

UEA- UNIVERSIDADE DO ESTADO DO AMAZONAS

EST- ESCOLA SUPERIOR DE TECNOLOGIA

ENGENHARIA DE CONTROLE E AUTOMAÇÃO

TIAGO MENDES IZUMISAWA

**DESENVOLVIMENTO DE UM CONTROLADOR PID, SISTEMA SUPERVISÓRIO
E MELHORIAS EM BANCADA NVT**

Manaus

2020

TIAGO MENDES IZUMISAWA

**DESENVOLVIMENTO DE UM CONTROLADOR PID, SISTEMA SUPERVISÓRIO
E MELHORIAS EM BANCADA NVT**

Projeto de pesquisa desenvolvido durante a disciplina de Trabalho de Conclusão de Curso II, e apresentada à banca avaliadora do Curso de Engenharia de Controle e Automação da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para obtenção do título de Engenheiro de Controle e Automação.

Orientador: Moisés Pereira Bastos, Me.

Manaus

2020

TIAGO MENDES IZUMISAWA

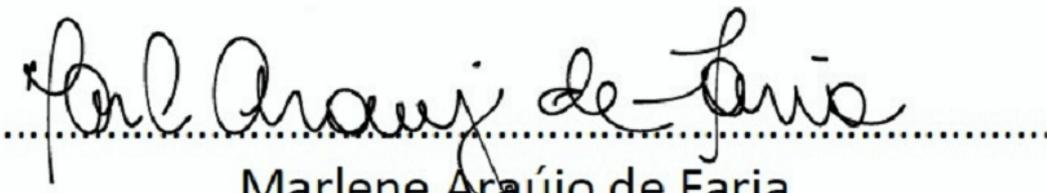
DESENVOLVIMENTO DE UM CONTROLADOR PID, SISTEMA SUPERVISÓRIO
E MELHORIAS EM BANCADA NVT

Projeto de pesquisa desenvolvido durante a disciplina de Trabalho de Conclusão de Curso II, e apresentada à banca avaliadora do Curso de Engenharia de Controle e Automação da Escola Superior de Tecnologia da Universidade do Estado do Amazonas, como pré-requisito para obtenção do título de Engenheiro de Controle e Automação.

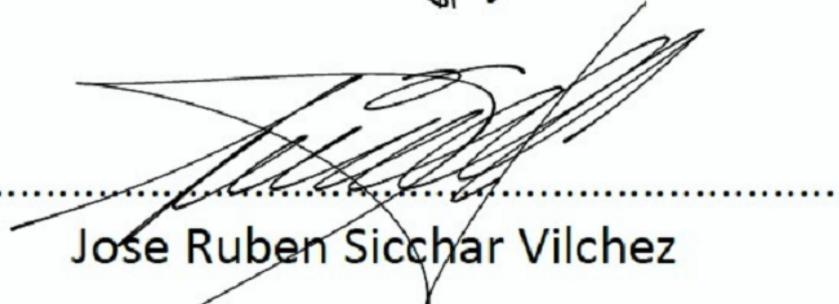
Aprovada em 09 de Dezembro de 2020.



.....
Cleto Cavalcante De Souza Leal



.....
Marlene Araújo de Faria



.....
Jose Ruben Sicchar Vilchez



.....
Moisés Pereira Bastos

AGRADECIMENTOS

Agradeço primeiramente a Deus que me deu força e saúde para desenvolver esse trabalho acadêmico, sem ele não estaria nessa etapa tão importante da minha vida.

Agradeço aos meus pais Tetsu Izumisawa e Rosenilda Mendes Izumisawa que deram muito apoio tanto financeiro como emocional para que este trabalho fosse desenvolvido.

Ao meu orientador, Me. Moisés Pereira Bastos, pelos direcionamentos e orientações durante este trabalho, por dar o seu melhor para transmitir os seus conhecimentos aos alunos e aos orientandos.

Agradeço também aos meus amigos, Lucas Belido Sena, Fausto José Costa Lemos, Lucas Almeida por me acompanhar nessa caminhada acadêmica e pelo apoio nos momentos que mais precisei.

Agradeço a UEA pela oportunidade de estudar em uma boa instituição de ensino e pelo suporte dado ao longo do ensino acadêmico.

RESUMO

A aplicação do controlador PID em conjunto com sistema supervisório tem grande importância para o processo industrial em médio e a longo prazo. Com esse objetivo este trabalho foi desenvolvido para aplicar controle contínuo em uma variável de processo e um sistema supervisório para monitorar um processo industrial que servirá de estudo acadêmico para discente e docentes da Escola Superior de Tecnologia – EST/UEA. O projeto de controlador PID foi desenvolvido no software MATLAB e executado no Arduino. Para o desenvolvimento do sistema supervisório foi utilizado o software Eclipse SCADA e nele foi criado telas que simula o funcionamento do protótipo. Para comunicar o sistema supervisório com o Arduino foi utilizado o protocolo modbus serial e com isso foi possível estabelecer a comunicação e o envio de variáveis para monitoramento de processo. A lógica do funcionamento do processo se deu através dos atuadores de processo como também os sensores, sendo o controle PID para a variável de temperatura e o controle contínuo para a variável de nível. Com isso obteve-se resultados satisfatório como o melhor rendimento da resposta do sistema de temperatura com a aplicação do controlador, o funcionamento adequado da malha de nível com o monitoramento em tempo real através do sistema supervisório.

Palavras-chave: Controle PID, Sistema Supervisório, Arduino.

ABSTRACT

The application of the PID controller in conjunction with a supervisory system is of great importance for the industrial process in the medium and long term. To this end, this work was developed to apply continuous control to a process variable and a supervisory system to monitor an industrial process that will serve as an academic study for students and teachers at the School of Technology - EST / UEA. The PID driver project was developed in MATLAB software and derived in Arduino. For the development of the supervisory system, Eclipse SCADA software was used and screens were created to simulate the prototype's operation. For the supervisory system with Arduino, the serial modbus protocol was used and with that it was possible to establish a communication and send variables for process monitoring. The logic of the process operation took place through the process actuators as well as the sensors, being the PID control for the temperature variable and the continuous control for the level variable. With this, a satisfactory result is obtained, such as the best performance of the temperature system response with the application of the controller, the proper functioning of the level mesh with real-time monitoring through the supervisory system.

Keywords: PID Control, Supervisory System, Arduino.

LISTA DE ILUSTRAÇÕES

Figura 1 - Diagrama de Blocos do Sistema de Controle Automático de Processo.....	17
Figura 2 - Representação do sinal de Controle On-Off.....	18
Figura 3 - Representação do sinal de Controle On-Off.....	20
Figura 4 - Bancada NVT.....	21
Figura 5 - Arduino Uno.....	23
Figura 6 - Representação de um Sistema Supervisório.....	24
Figura 7 - Componente IHM.....	25
Figura 8 - Fluxo de aquisição de dados.....	26
Figura 9 - Arquitetura de dois controladores no processo industrial.....	28
Figura 10 - Descrição geral da Bancada didática NVT.....	29
Figura 11 - Arquitetura do processo.....	30
Figura 12 - Funcionamento do processo original.....	31
Figura 13 - Arquitetura do protótipo com as alterações.....	32
Figura 14 - Funcionamento do processo com modificações.....	32
Figura 15 - Sensor Ultrassônico HC-SR04.....	34
Figura 16 - Sensor de Vazão.....	34
Figura 17 - Sensor de Nível de Água.....	34
Figura 18 - Sensor DS18B20.....	35
Figura 19 - Resistencia Elétrica 127v/550watt.....	36
Figura 20 - Minibomba 12v.....	36
Figura 21 - Mini Cooler 12v.....	37
Figura 22 – Servo Motor sg.....	37

Figura 23 - Arduino Uno.....	38
Figura 24 - Diagrama de blocos da malha de temperatura.....	39
Figura 25 - Tempo para temperatura variar 1°C.....	40
Figura 26 - Algoritmo para encontrar os parâmetros do controlador.....	41
Figura 27 - Conjunto de combinações de k, a e sobressinal.....	42
Figura 28 - Programação do controle PID no Arduino.....	44
Figura 29 - Alterações no reservatório de nível.....	45
Figura 30 - Código controle de nível on-off com a biblioteca modbus.....	46
Figura 31 - Código controle de temperatura PID com a biblioteca modbus.....	47
Figura 32 - Driver de comunicação.....	48
Figura 33 - Definição dos parâmetros de comunicação.....	48
Figura 34 - Temperatura do reservatório 1.....	49
Figura 35 - Nível no reservatório 2.....	49
Figura 36 - Medidor analógico de nível tanque 2.....	50
Figura 37 - Resposta do sistema com o controlador PID.....	52
Figura 38 - Resposta do sistema sem o cooler.....	52
Figura 39 - Resposta do sistema com o cooler.....	53
Figura 40 - Sistema supervisório do processo.....	54

SUMÁRIO

INTRODUÇÃO.....	12
1.1 TEMA.....	13
1.2 PROBLEMA DE PESQUISA.....	13
1.3 HIPÓTESE.....	13
1.4 JUSTIFICATIVA.....	13
1.5 OBJETIVOS GERAIS E ESPECÍFICOS.....	14
1.5.1 OBJETIVOS GERAIS.....	14
1.5.2 OBJETIVOS ESPECÍFICOS.....	14
1.6 METODOLOGIA.....	15
1.7 ORGANIZAÇÃO DO TRABALHO.....	16
2 REFERENCIAL TEÓRICO.....	17
2.1 Controle Automático.....	17
2.1.1 Controle On-Off.....	18
2.1.2 Controlador P.....	18
2.1.3 Controlador PI.....	19
2.1.4 Controlador PD.....	19
2.1.5 Controlador PID.....	20
2.1.6 Projeto de controladores PID com otimização computacional.....	20
2.2 Bancada NVT.....	21
2.3 Plataforma Arduino.....	22
2.3.1 Princípio de Funcionalidade.....	23
2.3.2 Linguagem C++.....	23
2.4 Sistema Supervisório.....	24
2.5 Interface Homem-Máquina (IHM).....	25

2.6 Aquisição de dados.....	25
2.7 Plataforma MATLAB.....	26
3 TRABALHOS RELACIONADOS.....	27
3.1 SISTEMA DE CONTROLE DE TEMPERATURA PARA UMA ESTUFA COM MONITORAMENTO VIA APLICATIVO.....	27
3.2 SISTEMA SUPERVISÓRIO E CONTROLE MULTIVARIÁVEL APLICANDO CONTROLADOR FUZZY PONDERADO-PID EM UM PROCESSO DE FERMENTAÇÃO ALCOÓLICA	28
3.3 DESENVOLVIMENTO E CONSTRUÇÃO DE UM PROTÓTIPO PARA A BANCADA DE TESTE DE LABORATÓRIO, DE UM SISTEMA SUPERVISÓRIO PARA MONITORAMENTO E CONTROLE DE NÍVEL, TEMPERATURA E VAZÃO DE UM PROCESSO INDUSTRIAL SIMULADO.....	28
4 METODOLOGIA.....	30
4.1 Arquitetura do Sistema Original.....	30
4.1.2 Arquitetura do Sistema com Alterações.....	31
4.2 Bancada NVT.....	32
4.2.1 Elementos da Bancada.....	33
4.2.1.1 Sensores.....	33
4.2.1.2 Atuadores.....	35
4.3 Sistema de Controle.....	38
4.3.1 Hardware de controle.....	38
4.3.2 Malha de controle de temperatura.....	39
4.3.3 Projeto de controladores PID com abordagem de otimização computacional.....	40
4.3.4 Aplicação do controlador PID na Malha de temperatura.....	43
4.4 Sistema Supervisório da bancada NVT.....	44
4.4.1 Biblioteca Modbus Serial.....	45
4.4.2 Código de Controle de Nível e Temperatura do processo.....	45

4.4.3 Comunicação Eclipse SCADA.....	47
4.4.4 Layout do Sistema supervisório.....	49
5 RESULTADOS OBTIDOS.....	50
5.1 Reservatório de Nível.....	50
5.2 Reservatório de Temperatura.....	51
5.2.1 Controle PID de temperatura.....	51
5.3 Sistema Supervisório.....	53
6 CONCLUSÃO.....	55
6.1 DIFICULDADES ENCONTRADAS.....	55
6.2 TRABALHOS FUTUROS.....	56
REFERÊNCIAS BIBLIOGRÁFICAS.....	57

INTRODUÇÃO

Os processos industriais nos dias de hoje têm exigido precisão e rendimento no manuseio de máquinas como também operações automáticas que a torna mais eficientes e produtivo em ambientes fabris. Segundo Equipe Brasil Escola (2020) a 3ª Revolução Industrial no século XX trouxe a automação e a robótica a fim de melhorar processos produtivos que exige repetição em um ambiente hostil para o ser humano trabalhar, com isso também surgiu os controladores PID no início do século XX que tem como função melhorar o rendimento e a precisão de um sistema industrial.

Em conjunto com a automatização de processos industriais surgiu o sistema supervisório que segundo SILVA (2018) são sistemas digitais de monitoração e operação da sistemas responsáveis pelo controle das variáveis do processo, onde as informações são atualizadas continuamente e armazenados em um banco de dados ou remotamente.

Neste contexto a Engenharia de Controle e Automação fez estudos acerca de projeto de controladores aplicado a uma planta industrial e dentro das disciplinas de Controle I,II e III, como também sistemas supervisórios dentro da disciplina de Automação e Supervisão de processos.

Dentro desses conceitos, o presente trabalho foi realizado com o objetivo de desenvolver um controlador PID de temperatura e um sistema supervisório na bancada NVT com o intuito de servir como base de estudo para a área de controle de processo como também de automação industrial.

1.1 TEMA

Desenvolvimento de um controlador PID, sistema supervisorio e melhorias em bancada NVT.

1.2 PROBLEMA DE PESQUISA

A bancada NVT localizada no laboratório de sistemas Microprocessados na EST/UEA está inativada e precisa de melhorias no Sistema de Controle aplicando um controlador PID e desenvolvendo um Sistema Supervisorio. Sendo assim há necessidade de melhorar o desempenho fisico do protótipo utilizando um Controle Contínuo na variável de temperatura.

1.3 HIPÓTESE

É possível o desenvolvimento de um controle PID contínuo na variável de temperatura e um Sistema Supervisorio para monitorar as variáveis de processo do sistema utilizando o Arduino Uno integrado a um *software* de aquisição de dados para análise do sistema e facilidade do usuário manusear o protótipo NVT em relação as entradas e saídas das variáveis de nível e temperatura.

1.4 JUSTIFICATIVA

A melhoria no protótipo NVT desenvolvido por Souza (2018) se justifica para fins acadêmicos com intuito de obter uma aplicação prática de um Sistema de Controle e Automação multivariável no laboratório de Sistemas Microprocessados de forma dinâmica e simples para o discente ou docente ter facilidade de acesso a bancada. Há necessidade também de o usuário utilizar o protótipo para adquirir conhecimento na interpretação de parâmetros do Controlador PID aplicado a grandeza fisica temperatura.

A pesquisa irá contemplar o conhecimento de disciplinas do curso de Engenharia de Controle e Automação, tais como: Instrumentação Industrial, Laboratório de Instrumentação Industrial, Controle e Automação I, II e III, Laboratório de Controle e Automação I, II e III Automação e Supervisão de Processo e Linguagem de Programação I e II.

1.5 OBJETIVOS GERAIS E ESPECÍFICOS

1.5.1 OBJETIVO GERAL

Desenvolver um controle PID contínuo com o objetivo de melhorar o desempenho do sistema de temperatura. O controlador PID irá ser projetado através do software MATLAB, onde o algoritmo do controlador vai ser desenvolvido e processado para que o Arduino Uno envie um sinal nos atuadores do sistema. Com isso o sistema supervisório mostrará em tempo real o desempenho do processo de temperatura e nível com o sinal PID de temperatura sendo aplicado, sendo assim será feito melhorias físicas na bancada NVT nas variáveis de processo para a aplicação do controle PID e a utilização do sistema supervisório.

1.5.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos são:

- a) Realizar o levantamento bibliográfico de artigos científicos e livros;
- b) Listar todo o material da bancada NVT e verificar quais componentes precisam ser substituídos para que se possa implementar o Controle PID;
- c) Realizar a montagem dos componentes a serem substituídos;
- d) Projetar o Controle PID;
- e) Implementar o sistema supervisório;
- f) Testar o Controlador PID no protótipo;

1.6 METODOLOGIA

O trabalho passará por um estudo minucioso das referências bibliográficas de estudo que foram escolhidos baseados em conteúdo conceituados e atuais, permitindo assim utilizar bases teóricas que são aplicáveis ao protótipo desenvolvido no laboratório. Há de se considerar que os métodos de estudo a ser utilizado nessa monografia se baseiam no hipotético-dedutivo, onde a coleta de informações será de forma qualitativa.

Inicialmente serão realizados os levantamentos bibliográficos baseados em aplicação de controlador PID em sistemas que controlam a variável de temperatura, com o objetivo de buscar métodos e componentes físicos capazes de projetar controle contínuo. Concluído essa etapa, será feito o estudo com fins de coletar conteúdos teóricos para obter respostas sobre a problemática do projeto.

Após o estudo ser realizado, serão feitas alterações de componentes físicos visando projetar o controlador PID, com isso será retirado do reservatório superior o mini cooler e colocado no reservatório inferior junto com a resistência elétrica e o sensor de temperatura. Para o sistema de nível foi acrescentado um suporte para o tanque superior e uma válvula de retenção que evita a água passar do tanque inferior para o tanque superior, além disso foi colocado uma torneira junto ao servo motor que irá permitir a passagem de água do reservatório inferior para o superior.

O desenvolvimento do Controle contínuo será realizado pelo software MATLAB, onde será utilizado o método de projeto de controladores PID por otimização computacional, depois de concluído essa etapa será aplicado o controle de temperatura no kit de prototipagem Arduino Uno.

Após a aplicação dos códigos do Controlador PID na malha de temperatura do protótipo, será desenvolvido um sistema supervisorio utilizando o software Elipse SCADA que será responsável pela monitoração dos processos de nível e temperatura da bancada, com isso será realizado um layout ilustrativo da bancada com dados em tempo real.

Por fim será analisado os resultados obtidos com a aplicação do controle PID e sem a ação do controlador como também as alterações físicas da bancada em relação as variáveis de nível e temperatura.

1.7 ORGANIZAÇÃO DO TRABALHO

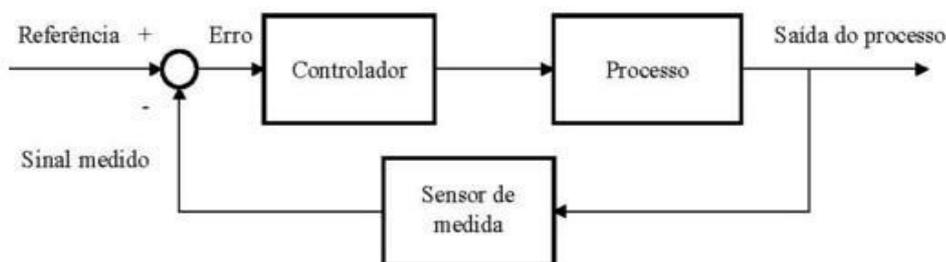
No capítulo 1 são descritos os seguintes itens deste trabalho: introdução, problema de pesquisa, hipótese, objetivo, justificativa e metodologia que foi utilizada para a conclusão do presente trabalho. No Capítulo 2 mostra o referencial teórico da monografia, com as pesquisas relacionadas com o tema do TCC, com o objetivo de o leitor ter o melhor entendimento. No Capítulo 3 são descritos alguns trabalhos relacionados de acordo com o tema tratado, onde foi realizado um comparativo pegando as principais características entre os mesmos e o trabalho apresentado nesta monografia. No Capítulo 4 trata da Metodologia utilizada onde são descritos os métodos e técnicas usadas para a conclusão do trabalho. No Capítulo 5 tem como título “Resultados Obtidos”. Neste tópico é tratado os principais resultados obtidos para conclusão da monografia. No Capítulo 6 se encontra a conclusão do trabalho, além das dificuldades encontradas e trabalhos futuros.

2 REFERENCIAL TEÓRICO

2.1 Controle Automático

Segundo Castrucci (2011), o objetivo da utilização dos Sistemas de Controle Automático é tornar as variáveis de processos a ser manipuladas mais estáveis e eficientes. A relação entre Processo, Controlador, Sensor, Referência e Saída do processo são os principais parâmetros utilizados para descrever a atuação de um Controle Automático em um processo Industrial conforme o diagrama de Blocos da figura 1.

Figura 1 – Diagrama de Blocos do Sistema de Controle Automático de Processo



Fonte: (CAMPOS, 2017)

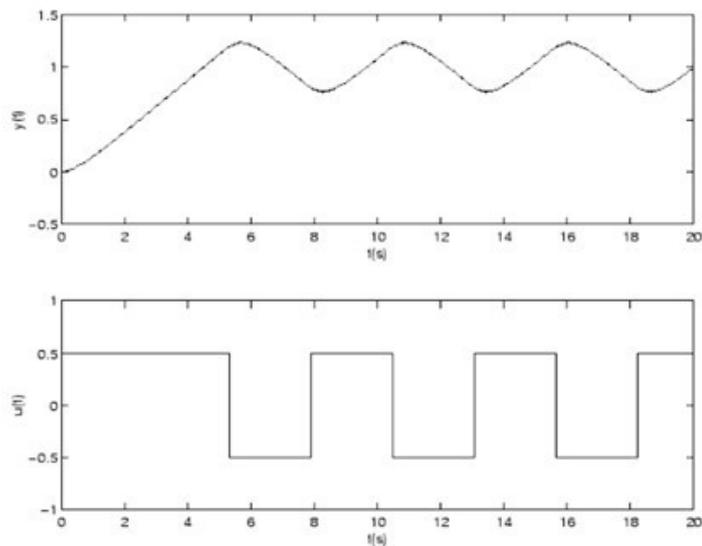
Conforme descreve Castrucci (2011), o Controle Automático tem a função de aperfeiçoar o desempenho das variáveis físicas a ser controlado. Para atuação do controlador no processo, deve-se ter conhecimento dos principais modelos de controladores utilizados na atualidade:

- Controle On-Off;
- Controlador P;
- Controlador PI;
- Controlador PD;
- Controlador PID.

2.1.1 Controle On-Off

De acordo com Barbosa *et al.* (2018) o controlador On-Off é uma técnica de controle que tem como característica principal assumir apenas dois valores na representação do sinal de controle como mostra na figura 2. O controle On-Off é bastante utilizado em projetos de Controle por apresentar uma simplicidade para modelar o controlador e por apresentar um custo baixo. Entretanto esse método possui desvantagens por não poder aplicar controle contínuo e conseqüentemente não é possível controlar o rendimento do processo, fazendo com que equipamento desgaste mais rapidamente.

Figura 2 – Representação do sinal de Controle On-Off



Fonte: (SLIDEPLAYER.COM, 2019)

2.1.2 Controlador P

Segundo o Zanolla *et al.*(2017), o Controlador Proporcional é um método de controle que deve ser usado moderadamente, pois tem como característica tornar o sistema mais oscilatório de acordo com o aumento do ganho proporcional, em contrapartida a ação do controlador faz com que melhore a resposta regime permanente, sendo bastante utilizado para

fazer a técnica do ajuste fino, que se caracteriza em aumentar o ganho K_p para diminuir o erro estacionário na resposta dinâmica do sistema. A equação que representa o controlador é dada por:

$$G(s) = K_p \quad (1)$$

2.1.3 Controlador PI

De acordo com Zanolla *et al.*(2017), o Controlador Proporcional-Integrativo é a junção do Controlador Proporcional com o Controlador Integrativo, tendo como característica tornar nulo o erro estacionário. Entretanto a utilização do controlador possibilita de aumentar a instabilidade devido ao aumento de ordem no sistema. A equação matemática do controlador PI é descrita da seguinte forma:

$$G(s) = K_p \left(1 + \frac{1}{sT_i} \right) \quad (2)$$

2.1.4 Controlador PD

Conforme descreve Zanolla *et al.*(2017), o Controlador Proporcional-Derivativo representa a junção do Controlador Proporcional com o Controlador Integrativo. A atuação desse controlador acontece para sistemas que apresentam resposta transitória insatisfatória, fazendo com que diminua as oscilações. Em contrapartida o Controlador PD amplifica os sinais de distúrbios. A equação matemática do controlador PD é representada por:

$$G(s) = K_p (1 + sT_d) \quad (3)$$

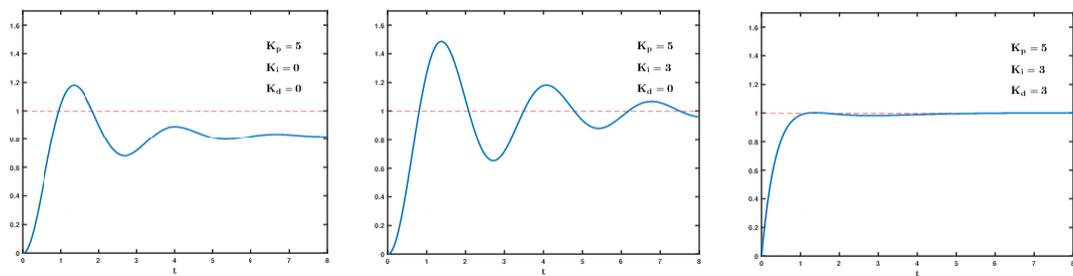
2.1.5 Controlador PID

Segundo Santos *et al.*(2016) a modelagem matemática do controlador PID é a soma das funções transferência proporcional, integral e derivativa como está descrito na equação 1.1. Onde K_p representa o ganho proporcional, T_i o tempo integral e T_d o tempo derivativo.

$$G(s) = K_p \left(1 + \frac{1}{sT_i} + sT_d \right) \quad (4)$$

De acordo com Torres *et al.*(2017) o controlador PID é um algoritmo de controle que atua no sistema com propósito de modificar o tempo de subida, diminuir o erro estacionário e melhorar a resposta transitória. Com isso a resposta temporal do sistema em regime transitório e permanente apresenta resultados satisfatórios como mostra a figura 3.

Figura 3 – Descrição da atuação do Controle P, PI e o PID



Fonte: (CAMPOS, 2017)

2.1.6 Projeto de controladores PID com otimização computacional

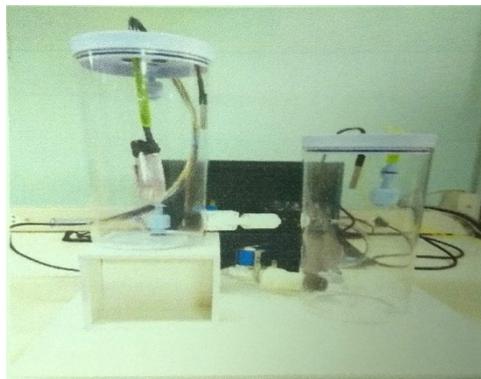
Segundo dos Santo *et al.*(2017), neste método de projeto de controladores PID é utilizado a ferramenta MATLAB para desenvolver um algoritmo que tem como objetivo encontrar a melhor combinação de k e a tendo um critério de comportamento do sistema para achar os melhores valores dessas variáveis, a equação que descreve o controlador em relação a essas variáveis está descrito abaixo.

$$G(s) = K \frac{(s+a)^2}{s}$$

2.2 Bancada NVT

Segundo Silva (2018), A bancada NVT é um protótipo didático localizado no Laboratório de sistemas microprocessados da EST/UEA que controla as variáveis de Nível, Vazão e Temperatura de um processo industrial simulado. O princípio de funcionamento se dá baseado na figura 4 sendo descrito da seguinte forma:

Figura 4 – Bancada NVT



Fonte: (AUTOR, 2019)

- O tanque superior irá aquecer um fluido com a resistência até chegar ao valor desejado;
- Após o fluido chegar à temperatura do valor desejado, a válvula Solenoide do tanque superior irá ser aberta para transportar o líquido pro tanque inferior;
- Quando o fluido chegar ao tanque inferior ocorrerá o processo de resfriamento do líquido utilizando um mini cooler até a temperatura alcançar o valor de setpoint;
- Ocorrendo o resfriamento do líquido, a válvula Solenoide do tanque inferior estará aberta para o fluido retornar ao tanque superior, onde irá retornar para o início do processo.

Os materiais utilizados para a construção da bancada são:

- 3 Sensores de nível;

- 1 Sensor de fluxo de Água;
- 2 Sensores de Temperatura à prova d'água;
- 1 Mini Cooler;
- 1 Resistência Aquecedora;
- 2 Válvulas Solenóides;
- 2 Minibombas;
- 7 Conexões hidráulicas;
- 1 Arduino Mega;
- 2 Recipientes;
- 1 Base de MDF.

2.3 Plataforma Arduino

Conforme descreve Silva (2018), a prototipagem Arduino é uma plataforma conhecida como computação física onde tem muita aplicação para prototipação de circuitos eletrônicos. O projeto Arduino iniciou em 2005 na cidade de Ivrea, Itália. Ele contém uma placa de microcontrolador Atmel AVR e um IDE de programação na linguagem C++

O Arduino possui portas de entradas e saídas, essas sendo tanto digital como analógica. O ambiente de programação possui código aberto, podendo ser alterado e reproduzido. Pode-se dizer que o Arduino é um pequeno computador que interage com o ambiente. A imagem física do Arduino é vista na figura 5.



Fonte: (STORE ARDUINO,2020).

2.3.1 Princípio de Funcionalidade

Segundo descreve Silva (2018), o Arduino é composto por um *hardware* e *software* onde o primeiro possui portas de entrada e saídas para receber e enviar sinais de atuação o segundo possui uma plataforma de programação onde é possível enviar funções para o Arduino executar.

O *hardware* da placa possui conectores que permitem energizar o projeto a ser desenvolvido no Arduino via USB ou por adaptador AC com tensão entre 9V e 12V.

A placa do Arduino possui 14 pinos de entrada e saída digitais que são usadas de acordo com a aplicação a ser utilizado e com a sua programação. Também há 6 pinos de entradas analógicas e 6 pinos de saídas analógicas.

2.3.2 Linguagem C++

A linguagem C++ foi desenvolvida e implementado no ano de 1972 por Dennis Richie. Essa linguagem de programação se tornou uma das mais utilizadas no mundo, por ser flexível e poderosa, sendo responsável pela criação de diversos softwares. (SILVA, 2018)

Segundo SILVA(2018), linguagem se caracteriza por ser estruturada, ou seja, o fluxo de execução do programa se apresenta por ser contínuo formado por sequência decisões e interações. Entretanto teve a necessidade de desenvolver uma linguagem mais robusta, com isso em 1980 surgiu a linguagem C++, que tem como característica ser orientada a objetos

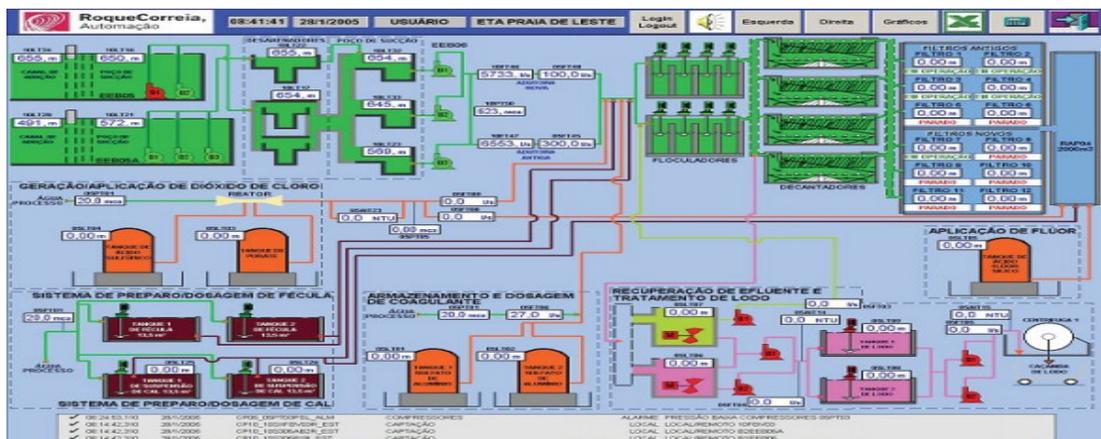
sendo nos dias de hoje a linguagem de programação de aplicações mais comum no sistema operativo Windows.

2.4 Sistema Supervisório

De acordo com Candido *et al.*(2013), o Sistema Supervisório permite a monitoração e o mapeamento do processo produtivo, onde as informações do processo são adquiridos através de equipamentos de aquisição de dados. Com isso as variáveis do processo que foram coletados serão manipuladas, analisadas e armazenadas para que em seguida seja apresentada ao usuário.

Os dados obtidos são organizados numa base de dados que pertence ao software de supervisão. São utilizados tags, que são todas as variáveis numéricas envolvidas na aplicação, que correspondem às variáveis do processo real, se comportando em relação ao controlador e o sistema conforme a figura 6.

Figura 6 – Representação de um Sistema Supervisório



Fonte: (Candido *et al.*, 2013)

2.5 Interface Homem-Máquina (IHM)

Segundo Braga (2015), a Interface Homem-Máquina é um sistema que busca unir ações ligadas tanto para o humano quando da máquina em busca de um objetivo que leve a

interação entre os dois elementos do sistema. A IHM funciona baseados em dados que são selecionados pelo usuário com o intuito de receber da máquina propriedades físicas que sejam interpretáveis para que a utiliza.

A IHM consiste em realizar tarefas através da colaboração entre o humano e o componente de automação utilizando *Displays* como mostra na figura 7, a Interface Homem-Máquina visa atender as necessidades e se acomodar as capacidades das pessoas a quem eles são endereçados (COSTALONGA *et al*, 2014).

Figura 7 - Componente IHM

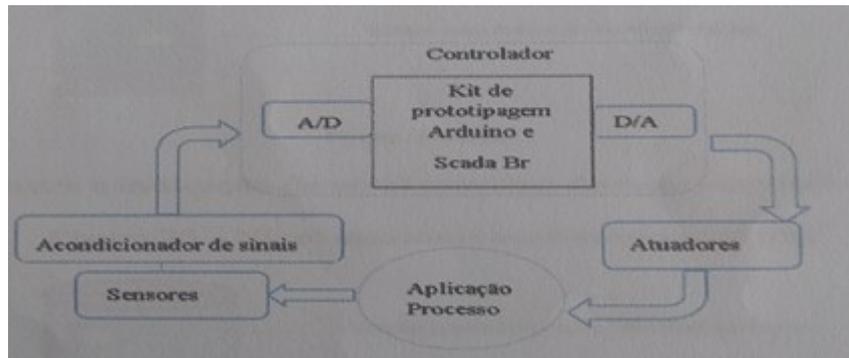


Fonte: (CUNHA *et al*, 2018)

2.6 Aquisição de dados

O processo de aquisição de dados se dá através de um fluxo onde o controlador irá enviar um sinal para os atuadores no processo que por este será monitorado sensores do sistema que de acordo com a aplicação pode receber sinais digitais ou sinais analógicos, sendo assim os sinais dos sensores retornará ao controlador para verificar a condição de funcionamento do processo como mostra a figura 8.

Figura 8 – Fluxo de aquisição de dados



Fonte: (Autor)

2.7 Plataforma MATLAB

Segundo dos Santo *et al.*(2017), a plataforma MATLAB é software que possui uma variedade de aplicações matemáticas voltadas para a área da engenharia, sendo uma das ferramentas mais utilizadas em simulação de sistemas de controle, inteligência artificial e análise de sinais.

Para a aplicação na área de controle, o MATLAB possui bibliotecas específicas para análise de resposta temporal de sistemas, projeto de controladores PID programação em blocos pela ferramenta Simulink.

3 TRABALHOS RELACIONADOS

3.1 SISTEMA DE CONTROLE DE TEMPERATURA PARA UMA ESTUFA COM MONITORAMENTO VIA APLICATIVO

No trabalho de dos Santo *et al.*(2017), o autor desenvolveu um controle PID para controlar a temperatura de uma estufa e um monitoramento via aplicativo. Foram utilizados para o sistema térmico um cooler 12V, uma lâmpada 12V e um sensor DS18B20 e um Arduino.

Foi utilizado o Software MATLAB para projetar o controle PID utilizando o Algoritmo de otimização computacional do OGATA onde e projetado um controlador através de combinações das variáveis k e a .

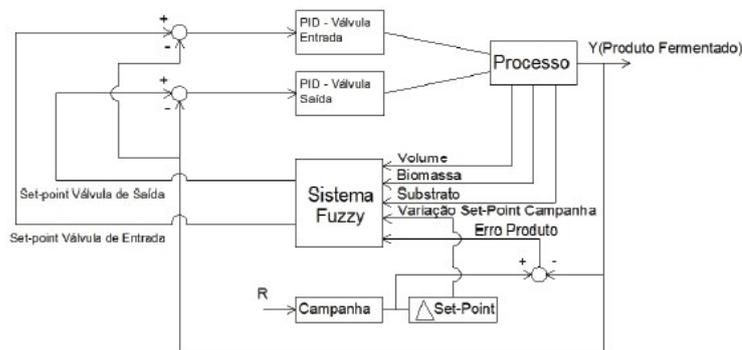
Após a aplicação do controle PID no sistema de temperatura foi desenvolvido uma plataforma móvel para monitorar a temperatura do sistema como também visualizar o gráfico de resposta do sistema com a aplicação do controlador PID

Esse artigo se relaciona com esse trabalho por ter aplicação de controle PID em um sistema que controla temperatura e utilização do algoritmo de otimização do OGATA com irá ser usado neste TCC.

3.2 SISTEMA SUPERVISÓRIO E CONTROLE MULTIVARIÁVEL APLICANDO CONTROLADOR FUZZY PONDERADO-PID EM UM PROCESSO DE FERMENTAÇÃO ALCOÓLICA

Foi desenvolvido no trabalho de Mendonça *et al.*(2015) um controlador PID em um processo de fermentação Alcoólico utilizando a arquitetura Fuzzy de controle nos estados de concentração de substrato, biomassa, produto e volume segundo a figura 9. A arquitetura desenvolvida para o aprendizado de padrões utilizando técnicas de Inteligência Artificial foi através do Software MATLAB.

Figura 9 – Arquitetura de dois controladores no processo industrial



Fonte: (MENDONÇA *et al*, 2015)

A aplicação de controle contínuo se dá através das variáveis de vazão de entrada e saída utilizando válvulas Solenoide proporcionais, com isso o sistema passou a controlar os estados da fermentação alcoólica e consequentemente podendo aplicar a rede Fuzzy de controle.

Após a arquitetura Fuzzy ser aplicada foi desenvolvido um sistema supervisor para os estados de processo de fermentação e com isso foi gerado uma tabela com os resultados do controle PID aplicado no sistema. De acordo com o relatório concluiu-se que controle contínuo no processo apresentou uma boa aplicação para esse estudo de caso.

O artigo descrito está relacionado nessa monografia por utilizar o controlador PID em um Processo Industrial assim com a bancada irá simular a aplicação de Controle Contínuo em Processo Industrial.

3.3 DESENVOLVIMENTO E CONSTRUÇÃO DE UM PROTÓTIPO PARA A BANCADA DE TESTE DE LABORATÓRIO, DE UM SISTEMA SUPERVISÓRIO PARA MONITORIAMENTO E CONTROLE DE NÍVEL, TEMPERATURA E VAZÃO DE UM PROCESSO INDUSTRIAL SIMULADO

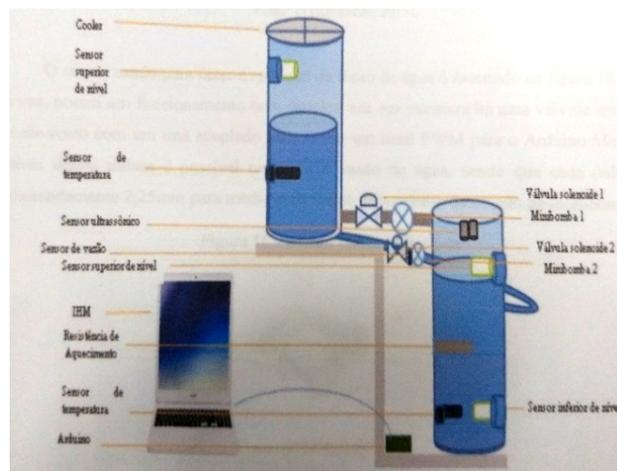
Na monografia de Silva (2018), foi desenvolvido um protótipo de simulação de um processo produtivo que controla as variáveis de Nível, Vazão e Temperatura com um Sistema Supervisor que monitora o funcionamento do Processo Industrial através do Software SCADA Br.

Primeiramente foi realizado um levantamento bibliográfico com o objetivo de escolher os componentes a ser utilizados para a construção da bancada NVT, conforme a figura 10. Com isso foram selecionados todos os sensores e acionamentos assim como a escolha de utilizar o Arduino Mega.

Após a construção da Bancada NVT, foi desenvolvido o Sistema Supervisório através do SCADA Br, assim foi realizado a aquisição de dados de todo o sistema e o monitoramento dos componentes que compõe todo o protótipo. Para controlar as variáveis de Nível, Vazão e Temperatura foram utilizados o controle On-Off.

Esta monografia servirá de base para o desenvolvimento desse trabalho, pois será projetado um controlador PID para a Bancada NVT.

Figura 10 – Descrição geral da Bancada didática NVT



Fonte: (SILVA, 2018)

4 METODOLOGIA

Ao decorrer do desenvolvimento deste trabalho, foi projetado um controle PID no processo de temperatura utilizando a plataforma de prototipagem Arduino para a atuação do controlador no sistema e um sistema supervisório que irá monitorar as variáveis de nível e temperatura da bancada, para ser realizado a comunicação entre Arduino e o sistema supervisório foi utilizado ao protocolo de comunicação Modbus Serial.

Para o desenvolvimento do sistema supervisório foi usado o software Elipse SCADA onde na aquisição de dados tem a função *master* (mestre) e o Arduino como *slave* (escravo), portanto um obedecerá aos comandos do outro, sendo assim foi montado um layout do processo onde é feito o monitoramento do protótipo.

4.1 Arquitetura do Sistema Original

A bancada NVT originalmente é composto por 2 reservatórios onde o primeiro tanque aquece o líquido com uma resistência elétrica e o segundo resfria a água com um mini cooler cada um determinado com o nível máximo e mínimo para a execução de cada processo. A figura 11 mostra a arquitetura do processo e a figura 12 demonstra o funcionamento do protótipo.

Figura 11 – Arquitetura do processo

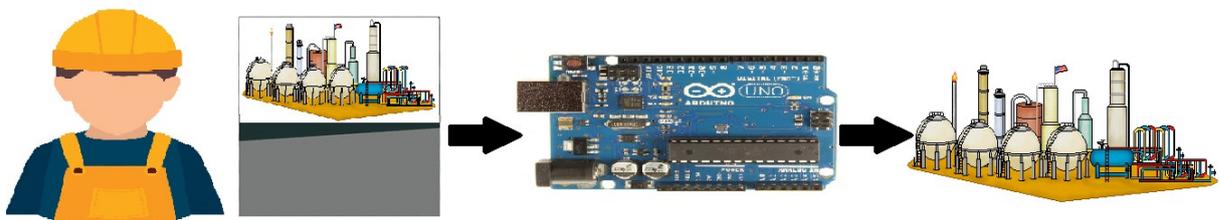


Figura: (Autor)

Figura 12 – Funcionamento do processo original

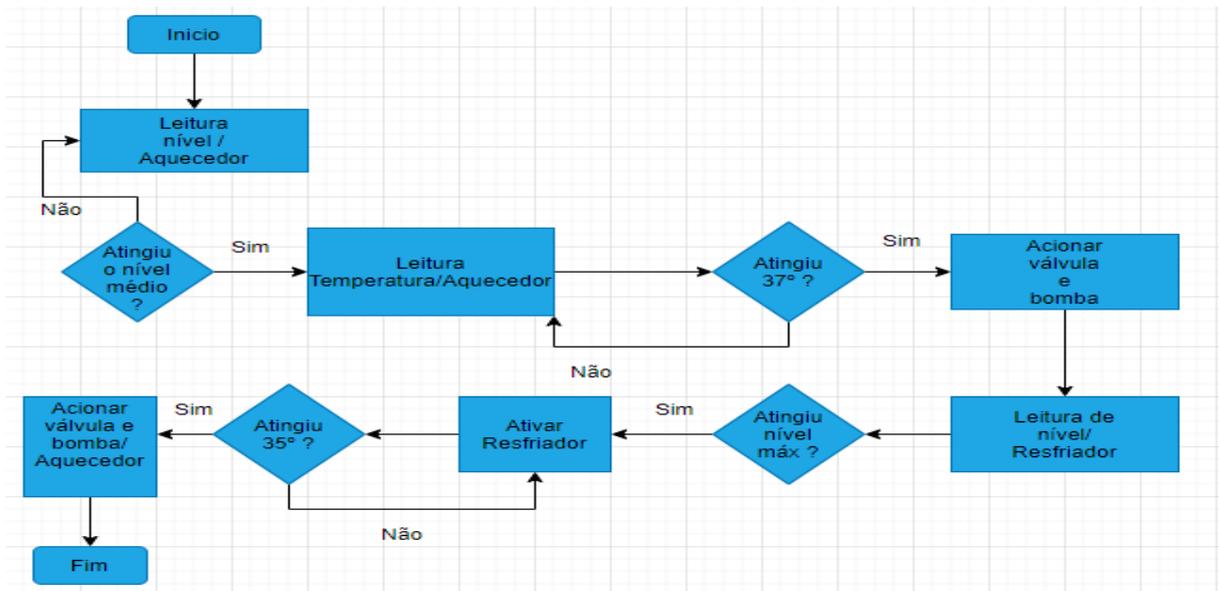


Figura: (Autor)

4.1.2 Arquitetura do Sistema com Alterações

Para aplicar o controle PID de temperatura foi realizado a alocação do mini cooler do tanque superior para o tanque inferior integrando junto com a resistência elétrica e o sensor de temperatura, formando assim a malha de temperatura. Para fazer o controle de nível foi adicionado um suporte e uma válvula de retenção, para impedir a passagem de água do tanque inferior para o superior. Assim como o acréscimo de uma torneira com acionamento por servo motor e uma torneira para o retornar à água para o tanque inferior. Sendo assim não houve a necessidade da utilização das válvulas solenoides, com isso foi utilizado apenas uma mini bomba. A figura 13 mostra a arquitetura do protótipo com as alterações e a figura 14 descreve o funcionamento do protótipo com as modificações.

Figura 13 – Arquitetura do protótipo com as alterações

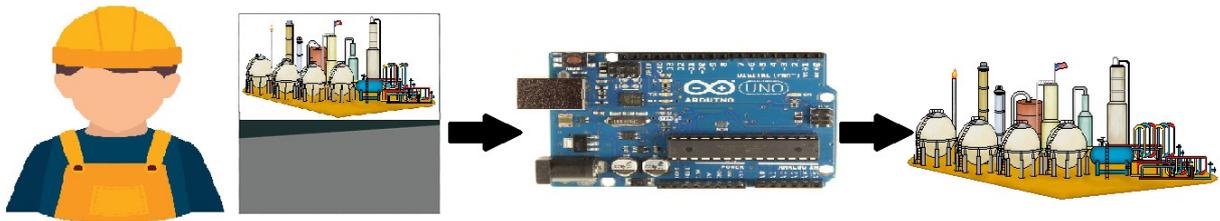


Figura: (Autor)

Figura 14 – Funcionamento do processo com modificações

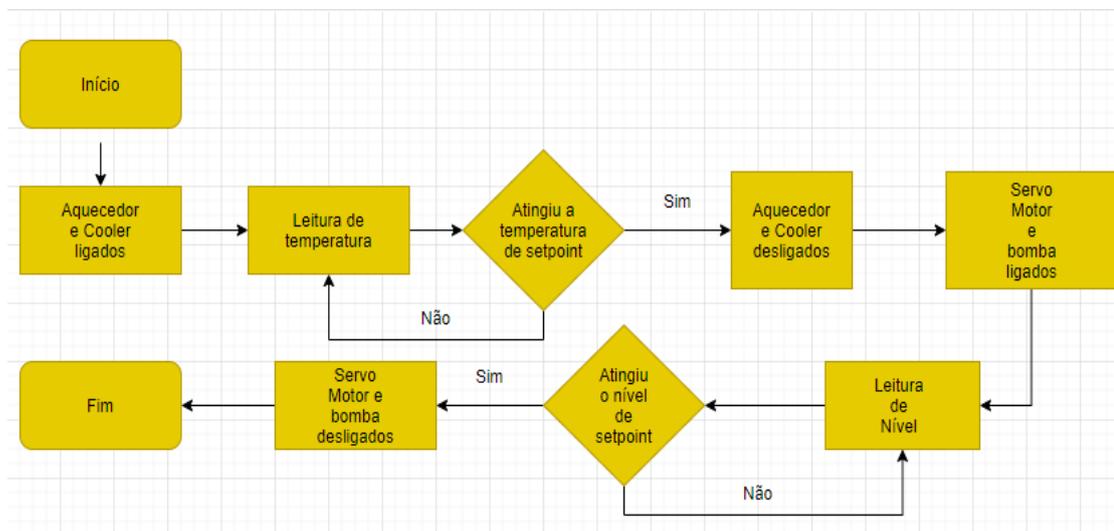


Figura: (Autor)

4.2 Bancada NVT

A bancada NVT é um protótipo didático que possibilita a aplicação de Automação e Controle em um processo Industrial que controla e monitora as variáveis de nível, vazão e temperatura.

O protótipo é composto pelos seguintes componentes:

- **Minibomba 1:** permite o bombeamento da água que se encontra no tanque superior para o inferior;
- **Minibomba 2:** permite o bombeamento da água que se encontra no tanque inferior para o tanque superior;

- **Válvula solenoide 1:** realiza o escoamento de água do tanque superior para o tanque inferior;
- **Válvula solenoide 2:** realiza o escoamento de água do tanque inferior para o tanque superior;
- **Tanque superior:** reservatório onde é realizado o resfriamento da água;
- **Tanque inferior:** reservatório onde é realizado o aquecimento da água;
- **Sensor de vazão:** faz a medição da bomba para o tanque superior;
- **Sensor inferior e superior de nível:** indica o nível máximo e mínimo do tanque inferior;
- **Sensor superior de nível:** indica o nível máximo do tanque superior;
- **Sensor ultrassônico:** realiza a medição do nível do tanque inferior;
- **Resistor de aquecimento:** realiza o aquecimento de água no tanque inferior;
- **Sensor DS18B20:** faz a medição da temperatura no reservatório inferior;
- **Cooler:** realiza o resfriamento da água no reservatório superior.

4.2.1 Elementos da Bancada

4.2.1.1 Sensores

Sensores são elementos físicos que sofrem estímulos externos de variáveis físicas como nível, vazão e temperatura, transformando esses estímulos em sinais elétricos correspondentes que podem ser transformados em valores numéricos dentro do protótipo Arduino.

Entre os sensores que compõe o protótipo estão o sensor ultrassônico HC-SR04 como mostra na figura 15, utilizado para medir o nível do reservatório superior, o sensor de temperatura DS18B20 para medição de temperatura de líquidos, o sensor de vazão que realiza a medição do fluxo de água proveniente da sucção da minibomba e o sensor de nível que realiza a medição máxima e mínima do nível de água. Neste trabalho serão utilizados o sensor

ultrassônico e o sensor de temperatura sendo o primeiro localizado na malha de temperatura e o segundo na malha de nível. Os sensores descritos podem ser vistos abaixo.

Figura 15 – Sensor Ultrassônico HC-SR04



Fonte: (FILIPEFLOP,2020).

Figura 16 – Sensor de Vazão



Fonte: (FILIPEFLOP, 2018).

Figura 17 – Sensor de Nível de Água



Fonte: (FILIPEFLOP,2020).

Figura 18 – Sensor DS18B20



Fonte: (MADEIRA,2020).

4.2.1.2 Atuadores

Os Atuadores são dispositivos que possuem a capacidade de alterar variáveis físicas de interesse em um processo através da ação de um controlador ou sistema de controle. Os atuadores que fazem parte da bancada estão a resistência elétrica de aquecimento que faz com que a água aumente sua temperatura, um mini cooler de 12 volts que é responsável pelo resfriamento do líquido e 2 Minibombas de 12v sendo uma responsável por impulsionar a água para o tanque superior para o inferior e outra com a função de compelir a água do reservatório inferior para o reservatório superior. Para este trabalho serão utilizados a resistência elétrica, o mini cooler e um servo motor SG90 em conjunto com uma torneira. Os atuadores utilizados podem ser vistos abaixo.

Figura 19 – Resistencia Elétrica 127v/550watt



Fonte: (Autor).

Figura 20 – Minibomba 12v



Fonte: (ALIBABA.COM,2020).

Figura 21 – Mini Cooler 12v



Fonte: (Autor).

Figura 22 – Servo Motor SG90



Fonte: (FILIPEFLOP,2020).

4.3 Sistema de Controle

4.3.1 Hardware de controle

Para aplicar o controle PID de temperatura foi utilizado o kit de prototipagem Arduino Uno, sendo a execução do controlador será feita através de códigos que enviará um sinal para a resistência elétrica e o mini cooler onde os mesmos atuaram de acordo com a lógica de programação feita neste trabalho. Também será usado o Arduino para realizar o controle de nível.

Figura 23 – Arduino Uno



Fonte: (STORE ARDUINO,2020).

4.3.2 Malha de controle de temperatura

Para aplicar o controle de temperatura foi necessária uma adequação do protótipo em relação a montagem original, com isso foi colocado a resistência elétrica no tanque superior junto com o cooler tendo assim maior precisão para controlar a temperatura do fluido e diminuir o erro causado pela retenção de calor causado pelo aquecedor mesmo após desligado.

Para desenvolver a malha de controle e projetar o controle PID de temperatura é preciso ter o modelo matemático da planta de temperatura. Para obter a função de transferência do sistema térmico foi feito estudos de modelagens térmicas que descreve a equação da planta relacionada ao aquecimento do fluido no reservatório, com isso foi obtido a seguinte função abaixo, onde T é a temperatura de saída, H é a taxa de entrada de calor, R é a resistência térmica e C é a capacidade térmica.

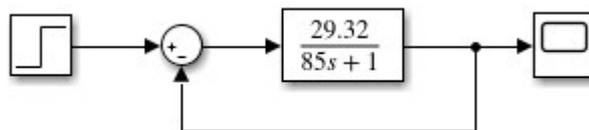
$$\frac{T(s)}{H(s)} = \frac{R}{RCs+1}$$

Sendo a capacidade térmica do reservatório $C = 2,9 \text{ Kcal/Kg}^\circ\text{C}$, a resistência elétrica análoga a resistência térmica e com isso $R = 29,32 \text{ }^\circ\text{C s/Kcal}$, a equação final será.

$$\frac{T(s)}{H(s)} = \frac{29,32}{85s+1}$$

Após desenvolver o modelo matemático relacionado ao aquecimento, o digrama de blocos da malha de temperatura será.

Figura 24 – Diagrama de blocos da malha de temperatura



Fonte: (Autor)

4.3.3 Projeto de controladores PID com abordagem de otimização computacional

Para o uso desse método foi utilizado o software MATLAB para elaborar o algoritmo onde definiu-se os parâmetros do controlador PID. A equação para esse projeto é dada por.

$$G(s) = K \frac{(s+a)^2}{s}$$

Para encontrar os valores de K e a foi determinado uma região de busca onde a combinação dos parâmetros do controlador aplicado a planta obtenha uma resposta com um sobressinal menor que 20%. A região de busca foi obtida através da medição de temperatura onde foi observado que a para a temperatura de aquecimento variar 1°C em um intervalo de tempo de aproximadamente 10s como mostra a figura abaixo.

Figura 25 – Tempo para temperatura variar 1°C

```
11:48:40.842 -> 27.87
11:48:51.445 -> 28.12
11:49:02.042 -> 28.94
11:49:12.636 -> 29.62
11:49:23.237 -> 30.37
11:49:33.836 -> 31.19
11:49:44.441 -> 32.13
11:49:55.082 -> 32.94
11:50:05.650 -> 33.88
```

Fonte: (Autor)

Com isso a região de busca de k e a foi.

$$1 \leq K \leq 10$$

$$1 \leq a \leq 10$$

A próxima etapa foi a elaboração do algoritmo no MATLAB que irá encontrar os melhores valores para os parâmetros do controlador PID, como mostra a figura abaixo.

Figura 26 – Algoritmo para encontrar os parâmetros do controlador

```

K = [1 2 3 4 5 6 7 8 9 10];
a = [1 2 3 4 5 6 7 8 9 10];

t = 0:0.01:5;
g = tf([29.32],[85 1]);
k = 0;
for i = 1:10;
    for j = 1:10;
        gc = tf(K(i)*[1 2*a(j) a(j)^2], [1 0]);
        G = gc*g/(1 + gc*g);
        y = step(G,t);
        m = max(y);
        if m < 2.20
            k = k+1;
            resposta(k,:) = [K(i) a(j) m];
        end
    end
end
end

```

Figura: (Autor)

A resposta das combinações de k, a e m (sobressinal) que o algoritmo gerou foi respectivamente.

Figura 27 – Conjunto de combinações de k, a e sobressinal

resposta =			3.0000	2.0000	1.0997
			3.0000	3.0000	1.1002
1.0000	1.0000	1.2106	3.0000	4.0000	1.1004
1.0000	2.0000	1.2149	3.0000	5.0000	1.1006
1.0000	3.0000	1.2163	3.0000	6.0000	1.1007
1.0000	4.0000	1.2170	3.0000	7.0000	1.1007
1.0000	5.0000	1.2175	3.0000	8.0000	1.1008
1.0000	6.0000	1.2177	3.0000	9.0000	1.1008
1.0000	7.0000	1.2180	3.0000	10.0000	1.1009
1.0000	8.0000	1.2181	4.0000	1.0000	1.0779
1.0000	9.0000	1.2183	4.0000	2.0000	1.0790
1.0000	10.0000	1.2183	4.0000	3.0000	1.0794
2.0000	1.0000	1.1332	4.0000	4.0000	1.0796
2.0000	2.0000	1.1355	4.0000	5.0000	1.0797
2.0000	3.0000	1.1362	4.0000	6.0000	1.0798
2.0000	4.0000	1.1366	4.0000	7.0000	1.0798
2.0000	5.0000	1.1368	4.0000	8.0000	1.0799
2.0000	6.0000	1.1369	4.0000	9.0000	1.0799
2.0000	7.0000	1.1371	4.0000	10.0000	1.0799
2.0000	8.0000	1.1371	5.0000	1.0000	1.0646
2.0000	9.0000	1.1372	5.0000	2.0000	1.0655
2.0000	10.0000	1.1372	5.0000	3.0000	1.0658
3.0000	1.0000	1.0982	5.0000	4.0000	1.0660
5.0000	5.0000	1.0661	7.0000	8.0000	1.0493
5.0000	6.0000	1.0661	7.0000	9.0000	1.0494
5.0000	7.0000	1.0661	7.0000	10.0000	1.0494
5.0000	8.0000	1.0662	8.0000	1.0000	1.0428
5.0000	9.0000	1.0662	8.0000	2.0000	1.0434
5.0000	10.0000	1.0662	8.0000	3.0000	1.0436
6.0000	1.0000	1.0552	8.0000	4.0000	1.0437
6.0000	2.0000	1.0560	8.0000	5.0000	1.0437
6.0000	3.0000	1.0562	8.0000	6.0000	1.0438
6.0000	4.0000	1.0564	8.0000	7.0000	1.0438
6.0000	5.0000	1.0564	8.0000	8.0000	1.0438
6.0000	6.0000	1.0565	8.0000	9.0000	1.0438
6.0000	7.0000	1.0565	8.0000	10.0000	1.0438
6.0000	8.0000	1.0565	9.0000	1.0000	1.0385
6.0000	9.0000	1.0566	9.0000	2.0000	1.0390
6.0000	10.0000	1.0566	9.0000	3.0000	1.0392
7.0000	1.0000	1.0482	9.0000	4.0000	1.0393
7.0000	2.0000	1.0489	9.0000	5.0000	1.0393
7.0000	3.0000	1.0491	9.0000	6.0000	1.0393
7.0000	4.0000	1.0492	9.0000	7.0000	1.0394
7.0000	5.0000	1.0493	9.0000	8.0000	1.0394
7.0000	6.0000	1.0493	9.0000	9.0000	1.0394
7.0000	7.0000	1.0493	9.0000	10.0000	1.0394
10.0000	1.0000	1.0350			
10.0000	2.0000	1.0354			
10.0000	3.0000	1.0356			
10.0000	4.0000	1.0357			
10.0000	5.0000	1.0357			
10.0000	6.0000	1.0357			
10.0000	7.0000	1.0358			
10.0000	8.0000	1.0358			
10.0000	9.0000	1.0358			
10.0000	10.0000	1.0358			

Fonte: (Autor)

De acordo com resultado que o programa gerou a melhor combinação dos valores de k e a para o menor valor de sobressinal foi

$$K = 3 \text{ e } a = 3 \text{ com sobressinal } m = 1.1002.$$

Então a equação final com os parâmetros k e a é.

$$G(s) = \frac{3s^2 + 18s + 27}{s}$$

Sabendo que o controlador PID em relação a Kd, Kp e Ki é.

$$G(s) = \frac{Kd s^2 + Kp s + Ki}{s}$$

Então os valores de Kd = 3, Kp = 18 e Ki = 27.

4.3.4 Aplicação do controlador PID na Malha de temperatura

A influência do controle PID na planta de temperatura foi desenvolvido no IDLE do Arduino que se inicia através da declaração dos ganhos do controlador junto com a aplicação do sinal Pwm, após isso foi elaborado a fórmula que descreve os parâmetros proporcional, derivativo e integrativo assim como a aplicação deles na resistência elétrica integrado junto ao cooler de acordo com a figura 28.

Figura 28 – Programação do controle PID no Arduino

```

#include <OneWire.h>
#define Tlcontrol 6
int DS18S20_Pin = 7;
OneWire ds(DS18S20_Pin);
float setpoint;
int cooler = 9;
float setin;
float error = 0;
float P = 0, I = 0, D = 0;
int controlePwm = 50;
float Kp = 18.00 , Ki = 27.00 , Kd = 3.00;
float PID;
float pastemperature;
void setup() {
  pinMode(cooler, OUTPUT);
  pinMode(Tlcontrol, OUTPUT);
  setin = 25.00;
  setpoint = 35.00;
  analogReference(EXTERNAL);
  Serial.begin(9600);
  delay(1000);
}

void loop() {
  float temperature = getTemp();
  Serial.println(temperature);
  error = setin - temperature;
  P = error*Kp;
  I = I + error*Ki;
  D = (pastemperature - temperature)*Kd;
  PID = P + I + D;
  controlePwm = (PID + 50);
  delay(500);
  if (getTemp() < setpoint){
    digitalWrite(Tlcontrol, HIGH);
    analogWrite(Tlcontrol, controlePwm);
    delay(5000);
    digitalWrite(cooler, HIGH);
    delay(5000);
  }

  if (getTemp() > setpoint){
    digitalWrite(cooler, HIGH);
    delay(5000);
    digitalWrite(Tlcontrol, LOW);
  }
}

```

Fig

ura: (Autor)

4.4 Sistema Supervisório da bancada NVT

O sistema supervisório do protótipo tem como função monitorar as variáveis de nível e temperatura, para isso foi realizados algumas alterações físicas da bancada em relação ao trabalho anterior. As mudanças se deram através da retirada das válvulas solenoides e o acréscimo de duas torneiras, uma válvula de retenção que impede a entrada de água no fluxo contrário, um suporte para elevar o segundo tanque que faz com que não venha água do primeiro tanque e um servo motor modelo SG90 que controla a passagem de água no segundo reservatório. A descrição das alterações pode ser vista na figura abaixo.

Figura 29 – Alterações no reservatório de nível

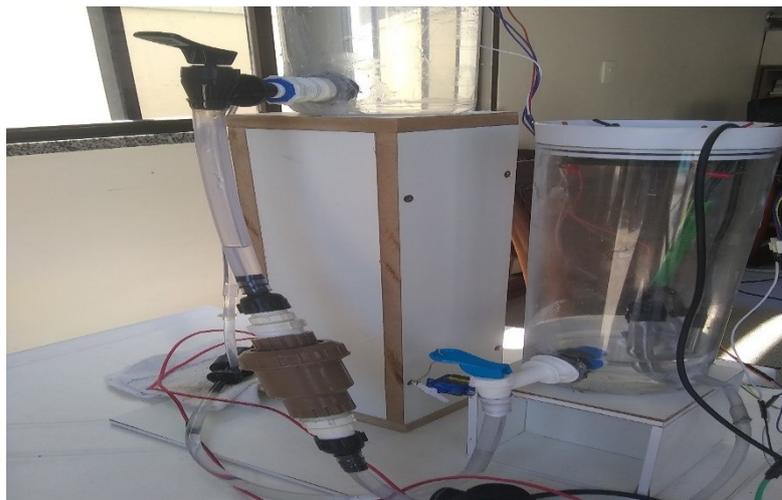


Figura: (Autor)

Para a montagem do sistema supervisório foi utilizado o software Eclipse SCADA que permite a comunicação entre o Arduino e a monitoração das variáveis de processo do protótipo, a comunicação se deu através do protocolo modbus, por meio de uma comunicação física USB. O Eclipse SCADA terá a função de *Master*, enquanto o Arduino terá a função de *slave*.

4.4.1 Biblioteca Modbus Serial

A utilização da biblioteca se deu através da instalação da biblioteca SimpleModbusSlave no IDLE do Arduino para a comunicação entre o Arduino e Eclipse SCADA. Nela constam as funções primordiais para o microcontrolador se comunicar com software de supervisão.

4.4.2 Código de Controle de Nível e Temperatura do processo

A bancada é composta por dois tanques, onde o primeiro tanque tem a função de controlar a temperatura do líquido e o segundo para controlar o nível. O controle de nível se dá através do fechamento e abertura da torneira que é acionado com um servo motor onde fica o primeiro reservatório. O segundo reservatório irá receber o nível de água do primeiro, onde

junto com a minibomba e um sensor ultrassônico pertencentes ao sistema de nível como mostra a figura 30.

O tanque de temperatura tem como função permitir que a temperatura água aqueça e chegue ao setpoint de modo estável como mostra a figura 31. O controle aplicado como é visto na secção anterior é um PID. Nesse Reservatório contém um cooler 12V, uma resistência elétrica e um sensor de temperatura.

Figura 30- Código controle de nível do processo

```
void loop() {
  digitalWrite(TRIG, LOW);
  delayMicroseconds(10);
  digitalWrite(TRIG, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG, LOW);
  duracao = pulseIn(ECHO, HIGH);
  distancia = duracao / 58;
  servo.write(120);
  delay(1000);
  servo.write(0);
  delay(1000);
  digitalWrite(bomba, HIGH);
  if(distancia < setpoint){
    servo.write(0);
    digitalWrite(bomba, LOW);
  }

  modbus_update();

  holdingRegs[DIS_VAL] = distancia;
}
```

Figura: (Autor)

Figura 31- Código controle de temperatura do processo

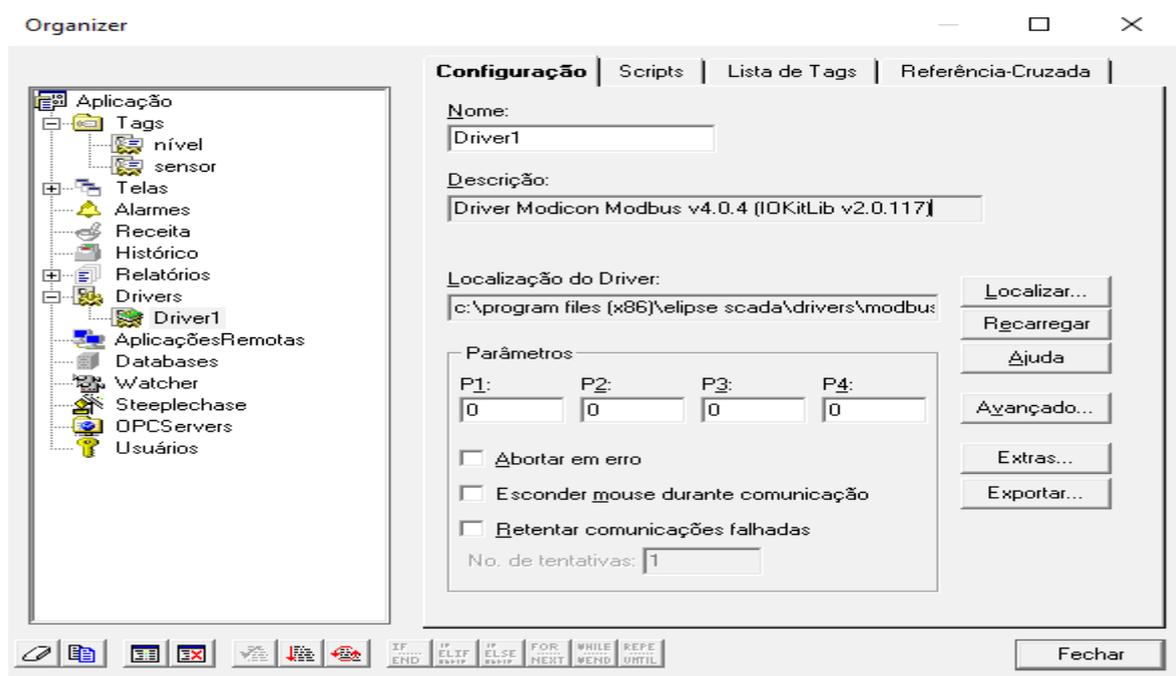
```
void loop() {  
  float temperature = getTemp();  
  Serial.println(temperature);  
  error = setin - temperature;  
  P = error*Kp;  
  I = I + error*Ki;  
  D = (pastemperature - temperature)*Kd;  
  PID = P + I + D;  
  controlePwm = (PID + 50);  
  delay(500);  
  if (getTemp() < setpoint){  
    digitalWrite(Tlcontrol,HIGH);  
    analogWrite(Tlcontrol, controlePwm);  
    delay(5000);  
    digitalWrite(cooler,HIGH);  
    delay(5000);  
  }  
  
  if (getTemp() > setpoint){  
    digitalWrite(cooler,HIGH);  
    delay(5000);  
    digitalWrite(Tlcontrol,LOW);  
  }  
}
```

Figura: (Autor)

4.4.3 Comunicação Eclipse SCADA

Para acontecer a interação do Arduino Uno com o Eclipse SCADA é necessário instalar o driver Modicon Modbus v4.0.4 (IOKitLib v2.0.117) que permitirá o software ter acesso a comunicação com o meio externo (Arduino) como mostra a figura 32.

Figura 32 – Driver de comunicação



Fonte: (Autor)

Logo após instalar o driver foi necessário realizar a configuração de comunicação de alguns parâmetros como por exemplo: a porta escolha da porta COM3 utilizada com a mesma do Arduino e a taxa de transmissão de 9600 bps. No algoritmo foi definido a data bit de 8 bits e um stop bit de 1 bit, com isso foi defino na configuração esses respectivos valores como mostra a figura 33.

Figura 33 – Definição dos parâmetros de comunicação

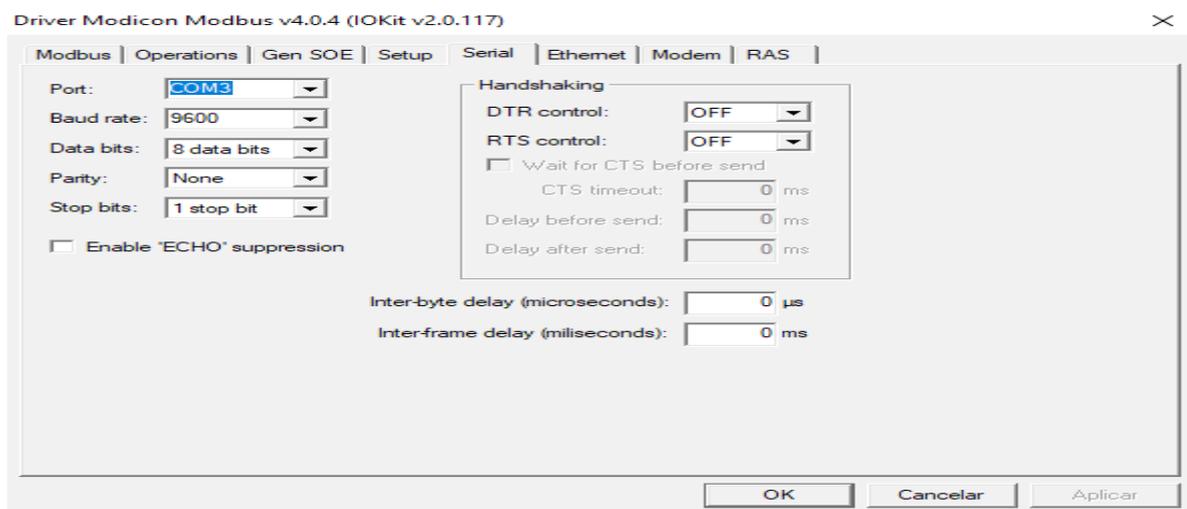
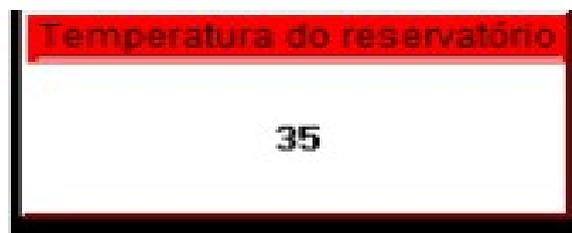


Figura: (Autor)

4.4.4 Layout do Sistema supervisorio

Nesta seção será apresentado todos os módulos do sistema supervisorio da bancada NVT como mostra a Figura 34, onde mostra a tela de medição em tempo real da temperatura do reservatório inferior com formato de tela.

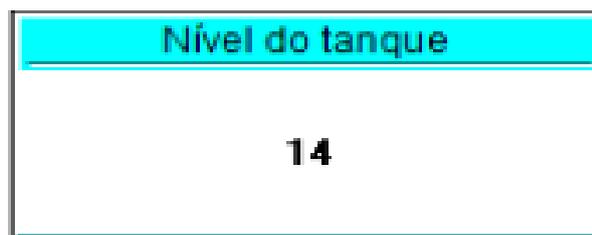
Figura 34 – Temperatura do reservatório 1



Fonte: (Autor)

Na figura 35 é possível identificar o nível do reservatório superior em tempo real.

Figura 35 – Nível no reservatório 2



Fonte: (Autor)

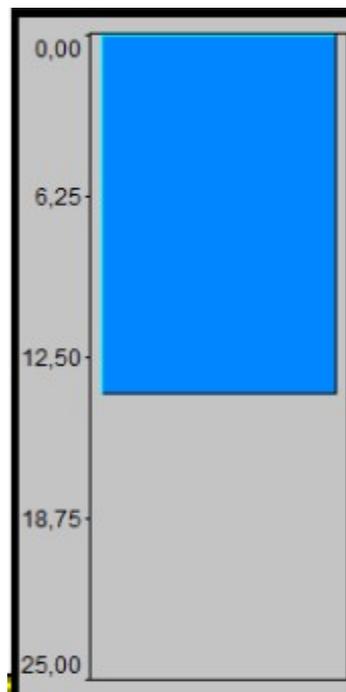
5 RESULTADOS OBTIDOS

Na seção presente são apresentados os resultados obtidos no desenvolvimento do projeto. Foram realizados ensaios no protótipo para ter embasamento teórico na avaliação do funcionamento do projeto. Para o controle PID de temperatura foram apresentados gráficos de resposta para que seja avaliado o desempenho do sistema. São apresentados códigos de algoritmos de controle de nível, temperatura e resultados do funcionamento do sistema supervisor do processo.

5.1 Reservatório de Nível

Foram realizados testes de medição com o sensor ultrassônico para verificar o seu funcionamento e o mesmo obteve respostas positivas como mostra a figura 36, sendo integrado junto ao sistema de nível a bomba, a válvula de retenção e o servo motor para acionar a torneira do tanque inferior.

Figura 36 – Medidor analógico de nível tanque 2



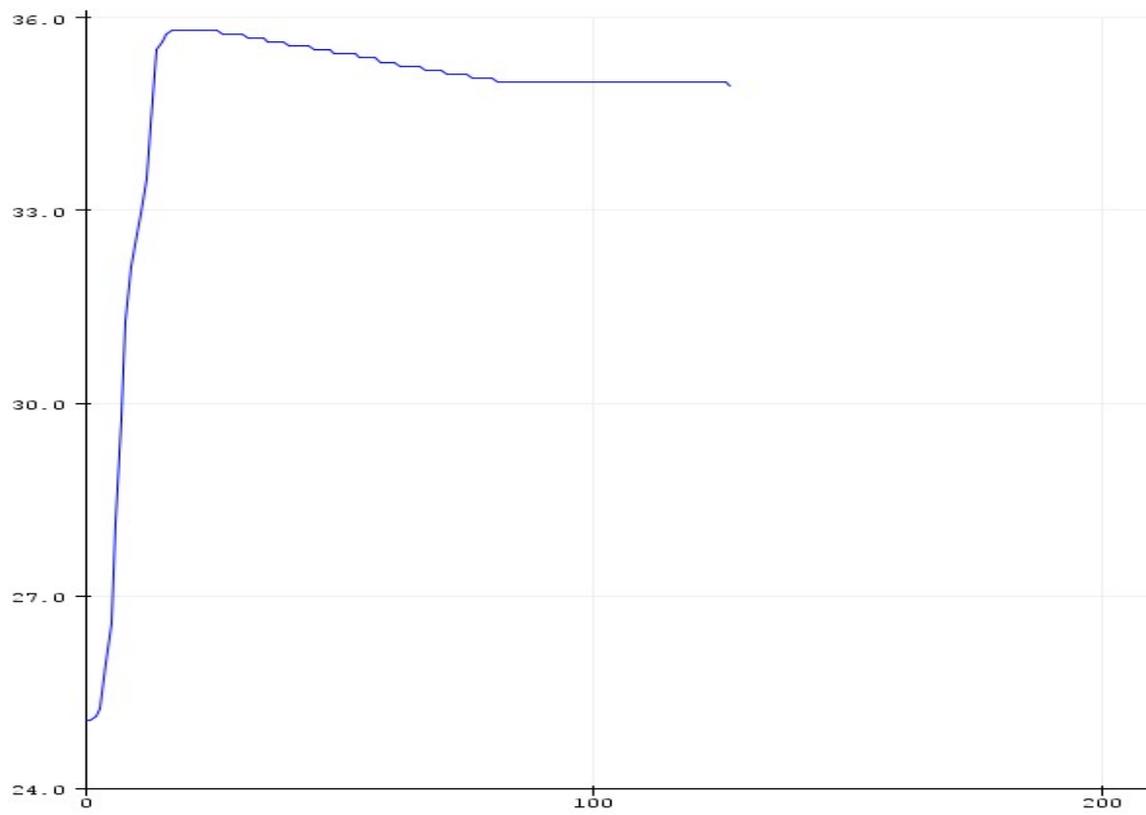
Fonte: (Autor)

5.2 Reservatório de Temperatura

O resultado esperado para esse reservatório é que a água aqueça a uma temperatura desejada de modo que diminua o erro estacionário e que obtenha uma resposta mais rápida em relação ao setpoint. Ao fazer os ensaios nesse tanque com a aplicação do controle PID o comportamento do sistema térmico se apresentou dentro do esperado, onde houve a diminuição do erro estacionário e uma resposta mais rápida em relação ao sistema térmico sem o controle PID.

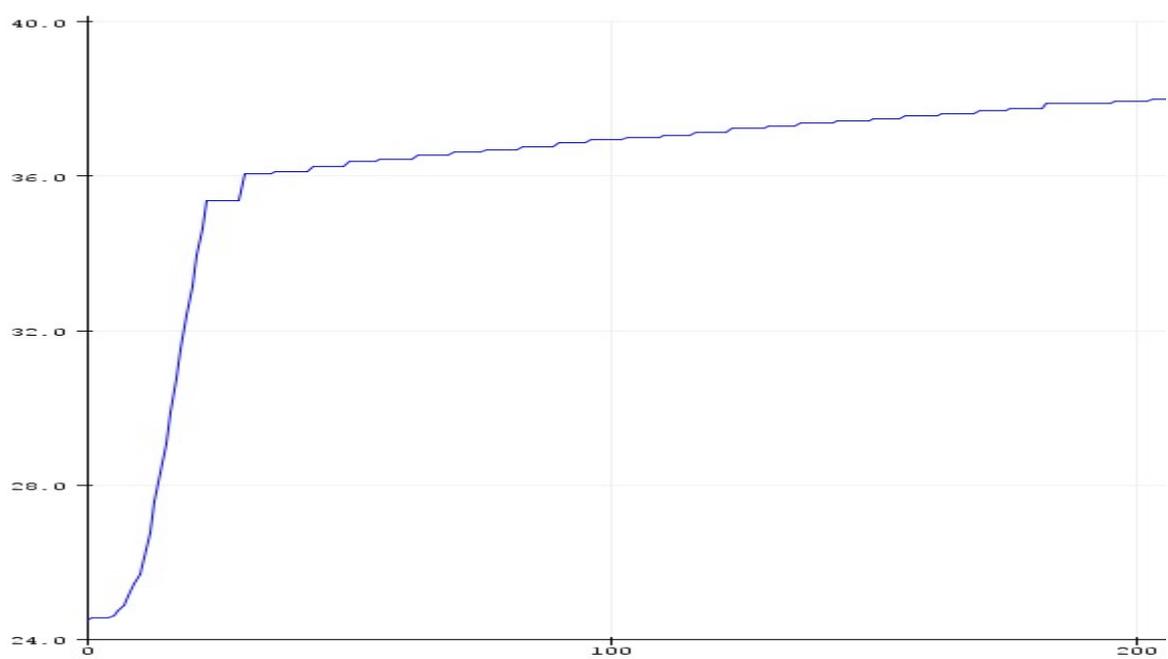
5.2.1 Controle PID de temperatura

A Aplicação do controle PID de temperatura no reservatório de aquecimento apresentou o resultado esperado, onde o gráfico do sistema térmico com o controle PID (Figura 37) apresentou uma resposta mais rápida para chegar à temperatura de setpoint e uma diminuição significativa do erro estacionário em relação ao sistema sem a utilização do cooler (Figura 38) e com a utilização do cooler (Figura 39).



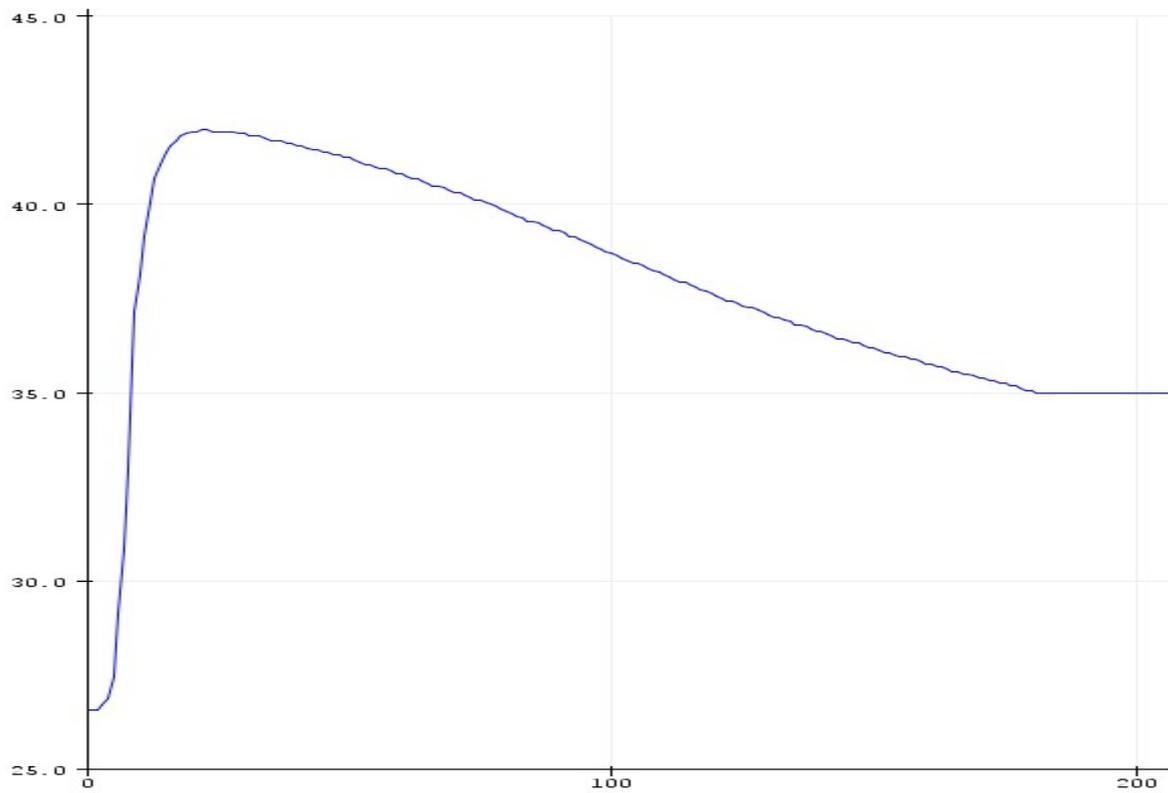
Fonte: (Autor)

Figura 38 – Resposta do sistema sem o cooler



Fonte: (Autor)

Figura 39 – Resposta do sistema com o cooler



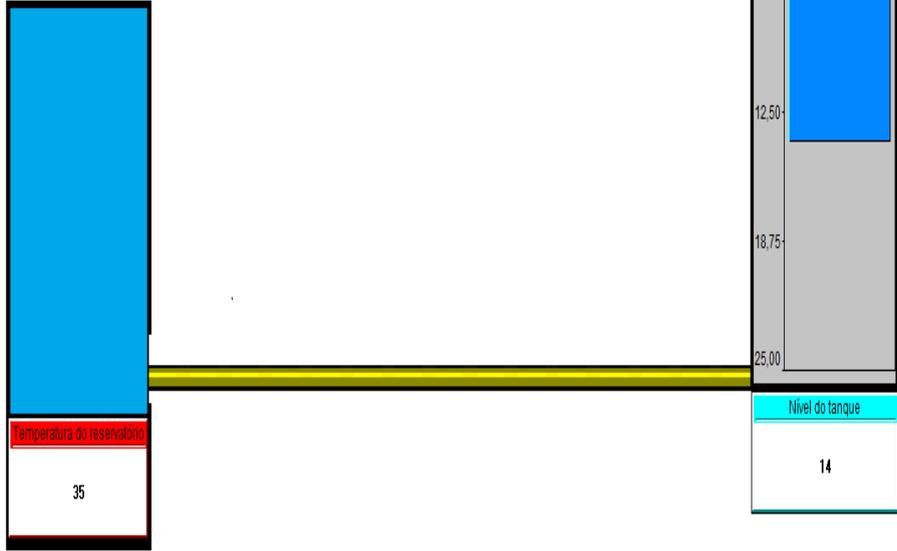
Fonte: (Autor)

5.3 Sistema Supervisório

Foram feitos ensaios com o uso do Eclipse SCADA e o resultado foi o esperado onde a comunicação entre o Arduino e o software ocorreu de forma correta e a leitura dos sensores para a monitoração dos processos não apresentou problemas como mostra a figura 40.

Figura 40 – Sistema supervisório do processo

Sistema Supervisório



Fonte: (Autor)

6 CONCLUSÃO

Neste trabalho tiveram como objetivos principais desenvolver um controlador PID na malha de temperatura e desenvolver um sistema supervisório para as variáveis de nível e temperatura, mas para a sua conclusão deveriam ser realizados ajustes no sistema térmico para que a medição de temperatura não gerasse erros de medição e conseqüentemente não correspondesse ao valor desejado pelo usuário.

Os ajustes no sistema térmico levaram a realocação física dos componentes da bancada em relação ao protótipo original e ao desenvolvimento de um controlador PID, que por sua vez foi projetado no MATLAB e aplicado na programação do Arduino e com isso foi gerado os resultados e pode-se avaliar uma melhoria no desempenho do sistema.

Para realizar o sistema supervisório no processo de nível foram feitos ajustes com a inclusão de duas torneiras, um servo motor, uma válvula de retenção e um suporte e com isso foi realizado o desenvolvimento do sistema supervisório de nível, fazendo com que o sistema de nível seja monitorado em tempo real e por ultimo foi realizado sistema supervisório de temperatura onde também ocorreu o monitoramento na variável térmica, com isso abre possibilidades de ajustes para trabalhos futuros como a aplicação do controle PID multivariável de nível e vazão melhorando assim o desempenho nessas variáveis.

6.1 DIFICULDADES ENCONTRADAS

As principais dificuldades encontradas se deram através da comunicação entre o Eclipse SCADA e o Arduino Uno para realização de teste de leitura tanto para a variável de nível como para a variável de temperatura isso se deu pela inexperiência com a utilização do software envolvido quando se trata dos protocolos de comunicação. Na utilização do servo motor para realizar a abertura o fechamento da torneira foram também uma das maiores dificuldades devido à falta de experiência do projetista.

6.2 TRABALHOS FUTUROS

Nesta seção são apresentadas propostas de trabalho, visando a melhoria para a funcionalidade do protótipo, integrando com o trabalho realizado neste TCC.

Há necessidade de acrescentar uma válvula proporcional para os dois reservatórios como também a utilização de uma bomba mais potente fazendo com que tenha mais precisão para implementar o controle PID de vazão e nível.

Implementar um método de controle PID de temperatura utilizando Redes-Neurais, Lógica-Fuzzy ou Controle Adaptativo fazendo com que sirva de exemplo de estudos na área de Inteligência Artificial.

Outra Proposta é a utilização de remoto através da internet das coisas da bancada através de smartphome, com isso torna mais didático o estudo de controle de processo como também de sistema supervisório.

REFERÊNCIAS BIBLIOGRÁFICAS

dos Santos, Orlem Lima. "Sistema de Controle de Temperatura para uma Estufa com Monitoramento via Aplicativo." *RCT-Revista de Ciência e Tecnologia* 3.4 (2017).

Mendonça, Márcio, et al. "Supervisory System and Multivariable Control Applying Weighted Fuzzy-PID Logic in an Alcoholic Fermentation Process." *Semina Ciências Exatas e Tecnológicas* 36.2 (2015): 95-108.

Barbosa, Célia Maria Dantas, et al. "AUTOMAÇÃO DE UM PROCESSO ALTERNATIVO DA PASTEURIZAÇÃO DO LEITE UTILIZANDO AÇÃO DE CONTROLE ON-OFF." *REVISTA DE ENGENHARIA E TECNOLOGIA* 10.1 (2018): páginas-41.

CASTRUCCI, Plínio de Lauro; BITTAR, Anselmo; SALES, Roberto Moura. **Controle Automático**. Rio de Janeiro: LTC, 2011. 284-285 p.

Zanolla, Leandro, et al. "Implementação com Validação Real de um Controle Proporcional, Integral e Derivativo na Plataforma Robótica RoboDeck." (2017).

Torres, Winnie Lima, et al. "Mathematical Modeling and PID Controller Parameter Tuning in a Didactic Thermal Plant." *IEEE Latin America Transactions* 15.7 (2017): 1250-1256.

SANTOS, ORLEM LD, JODSM JUNIOR, and MENDELSSON RM NEVES. "Sistema de controle de temperatura para uma estufa com monitoramento via aplicativo." *RCT* 1.1 (2016).

Zhao, CheahWai, JayanandJegatheesan, andSonCheeLoon.
"Exploringiotapplicationusingraspberrypi." *InternationalJournalof Computer Networks andApplications* 2.1 (2015): 27-34.

Agrawal, Nikhil, andSmitaSinghal. "Smartdripirrigation system usingraspberrypiandarduino." *InternationalConferenceonComputing, Communication & Automation*. IEEE, 2015.

Campos, Gabriel Simões Coelho de. "Desenvolvimento de bancada didática para controle de nível de 4 tanques, utilizando placa de aquisição de dados NIUSB6009, controlador PID e sistemas supervisórios programados no LabVIEW." (2017).

OGATA, K. Engenharia de Controle Moderno. 5. ed. São Paulo: Pearson Prentice Hall, 2010.

Silva, Márcia de Souza, Desenvolvimento e construção de um protótipo para Bancada de teste de Laboratório, de um Sistema Supervisório para monitoramento e controle de Nível, Temperatura e Vazão de um Processo Industrial simulado. 2018. 55f. Trabalho de Conclusão de Curso – Universidade do Estado do Amazonas, Manaus, 2018.

Estudos e sintonia de controladores industriais. Disponível em:

<<https://player.slideplayer.com.br/1/67593/#>>. Acesso em: 15 nov. 2019.

da Silva Candido, Alessandro, Eduardo Araujo Pereira, and Glaucia Regina Almeida.

"Aquisição de dados do Floboss via sistema supervisorio." *Caderno de Graduação-Ciências Exatas e Tecnológicas-UNIT* 1.3 (2013): 41-48.

Braga, Alexandre. "Interface: Sistema orientado à visualização." *Revista GEMInIS* 6.1 (2015): 125-140.

Costalonga, T., et al. "Interface Homem-Máquina utilizando sensor Kinect para control de um Quadrotor AR. Drone Parrot." *XX Congresso Brasileiro de Automática*. 2014.

Cunha, Maxsuel Ferreira, Thais Christine Borges da Silva, and Carlos Alberto de Souza.

"PLATAFORMA DIDÁTICA TÉRMICA MULTIDISCIPLINAR PARA CONTROLE, AUTOMACAO E REDES DE SENSORES."

STORE ARDUINO. Loja Arduino Site.2020. Disponível em:

<<https://store.arduino.cc/usa/arduino-uno-rev3>>. Acesso em 14 de outubro 2020.

ALIBABA, Loja virtual.2020. Disponível em: <https://www.alibaba.com/product-detail/dc-brushless-bomba-12v-Cooling-bomba_958641065.html?spm=a2700.galleryofferlist.normal_offer.d_title.2a6766castbSY5>.

Acesso em 14 de outubro 2020.

FILIFEFLOP, Loja virtual.2020. Disponível em: <<https://www.filieflop.com/br>> Acesso em 14 de outubro 2020.

VIDADESILÍCIO, Loja virtual.2020. Disponível em: <<https://portal.vidadesilicio.com.br/sensor-de-temperatura-ds18b20>>. Acesso em 14 de outubro de 2020.

