



CONTROLE DE NÍVEL E VAZÃO EM CASCATA UTILIZANDO ARDUINO®

Marcelo Pereira de Oliveira¹, Silas Martins Sousa¹, Gabriele Fernanda de Souza¹, Marcelo Nunes Pereira¹, Nalberth Carvalho Barbosa¹ e Carlos Renato Borges dos Santos¹

¹ IFTM – Instituto Federal do Triângulo Mineiro – *campus* Paracatu

Resumo - O objetivo deste documento é apresentar os resultados dos testes do controle de nível e vazão em cascata utilizando Arduino® em uma planta didática de baixo custo. O controle em cascata consiste em um controlador primário (mestre), que regula um controlador secundário (escravo), aprimorando a velocidade de resposta e minimizando os distúrbios causados pela malha secundária. Esse tipo de controle é muito utilizado em sistemas industriais em que duas variáveis medidas influenciam no controle de uma malha de controle. Para a construção do protótipo da planta didática foi utilizado um Arduino® Mega 2560 como controlador, implementado a lógica de PID no mesmo, como sensor de nível foi utilizado laser-TOF200F, como sensor de vazão foi utilizado YF-S301. Uma planta foi construída para testar o sistema de controle. Testes de desempenho do sistema foram realizados e os estes mostraram a capacidade do projeto de controle em receber o *setpoint* e ser capaz de efetuar as correções necessárias.

Palavras-Chave – Arduino®, Cascata, Controle, Nível, PID, Vazão.

LEVEL AND FLOW CONTROL IN CASCADE USING ARDUINO®

Abstract - The objective of this document is to present the results of level and flow control tests in cascade using Arduino® in a low-cost teaching plant. Cascade control consists of a primary controller (master), which regulates a secondary controller (slave), improving response speed and minimizing disturbances caused by the secondary loop. This type of control is widely used in industrial systems in which two measured variables influence the control of a control loop. To build the prototype of the teaching plant, an Arduino® Mega 2560 was used as a controller, PID logic was implemented in it, a laser-TOF200F was used as a level sensor, and a YF-S301 was used as a flow sensor. A plant was built to test the control system. System performance tests were carried out and

these showed the ability of the control design to receive the *setpoint* and be able to make the necessary corrections.

Keywords - Arduino®, Cascade, Control, Flow, Level, PID.

NOMENCLATURA

<i>CI</i>	Circuito Integrado
<i>PWM</i>	Pulse Width Modulation (Modulação por Largura de Pulso)
<i>I2C</i>	Barramento serial Barramento multimestre desenvolvido pela Philips
<i>PID</i>	Proporcional Integral Derivativo
<i>SETPOINT</i>	Valor pedido
<i>MV</i>	Variável Manipulada (saída de controle)
<i>PV</i>	Variável de Processo (valor medido)
<i>SDA</i>	Serial Data (dados seriais)
<i>SCL</i>	Serial Clock (clock serial)
<i>Vcc</i>	Tensão Contínua
<i>GND</i>	Terra
<i>Time-of-flight</i>	Tempo de voo
<i>Mm</i>	Milímetro
<i>L/min</i>	Litros por minuto

I. INTRODUÇÃO

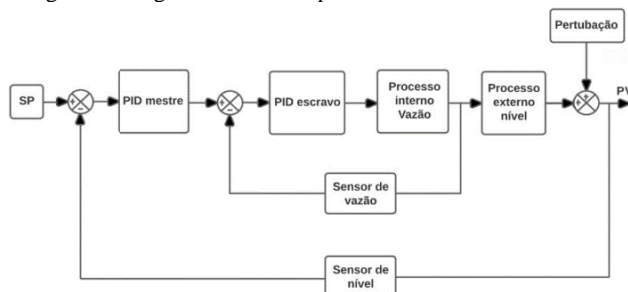
Este estudo abrange uma metodologia prática para controle PID eficaz de nível e vazão em cascata, utilizando várias variáveis em um pequeno projeto de eletrônica embarcada com controle digital de processos. Alguns testes de controle PID demandam de bancadas complexas e caras tendo como exemplo a planta didática PDH-ATK 1001 (*Authomathika*), expostos isso, definimos a construção de uma bancada simples e de baixo custo quando comparada a anterior.

O controle de nível e vazão serão processados pelo microcontrolador CI, Arduino® Mega 2560, uma plataforma de prototipagem eletrônica programável, possibilitando estudos educacionais de alto nível [1] para realizar o controle PID, inserindo-se o *set point* desejado para o controlador primário (mestre) que escreve no controlador secundário (escravo).

O fluido em análise é a água, por não ser um líquido invasivo que danifique os dispositivos eletrônicos do protótipo. A utilização dos sensores de nível e vazão são destinados a leitura da distância (mm) e vazão em (L/min) para regulação modulação PWM, por meio do controle PID.

O controle em cascata inicia inserindo-se o *setpoint* desejado para o controlador primário (mestre). A saída deste controlador primário será justamente a entrada do controlador secundário (escravo). A adição de uma malha no sistema, tem o objetivo de diminuir a influência de uma segunda variável física que também influencia a variável de processo. Além disso, aumenta a estabilidade e capacidade de rejeitar perturbações externas [4]. A Figura 1 mostra um diagrama de blocos típica desse sistema de controle.

Figura 1: Diagrama de blocos para controle de Nível e Vazão.



II. MATERIAIS E MÉTODOS

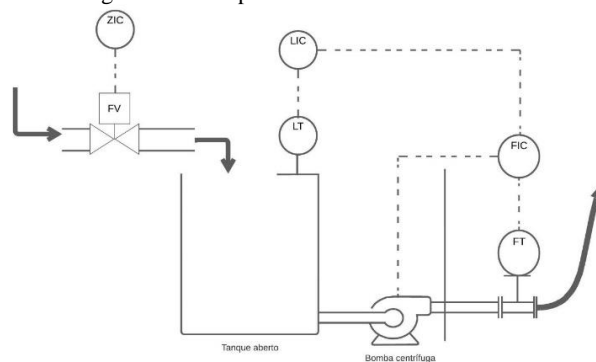
Utilizando-se do microcontrolador Arduino® Mega 2560, este trabalho consistiu em realizar o controle de alta performance a nível educacional, através de comandos para a válvula ligada ao servo motor de enchimento do tanque e para bomba de controle do nível de água, conforme as leituras e armazenamento de dados via sensor de nível a *laser* TOF200F [10] e vazão de água YF-S301[7]. Através desses componentes principais possibilitou a aplicação para eletrônica embarcada e controle de processos, bem como análise de perturbações na malha de controle. A Tabela 1 mostra a lista de equipamentos utilizada no protótipo.

Tabela 1: Lista de Materiais utilizados.

Item	Quantidade
Arduino (Mega 250)	1
Tanque	1
Mini bomba D'água 12V 800L/h coluna 5m	1
Fonte ATX - LC-8460BTX	1
Ponte H L298N	1
Sensor a <i>laser</i> -TOF200F	1
Sensor de fluxo YF-S301	1
Servo Motor Mg995 13kg	1
Suporte Acrílico [fundo vazado]	1
Resina de Poliéster	1
Torneira 1/4 de volta	1
Cabos	vários

Para iniciar o projeto foi desenvolvido o diagrama de instrumentação do sistema, também chamado de *P&ID*, este é mostrado na Figura 2. Ele auxilia a entender o sistema de controle em cascata implementado.

Figura 2: P&ID para controle de Nível e Vazão



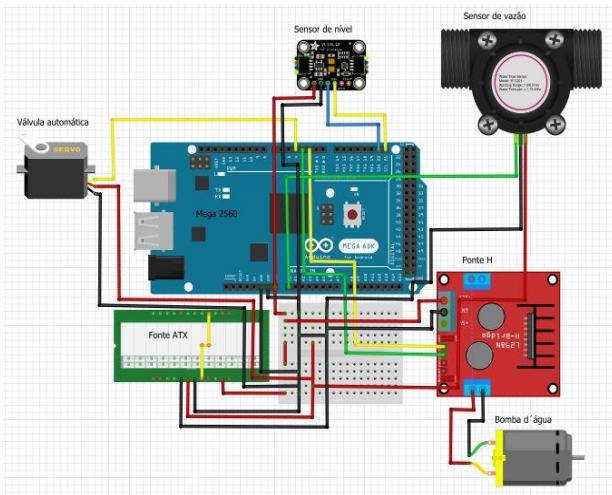
A. Arduino® Mega 2560

O microcontrolador é baseado no ATmega 2560, com 54 pinos de entrada/saída digital, oscilador cristal de 16 MHz e conexão USB, desenvolvida para projetos mais complexos [2], tal como aplicado neste trabalho. Desse modo, com o Arduino® Mega 2560, utilizou-se de programação específica para cada tipo de sensores e motores, a fim de realizar o controle destes.

As ligações efetuadas para controle da planta foram, PWM controle da bomba, pino 8, para o sensor de nível, as pinagens

SDA e SCL, para a válvula, utilizou-se o pino 9, para o sensor de vazão, o pino 2, utilizou-se a ligação em GND, 5 Vcc e uma alimentação externa 12 Vcc. Assim como podem ser vistas no exemplo de ligação na figura 3, o circuito foi desenhado no software gratuito *Fritzing*[®]. Para desenvolver a placa eletrônica foi utilizado o software Easyeda[®] também gratuito, esta placa foi usada como shield do Arduino[®] a fim de evitar mal contato nas ligações com os demais componentes.

Figura 3: Ligações no Arduino[®] para controle em Cascata



B. Mini bomba D'água 12 Vcc

A utilização da bomba destinada para receber energia potencial e transformar parte dessa energia em energia cinética e pressão (força) transferindo essas duas energias ao fluido analisado, neste caso a água. Com a principal finalidade de manter o nível no tanque conforme o valor de *setpoint* determinado na interface do supervisor.

C. Servo Motor Mg995

O servo Motor [12] foi utilizado para construção da válvula automática, sendo utilizado para controlar a válvula (torneira 1/4 de volta) de forma precisa. Desse modo, a válvula automática é composta por três componentes principais, corpo da válvula (torneira 1/4 de volta), atuador (servo motor) e controlador que será feito pelo Arduino[®]. O corpo da válvula bloqueia a passagem do fluido. O atuador (servo motor) controla a posição da válvula de forma rotacional.

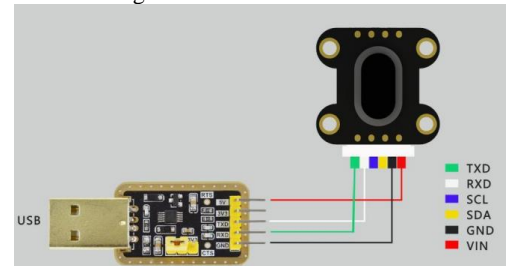
D. Sensor a laser TOF200F

Os sensores foram aplicados como feedback (realimentação) para a malha fechada de controle em cascata. O sensor a laser TOF200F [10] (*Time-of-flight*) foi fixado no topo do tanque, este por sua vez mede a distância até o líquido, que posteriormente é convertida em porcentagem de nível pelo controlador.

O princípio de funcionamento do sensor é baseado no tempo de voo (trânsito) do *laser* que foi emitido e retornou após encontrar um anteparo. Para comunicação do sensor de nível foi utilizado o protocolo I2C, conectado aos pinos SDA

para recebimento de dados, e o SCL para clock de sincronização dos sistemas, dispostos conforme figura 4.

Figura 4: Sensor a laser TOF200F



E. Sensor de fluxo YF-S301

Já o sensor de fluxo YF-S301[7], foi instalado após a saída da bomba instalada no tanque para detectar o fluxo do líquido. É um medidor com excelente precisão. O princípio de funcionamento se baseia no efeito *Hall*, esse é um fenômeno físico em que uma tensão elétrica é gerada perpendicularmente ao campo magnético aplicado e a corrente elétrica que o atravessa. Mensurando unidades de volume por unidade de tempo (L/min). Disposto conforme a figura 5.

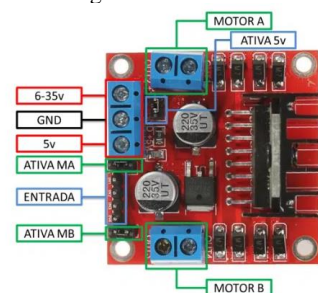
Figura 5: Sensor de Fluxo YF-S301



F. Ponte H L298N

A ponte H é um CI, conforme figura 6 a seguir, utilizado como drive de motores tanto para DC como de passo amplamente empregado nos projetos de eletrônica embarcada [5]. Capaz de controlar direção e velocidade de até dois motores DC ao mesmo tempo, contém proteção de sobretensão, e capacidade de 2 amperes por canal. Os controles de velocidade podem ser feitos pelas entradas *enable* ou podemos controlar a rotação e sentido pelas entradas (IN1, IN2, IN3 e IN4) [6], para controlar a velocidade do motor é usado um sinal *PWM* [9].

Figura 6: Ponte H L298N



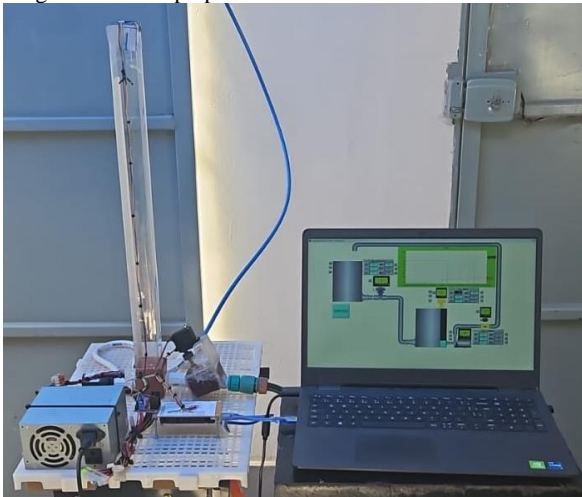
G. Supervisório Scada

Para inserção dos valores do PID e *setpoint* foi utilizado o supervisório Elipse SCADA®, O Elipse SCADA® [11] alia alto desempenho e grande versatilidade, representados em seus diversos recursos que facilitam e agilizam a tarefa de desenvolvimento de sua aplicação. Totalmente configurável pelo usuário, permite a monitoração de variáveis em tempo real, através de gráficos e objetos que estão relacionados com as variáveis físicas de campo. Também é possível fazer acionamentos e enviar ou receber informações para equipamentos de aquisição de dados. Além disto, através de sua exclusiva linguagem de programação, o Elipse Basic, é possível automatizar diversas tarefas a fim de atender as necessidades específicas de sua empresa.

H. Protótipo para o controle de Nível e Vazão em Cascata

A conexão dos materiais supracitados bem como os demais, descritos na Tabela 1, resultou no *hardware* desse trabalho com intuito de analisar o controle de nível e vazão em cascata, conforme visto na figura 7. Além disso, acrescentou-se uma régua em escala de porcentagem para acompanhar as leituras visuais comparadas aos resultados via supervisório.

Figura 7: Protótipo para controle de nível e vazão em cascata



I. Método de controle

O controle em cascata pode ser definido como uma aplicação onde há um controlador primário (mestre) regulando um controlador secundário (escravo) aprimorando a velocidade de resposta e minimizando os distúrbios causados na malha primária [8]. O funcionamento é através de dois controladores PID convencionais que controlam o nível e a

vazão, compondo duas malhas fechadas, já a válvula é controlada de forma condicional. Sendo assim as perturbações do nível gerada no tanque será amenizada pelo controle escravo de forma muito mais rápida, diminuindo as oscilações no controle do nível do tanque [4]. O controle se baseia na figura 2. Note que a malha de vazão, a malha secundária (escrava), é uma malha interna e apresenta uma dinâmica mais rápida [14]. Por sua vez, a malha de nível, primária (mestre), é externa e possui dinâmica mais lenta em relação à primeira [13]. Isto posto, as perturbações do nível gerada no tanque será amenizada pelo controle escravo de forma muito mais rápida, diminuindo as oscilações no controle do nível do tanque [4].

A forma de controle para todos os dispositivos de eletrônica embarcada como fora supracitado foi através da programação e calibração para cada dispositivos separadamente, e posteriormente unificou a programação para aplicação no protótipo final. Para análise visual e gráfica dos resultados foi implementado a comunicação com supervisório Elipse SCADA®. O qual é possível visualizar e inserir o valor para o *setpoint* e PID, além de analisar os gráficos, para nível e válvula, conforme as perturbações geradas no sistema, de acordo com a figura 8 e 9.

Figura 8: Página principal no supervisório SCADA

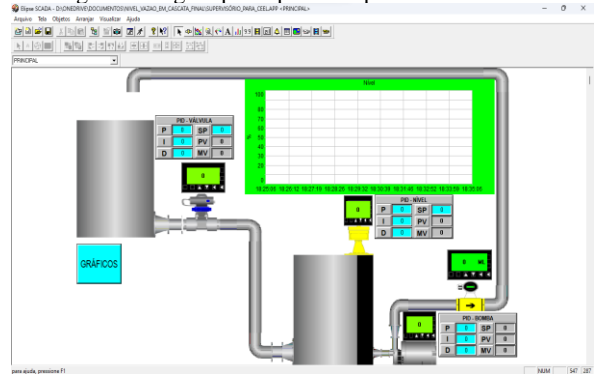
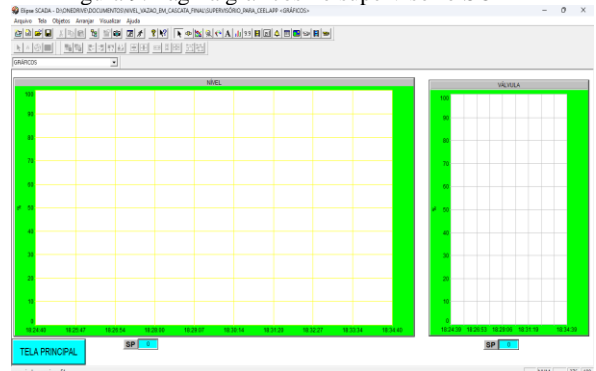


Figura 9: Página gráficos no supervisório SCADA



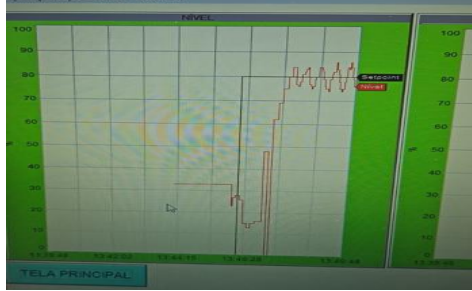
III. RESULTADOS E DISCUSSÕES

As primeiras simulações foram realizadas de forma individual, para calibração de cada componente que compõem a eletrônica embarcada do projeto. Para o sensor de nível foi parametrizado o valor de dois pontos, 0% = 640mm e 100% = 110mm, por causa de oscilações na leitura foi aumentado o intervalo de leitura do sensor. O sensor de vazão mensura o

valor calculado da frequência gerada X área do sensor, resultando na medida do fluxo em (l/min), com isso gera a modulação PWM para a bomba, a qual faz o controle do nível da água no tanque, conforme o valor de *setpoint* inserido pelo usuário via supervisor.

Para os testes realizados antes da implementação do PID, considerando apenas o controle condicional, obteve-se os resultados conforme figura 10.

Figura 10: Gráfico do nível para controle condicional



A configuração do PID para o projeto foi realizada com o PID direto para o nível, para o controle de nível quando o *set point* está menor do que o nível medido a MV é aumentada, e o PID reverso para a vazão, para o controle de vazão quando o *set point* está menor do que a vazão medida a MV deve ser diminuída. Foram realizados vários ajustes nos PID para testar o comportamento das malhas de controle. Apesar das dificuldades encontradas os resultados foram satisfatórios, na qual foi possível visualizar a variação nos dois controles da malha em cascata. Os comportamentos estão demonstrados nas conforme figuras 11, 12 e 13.

Figura 11: Comportamento do nível

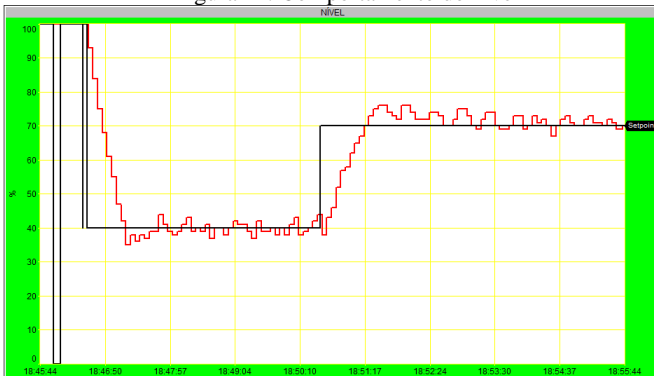


Figura 12: Comportamento do nível

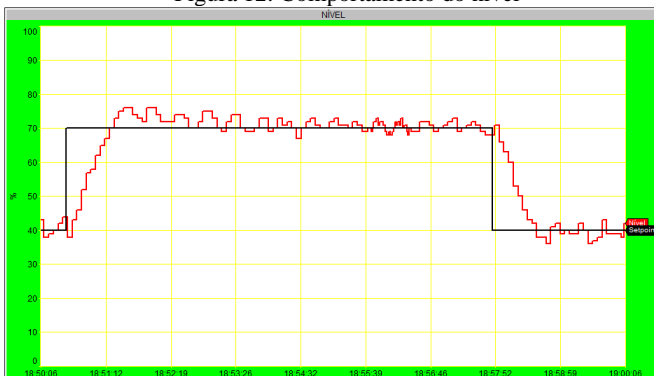
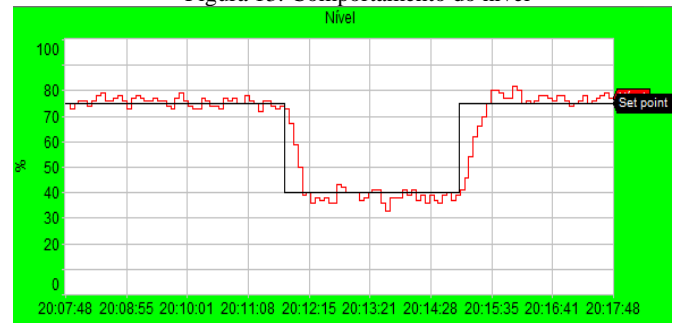


Figura 13: Comportamento do nível



IV. CONCLUSÃO

Conforme gráficos apresentados, o projeto obteve êxito em seu funcionamento, onde é possível verificar o comportamento do controle de nível e vazão em cascata mediante perturbações e alteração de *setpoint*. Atingindo assim o objetivo inicial do projeto, ser de baixo custo e utilizando o Arduino® como controlador PID. Nos gráficos apresentados, o sistema de controle em cascata foi capaz de controlar o nível de forma satisfatória, onde é possível verificar os parâmetros de um controle PID, que são: Tempo de subida, ultrapassagem, tempo de acomodação e estabilidade. Este trabalho mostrou uma alternativa viável ao CLP, para implementação do controle PID em cascata em microcontroladores de baixo custo. Isso pode ser importante para aplicações educacionais utilizando controle de nível e vazão.

Para projetos futuros pode-se utilizar bombas em corrente alternada e acioná-las via inversores de frequência, sendo que o mesmo princípio deste projeto pode ser usado e irá enviar a referência de velocidade que o motor deverá ser acionado. Em se tratando do protótipo construído pode-se inserir uma válvula de retenção na saída da bomba, para não gerar perturbações no sistema devido ao retorno do fluido quando a bomba para.

REFERÊNCIAS

- [1] C. R. B. Santos. Fundamentos de projetos eletrônicos utilizando o Arduino® - Teoria Digital. Editora LTC, 2021.
- [2] Arduino® Mega 2560 Ver 3. Acesso em 28 de abril de 2023 em: <https://docs.Arduino.cc/hardware/mega-2560>.
- [3] Corripio, S. & (s.d). Controle em Cascata Seborg. (s.d). Acesso em 15 de maio 2023 em: https://www.professores.uff.br/ninoska/wp-content/uploads/sites/57/2017/11/Aula-25-Controlle-Cascata2017_2sem17.pdf.
- [4] Passetti, G., Rico, J. E. L. N. Tecnologia fundation fieldbus aplicada no controle avançado da vazão e do nível de uma coluna de destilação piloto. Acesso em 17 de maio de 2023 em:

http://www.portalabpg.org.br/PDPetro/3/trabalhos/IBP0427_05.pdf.

[5] Thomsen, A. Motor DC com driver Ponte H L298N. Acesso em 30 de maio de 2023 em: <https://www.makerhero.com/blog/motor-dc-Arduino@-ponte-h-l298n/>.

[6] DATASHEET L298N. Acesso em 30 de maio de 2023 em: <portal.vidadesilicio.com.br/wp-content/uploads/2017/05/L298N.pdf>.

[7] Informações YF-S301 0,3-10LPM. Acessado em 28 de abril de 2023 em: <http://seazhongjiang.com/product/showproduct.php?id=26> .

[8] Simulador de controle de Nível e Vazão com blocos PID - Estratégia Cascata. Acessado em 30 de maio de 2023 em: <https://www.youtube.com/watch?v=xhADxVHCwx8>.

[9] Controle de velocidade de motor DC com PWM no Arduino® UNO. Acessado em 28 de abril de 2023 em: <https://www.makerhero.com/blog/controle-motor-dc-pwm-Arduino@-uno/>.

[10] DATA SHEET TOF200F. Acessado em 04 de maio de 2023 em: <http://myosuploads3.banggood.com/products/20210224/20210224033416TOF200F.pdf>.

[11] Manual do Elipse SCADA. Acessado em 04 de maio de 2023 em: <https://www.elipse.com.br/downloads/?cat=69&key=&language=ptbr>.

[12] DATA SHEET SERVO MOTOR MG995. Acessado em 04 de maio de 2023 em: <https://www.arduocore.com.br/servo-motor-mg995->

[13] Simulador de controle de Nível e Vazão com blocos PID - Estratégia Cascata. Acessado em agosto de 2023 em: <https://www.youtube.com/watch?v=xhADxVHCwx8>.

[14] Simulador de controle de Nível e Vazão com blocos PID - Estratégia Cascata. Acessado em agosto de 2023 em: <https://www.youtube.com/watch?v=xhADxVHCwx8>.