

PROTÓTIPO DE CONTROLE DE TEMPERATURA PARA MICROERVEJARIAS ARTESANAIS UTILIZANDO ARDUINO NANO E SUPERVISÓRIO ELIPSE SCADA

TEMPERATURE CONTROL PROTOTYPE FOR CRAFT MICROBREWERIES USING ARDUINO NANO AND ELIPSE SCADA INTERFACE

PROTÓTIPO DE CONTROL DE TEMPERATURA PARA MICROCERVECERÍAS ARTESANALES USANDO ARDUINO NANO Y LA INTERFAZ ELIPSE SCADA

Rafaela Martins Meneleu Pedroso¹
Alex Franco Ferreira²

RESUMO: Este trabalho apresenta o desenvolvimento de um protótipo de baixo custo para controle de temperatura em câmaras frias de microcervejarias, visando reduzir perdas causadas por variações térmicas fora do intervalo ideal. O projeto surgiu a partir da demanda de uma cervejaria artesanal de Volta Redonda (RJ), que enfrentava perdas de produção pela falta de monitoramento térmico contínuo. O sistema utiliza um microcontrolador Arduino Nano, um sensor NTC 10k, LEDs de sinalização, um display LCD 16x2 e integra um supervisório desenvolvido na plataforma Elipse SCADA, utilizando o protocolo Modbus para comunicação. A leitura de temperatura é baseada na equação de Steinhart-Hart, garantindo alta precisão na conversão dos dados analógicos. O sistema aciona alertas visuais conforme a faixa de temperatura lida, permitindo rápida intervenção dos operadores. Os testes realizados validaram o funcionamento e confiabilidade do protótipo, demonstrando sua viabilidade como alternativa ao uso de CLPs em pequenas produções. Os resultados estão alinhados com a literatura, que defende o uso de sistemas automatizados para garantir a segurança alimentar e eficiência dos processos artesanais. Conclui-se que a aplicação de tecnologias acessíveis, como o Arduino, pode representar um avanço significativo para o setor de microcervejarias, unindo inovação, controle e sustentabilidade operacional.

2893

Palavras-chave: Arduino Nano. Controle de Temperatura. Micro cervejaria Artesanal.

ABSTRACT: This paper presents the development of a low-cost prototype for temperature control in cold chambers of microbreweries, aiming to reduce losses caused by thermal variations outside the ideal range. The project emerged from the demand of a craft brewery in Volta Redonda (RJ), which faced production losses due to the lack of continuous thermal monitoring. The system uses an Arduino Nano microcontroller, a 10k NTC sensor, signaling LEDs, a 16x2 LCD display, and integrates a supervisory system developed on the Elipse SCADA platform, using the Modbus protocol for communication. Temperature reading is based on the Steinhart-Hart equation, ensuring high accuracy in converting analog data. The system triggers visual alerts according to the temperature range read, allowing quick intervention by operators. Tests validated the prototype's operation and reliability, demonstrating its viability as an alternative to using PLCs in small-scale productions. The results align with the literature, which advocates the use of automated systems to ensure food safety and efficiency in artisanal processes. It is concluded that the application of accessible technologies, such as Arduino, can represent a significant advancement for the microbrewery sector, combining innovation, control, and operational sustainability.

Keywords: Arduino Nano. Temperature Control. Craft Microbrewery.

¹Engenheira Mecânica, discente do curso de pós-graduação em Automação Industrial da Universidade de Vassouras.

²Engenheiro Eletricista, docente do curso de pós-graduação em Automação Industrial da Universidade de Vassouras.

RESUMEN: Este trabajo presenta el desarrollo de un prototipo de bajo costo para el control de temperatura en cámaras frías de microcervecías, con el objetivo de reducir las pérdidas causadas por variaciones térmicas fuera del rango ideal. El proyecto surgió a partir de la demanda de una cervecía artesanal de Volta Redonda (RJ), que enfrentaba pérdidas de producción debido a la falta de monitoreo térmico continuo. El sistema utiliza un microcontrolador Arduino Nano, un sensor NTC 10k, LEDs de señalización, una pantalla LCD 16x2 e integra un sistema de supervisión desarrollado en la plataforma Elipse SCADA, utilizando el protocolo Modbus para la comunicación. La lectura de temperatura se basa en la ecuación de Steinhart-Hart, asegurando alta precisión en la conversión de datos analógicos. El sistema activa alertas visuales según el rango de temperatura leído, permitiendo una rápida intervención por parte de los operadores. Las pruebas realizadas validaron el funcionamiento y la confiabilidad del prototipo, demostrando su viabilidad como alternativa al uso de PLCs en producciones pequeñas. Los resultados están alineados con la literatura, que defiende el uso de sistemas automatizados para garantizar la seguridad alimentaria y la eficiencia de los procesos artesanales. Se concluye que la aplicación de tecnologías accesibles, como Arduino, puede representar un avance significativo para el sector de las microcervecías, uniendo innovación, control y sostenibilidad operativa.

Palabras clave: Arduino Nano. Control de Temperatura. Microcervecía Artesanal.

INTRODUÇÃO

2894

Desde o final dos anos 90 e início dos anos 2000, a cultura de cervejas artesanais vem se espalhando no Brasil. No entanto, a partir de 2010, observou-se um aumento significativo das microcervejarias e fábricas-bar por todo o país. Segundo informações do Ministério da Agricultura e Pecuária (2024), esse setor continua em crescimento, com uma alta de 11,6% em 2022, passando de pouco mais de 200 para 1.729 cervejarias registradas no Brasil, com expectativas de crescimento ainda maior.

Com o aumento da produção, especialmente pela possibilidade de variadas receitas, surge a necessidade de um controle mais rigoroso desse processo.

Os artigos selecionados abordam a necessidade de controle na produção de microcervejarias, visto que o crescimento da produção de cervejas especiais por processo artesanal exige cuidados especiais. Benacchio, et al. (2023), por exemplo, enfatizam a importância da segurança alimentar e a mitigação de erros humanos. Eles argumentam que, independentemente do porte do processo, o controle assegura a qualidade do produto final e do processo como um todo, evitando desperdícios de matéria-prima, descarte de não

conformidades, consumo excessivo de água, energia e outros insumos, além de minimizar o retrabalho.

Por se tratar de um produto alimentício, o cuidado com a segurança alimentar é de extrema importância. Como solução, Fernandes e Franzen (2011) e Schultz et al. (2020) propõem a automatização do processo de fabricação das microcervejarias. Nos artigos analisados, os autores utilizam Controladores Lógicos Programáveis (CLPs) e o protocolo Modbus, um dos mais empregados em automação industrial, somado a um IHM – Interface Homem-Máquina, proporcionando assim um controle de produção viável e versátil, adequado à diversidade de receitas produzidas. Dois artigos apresentaram resumos praticamente idênticos sobre a metodologia utilizada.

Considero viável automatizar parte do processo de produção de cervejas artesanais em microcervejarias utilizando um microcontrolador como o Arduino, embora reconheça suas limitações, uma vez que sua principal aplicação é em prototipagem, e não em processos industriais. A proposta busca ser mais acessível e viável do que o uso de um CLP.

O problema identificado, que levou à escolha de qual parte do processo priorizar na construção de um protótipo, surgiu dos donos de uma cervejaria no interior do estado do Rio de Janeiro, na cidade de Volta Redonda, chamada HOPERÁRIOS. Eles enfrentam uma elevada perda de produção, devido à falta de monitoramento detalhado das variações de temperatura, além de um pequeno intervalo de controle, que varia de -3°C a 3°C .

Este projeto propõe a criação de um protótipo acessível e de baixo custo para o controle de câmaras frias de microcervejarias, com o objetivo de melhorar o controle do processo de resfriamento e conservação do produto, minimizando perdas de produção causadas por variações de temperatura fora do intervalo desejado.

Durante a realização do projeto, pesquisas em artigos tiveram um papel fundamental no andamento do trabalho e na elaboração deste artigo. O objetivo deste estudo foi criar um protótipo de baixo custo, acessível a microcervejarias.

Para a elaboração do projeto, foi desenvolvido o fluxograma apresentado na Figura 1.

Figura 1: FLUXOGRAMA



Fonte: Autores, 2025

2896

LISTA DE MATERIAIS PARA O PROJETO

Para a construção do controlador de temperatura, foram selecionados materiais de baixo custo que garantissem a eficiência na leitura, no processamento dos dados coletados e na criação de um sistema embarcado acessível que atendesse à solicitação do projeto. Os materiais selecionados foram:

Arduino Nano
 Display OLED 128x64 I2C
 Sonda NTC 10k
 Resistores 10kΩ ±1%
 LEDs (Verde, Amarelo, Vermelho)
 Resistores de 220Ω para LEDs
 Protoboard e Jumpers
 Fonte de 5V

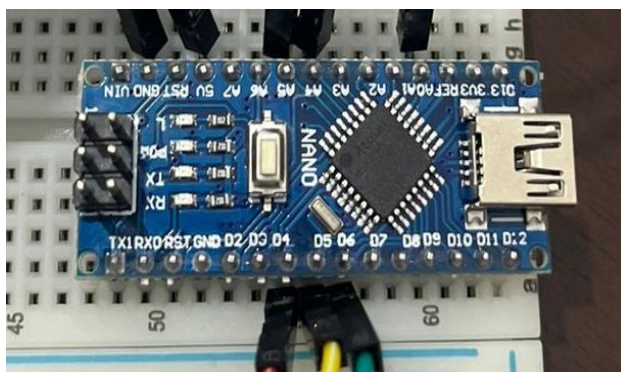
PRINCIPAIS COMPONENTES

O Arduino é um microcontrolador de placa única e código aberto, utilizado para a criação de pequenos projetos de baixo custo. Ele se destaca por ser acessível, flexível, independente e de fácil utilização. A placa Arduino lê entradas, como sensores, e transforma essas informações em saídas, como acender um LED ou exibir valores de parâmetros em uma tela LCD, por exemplo. O Arduino processa o envio de instruções do usuário por meio de programação. Para isso, utiliza-se a linguagem de programação Arduino (baseada em *Wiring*) e o software Arduino IDE, baseado em *Processing* (Arduino, 2022).

A Figura 2 mostra um Arduino Nano ligado a um *protoboard*. Esta versão compacta da plataforma Arduino é construída em torno do microcontrolador ATmega328P, oferecendo uma alternativa eficaz para projetos que exigem um espaço físico reduzido, preservando a maior parte das funções das versões maiores. A imagem exhibe a localização dos pinos de entrada e saída digital (I/O), alimentação (5V e GND), além da conexão USB, utilizada para programar e comunicar com um computador. Essa configuração é frequentemente empregada em projetos de automação, controle e prototipagem eletrônica.

2897

Figura 2 – Arduino Nano



Fonte: Autores, 2025

Como neste projeto a ideia foi realizar a leitura de temperatura – entrada de dados e processar para saída através de LEDs, tela LCD e, posteriormente, supervisorio, foi escolhido o modelo Nano, conforme a Figura 2, por ser mais compacto e permitir a instalação do protótipo diretamente na parede da câmara fria, sem ocupar grandes espaços.

Sensores de temperatura são dispositivos que apresentam variações em uma grandeza elétrica proporcional a uma grandeza física, como a resistência elétrica em função da temperatura, desde que essa relação seja previamente conhecida. Dentre esses sensores, destacam-se os termistores, componentes sensíveis cuja resistência sofre variações significativas mesmo com pequenas mudanças de temperatura, devido à modificação na concentração de portadores de carga. Os termistores do tipo NTC (*Negative Temperature Coefficient*) possuem resistência inversamente proporcional à temperatura e são amplamente utilizados em circuitos eletrônicos para controle, medição e polarização, devido à sua alta sensibilidade térmica e facilidade de fabricação (Addtherm, 2017).

A Figura 3 mostra um sensor de temperatura do tipo NTC, envolto em uma carcaça de metal com um cabo extensor, frequentemente empregado em setores industriais e laboratórios. Este modelo de sensor é projetado para resistir a condições desfavoráveis, possibilitando medições precisas mesmo em áreas de difícil alcance. A ligação com fios permite que ele seja facilmente conectado a microcontroladores como o Arduino, tornando-o ideal para usos que necessitam de monitoramento térmico constante e confiável em sistemas de automação, controle e medição.

2898

Figura 3: SONDA NTC 10K



Fonte: Autores, 2025

O display LCD 16x2 com interface I²C (*Inter-Integrated Circuit*) é amplamente utilizado em projetos de sistemas embarcados devido à sua simplicidade de uso, baixo custo e boa capacidade de exibição de dados em tempo real. Composto por duas linhas de 16 caracteres cada,

esse tipo de display permite apresentar informações relevantes do sistema, como medições de sensores, status de operação ou mensagens de alerta. A principal vantagem do modelo com comunicação I2C está na redução significativa do número de conexões necessárias com o microcontrolador: enquanto a versão paralela convencional requer até sete pinos digitais, o modelo I2C utiliza apenas dois (SDA e SCL), o que libera portas para outros periféricos e simplifica a montagem de protótipos (FILIFELOP, 2021).

No contexto deste trabalho, o display LCD I2C foi integrado ao Arduino Nano para exibir, de forma contínua e legível, as informações obtidas pelo sensor NTC, como temperatura em graus Celsius, tensão e resistência elétrica. A exibição em tempo real auxilia na tomada de decisões rápidas em ambientes produtivos, como microcervejarias artesanais, onde o monitoramento térmico constante é essencial para a conservação do produto final.

A Figura 4 apresenta o display LCD 16x2 utilizado no desenvolvimento do protótipo. Esse módulo é responsável por exibir em tempo real informações essenciais do sistema, como temperatura, tensão e resistência lidas pelo sensor NTC, facilitando a interpretação direta pelo operador. O modelo exibido na Figura 4 é compatível com a interface I2C, o que permite sua comunicação com o microcontrolador Arduino utilizando apenas dois pinos (SDA e SCL), otimizando o uso das portas digitais disponíveis e simplificando a montagem do circuito. Além disso, o display possui iluminação *backlight*, o que garante boa visibilidade mesmo em ambientes com baixa luminosidade, sendo uma escolha comum em aplicações embarcadas e sistemas de automação de baixo custo.

2899

Figura 4: Display LCD 16x2



Fonte: Autores, 2025

MONTAGEM

A montagem do protótipo foi realizada na plataforma circuitio.io, que é online e gratuita. O uso dessa plataforma auxilia na conferência tanto da montagem quanto dos componentes utilizados. O protótipo apresenta uma montagem simples e de fácil entendimento. O microprocessador Arduino Nano recebe as informações da sonda NTC e lê os dados variáveis, que são valores de resistência interpretados pela programação. A partir da equação de Steinhart-Hart, o microcontrolador determina a temperatura do termistor com base na resistência (Addtherm, 2017).

Ainda na programação, os valores de temperatura interpretados são analisados para obter uma análise baseada em um intervalo de temperatura, que é a faixa aceitável para o bom funcionamento da câmara fria. O display LCD exibe a leitura da temperatura em tempo real, enquanto três LEDs de alto brilho (verde, amarelo e vermelho) acendem conforme a faixa de temperatura determinada na programação. Esses LEDs têm a função de fornecer um alerta visual, permitindo que qualquer funcionário da cervejaria perceba rapidamente a situação, mesmo em ambientes barulhentos, contribuindo para ações rápidas e eficazes, e evitando a perda de produção.

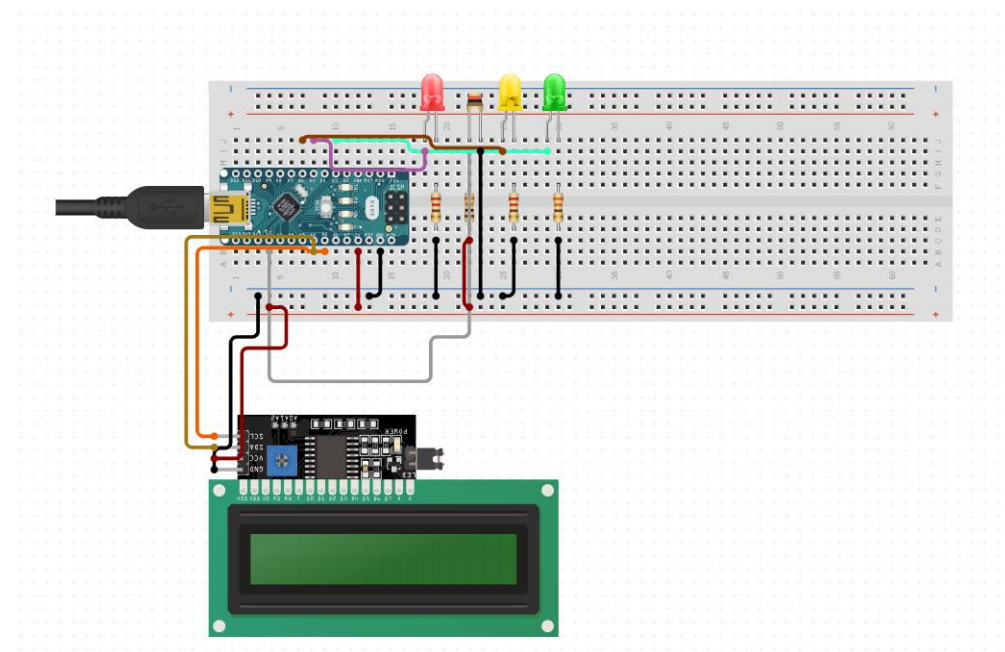
2900

Os parâmetros foram definidos com base nos dados fornecidos pelos donos da cervejaria. Para o funcionamento ideal da câmara fria, o intervalo de temperatura deve variar entre -3°C e 3°C . Os LEDs foram programados para acender de acordo com as seguintes condições: LED Verde – Dentro da faixa, Temperatura Normal; LED Vermelho – Temperatura $< -3^{\circ}\text{C}$, indicando risco de congelamento ou Temperatura $> 3^{\circ}\text{C}$, alertando para Câmara Quente.

A Figura 5 apresenta o esquema de montagem do protótipo desenvolvido para o controle de temperatura utilizando o microcontrolador Arduino Nano. O circuito foi construído sobre uma *proto-board*, conectando-se aos principais componentes do sistema: três LEDs indicadores (verde, amarelo e vermelho), que representam visualmente a faixa de temperatura medida; resistores de proteção para os LEDs; e um display LCD 16x2 com interface I2C, responsável por exibir em tempo real os valores de temperatura, tensão e resistência obtidos do sensor NTC 10k. A comunicação entre os componentes foi realizada por meio de jumpers, com alimentação fornecida via cabo USB. Essa configuração representa uma implementação prática e didática de um sistema embarcado de monitoramento, sendo útil tanto para fins acadêmicos quanto para

aplicações em microcervejarias artesanais que buscam soluções de automação acessíveis e funcionais.

Figura 5: Esquema montado na plataforma CIRCUITO.IO



2901

Fonte: Dos Autores, 2025

EQUAÇÃO STEINHART-HART

O NTC – Coeficiente de Temperatura Negativo é um termistor que faz a leitura a partir da resistência da sonda. Conforme a temperatura aumenta, a resistência diminui (Addtherm, 2017). A partir desse dado, utiliza-se a equação de parâmetro β , uma equação derivada da equação de Steinhart-Hart. A equação de Steinhart-Hart é uma expressão empírica que determina de forma mais precisa a relação entre temperatura e resistência de termistores NTC, pois oferece uma boa aproximação da temperatura real, sendo útil em toda a faixa de temperatura de trabalho do sensor (Addtherm, 2017).

A equação de Steinhart-Hart é expressa como segue:

Fatoorehchi et al. (2019) afirmam que a sentença da equação Steinhart-Hart pode ser observada na Equação 1.

$$\frac{1}{T} = a + \beta * \ln(R) + c * \ln(R)^3 \quad (I)$$

Onde,

T é a temperatura a ser medida (em Kelvin);

R é a resistência obtida em T (em ohms);

A , β , e C são os coeficientes de Steinhart-Hart, que variam dependendo do tipo e modelo do termistor e da faixa de temperatura de interesse.

A Equação de parâmetro β

De acordo com Fatoorehchi et. Al.(2019) a equação de parâmetro β ,é uma equação derivada da equação de Steinhart-Hart, apresentada na Equação 2.

$$\frac{1}{T} = \frac{1}{T_o} + \frac{1}{\beta} * \ln\left(\frac{Rt}{R_o}\right) \quad (2)$$

Onde,

T é a temperatura a ser medida (em Kelvin);

T_o é a temperatura de referência (em Kelvin) de 25 °C (25 °C = 298.15 K);

R_o é a resistência do termistor em T_o ;

R_t é a resistência do termistor;

β é o parâmetro Beta ou B, fornecido pelo fabricante em sua especificação.

2902

PROGRAMAÇÃO DO ARDUÍNO

A linguagem de programação utilizada no Arduino foi C++, que é de fácil compreensão e permite uma conexão direta com o microcontrolador por meio da IDE e conexão USB. O programa utiliza bibliotecas específicas para cada componente eletrônico empregado no projeto. Nesta programação, foram utilizadas bibliotecas para o display LCD I2C, para os LEDs, para o

termistor (sonda NTC) e uma biblioteca Modbus, a fim de permitir que o Arduino se comunicasse com o programa Elipse SCADA, utilizado para a criação de um supervisório.

Para que a sonda NTC pudesse mostrar a temperatura em graus Celsius, tanto no display quanto no supervisório, foi necessário codificar uma fórmula que fizesse a relação matemática entre a variação de resistência da sonda NTC, lida pelo Arduino, e a temperatura equivalente em graus Celsius. Para isso, foi utilizada a fórmula de Steinhart-Hart, descrita na Equação 1.

Com a leitura em Celsius, obtida pela fórmula de Steinhart-Hart, foram programadas fórmulas matemáticas de relações condicionais do resultado, com base nas faixas de temperatura, para análise do estado da câmara fria da cervejaria. Essas faixas foram então atribuídas às cores dos LEDs utilizados.

Por fim, no software Elipse SCADA, foi criado um supervisório com protocolo Modbus, que viabilizou o acesso fácil ao sistema de controle de temperatura da câmara fria, tanto para os funcionários quanto para os donos da cervejaria, garantindo um melhor controle da conservação do produto final – no caso, os barris de cerveja prontos para consumo.

CONEXÃO ENTRE ELIPSE E ARDUINO

2903

A conexão entre o software Elipse SCADA e o Arduino foi feita configurando a biblioteca Modbus na programação do Arduino e declarando as tags necessárias para identificação das variáveis a serem lidas. No Elipse SCADA, foi necessária uma configuração para atribuir as tags às variáveis que deveriam ser explicitadas no painel do supervisório. Foram então definidas as variáveis de temperatura a serem mostradas em tempo real, os LEDs como aviso por meio de texto alertando sobre a faixa de temperatura alcançada, e a situação em que se encontrava a câmara em determinado momento.

CÓDIGO FONTE ADOTADO AO PROJETO

O código fonte realizado na IDE do Arduino é apresentado a seguir.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Modbusino.h>
/*****
```

ARDUINO - COMUNICAÇÃO SERIAL (MODBUS)

<https://github.com/stephane/modbusino>

```
*****/  
  
/* Inicializa o ID do Dispositivo*/  
ModbusinoSlave modbusino_slave(1);  
  
/* Aloca o Número de Registradores*/  
uint16_t tab_reg[10]; // quantos tags eu vou transportar  
  
#define PIN_NTC      Ao  
  
#define VCC          5.0    // tensão real medida da placa  
  
#define R_SERIE      10000.0 // resistor fixo (10 kΩ)  
  
#define LCD_ADDRESS  0x27    // 0x27 ou 0x3F, confira no seu módulo  
  
#define LCD_COLS     16  
  
#define LCD_ROWS     2  
  
/* Variaveis do Projeto */  
const int LED_VD_PIN = 5;  
const int LED_VM_PIN = 6;  
int Leitura_PIN_NTC = 0;  
  
// Coeficientes Steinhart-Hart de NTC 10 kΩ (curva "B3950").  
const float A_COEF = 1.009249522e-03;  
const float B_COEF = 2.378405444e-04;  
const float C_COEF = 2.019202697e-07;  
  
// Instância do display  
LiquidCrystal_I2C lcd(LCD_ADDRESS, LCD_COLS, LCD_ROWS);  
  
// _____  
  
// Funções utilitárias  
  
// _____  
  
/* Faz média de n leituras ADC para reduzir ruído */  
int mediaADC(byte n = 10, byte delay_ms = 3) {  
    long soma = 0;  
    for (byte i = 0; i < n; i++) {  
        soma += analogRead(PIN_NTC);  
        delay(delay_ms);  
    }  
}
```

```

}
return soma / n;
}
/* Converte leitura ADC → resistência da NTC em ohms */
float adcParaResistencia(int adc) {
    float v_adc = adc * VCC / 1023.0;          // tensão no divisor
    return (VCC * R_SERIE / v_adc) - R_SERIE;   // álgebra do divisor
}
/* Converte resistência → temperatura (°C) via Steinhart-Hart */
float resistenciaParaCelsius(float r_ntc) {
    float lnR = log(r_ntc);
    float tempK = 1.0 / (A_COEF + B_COEF * lnR + C_COEF * pow(lnR, 3));
    float tempCelsius = tempK - 273;
    return tempcelsius;
}
// -----
void setup() {
    /* Definindo a taxa de transferencia em bauds */
    modbusino_slave.setup(9600);
    /* Definindo o comportamento dos pinos */
    // Configura os pinos dos LEDs como saída
    pinMode(LED_VM_PIN, OUTPUT);
    pinMode(LED_VD_PIN, OUTPUT);
    analogReference(DEFAULT); // usa Vcc como referência (5 V)
    lcd.init();
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print("NTC 10k - Nano");
    lcd.setCursor(0, 1);
    lcd.print("Inicializando...");
    delay(1500);
    lcd.clear();
}

```

```
}  
//  
void loop() {  
    /*Leitura da sonda*/  
    int Leitura_NTC = 0;  
    /* 1. Leitura média do ADC */  
    int adc = mediaADC(10);          // 10 amostras  
    /* 2. Conversões */  
    float v_adc    = adc * VCC / 1023.0;    // tensão na NTC  
    float r_ntc    = adcParaResistencia(adc);  
    float tempC    = resistenciaParaCelsius(r_ntc);  
    // 3. CONTROLAR OS LEDS E DEFINIR O TEXTO DE STATUS  
    if (temp_celsius < -3) {  
        status_text = "Status: Risco Congelamento";  
        digitalWrite(LED_VM_PIN, HIGH);  
        digitalWrite(LED_VD_PIN, LOW);  
    } else if (temp_celsius >= -3 && temp_celsius <= 2) {  
        status_text = "Status: Câmara Normal";  
        digitalWrite(LED_VM_PIN, LOW);  
        digitalWrite(LED_VD_PIN, HIGH);  
    } else { temp_celsius > 3  
        status_text = "Status: Câmara Quente";  
        digitalWrite(LED_VD_PIN, LOW);  
        digitalWrite(LED_VM_PIN, HIGH);  
    }  
    /* Exemplo de como passar valores para os registradores  
    tab_reg[0] = 600;  
    Exemplo de como recuperar valores dos registradores  
    Variavel = tab_reg[1];  
    */  
    while(1)  
    {
```

```
Leitura_NTC = analogRead(PIN_NTC); // LENDO O VALOR NO NTC
Serial.println(Leitura_PIN_NTC); //ESCREVENDO O VALOR DO NTC NA SERIAL
//Passando a leitura do NTC para o modbus
tab_reg[0] = Leitura_PIN_NTC;
//Passando a sinalização dos leds para o modbus
tab_reg[6] = LED_VD_PIN;
tab_reg[7] = LED_VM_PIN;
/* 3. Envio ao display */
lcd.setCursor(0, 0);
lcd.print("T: ");
lcd.print(tempC, 1);    // uma casa decimal
lcd.print((char)223);    // símbolo grau
lcd.print("C ");        // limpa sobranes

lcd.setCursor(0, 1);
lcd.print("V:");
lcd.print(v_adc, 2);
lcd.print(" R:");
if (r_ntc < 1000)
    lcd.print(r_ntc, 0);    // mostra em ohms
else {
    lcd.print(r_ntc / 1000.0, 1); // em kΩ
    lcd.print("k");
}
lcd.print(" ");          // limpa resto

delay(1000); // atualiza a cada 1 s
}
}
```


SUPERVISÓRIO

O Elipse SCADA, nas versões anteriores ao Elipse E3, é um programa voltado para supervisão e controle, criado pela Elipse Software, com foco na automação de processos industriais e de infraestrutura. Essa versão tradicional do SCADA se destacou pela interface simples e eficiente, sendo amplamente adotada em aplicações industriais de menor e médio porte, especialmente para a supervisão de variáveis físicas, como temperatura, pressão, nível e vazão. Sua principal função é permitir a visualização em tempo real dos processos industriais, oferecendo funcionalidades como alarmes, registros, relatórios e comandos, tanto manuais quanto automáticos (ELIPSE SOFTWARE, 2007).

Embora seja uma versão mais antiga, o Elipse SCADA ainda é compatível com diversos protocolos de comunicação, como Modbus RTU e Modbus TCP/IP, o que possibilita sua integração com controladores lógicos programáveis (CLPs), microcontroladores e outros equipamentos industriais. Essa capacidade torna-o adequado para projetos de automação de baixo custo, como aqueles que utilizam o Arduino. Devido à sua leveza e simplicidade, o SCADA clássico da Elipse ainda é uma alternativa eficaz para supervisão em tempo real em projetos educacionais ou industriais de complexidade moderada.

2908

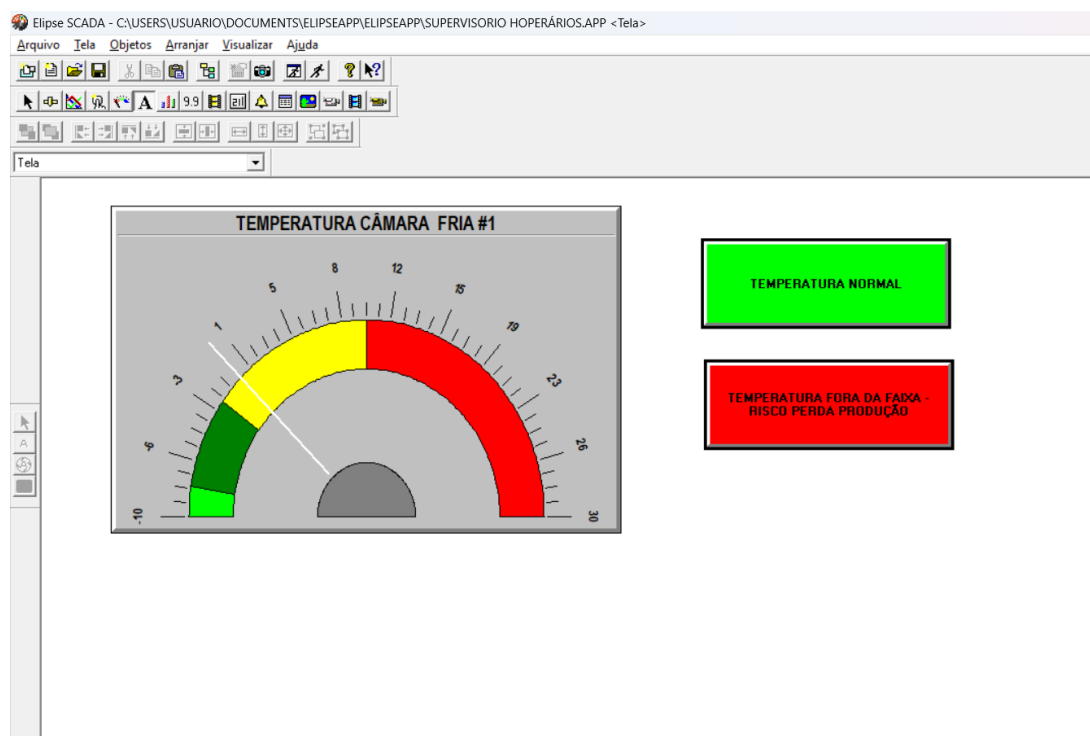
No projeto desenvolvido neste estudo, o Elipse SCADA foi empregado como sistema de supervisão para o controle de temperatura de uma câmara fria em uma microcervejaria, realizando a comunicação com o Arduino Nano por meio do protocolo Modbus, o que possibilitou a visualização das variáveis em tempo real e contribuiu para o controle eficiente do processo.

Na tela criada, o usuário do supervisório pôde monitorar o valor da temperatura lida pela sonda NTC 10K. Dependendo da faixa de temperatura lida, acendiam-se no painel uma caixa verde contendo a mensagem “Temperatura Normal” e outra vermelha contendo a mensagem “Temperatura fora da faixa – Risco de perda de produção”.

A Figura 6 apresenta a interface gráfica do sistema supervisório desenvolvido no software Elipse SCADA para monitoramento da temperatura da câmara fria. O painel inclui um mostrador analógico semicircular que exibe a variação de temperatura em tempo real, destacando as faixas de operação com cores indicativas: verde para condição normal, amarelo para atenção e vermelho para situação crítica fora da faixa aceitável. À direita do mostrador,

dois indicadores visuais em formato de caixas sinalizam de forma textual o estado da temperatura, facilitando a interpretação rápida pelos operadores. Essa interface foi projetada para proporcionar clareza e agilidade na tomada de decisões, permitindo que qualquer colaborador da microcervejaria identifique com facilidade desvios no processo de resfriamento, reduzindo o risco de perdas na produção.

Figura 6: Tela Elipse SCADA



Fonte: Autores, 2025

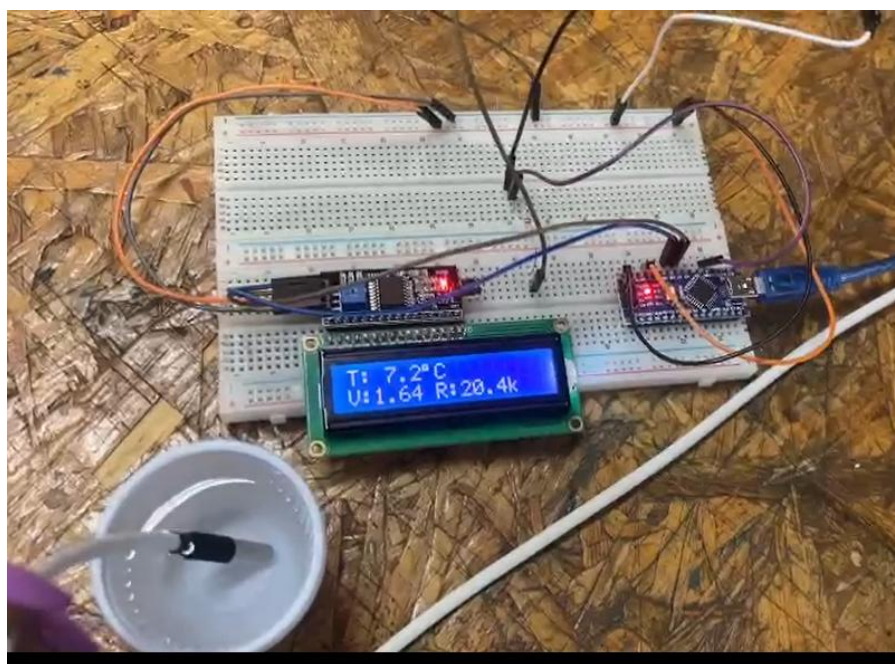
RESULTADOS

Após a montagem do protótipo, este foi submetido a testes de temperatura variáveis para observação da precisão de leitura e estabilidade dos dados. Foi possível observar o acompanhamento em tempo real, demonstrando eficiência no controle de temperatura para aplicação em câmaras frias.

Os testes foram realizados com medições de temperatura utilizando termômetro, comparando as leituras apresentadas no display LCD. Um dos testes foi realizado introduzindo a sonda em um recipiente com água gelada, que indicou uma leitura de 7,2°C.

A Figura 7 ilustra o protótipo criado para coleta de temperatura utilizando um sensor NTC acoplado a um microcontrolador Arduino Nano. O circuito foi montado em uma protoboard e conta com um display LCD 16x2 que mostra, em tempo real, a temperatura (em °C), a tensão de entrada (V) e a resistência do termistor (Ω). O sensor NTC foi colocado em um recipiente com líquido para medir de forma prática sua resposta térmica, demonstrando a conexão entre os dados analógicos obtidos e o processamento digital realizado pelo Arduino. Essa configuração exemplifica uma aplicação prática de sistemas embarcados voltados para instrumentação e controle de variáveis físicas.

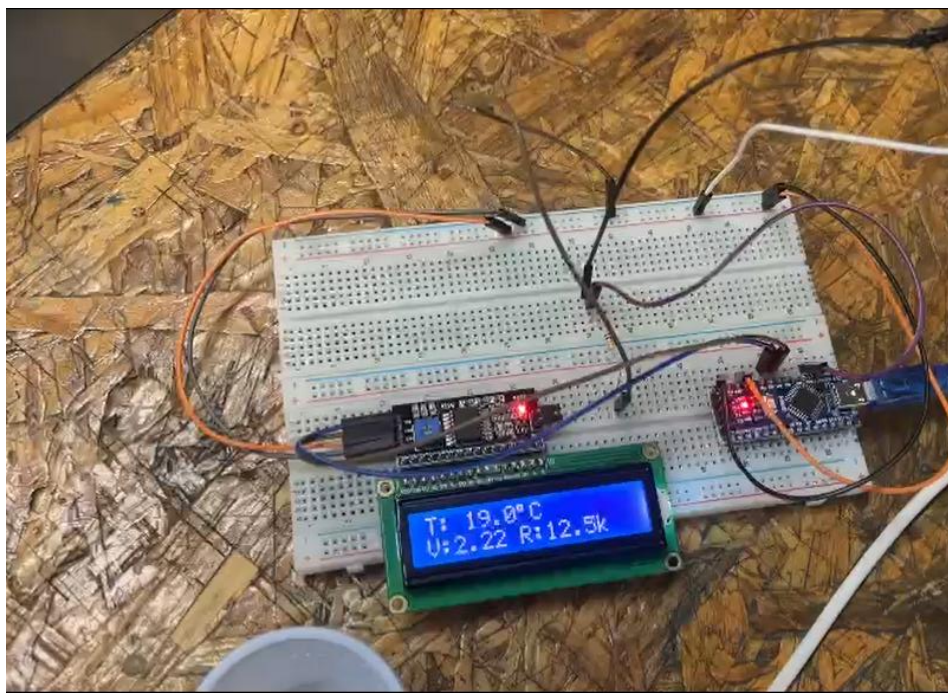
Figura 7: Teste com sonda em água gelada



Fonte: Autores, 2025

O teste realizado retirando a sonda do recipiente com água gelada e colocando-a em contato com o corpo humano mostrou um aumento considerável de temperatura, com a leitura de 19°C, conforme ilustrado na Figura 8.

Figura 8: Teste em temperatura ambiente



Fonte: Autores, 2025

2911

CONCLUSÃO

O desenvolvimento do protótipo de controle de temperatura apresentado neste trabalho demonstrou ser uma solução viável, de baixo custo e tecnicamente eficaz para aplicação em microcervejarias artesanais. Utilizando o microcontrolador Arduino Nano, sensor NTC 10k, display LCD e LEDs de sinalização, o sistema foi capaz de monitorar e sinalizar variações térmicas em tempo real, contribuindo para a preservação da qualidade do produto e redução de perdas por falhas no processo de resfriamento. A comunicação com o software supervisor Elipse SCADA, por meio do protocolo Modbus, proporcionou uma interface funcional e intuitiva para os operadores, reforçando o potencial de aplicação de sistemas embarcados em ambientes produtivos de pequena escala.

Os testes realizados confirmaram a precisão da leitura de temperatura com base na equação de Steinhart-Hart, além da eficácia do sistema em fornecer alertas visuais rápidos e compreensíveis. A proposta está alinhada com os estudos de Fernandes e Franzen (2011) e Schultz et al. (2020), que destacam os benefícios da automação para aumentar a eficiência e a

segurança na produção artesanal. Além disso, o projeto responde às preocupações apontadas por Benacchio, Padilha e Arrieche (2023) em relação à segurança alimentar e ao controle rigoroso de variáveis críticas no processo. Dessa forma, o trabalho evidencia a viabilidade do uso do Arduino como alternativa prática ao CLP em projetos de automação de baixo custo, promovendo inovação, acessibilidade e confiabilidade para microcervejarias em expansão.

REFERÊNCIAS BIBLIOGRÁFICAS

ADDTHERM. **O que são termistores NTC e como utilizá-los?** 2017. Disponível em: <https://www.addtherm.com.br/blog/o-que-sao-termistores-ntc/>. Acesso em: 15 jun. 2025.

ARDUINO. **Arduino Nano – Official Documentation.** 2022. Disponível em: <https://www.arduino.cc/en/Guide/ArduinoNano>. Acesso em: 15 jun. 2025.

BENACCHIO, J. C.; PADILHA, L. B.; ARRIECHE, L. D. A. **Automação aplicada ao processo de produção de cervejas artesanais em microcervejarias: segurança alimentar e minimização de perdas.** *Revista Brasileira de Tecnologia Aplicada*, v. 6, n. 1, p. 89–102, 2023.

ELIPSE SOFTWARE. **Elipse SCADA – Sistema de supervisão e controle.** Porto Alegre: Elipse Software, 2007. Disponível em: <https://www.elipse.com.br>. Acesso em: 10 jun. 2025.

FERNANDES, D. M.; FRANZEN, J. A. **Automação de microcervejarias com uso de CLP e IHM aplicados ao processo de brassagem.** *Revista Ciências Exatas e Tecnológicas*, v. 4, n. 2, p. 25–32, 2011.

FATTOREHCHI, M. et al. **Steinhart–Hart equation for thermistor calibration: theory and application.** *Journal of Physics: Conference Series*, v. 1234, p. 012012, 2019.

SCHULTZ, P. C. et al. **Sistemas automatizados no controle de temperatura em fermentações cervejeiras artesanais.** *Anais do Congresso Brasileiro de Engenharia de Produção*, v. 12, p. 1–7, 2020.