, consolidando informações provenientes de diferentes pontos do processo e disponibilizando-as ao operador por meio de interfaces gráficas. Essa estrutura possibilita centralização das informações operacionais e ampliação da capacidade de acompanhamento do sistema.

A supervisão envolve a observação contínua das variáveis de processo, enquanto a aquisição de dados refere-se à coleta sistemática dessas informações para armazenamento, análise e apoio à tomada de decisão. Assim, o SCADA não substitui o controle automático executado pelo CLP, mas complementa-o, fornecendo visibilidade, rastreabilidade e suporte operacional (BOYER, 2004).

### 2.3.2 Arquitetura e Componentes de um Sistema SCADA

Segundo [Boyer \(2004\)](#), a arquitetura tradicional de um sistema SCADA é composta por Interface Homem-Máquina (IHM), Unidade Terminal Mestra (MTU), Unidades Terminais Remotas (RTUs) e infraestrutura de comunicação.

A IHM é responsável pela visualização das variáveis, apresentação de alarmes e envio de comandos. A MTU atua como núcleo central do sistema, consolidando dados das unidades remotas. As RTUs realizam a aquisição direta das variáveis em campo, podendo estar conectadas a sensores, atuadores ou controladores locais.

A comunicação entre esses elementos é essencial para a troca de dados em tempo real entre campo e centro de supervisão. Essa estrutura reforça a separação entre a camada de controle e a camada de supervisão, aumentando confiabilidade e organização do sistema.

Do ponto de vista evolutivo, a arquitetura SCADA passou por diferentes gerações, incorporando redes distribuídas, protocolos padronizados e, mais recentemente, integração com IoT e computação em nuvem ([YADAV; PAUL, 2021](#)). Essa evolução amplia interoperabilidade, mas também introduz desafios relacionados à segurança cibernética.

### 2.3.3 Função do Operador no Processo

No contexto dos sistemas SCADA, o operador desempenha papel estratégico na supervisão das operações industriais. Conforme descrito por ([BOYER, 2004](#)), o sistema fornece visão consolidada do processo, permitindo acompanhamento de variáveis críticas, identificação de anomalias e execução de comandos quando necessário.

Entretanto, o controle contínuo permanece sob responsabilidade do controlador local. O SCADA atua como ferramenta de suporte à decisão, disponibilizando dados em tempo real, alarmes e registros históricos. Essa distinção entre controle automático e supervisão é fundamental para compreensão da arquitetura hierárquica da automação industrial.

### 2.3.4 Integração entre CLP e SCADA

A integração entre CLPs e sistemas SCADA constitui elemento central da automação industrial contemporânea. Enquanto o CLP executa lógicas determinísticas e malhas de controle, o SCADA consolida dados, armazena históricos e fornece interface ao operador.

Segundo ([YADAV; PAUL, 2021](#)), a arquitetura SCADA moderna inclui unidades remotas, unidade mestra, IHM, banco de dados histórico e rede de comunicação. Nesse contexto, os CLPs podem atuar como substitutos ou complementos das RTUs, realizando interface direta com sensores e atuadores.

A comunicação ocorre por meio de protocolos industriais padronizados, entre os quais se destacam Modbus, DNP3 e IEC 60870-5-101 (YADAV; PAUL, 2021). O Modbus é amplamente empregado em aplicações de menor porte e ambientes didáticos, devido à sua simplicidade e ampla compatibilidade.

No cenário brasileiro, (VIEIRA; REIS, 2025) indicam que a integração CLP-SCADA pode reduzir tempos de parada e aumentar eficiência operacional, decorrentes da maior visibilidade das variáveis e da gestão centralizada de alarmes. A evolução para arquiteturas baseadas em IoT e nuvem amplia a interoperabilidade, mas exige atenção à segurança cibernética e à modernização de sistemas legados (VIEIRA; REIS, 2025).

### 2.3.5 Interfaces Homem-Máquina e Gestão de Alarmes

A Interface Homem-Máquina constitui o meio de interação entre operador e sistema supervisor. Conforme definido na norma ANSI/ISA-101.01-2015, a IHM corresponde ao conjunto de hardware e software utilizado para monitorar e interagir com o sistema de controle (INTERNATIONAL SOCIETY OF AUTOMATION, 2015).

A norma estabelece ciclo de vida estruturado para desenvolvimento da interface, contemplando filosofia, guia de estilo, implementação e melhoria contínua. Recomenda-se clareza na apresentação das informações, consistência visual, uso criterioso de cores e organização hierárquica das telas.

A gestão de alarmes é regulamentada pela norma ANSI/ISA-18.2, que define alarme como condição anormal que requer ação do operador (INTERNATIONAL SOCIETY OF AUTOMATION, 2016). A norma recomenda classificação por prioridade, eliminação de alarmes redundantes e definição clara de ação associada.

### 2.3.6 Aplicação do ScadaBR como Plataforma Open Source

O ScadaBR é um sistema supervisor *open source* baseado em Java, executado sobre servidor de aplicações e acessível via navegador web (SCADABR..., 2010).

Sua arquitetura funcional organiza-se em três elementos principais: *Data Sources*, *Data Points* e interface de visualização. Os *Data Sources* representam os protocolos de comunicação com dispositivos de campo, como Modbus e OPC. Os *Data Points* correspondem às variáveis monitoradas ou controladas, podendo assumir natureza binária, numérica ou multiestado. Cada ponto armazena histórico temporal, possibilitando análise posterior.

O sistema permite atualização em tempo real, construção de telas gráficas, exibição de gráficos históricos e geração de relatórios. Possui mecanismo de detecção de eventos e classificação de alarmes em diferentes níveis de severidade, alinhando-se às recomendações da ISA-18.2 (INTERNATIONAL SOCIETY OF AUTOMATION, 2016).

A escolha do ScadaBR fundamenta-se em sua disponibilidade *open source*, suporte a protocolos industriais e possibilidade de simulação por meio de *Virtual Data Source*, permitindo aplicação prática dos conceitos teóricos discutidos anteriormente.

## 2.4 Instrumentação para Processos em Batelada

A implementação de sistemas automatizados em processos industriais depende diretamente da adequada seleção e integração de sensores e atuadores. Em processos em batelada, nos quais a execução ocorre por etapas sequenciais e condicionadas, a instrumentação assume papel estruturante, pois fornece as informações necessárias ao controlador lógico programável e executa as ações físicas determinadas pela lógica de controle.

### 2.4.1 Sensores e Variáveis de Processo

Os sensores têm como função converter grandezas físicas do processo em sinais elétricos compatíveis com o sistema de controle. Em processos cervejeiros, destacam-se como variáveis críticas a temperatura, o nível e, em determinadas configurações, a vazão.

A temperatura constitui a variável mais sensível durante as etapas de mosturação e fervura, pois influencia diretamente as reações enzimáticas e, conseqüentemente, as características sensoriais da cerveja (BRIGGS *et al.*, 2004). Pequenas variações térmicas podem alterar a eficiência da conversão enzimática, impactando corpo, teor alcoólico e perfil final do produto. Dessa forma, sensores térmicos devem apresentar precisão, estabilidade e adequada resposta dinâmica.

Além das variáveis analógicas, sensores discretos desempenham função relevante em sistemas sequenciais. Sensores de chama, chaves de nível mínimo e dispositivos de fim de curso são exemplos de entradas digitais que garantem permissivos operacionais e condições de segurança. Conforme discutido por (SILVEIRA; SANTOS, 1998), em sistemas de controle discreto o CLP toma decisões com base no estado lógico das entradas, tornando essencial a confiabilidade dessas medições para a integridade do processo.

### 2.4.2 Atuadores e Elementos Finais de Controle

Os atuadores constituem os elementos finais responsáveis por modificar o estado físico do processo. Em sistemas de produção cervejeira automatizados, destacam-se resistências elétricas ou queimadores para aquecimento, bombas para transferência de mosto e válvulas solenoides para controle de fluxo.

O acionamento desses dispositivos ocorre a partir de comandos enviados pelo CLP, por meio de saídas digitais ou analógicas. Em sistemas organizados por etapas, os

atuadores operam condicionados a permissivos lógicos e intertravamentos, assegurando que determinadas ações ocorram apenas quando requisitos previamente estabelecidos forem atendidos.

### 2.4.3 Tratamento de Sinais e Confiabilidade

Em ambientes industriais, os sinais provenientes de sensores podem sofrer interferências elétricas, ruídos e oscilações transitórias. O uso de técnicas de condicionamento e filtragem torna-se necessário para evitar decisões incorretas na lógica de controle.

No estudo de automação de microcervejaria analisado, foram empregados filtros de média móvel, banda morta e mecanismos de validação de comunicação com o objetivo de reduzir instabilidades nas leituras (ROCHA FERNANDES; FRANZEN, 2011).

Em síntese, a instrumentação representa a base física sobre a qual o controle sequencial é implementado. Sensores fornecem as informações para tomada de decisão, enquanto atuadores executam as ações determinadas pelo controlador. A qualidade desses elementos é determinante para repetibilidade, segurança e desempenho do processo cervejeiro automatizado.

## 2.5 Processos em Batelada na Automação Industrial

### 2.5.1 Evolução Histórica dos Processos em Batelada

Os processos em batelada possuem origem histórica fortemente associada às indústrias química, farmacêutica e alimentícia, setores nos quais a produção ocorre em quantidades finitas e segue uma sequência definida de operações até a obtenção do produto final. Em comparação a processos contínuos, a operação em batelada organiza a produção em ciclos, nos quais cada lote é conduzido conforme parâmetros previamente estabelecidos em uma receita (LIBERALESO, 2010).

Nesse contexto, a *International Society of Automation* desenvolveu a norma ANSI/ISA-88, publicada inicialmente em 1995, com o objetivo de padronizar terminologia e modelos para o controle de processos em batelada (INTERNATIONAL SOCIETY OF AUTOMATION, 1995). A norma estabeleceu a separação entre o modelo físico do processo e o modelo procedural, criando base conceitual para modularização do controle, maior reutilização de código e integração mais consistente entre controladores e supervisão.

### 2.5.2 Conceito e Características dos Processos em Batelada

Processos em batelada são caracterizados pela produção de quantidades finitas de produto por meio da execução sequencial de etapas definidas em uma receita, com

encerramento do ciclo antes do início de um novo lote ([INTERNATIONAL SOCIETY OF AUTOMATION, 1995](#); [LIBERALESO, 2010](#)).

Do ponto de vista da dinâmica de processos, sistemas em batelada apresentam comportamento predominantemente transitório, pois variáveis como temperatura, concentração, nível e pressão evoluem ao longo do tempo em função da progressão das etapas do processo ([SEBORG; EDGAR; MELLICHAMP, 2022](#)). Como consequência, o controle não se restringe à manutenção de um ponto de operação fixo, mas envolve conduzir o sistema ao longo de diferentes condições operacionais planejadas.

Outra característica relevante é a orientação por estados e condições de transição. Em processos contínuos, prevalece o controle regulatório associado à compensação de perturbações. Em processos em batelada, o controle inclui organização sequencial, na qual o avanço entre etapas depende do atendimento de critérios definidos, como tempo programado, alcance de variável de processo ou validação de permissivos operacionais ([LIBERALESO, 2010](#)).

### 2.5.3 Estrutura Funcional do Processo Cervejeiro como Processo em Batelada

A produção de cerveja constitui um exemplo clássico de processo em batelada, pois ocorre por meio da execução sequencial de etapas bem definidas, orientadas por receita e conduzidas em lotes. Cada produção corresponde a uma batelada cujos parâmetros de tempo, temperatura e composição são previamente estabelecidos conforme o estilo desejado ([BRIGGS \*et al.\*, 2004](#)).

De forma geral, o processo cervejeiro pode ser dividido em etapas como moagem do malte, mosturação, filtração (*lautering*), fervura do mosto, separação de *trub* (*whirlpool*), resfriamento e fermentação. Cada etapa apresenta objetivos tecnológicos específicos e condições operacionais próprias, compondo uma sequência dependente ([BRIGGS \*et al.\*, 2004](#)).

A sequência operacional simplificada do processo adotado neste trabalho é apresentada na Figura 2, evidenciando o encadeamento lógico das fases que compõem a batelada.



Figura 2 – Fluxograma simplificado do processo cervejeiro

Fonte: Elaborado pelo autor (2026).

Sob a perspectiva da ANSI/ISA-88, o processo pode ser modelado pela separação entre modelo físico e modelo procedural. No modelo físico, os equipamentos envolvidos, como tanques e sistemas de aquecimento e transferência, são organizados em unidades e módulos. No modelo procedural, as etapas do processo são estruturadas em procedimentos, operações e fases, permitindo implementação modular no controlador ([INTERNATIONAL SOCIETY OF AUTOMATION, 1995](#)).

A produção cervejeira é adotada neste trabalho como aplicação por reunir as características clássicas de um processo em batelada: execução orientada por receita, comportamento dinâmico transitório e organização estruturada em fases dependentes (como mosturação e fervura). Essa estrutura permite a aplicação direta da modelagem procedural proposta pela norma ANSI/ISA-88, viabilizando a simulação e o monitoramento integrado do controle sequencial utilizando plataformas de arquitetura aberta.

#### 2.5.4 Sequenciamento por Etapas na Automação

O controle de processos em batelada é estruturado por meio de sequenciamento de etapas, no qual o sistema evolui entre estados operacionais conforme condições previamente definidas. A ANSI/ISA-88 estabelece que o modelo procedural de um processo em batelada deve ser organizado em procedimentos, operações e fases, sendo a fase a menor unidade de execução responsável por uma ação específica do processo. As transições entre fases podem ocorrer por tempo, por condição de variável de processo ou por validação de permissivos ([INTERNATIONAL SOCIETY OF AUTOMATION, 1995](#)).

Na prática, esse modelo é implementado em CLPs por meio de lógica sequencial baseada em estados. No estudo de automação de microcervejaria analisado, a operação foi organizada como uma estrutura de estados, em que etapas como aquecimento, brasagem, fervura e resfriamento são tratadas individualmente, com critérios claros de início e término ([ROCHA FERNANDES; FRANZEN, 2011](#)).

### 2.5.5 Controle Baseado em Tempo versus Controle Baseado em Eventos

Em processos em batelada, as transições entre etapas podem ser estruturadas principalmente por critérios temporais ou por eventos. No controle baseado em tempo, temporizadores determinam a duração de uma fase, e a transição ocorre após o decurso de um intervalo parametrizado. Essa abordagem é comum quando a duração da operação é determinante para o resultado, como em etapas térmicas e reações que exigem permanência por período específico (SEBORG; EDGAR; MELLICHAMP, 2022).

No controle baseado em eventos, a transição ocorre quando uma condição específica é atendida, como atingir temperatura, nível ou pressão desejados. Essa abordagem é relevante em sistemas dinâmicos sujeitos a perturbações e variações de condições iniciais (SEBORG; EDGAR; MELLICHAMP, 2022).

No processo cervejeiro, ambas as estratégias aparecem de forma complementar. A manutenção de patamares térmicos pode ser delimitada por tempo, enquanto o avanço de etapa pode depender do alcance de uma condição de processo, como uma temperatura alvo (BRIGGS *et al.*, 2004).

### 2.5.6 Intertravamentos e Segurança Operacional

Em sistemas automatizados, o controle lógico deve incorporar mecanismos de proteção que previnam condições inseguras de operação. Intertravamentos são estruturas lógicas que impedem a execução de determinada ação quando condições permissivas não são atendidas.

Conforme (SILVEIRA; SANTOS, 1998), em sistemas de controle discreto o CLP toma decisões com base nas entradas recebidas, executando saídas apenas quando condições definidas são satisfeitas. Essa lógica condicional fundamenta intertravamentos, pois permite bloquear ações potencialmente perigosas diante de estados indesejados.

Em aplicação cervejeira, intertravamentos típicos incluem bloqueio de aquecimento sem nível mínimo, impedimento de transferência sem confirmação de válvula aberta e geração de alarmes quando limites máximos de temperatura são excedidos. A integração entre CLP e supervisão reforça esse comportamento ao combinar bloqueio automático com notificação ao operador.

### 2.5.7 Complicações Típicas em Processos em Batelada

Embora processos em batelada apresentem flexibilidade produtiva e boa adequação a sistemas orientados por receita, sua natureza transitória e sequencial os torna sensíveis a falhas de instrumentação, desvios dinâmicos e erros de sequenciamento.

### 2.5.7.1 Falhas de Instrumentação

Sensores são a base de informação para tomada de decisão do controlador. Em processo cervejeiro, medições de temperatura são críticas em mosturação e fervura; variações podem alterar a atividade enzimática e a extração de compostos, impactando o perfil sensorial do produto (BRIGGS *et al.*, 2004).

### 2.5.7.2 Desvios Dinâmicos e Inércia Térmica

Processos em batelada apresentam variáveis evoluindo continuamente, o que pode resultar em atrasos de resposta, *overshoot* e oscilações diante de perturbações (SEBORG; EDGAR; MELLICHAMP, 2022). No processo cervejeiro, a inércia térmica do mosto pode produzir atraso entre acionamento do aquecimento e estabilização da temperatura.

### 2.5.7.3 Problemas de Sequenciamento e Erro Operacional

Como as etapas são interdependentes, erros de sincronização ou execução fora de ordem podem gerar perdas de matéria-prima e riscos operacionais. A máquina de estados implementada no controlador busca evitar inconsistências, mas falhas de parametrização, intervenções manuais inadequadas ou ausência de permissivos podem comprometer o processo (ROCHA FERNANDES; FRANZEN, 2011).

### 2.5.7.4 Impacto na Qualidade e Repetibilidade

A repetibilidade entre lotes é desafio relevante em produção cervejeira, pois variações pequenas de tempo e temperatura podem alterar teor alcoólico, perfil aromático e corpo da cerveja (BRIGGS *et al.*, 2004).

## 2.6 Síntese e Relação com o Trabalho Proposto

Os fundamentos discutidos ao longo deste capítulo indicam que a automação de processos em batelada requer integração consistente entre instrumentação, estratégias de controle e supervisão operacional. No cenário industrial, a consolidação de normas como a ANSI/ISA-88 reforça a importância da padronização e da modularização para o desenvolvimento de sistemas mais flexíveis, reutilizáveis e seguros (INTERNATIONAL SOCIETY OF AUTOMATION, 1995).

A análise do processo cervejeiro evidenciou sua aderência à lógica de execução típica de processos em batelada, com etapas operacionais definidas e parâmetros associados à receita. A adoção do OpenPLC como plataforma de controle lógico permite implementar a lógica sequencial em ambiente *open source*, com estrutura compatível com práticas industriais. De forma complementar, a utilização do ScadaBR como sistema supervisório

possibilita monitoramento, parametrização operacional, registro histórico e gestão de alarmes.

Dessa forma, os conceitos apresentados no Capítulo 2 convergem para uma arquitetura coerente com o desenvolvimento do protótipo proposto.

## 3 Modelagem, Implementação e Supervisão do Processo Cervejeiro em Batelada

### 3.1 Arquitetura Geral do Sistema

O sistema desenvolvido foi estruturado segundo uma arquitetura hierárquica típica de automação industrial, composta por três camadas funcionais: campo, controle e supervisão (BOYER, 2004). Essa organização permite separar aquisição de dados, execução determinística do controle e visualização operacional do processo.

A arquitetura implementada é apresentada na Figura 3, evidenciando o fluxo de informação desde a camada de campo até o nível supervísório, bem como o protocolo de comunicação empregado na integração dos sistemas.

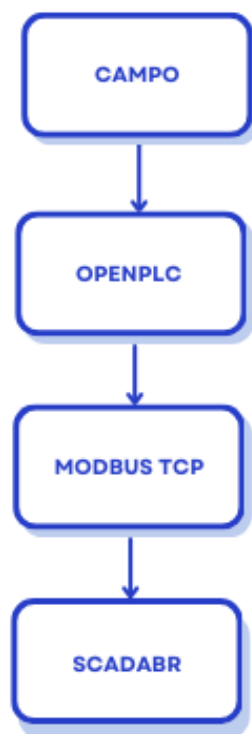


Figura 3 – Arquitetura hierárquica do sistema desenvolvido.

Fonte: Elaborado pelo autor (2026).

Na camada de campo encontra-se a variável de processo principal, representada pela temperatura do mosto (TEMPERATURA), modelada como variável analógica do tipo REAL. Essa camada corresponde ao nível físico do processo, no qual sensores e atuadores

interagem diretamente com o sistema produtivo.

A camada de controle foi implementada no OpenPLC, que opera como um *soft-PLC* executado por um *runtime* responsável por carregar, iniciar e manter a execução cíclica do programa de controle. A lógica foi desenvolvida em linguagem Ladder conforme a padronização estabelecida pela IEC 61131-3 (GUIMARÃES, 2005), garantindo aderência às práticas consolidadas de programação de controladores industriais. Nessa camada ocorre a execução cíclica do programa, leitura das entradas, processamento das regras de controle e atualização das saídas.

Embora nesta implementação tenha sido utilizado o OpenPLC executado em ambiente computacional convencional, a arquitetura proposta é flexível e poderia ser adaptada para execução em diferentes plataformas embarcadas, como microcontroladores industriais, dispositivos baseados em Raspberry Pi ou outros sistemas de baixo custo compatíveis com execução de *soft-PLCs*. O OpenPLC, por sua natureza aberta, permite implantação em múltiplos sistemas operacionais e hardwares, ampliando as possibilidades de aplicação. Além disso, a plataforma suporta diferentes linguagens de programação padronizadas pela IEC 61131-3, como Structured Text (ST), Function Block Diagram (FBD) e Sequential Function Chart (SFC), permitindo maior flexibilidade na modelagem da lógica de controle conforme a complexidade e os requisitos do processo.

O ambiente OpenPLC disponibiliza ainda uma interface acessível via navegador, permitindo acompanhar o estado do controlador, o programa ativo e os registros de execução (*runtime log*). O sistema também permite o envio e atualização de programas compilados, com verificação do log de compilação e posterior execução do projeto carregado (OPENPLC..., 2022). Esses recursos contribuem para maior rastreabilidade e transparência no processo de desenvolvimento e validação da lógica de controle.

A camada de supervisão foi realizada por meio da plataforma ScadaBR, sistema SCADA responsável pelo monitoramento em tempo real, registro histórico de dados e gerenciamento de alarmes. Trata-se de uma solução *open source* voltada ao desenvolvimento de IHM, amplamente empregada em aplicações industriais e acadêmicas (SCADABR..., 2010). Sua arquitetura permite a configuração de *data sources*, definição de *data points* e construção de telas de operação, possibilitando a visualização estruturada das variáveis de processo.

A comunicação entre o controlador e o sistema supervisor foi estruturada por meio do protocolo Modbus TCP (YADAV; PAUL, 2021), garantindo interoperabilidade e troca de dados em rede *Ethernet* industrial.

No contexto dessa arquitetura, o ScadaBR permite ajustar o período de atualização (*polling*) e parâmetros de transporte TCP, incluindo configurações com *keep-alive*, favorecendo estabilidade na leitura cíclica dos registradores. O sistema também disponibi-

liza mecanismos de diagnóstico, registrando falhas de comunicação por *timeout* ou erro de conexão, o que contribui para avaliação da disponibilidade do sistema supervisório (SCADABR. . . , 2010).

Essa organização encontra-se alinhada à separação funcional recomendada pela ANSI/ISA-88 para processos em batelada (INTERNATIONAL SOCIETY OF AUTOMATION, 1995), na qual o controle básico é desacoplado da camada supervisória, permitindo modularidade, escalabilidade e maior robustez operacional. Dessa forma, a integração OpenPLC–ScadaBR viabiliza uma implementação completa de controle e supervisão em arquitetura aberta, desde a execução do controle sequencial até a aquisição e visualização estruturada das variáveis do processo.

### 3.2 Modelagem Procedural segundo a ANSI/ISA-88

A modelagem do processo foi estruturada conforme os princípios da ANSI/ISA-88 (INTERNATIONAL SOCIETY OF AUTOMATION, 1995). No modelo procedural, o processo foi decomposto em fases discretas representadas pela variável inteira `Estado_Atual`, implementando uma máquina de estados finitos.

Cada valor assumido por essa variável corresponde a uma etapa tecnológica da produção cervejeira, conforme apresentado na Tabela 1.

Tabela 1 – Correspondência entre estados e fases do processo

| Estado | Fase                  |
|--------|-----------------------|
| 1      | Moagem                |
| 2      | Mistura               |
| 3      | Aquecimento até 62°C  |
| 4      | Patamar 62°C          |
| 5      | Aquecimento até 72°C  |
| 6      | Patamar 72°C          |
| 7      | Filtração             |
| 8      | Aquecimento até 100°C |
| 9      | Fervura               |
| 10     | Whirlpool             |
| 11     | Resfriamento até 20°C |
| 12     | Transferência         |
| 13     | Fermentação           |

Fonte: Elaborado pelo autor (2026).

Essa estrutura demonstra aderência ao conceito de fase como menor unidade executável do processo em batelada (INTERNATIONAL SOCIETY OF AUTOMATION, 1995).

### 3.3 Implementação da Lógica de Controle em Ladder

A lógica de controle foi implementada em linguagem Ladder utilizando estrutura de máquina de estados, temporizadores do tipo TON e blocos funcionais matemáticos. A variável `Estado_Atual` define a etapa ativa do processo, enquanto as transições ocorrem mediante condições temporais ou baseadas em variáveis físicas.

#### 3.3.1 Inicialização e Transição de Estados

O sistema inicia no estado 0 (*standby*). O comando **RESET** força a atribuição do valor zero à variável `Estado_Atual`. O comando **START** promove a transição para o estado 1, desde que o processo esteja em condição inicial.

A lógica implementada em linguagem Ladder para essa etapa é apresentada na Figura 4. Observa-se que o sinal **RESET** aciona um bloco **MOVE** responsável por escrever o valor 0 na variável `Estado_Atual`. De forma análoga, quando o comando **START** é acionado e a comparação lógica confirma que o estado atual é igual a zero, um novo bloco **MOVE** atribui o valor 1 à variável de estado, caracterizando a transição para a próxima fase do processo.

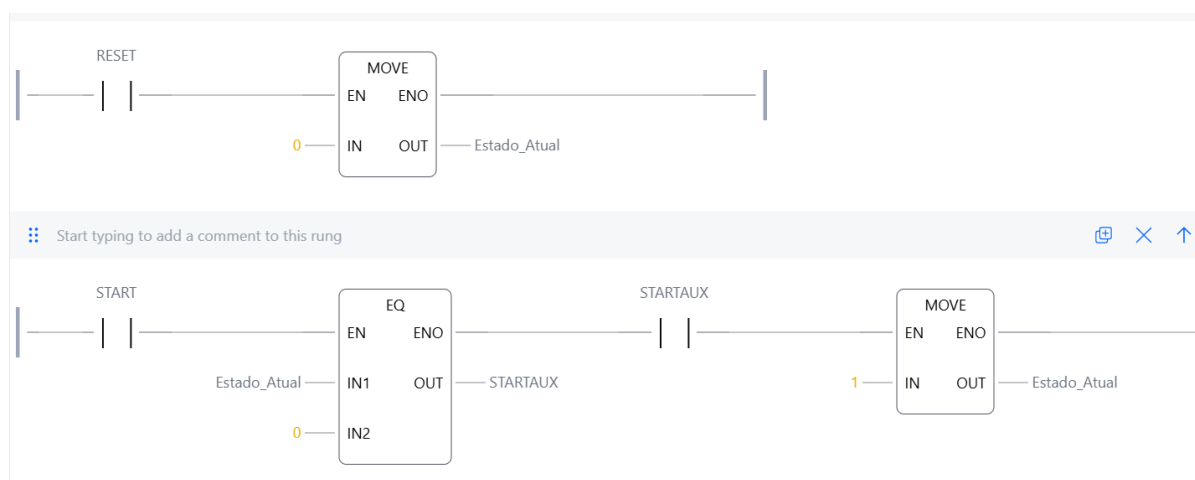


Figura 4 – Lógica de inicialização e transição do estado 0 para o estado 1.

Fonte: Elaborado pelo autor (2026).

Essa estrutura assegura reinicialização segura, evita transições indevidas e garante execução determinística do ciclo sequencial implementado no controlador.

#### 3.3.2 Transições Temporizadas

Etapas como moagem e mistura utilizam temporizadores do tipo TON para garantir permanência controlada na fase ativa do processo.

A lógica implementada para a etapa de moagem é apresentada na Figura 5. Inicialmente, um bloco de comparação (EQ) verifica se a variável `Estado_Atual` é igual a 1, condição que caracteriza a ativação da fase de moagem. Quando essa condição é verdadeira, a saída `MOAGEM` é energizada.

Com a etapa ativa, o temporizador do tipo TON é acionado, permanecendo ativo durante o tempo pré-programado. Ao término do intervalo especificado, a saída Q do temporizador é energizada, habilitando um bloco MOVE que escreve o valor 2 na variável `Estado_Atual`, promovendo a transição para a próxima fase do processo.

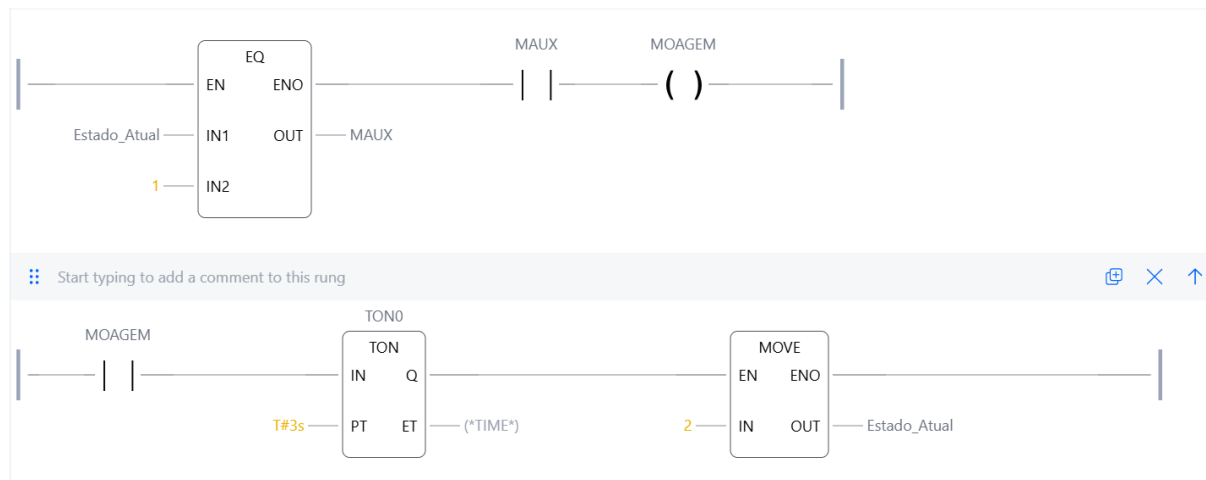


Figura 5 – Transição temporizada utilizando temporizador TON.

Fonte: Elaborado pelo autor (2026).

Essa estratégia assegura controle determinístico do tempo de permanência na etapa, impedindo avanço prematuro da sequência e garantindo coerência com a lógica de máquina de estados implementada no controlador.

### 3.3.3 Transições Baseadas em Temperatura

Nas etapas de aquecimento, a transição de estado ocorre mediante o alcance de uma condição física previamente estabelecida, neste caso, a temperatura de referência do processo.

A lógica implementada é apresentada na Figura 6. O bloco de comparação do tipo GE (*greater or equal*) verifica se a variável analógica `TEMPERATURA` é maior ou igual ao valor de referência (62°C). Quando essa condição é satisfeita, a variável auxiliar `AUX1` é energizada, habilitando o bloco MOVE responsável por escrever o valor 4 na variável `Estado_Atual`, promovendo a transição para a etapa subsequente do processo.



Figura 6 – Transição condicionada ao atingimento de 62°C.

Fonte: Elaborado pelo autor (2026).

Essa estratégia caracteriza uma transição orientada a evento físico, na qual a mudança de estado depende de uma condição de processo medida em tempo real, diferentemente das transições puramente temporizadas. Tal abordagem está alinhada ao conceito de término de fase baseado em critério de variável de processo.

### 3.3.4 Manutenção de Patamares Térmicos

Após atingir a temperatura alvo, o sistema permanece no patamar por intervalo temporizado, conforme ilustrado na Figura 7.

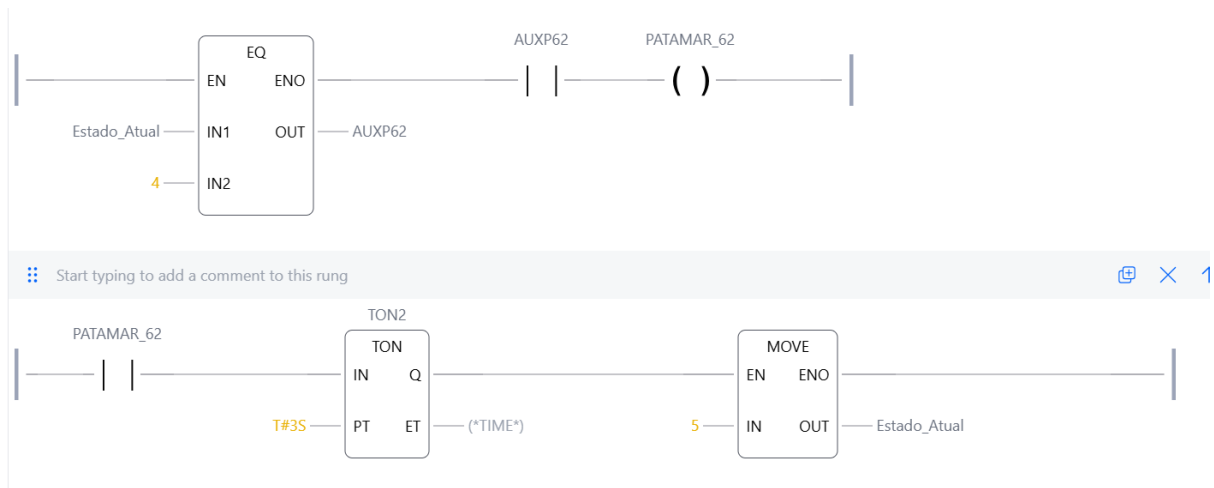


Figura 7 – Manutenção temporizada do patamar térmico.

Fonte: Elaborado pelo autor (2026).

### 3.3.5 Modelo Discreto de Simulação Térmica

Para fins didáticos e de validação da lógica de controle, foi implementado um modelo discreto simplificado de evolução térmica da variável **TEMPERATURA**. O modelo

baseia-se em incrementos e decrementos periódicos aplicados à variável, simulando o efeito de aquecimento e resfriamento ao longo do tempo.

A Figura 8 apresenta a lógica de incremento da temperatura. Um sinal periódico de 1 segundo (**Tick1s**) é utilizado como base temporal. Quando o aquecedor está ativo, um bloco **ADD** soma um valor fixo à variável **TEMPERATURA**, caracterizando aumento gradual e discreto da temperatura simulada.

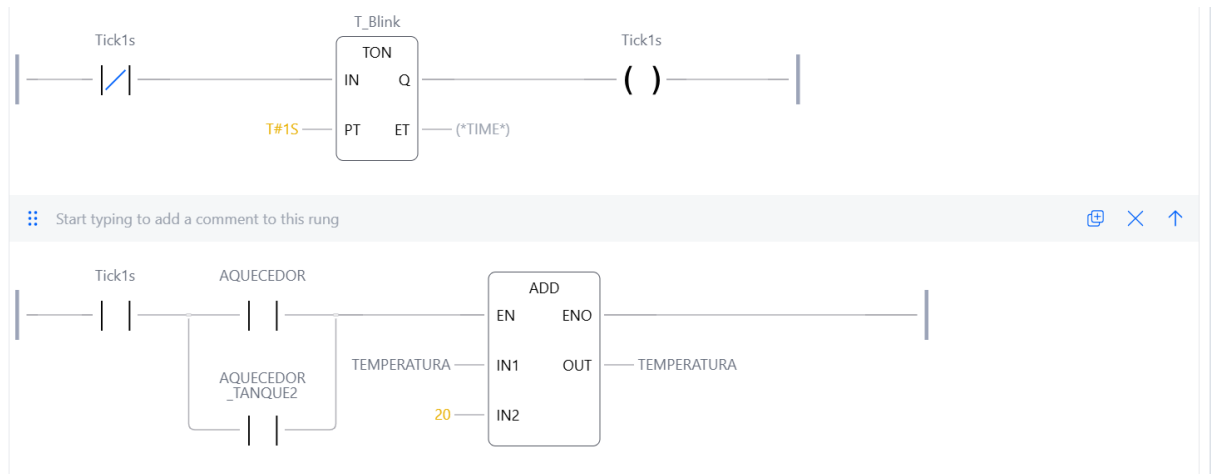


Figura 8 – Incremento discreto da variável **TEMPERATURA** quando o aquecedor está ativo.

Fonte: Elaborado pelo autor (2026).

De forma análoga, a Figura 9 apresenta a lógica de resfriamento. Quando o sistema de resfriamento (**COOLER**) está ativo, um bloco **SUB** subtrai um valor fixo da variável **TEMPERATURA** a cada pulso temporal, representando redução progressiva da temperatura.

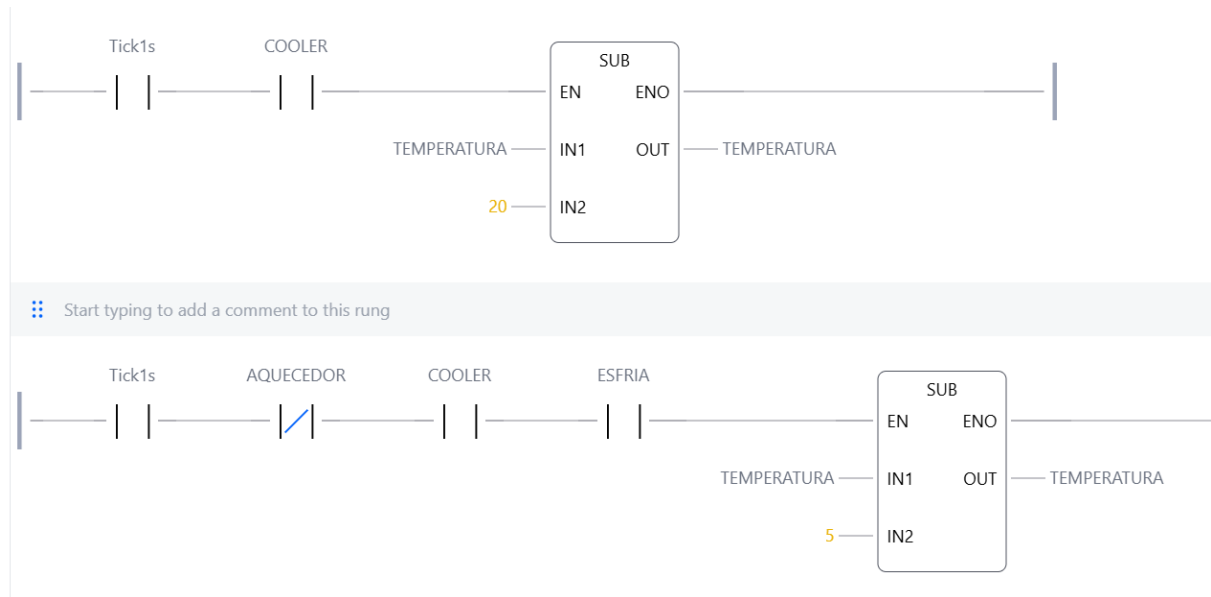


Figura 9 – Decremento discreto da variável TEMPERATURA quando o sistema de resfriamento está ativo.

Fonte: Elaborado pelo autor (2026).

Esse modelo caracteriza uma aproximação discreta de um sistema térmico de primeira ordem, permitindo testar transições condicionadas à temperatura sem necessidade de (*hardware*) físico. Embora simplificado, o modelo é suficiente para validar a lógica sequencial e os intertravamentos implementados no controlador.

### 3.4 Comunicação Modbus TCP entre OpenPLC e ScadaBR

As variáveis implementadas no OpenPLC foram explicitamente mapeadas para endereços físicos de saída, permitindo sua leitura e supervisão via protocolo Modbus TCP. A Tabela 2 apresenta o mapeamento completo das variáveis utilizadas no sistema, incluindo tipo de dado, endereço lógico e função operacional no processo.

Tabela 2 – Mapeamento das variáveis do OpenPLC para supervisão via Modbus TCP

| <b>Variável</b>   | <b>Tipo</b> | <b>Endereço</b> | <b>Função no Processo</b>                                         |
|-------------------|-------------|-----------------|-------------------------------------------------------------------|
| RESET             | BOOL        | %QX0.0          | Reinicialização do ciclo produtivo, retornando ao estado inicial. |
| START             | BOOL        | %QX0.1          | Comando de partida da batelada.                                   |
| MOAGEM            | BOOL        | %QX0.2          | Indica execução da etapa de moagem.                               |
| MISTURA           | BOOL        | %QX0.3          | Indica execução da etapa de mistura.                              |
| AQUECEDOR         | BOOL        | %QX0.4          | Acionamento do sistema de aquecimento do tanque principal.        |
| BOMBA             | BOOL        | %QX0.5          | Controle da bomba de transferência.                               |
| COOLER            | BOOL        | %QX0.6          | Acionamento do sistema de resfriamento.                           |
| FERMENTACAO       | BOOL        | %QX0.7          | Indicação da etapa de fermentação.                                |
| FINISHED          | BOOL        | %QX1.0          | Indica finalização completa do ciclo produtivo.                   |
| AQUECEDOR_TANQUE2 | BOOL        | %QX1.1          | Acionamento do aquecimento do segundo tanque.                     |
| Estado_Atual      | INT         | %QW0            | Variável inteira responsável pela máquina de estados do processo. |
| TEMPERATURA       | INT         | %QW2            | Variável numérica que representa a temperatura simulada do mosto. |

Fonte: Elaborado pelo autor (2026).

A integração entre o OpenPLC e o ScadaBR foi estruturada por meio do protocolo Modbus TCP, operando em arquitetura cliente-servidor. O OpenPLC atua como servidor Modbus, disponibilizando os registradores mapeados, enquanto o ScadaBR atua como cliente (mestre), realizando a leitura periódica dos endereços configurados.

A Figura 10 apresenta a configuração do data source Modbus TCP no ScadaBR, evidenciando parâmetros como endereço IP, porta de comunicação (1502), identificação do escravo (ID = 1) e mapeamento das bobinas e registradores holding correspondentes às variáveis do controlador.

| Nome         | Tipo de dado | Status | Escravo | Faixa                  | Offset (baseado em 0) |
|--------------|--------------|--------|---------|------------------------|-----------------------|
| AQUECEDOR    | Binário      |        | 1       | Bobina (Coil Status) 4 |                       |
| BOMBA        | Binário      |        | 1       | Bobina (Coil Status) 5 |                       |
| COOLER       | Binário      |        | 1       | Bobina (Coil Status) 6 |                       |
| Estado_Atual | Númerico     |        | 1       | Registrador holding 0  |                       |
| FERMENTACAO  | Binário      |        | 1       | Bobina (Coil Status) 7 |                       |
| FERVURA      | Binário      |        | 1       | Bobina (Coil Status) 9 |                       |
| FINISHED     | Binário      |        | 1       | Bobina (Coil Status) 8 |                       |
| MISTURA      | Binário      |        | 1       | Bobina (Coil Status) 3 |                       |
| MOAGEM       | Binário      |        | 1       | Bobina (Coil Status) 2 |                       |
| RESET        | Binário      |        | 1       | Bobina (Coil Status) 0 |                       |
| START        | Binário      |        | 1       | Bobina (Coil Status) 1 |                       |
| TEMPERATURA  | Númerico     |        | 1       | Registrador holding 2  |                       |

Figura 10 – Configuração do (*data source*) Modbus TCP no ScadaBR para comunicação com o OpenPLC.

Fonte: Elaborado pelo autor (2026).

A atualização cíclica das variáveis permite monitoramento em tempo real do estado do processo, incluindo variáveis discretas (bobinas) e variáveis numéricas (registradores holding). Essa estrutura garante interoperabilidade entre controlador e sistema supervisorio, assegurando consistência entre execução lógica e visualização operacional.

### 3.5 Supervisão e Monitoramento

O sistema supervisorio foi implementado na plataforma ScadaBR, permitindo visualização em tempo real das variáveis do processo, acompanhamento da evolução térmica, registro histórico e configuração de alarmes.

A interface desenvolvida possibilita interação com comandos de operação (START e RESET), visualização do estado atual da máquina de estados e monitoramento da variável de temperatura simulada. O sistema também registra dados históricos para posterior análise quantitativa, garantindo rastreabilidade da execução da batelada.

## 4 Resultados

Este capítulo apresenta a análise dos resultados obtidos a partir da implementação do sistema de automação do processo cervejeiro em batelada, desenvolvido integralmente em ambiente open source, utilizando o OpenPLC como controlador lógico programável e o ScadaBR como sistema supervisor.

A avaliação contempla a validação do sequenciamento da máquina de estados, a análise quantitativa da evolução térmica, a verificação da comunicação Modbus TCP e a funcionalidade do sistema de alarmes.

### 4.1 Validação do Sequenciamento da Máquina de Estados

A lógica de controle foi estruturada como máquina de estados, na qual a variável *Estado\_Atual* representa a fase ativa do processo.

Durante os testes realizados, observou-se progressão consistente entre as treze fases modeladas, respeitando rigorosamente os critérios de transição definidos na programação do CLP.

A ativação dos atuadores ocorreu de forma condicional e coerente com o estado ativo, não sendo identificadas transições indevidas ou inconsistências operacionais.

A representação gráfica do processo durante a execução da batelada pode ser observada na Figura 11, na qual se verifica a correspondência entre o estado lógico do controlador e a interface supervisória.

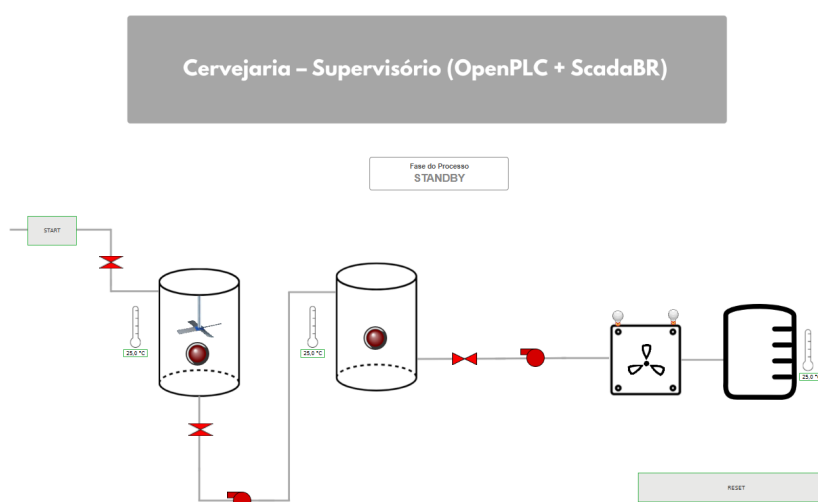


Figura 11 – Tela supervisória desenvolvida no ScadaBR durante execução da batelada.

Fonte: Elaborado pelo autor (2026).

## 4.2 Análise Quantitativa da Evolução Térmica

A variável **TEMPERATURA** foi registrada ao longo da execução completa da batelada, permitindo análise temporal do comportamento térmico do processo.

Conforme ilustrado na Figura 12, foram observadas rampas de aquecimento até 62 °C e 72 °C, permanência controlada nesses patamares, aquecimento até aproximadamente 100 °C na etapa de fervura e posterior resfriamento até cerca de 20 °C.

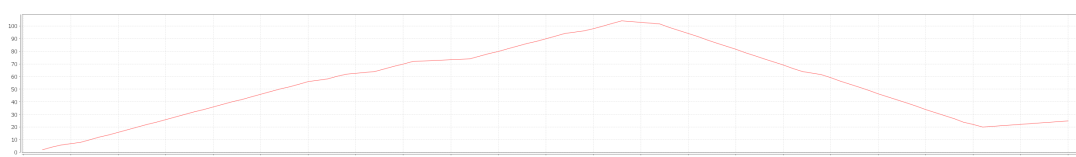


Figura 12 – Curva térmica registrada durante execução completa do ciclo produtivo.

Fonte: Elaborado pelo autor (2026).

## 4.3 Correlação entre Acionamento e Evolução Térmica

A relação entre o acionamento do aquecedor e a variação da temperatura pode ser observada na Figura 13, que apresenta a sobreposição do sinal do atuador com a curva térmica do processo.

Observa-se que o incremento térmico ocorre exclusivamente quando o aquecedor encontra-se energizado, validando a lógica condicional implementada no controlador.

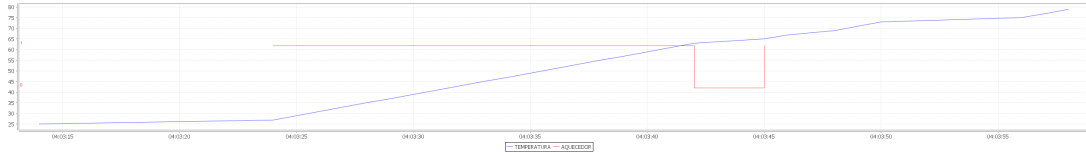


Figura 13 – Correlação entre acionamento do aquecedor e evolução da temperatura.

Fonte: Elaborado pelo autor (2026).

## 4.4 Análise Temporal das Etapas

Tabela 3 – Duração observada das principais etapas em escala reduzida

| Fase          | Temp. Alvo | Tempo Prog. (s) | Tempo Obs. (s) |
|---------------|------------|-----------------|----------------|
| Patamar 62 °C | 62         | 4,0             | 4,1            |
| Patamar 72 °C | 72         | 3,0             | 3,0            |
| Fervura       | 100        | 6,0             | 6,2            |
| Resfriamento  | 20         | —               | 35,4           |

As variações observadas foram inferiores a 3%, atribuídas ao ciclo de varredura do CLP e à discretização da simulação.

## 4.5 Validação da Comunicação Modbus TCP

A comunicação Modbus TCP operou com tempo de *polling* de 500 ms, garantindo atualização contínua do sistema supervisório sem inconsistências nos registradores.

Durante os testes realizados, não foram observadas perdas de comunicação, atrasos perceptíveis ou divergências entre os valores lidos no ScadaBR e os valores internos do OpenPLC, evidenciando sincronismo adequado entre as camadas de controle e supervisão.

## 4.6 Avaliação do Sistema de Alarmes

Com o objetivo de validar o sistema de gestão de eventos, foram simuladas condições anormais associadas à variável **TEMPERATURA**, incluindo ultrapassagem de limite superior configurado.

O registro desses eventos pode ser observado na Figura 14, que apresenta o histórico de alarmes gerados pelo supervisório durante a simulação. Observa-se a geração de evento classificado como crítico quando a temperatura ultrapassou 105 °C, confirmando a correta parametrização dos limites e o funcionamento adequado do mecanismo de detecção e registro de anomalias.



| Alarmes pendentes - 100 |                 |          |                              |                      |  |
|-------------------------|-----------------|----------|------------------------------|----------------------|--|
| Id                      | Nivel de alarme | Tempo    | Mensagens                    | Reconhecer todos     |  |
|                         |                 |          |                              | Tempo de inatividade |  |
| 1593                    |                 | 04:50:14 | TEMPERATURA excedeu 0,0 °C   | Ativo                |  |
| 1591                    |                 | 04:50:14 | TEMPERATURA excedeu 105,0 °C | Ativo                |  |

Figura 14 – Registro histórico de eventos configurados no supervisório.

Fonte: Elaborado pelo autor (2026).

O sistema permitiu ainda o reconhecimento e silenciamento dos alarmes pelo operador, garantindo rastreabilidade e controle das condições operacionais simuladas.

## 5 Considerações Finais

O trabalho demonstrou a viabilidade técnica da automação de um processo cervejeiro em batelada utilizando exclusivamente plataformas *open source*.

A máquina de estados garantiu execução determinística das treze fases modeladas, enquanto a integração via Modbus TCP assegurou sincronismo adequado entre controlador e supervisor.

Os resultados experimentais evidenciaram coerência entre lógica programada e comportamento térmico simulado, com variações temporais inferiores a 3%.

Apesar da utilização de um modelo térmico simplificado e da ausência de controle PID refinado, o sistema atendeu aos objetivos propostos e apresentou funcionamento consistente dentro dos limites definidos para o projeto.

Como continuidade do trabalho, recomenda-se aprofundar a análise da comunicação em rede, especialmente no que se refere ao tempo de resposta do protocolo Modbus TCP e sua influência na dinâmica do sistema supervisor. Sugere-se, em particular, a avaliação do impacto da variação do tempo de polling na atualização das variáveis monitoradas, analisando como diferentes taxas de leitura podem afetar a resposta da interface, a carga de comunicação e o desempenho geral do sistema. Avaliações relacionadas à latência, perda de pacotes e confiabilidade da transmissão podem contribuir para uma compreensão mais detalhada do desempenho da arquitetura cliente-servidor adotada.

Além disso, sugere-se a exploração de cenários de falhas em ambiente físico real, permitindo analisar o comportamento do sistema diante de perturbações, falhas de sensores ou atuadores e interrupções de comunicação.

Complementarmente, sugerem-se:

- Implementação de controle PID;
- Modelagem térmica baseada em balanço de energia;
- Parametrização dinâmica de receitas;
- Integração com banco de dados externo;
- Implementação em bancada física.

# Referências

BOLTON, William. *Programmable Logic Controllers*. 6. ed. Oxford: Newnes/Elsevier, 2015. Citado 4 vezes nas páginas 11, 12.

BOYER, Stuart A. *SCADA: Supervisory Control and Data Acquisition*. 3. ed. Research Triangle Park, NC: ISA – The Instrumentation, Systems, e Automation Society, 2004. ISBN 1-55617-877-8. Citado 5 vezes nas páginas 14, 15, 24.

BRIGGS, Dennis E. *et al. Brewing: Science and Practice*. Cambridge: Woodhead Publishing, 2004. ISBN 978-1-85573-490-6. Citado 6 vezes nas páginas 17, 19, 21, 22.

DESMARAIS, Milene Lagoas de Almeida. *Automação de uma unidade de destilação em batelada (UDB)*. 2008. Monografia (Trabalho de Conclusão de Curso (Engenharia Química)) – Universidade Federal do Rio de Janeiro, Rio de Janeiro. Citado 2 vezes nas páginas 12, 13.

FOGLIATTO, Flávio S.; NIANG, Ndèye. Controle multivariado de processos em batelada com duração variável. *Production*, SciELO Brasil, v. 18, n. 2, p. 240–259, 2008. Citado 1 vez na página 11.

FRANCHI, Claiton Moro. *Controle de Processos Industriais*. São Paulo: Érica, 2011. Citado 1 vez na página 11.

GUIMARÃES, Hugo Casati Ferreira. *Norma IEC 61131-3 para programação de controladores programáveis: estudo e aplicação*. 2005. Monografia (Projeto de Graduação) – Universidade Federal do Espírito Santo, Vitória. Citado 3 vezes nas páginas 13, 25.

INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-101.01-2015: Human Machine Interfaces for Process Automation Systems*. Research Triangle Park, NC: ISA, 2015. Approved 9 July 2015. Citado 1 vez na página 16.

INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-18.2-2016: Management of Alarm Systems for the Process Industries*. Research Triangle Park, NC: ISA, 2016. Citado 2 vezes na página 16.

INTERNATIONAL SOCIETY OF AUTOMATION. *ANSI/ISA-88.01-1995: Batch Control – Part 1: Models and Terminology*. Research Triangle Park, NC: ISA, 1995. Formerly ANSI/ISA-S88.01-1995. Citado 8 vezes nas páginas 18–20, 22, 26.

LIBERALESO, Alan. Norma ANSI/ISA S88 e a utilização no controle de bateladas. *InTech*, Brasil, n. 126, p. 24–36, out 2010. Citado 3 vezes nas páginas 18, 19.

OPENPLC PROJECT. *OpenPLC User Guide*. 2022. VisionFive2 OpenPLC Documentation. Citado 1 vez na página 25.

ROCHA FERNANDES, Anita Maria da; FRANZEN, Tiago Alexandre. Automação e Controle em uma Micro Cervejaria Artesanal. *In: ANAIS do VIII Simpósio de Excelência em Gestão e Tecnologia (SEGeT)*. Resende, RJ, 2011. Citado 3 vezes nas páginas 18, 20, 22.

SANTANA, Joao Vitor R. *et al.* Controlador lógico programável open source para aplicações de IIoT. *In: ANAIS do Congresso Brasileiro de Automática (CBA 2024)*. Belo Horizonte, MG, 2024. Citado 2 vezes na página 14.

SCADABR. *ScadaBR 0.7: Sistema Open-Source para Supervisão e Controle – Manual do Software*. Brasil, 2010. Manual do software. Citado 3 vezes nas páginas 16, 25, 26.

SEBORG, Dale E.; EDGAR, Thomas F.; MELLICHAMP, Duncan A. *Introdução à Modelagem e Dinâmica para Controle de Processos*. Rio de Janeiro: LTC, 2022. Citado 4 vezes nas páginas 19, 21, 22.

SILVEIRA, Paulo Rogério da; SANTOS, Winderson E. dos. *Automação e Controle Discreto*. São Paulo: Érica, 1998. ISBN 85-7194-591-8. Citado 2 vezes nas páginas 17, 21.

SOUSA, Renata; TAIRA, Gustavo Ryuji; PARK, Song Won. Integração do sistema ciber-físico para sistema de programação, intertravamento e controle de um reator batelada. *The Journal of Engineering and Exact Sciences*, v. 5, n. 5, p. 424–432, 2019. Citado 2 vezes na página 13.

VIEIRA, Emerson Thomaz; REIS, Ilmar Duarte. Análise da aplicação de sistemas de automação industrial baseados em CLPs e SCADA no Brasil: arquitetura, protocolos de comunicação e impactos na eficiência produtiva. *Revista Contemporânea*, v. 5, n. 10, e9483, 2025. Citado 2 vezes na página 16.

YADAV, Geeta; PAUL, Kolin. Architecture and security of SCADA systems: A review. *International Journal of Critical Infrastructure Protection*, Elsevier, v. 34, p. 100433, 2021. Citado 4 vezes nas páginas 15, 16, 25.