

**UNIVERSIDADE DE CAXIAS DO SUL
ÁREA DO CONHECIMENTO DE CIÊNCIAS EXATAS E
ENGENHARIAS**

WILLIAN PAIM GOBETTI

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA CONTROLE DE
NÍVEL DE ÓLEO SOLÚVEL COM ENVIO DA INFORMAÇÃO POR
MEIO DE IOT**

CAXIAS DO SUL

2024

WILLIAN PAIM GOBETTI

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA CONTROLE DE
NÍVEL DE ÓLEO SOLÚVEL COM ENVIO DA INFORMAÇÃO POR
MEIO DE IOT**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia de Controle e Automação na Área do Conhecimento de Ciências Exatas e Engenharias da Universidade de Caxias do Sul.

Orientador: Prof. Dr. Alexandre Mesquita

CAXIAS DO SUL

2024

WILLIAN PAIM GOBETTI

**DESENVOLVIMENTO DE UM PROTÓTIPO PARA CONTROLE DE
NÍVEL DE ÓLEO SOLÚVEL COM ENVIO DA INFORMAÇÃO POR
MEIO DE IOT**

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia de Controle e Automação na Área do Conhecimento de Ciências Exatas e Engenharias da Universidade de Caxias do Sul.

Aprovado(a) em 27/11/2024

BANCA EXAMINADORA

Prof. Dr. Alexandre Mesquita
Universidade de Caxias do Sul - UCS

Prof. Dr. Ricardo Vargas Dorneles
Universidade de Caxias do Sul - UCS

Prof. Ma. Patricia Giacomelli
Universidade de Caxias do Sul - UCS

AGRADECIMENTOS

Gostaria de expressar minha profunda gratidão a todos que contribuíram para a realização deste trabalho.

Primeiramente, agradeço a Deus por me conceder saúde, sabedoria e perseverança ao longo desta jornada acadêmica.

À minha esposa, Franciele, meu eterno agradecimento por seu amor, compreensão e apoio inabalável. Sua presença foi fundamental para que eu pudesse chegar até aqui.

À minha filha, Antonella, que, mesmo sem saber, me inspirou a dar o meu melhor em cada etapa deste processo. Sua presença ilumina meus dias e me inspira a ser melhor a cada dia.

Aos meus amigos, que compartilharam momentos de alegria e me ofereceram palavras de incentivo quando mais precisei. Sua amizade foi essencial para manter minha motivação.

Aos meus professores, especialmente Alexandre Mesquita, por sua orientação valiosa, paciência e dedicação. Suas contribuições foram fundamentais para o desenvolvimento deste trabalho.

Por fim, agradeço a todos que, de alguma forma, participaram desta caminhada e contribuíram para a concretização deste projeto.

"Lembre-se que as pessoas podem tirar tudo de você, menos o seu conhecimento."

Albert Einstein

RESUMO

O óleo solúvel ou fluido de corte é um elemento importante para o processo de usinagem no Comando Numérico Computadorizado (CNC). Dentre suas principais finalidades estão a refrigeração na região de remoção do material, a lubrificação da superfície de atrito e proteger a máquina e as peças usinadas de corrosões e oxidações. É de grande importância o monitoramento do nível do fluido de corte em seu reservatório para que não haja a falta do mesmo. O presente trabalho teve como objetivo desenvolver um protótipo, que automatiza o abastecimento, mede o nível do reservatório e transmite essas informações utilizando recursos e protocolos de *Internet of Things* (IoT). Para alcançar esses objetivos, foi desenvolvido um sistema que integra um sensor laser para monitorar continuamente o nível do fluido, uma eletro-válvula responsável pela abertura e fechamento automáticos do abastecimento. E gestão da aquisição e processamento dos dados é realizada por um microcontrolador ESP32, com transmissão via Wi-Fi. Este protótipo foi instalado com sucesso em uma máquina (CNC), incluindo dispositivos de fixação customizados para o sensor laser e a eletro-válvula, além de uma caixa plástica para abrigar o microcontrolador. Este sistema demonstrou eficácia e confiabilidade, validando os objetivos propostos e oferecendo uma solução prática e moderna para o monitoramento e controle do fluido de corte em ambientes de usinagem.

Palavras-chave: Controle de Nível. Sensor Laser. ESP32. IoT.

ABSTRACT

Soluble oil or cutting fluid is an important element for the machining process in CNC. Its main purposes include cooling the material removal region, lubricating the friction surface and protecting the machine and machined parts from corrosion and oxidation. Monitoring the level of the cutting fluid in its reservoir is of great importance to ensure that there is no shortage. This work aimed to develop a prototype that automates the supply, measures the reservoir level and transmits this information using IoT resources and protocols. To achieve these objectives, a system was developed that integrates a laser sensor to continuously monitor the fluid level, an electro-valve responsible for automatically opening and closing the supply. And the management of data acquisition and processing is performed by an ESP32 microcontroller, with transmission via Wi-Fi. This prototype was successfully installed in a machine (CNC), including customized fixing devices for the laser sensor and the electro-valve, in addition to a plastic box to house the microcontroller. This system demonstrated effectiveness and reliability, validating the proposed objectives and offering a practical and modern solution for monitoring and controlling cutting fluid in machining environments.

Keywords: Level Control. Laser Sensor. ESP32. IoT.

LISTA DE FIGURAS

Figura 1 – Exemplo de máquinas CNC	15
Figura 2 – Sensor laser	18
Figura 3 – Esquema de um medidor de nível por laser	19
Figura 4 – Bloco básico de IoT	21
Figura 5 – Esquema dos periféricos integrados para microcontrolador	22
Figura 6 – Microcontroladores Arduino UNO, ESP8266 e ESP32	22
Figura 7 – <i>Publish/subscribe</i>	23
Figura 8 – Exemplo de Dashboard Tago	26
Figura 9 – Setor de emulsão do óleo.	27
Figura 10 – Transbordamento do óleo por super abastecimento.	28
Figura 11 – Gráfico de Indicadores.	28
Figura 12 – Representação do sistema.	29
Figura 13 – Invólucro de proteção do sensor.	31
Figura 14 – Etapas de funcionamento do sensor.	32
Figura 15 – Válvula solenoide para abastecimento.	33
Figura 16 – Menu apresentado no display.	34
Figura 17 – Token gerado na pagina da Tago.io.	35
Figura 18 – Tabela de configuração do tanque da máquina	36
Figura 19 – Indicador do nível do tanque.	37
Figura 20 – Indicador de status da eletroválvula.	37
Figura 21 – Indicador total litros.	37
Figura 22 – Indicador Status do Sistema e Total Litros.	38
Figura 23 – Fluxograma de funcionamento.	39
Figura 24 – Centro de Usinagem Mazak 5000.	41
Figura 25 – Instalação do projeto.	42
Figura 26 – Teste de calibração do sensor.	43
Figura 27 – Teste de calibração de Nível.	44
Figura 28 – Nível apos o abastecimento manual.	44

LISTA DE TABELAS

Tabela 1 – Classificação das medidas de nível (direta)	17
Tabela 2 – Classificação das medidas de nível (indireta)	17
Tabela 3 – Métodos HTTP	24
Tabela 4 – Principais códigos de status HTTP	25
Tabela 5 – Dimensões do tanque e volume em Litros	41
Tabela 6 – Resultado das Medições	43

LISTA DE ABREVIATURAS E SIGLAS

ADC	<i>Analog Digital Converter</i>
CAD	<i>Computer Aided-Design</i>
CAM	<i>Computer-Aided Manufacturing</i>
CNC	Comando Numérico Computadorizado
CoAP	<i>Constrained Application Protocol</i>
DAC	<i>Digital-to-Analog Converter</i>
GPIO	<i>General-Purpose Input/Output</i>
HTTP	<i>Hypertext Transfer Protocol</i>
IEEE	<i>Institute of Electrical and Electronic Engineers</i>
IDE	<i>Integrated Development Environment</i>
IoT	<i>Internet of Things</i>
I2C	<i>Circuito Inter-integrado</i>
LAN	Rede Local
LiDAR	<i>Light Detection and Ranging</i>
MQTT	<i>Message Queue Telemetry Transport</i>
NFC	<i>Near Field Communication</i>
PAM	<i>Bluetooth</i>
PID	Controle Proporcional Integral Derivativo
PWM	Pulse Width Modulation
RAM	<i>Random Access Memory</i>
RFID	<i>Radio Frequency Identification</i>
ROM	<i>Read Only Memory</i>
ToF	<i>Time-of-Flight</i>
UART	<i>universal asynchronous receiver/transmitter</i>
Wi-Fi	<i>Wireless Fidelity</i>
3D	Tridimensional

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Justificativa	12
1.2	Objetivo Geral	13
1.3	Objetivos Específicos	13
1.4	Estrutura do Trabalho	13
1.5	Limitações do Trabalho	13
2	REVISÃO BIBLIOGRÁFICA	14
2.1	Revisão sobre Máquinas de CNC	14
2.2	Óleo refrigerante	15
2.2.1	Controle no Nível e Consumo do Óleo	16
2.2.1.1	Métodos de controle	16
2.2.1.2	Controle de consumo e desperdício	16
2.3	Sensores	17
2.3.1	Sensor laser	18
2.3.1.1	Sensor pressão	19
2.3.1.2	Comparação entre sensores	19
2.4	Internet das Coisas	20
2.5	Microcontroladores	20
2.6	Sistemas de Comunicação	22
2.6.1	I2C	22
2.6.2	MQTT	23
2.6.3	HTTP	24
2.6.4	JSON	25
2.6.5	BLUETOOTH	25
2.7	Tecnologia de Comunicação	25
2.7.1	Wi-Fi	25
2.8	Plataforma Tago.io	26
3	METODOLOGIA	27
3.1	Cenário Atual	27
3.1.1	Impacto Financeiro	28
3.2	Projeto do sistema de medição de nível	29
3.3	<i>Hardware</i>	30
3.4	Sensor Laser	30
3.5	Eletroválvula e Controle	33

3.6	Display LCD	34
3.7	Dashboard Tago.io	34
3.8	<i>Firmware</i>	38
4	RESULTADOS	41
4.1	Máquina e instalação	41
4.2	Teste de medição do sensor	42
4.3	Testes de volume	43
4.4	Comportamento da válvula	44
5	CONSIDERAÇÕES FINAIS	45
	REFERÊNCIAS	46
	APÊNDICE A – DESENHO INVÓLUCRO SENSOR	49
	APÊNDICE B – CONFIGURAÇÃO DA PLATAFORMA TAGO.IO . .	50
	APÊNDICE C – ESQUEMÁTICO	55
	APÊNDICE D – FIRMWARE	57
	APÊNDICE E – TABELA DE MEDIDA	64
	APÊNDICE F – TABELA DE CUSTO	65
	APÊNDICE G – PROJETO INSTALADO	66

1 INTRODUÇÃO

O avanço da automação de processos industriais tem sido um catalisador fundamental para a otimização da produção, redução de custos e aumento da competitividade das empresas. Nos processos de usinagem a automação proporciona precisão, eficiência e flexibilidade sem precedentes, além de promover um ambiente de trabalho mais seguro e ergonômico para os operadores. Automação de processos manuais representa o ápice da eficiência operacional, capacitando empresas a eliminar tarefas repetitivas e propensas a erros, enquanto liberam recursos humanos para atividades de maior valor agregado (MARTINS, 2011).

Usinagem é um processo de fabricação que envolve a remoção de material de uma peça bruta para dar forma e criar uma peça finalizada com as dimensões desejadas. O óleo é frequentemente utilizado como um lubrificante durante a usinagem para reduzir o atrito entre a ferramenta de corte e a peça. Isso ajuda a evitar o desgaste excessivo da ferramenta e da peça, prolongando sua vida útil e mantendo a qualidade das superfícies usinadas (ALMEIDA, 2015).

Para medir o nível de fluidos em um sistema de usinagem podem ser usados dois métodos de medição: o direto e o indireto. Medição direta é uma abordagem simples podendo aplicar régua de medição, que consiste em inserir uma régua graduada diretamente no reservatório e realizar a leitura manualmente (PEREIRA, 2018).

A medição indireta pode ser realizada através do uso de sensor de nível, que utiliza tecnologia eletrônica para detectar e medir o nível do líquido. Por fim, tem-se o uso de boia de medição, que consiste em uma boia flutuante que se desloca verticalmente de acordo com o nível do líquido, permitindo a medição mais precisa (JUNIOR, 2019).

O presente trabalho propõe uma utilização da Internet das Coisas IoT para fornecer informações em tempo real sobre os níveis do óleo solúvel, permitindo uma gestão mais eficaz dos recursos e potencialmente levando a economias significativas. Além disso, com a implementação deste protótipo, espera-se contribuir para a literatura existente sobre sistemas automatizados de controle e monitoramento na indústria 4.0.

1.1 JUSTIFICATIVA

O desenvolvimento e aplicação de tecnologias avançadas para aprimorar a eficiência e precisão dos processos industriais é um dos principais focos da Engenharia de Controle e Automação. Em particular, o controle do nível de óleo solúvel em reservatórios de máquinas CNC é um aspecto crítico que afeta a operação eficiente destas máquinas e, conseqüentemente, a produtividade das indústrias que as utilizam. De acordo com (MAZAK, 2017), manter o nível adequado de óleo solúvel é essencial para assegurar a qualidade da usinagem, prevenir desgaste

excessivo das ferramentas e garantir a segurança do operador.

Controlar o nível e consumo do óleo solúvel em máquinas CNC agrega vantagens como a redução de custos, melhoria na eficiência operacional, qualidade das peças aprimorada, segurança no ambiente de trabalho e conformidade ambiental. Essas vantagens são fundamentais para uma operação industrial eficaz e sustentável (SHIRMER, 2024).

1.2 OBJETIVO GERAL

Este trabalho tem como objetivo apresentar um método de controle de nível de baixo custo para reservatórios de máquinas CNC para a empresa especializada em usinagem de ligas de aço, e apresentar essas informações via IoT. A partir do objetivo geral, identificou-se os objetivos específicos que se seguem:

1.3 OBJETIVOS ESPECÍFICOS

1. Realizar revisão bibliográfica sobre o tema para identificar sistemas e soluções já existentes similares ou próximos à proposta aqui apresentada.
2. Desenvolver um sistema de monitoramento de nível do reservatório em tempo real.
3. Realizar o abastecimento de forma automática do reservatório com critérios previamente definidos.
4. Apresentar estas informações de nível e consumo em uma plataforma IoT.

1.4 ESTRUTURA DO TRABALHO

O presente trabalho está organizado da seguinte forma:

- O Capítulo 2 apresenta a revisão bibliográfica dos conceitos de CNC, e também utilizados para a aquisição e tratamento dos dados.
- O Capítulo 3 apresenta a metodologia de desenvolvimento para o projeto proposto.

1.5 LIMITAÇÕES DO TRABALHO

No presente trabalho está delimitado o escopo em realizar a medição do nível do reservatório e o abastecimento automático, bem como o envio dos dados, utilizando sistema de IoT e apresentar em um *dashboard* de proteção.

O equipamento que receberá o sistema é de modelo específico. Não é almejada a escalabilidade da solução desenvolvida para outros equipamentos com função semelhante.

2 REVISÃO BIBLIOGRÁFICA

Neste capítulo serão apresentados os conceitos estudados para contextualização do projeto, demonstrando os conceitos sobre usinagem em máquinas CNC, óleo como elemento importante do processo, tipos de sensores para medir distâncias, microcontroladores, protocolos de comunicação e os benefícios de um sistema remoto para apresentar essas informações.

2.1 REVISÃO SOBRE MÁQUINAS DE CNC

As máquinas de Controle Numérico por Computador são ferramentas automatizadas controladas por um computador que executa comandos programados para controlar o movimento de máquinas e ferramentas (HENRIQUES, 2021). Essas máquinas são amplamente utilizadas na indústria para a fabricação de peças e componentes. Entre principais características das máquinas de CNC estão:

- **Precisão e Repetibilidade:** As máquinas CNC são capazes de produzir peças com alta precisão e repetibilidade, essencial para indústrias que requerem tolerâncias estritas, como a aeroespacial e a de dispositivos médicos.
- **Automação e Eficiência:** Ao automatizar processos de usinagem, as máquinas CNC aumentam a eficiência e reduzem a necessidade de intervenção manual, permitindo operações contínuas e a produção em massa.
- **Flexibilidade:** As CNCs podem ser programadas para fabricar uma ampla variedade de peças, tornando-as extremamente versáteis. Mudanças de programação podem ser realizadas rapidamente para produzir diferentes peças sem a necessidade de modificar fisicamente a máquina.
- **Integração com CAD/CAM:** A integração com softwares de *Computer Aided-Design* (CAD) e de *Computer-Aided Manufacturing* (CAM) permite a criação de designs complexos que podem ser diretamente traduzidos em instruções de usinagem (ALMEIDA, 2015).

Dentre os principais processos na usinagem, podemos destacar os de furação, torneamento e fresamento. Estes processos são realizados por equipamentos CNC específicos, cada um com suas particularidades e aplicações. A furação, por exemplo, é utilizada para criar orifícios precisos em materiais sólidos, sendo essencial em indústrias como a automotiva e a aeroespacial. O torneamento, por sua vez, envolve a rotação da peça a ser usinada contra uma ferramenta de corte fixa, permitindo a fabricação de peças cilíndricas e simétricas, como eixos e pinos. Já o fresamento utiliza uma ferramenta rotativa com múltiplas arestas cortantes para

remover material, ideal para criar superfícies planas, ranhuras e contornos complexos (HENRIQUES, 2021).

Estes processos são realizados em máquinas CNC de alta precisão, que garantem a repetibilidade e a qualidade das peças produzidas, como mostrado na Figura 1.

Figura 1 – Exemplo de máquinas CNC



Fonte: Adaptado de CIMM (2022)

A Figura 1 mostra algumas destas versões sendo a) Torno CNC; b) Fresadora CNC e c) Centro de usinagem.

2.2 ÓLEO REFRIGERANTE

No processo de usinagem, a formação do cavaco gera uma quantidade significativa de energia, resultante do atrito entre a ferramenta de corte e a peça usinada. Esse atrito e o calor produzido durante o processo são os principais responsáveis pelo desgaste da ferramenta de corte e pela degradação da superfície usinada, o que pode aumentar as dimensões da peça trabalhada (MACHADO, 2023).

O óleo de refrigeração, também conhecido como fluido de corte ou refrigerante, desempenha um papel crucial no funcionamento eficiente e seguro das máquinas de CNC (RAIDEN, 2024). Ele é projetado para cumprir várias funções essenciais durante os processos de usinagem:

- **Aumento da Vida Útil das Ferramentas:** A lubrificação e a refrigeração adequadas reduzem o desgaste das ferramentas de corte, prolongando sua vida útil e reduzindo os custos operacionais associados à substituição frequente de ferramentas.
- **Melhoria da Precisão e Qualidade do Produto:** A manutenção de temperaturas estáveis e a lubrificação eficiente garantem que as peças usinadas mantenham suas dimensões precisas e acabamentos de alta qualidade.

- **Eficiência Operacional:** A remoção eficiente de cavacos e a prevenção de sobreaquecimento permitem que a máquina opere em sua capacidade máxima sem interrupções, aumentando a produtividade (RAIDEN, 2024).
- **Segurança:** A gestão eficaz do calor e a prevenção de incêndios associados ao atrito excessivo tornam o ambiente de trabalho mais seguro para os operadores.

2.2.1 Controle no Nível e Consumo do Óleo

O controle de nível de óleo solúvel nas máquinas de CNC é crucial para manter a eficiência, a qualidade do trabalho e a durabilidade do equipamento (DINIZ, 2001). Manter o nível adequado de óleo solúvel é essencial para garantir uma lubrificação contínua das ferramentas de corte e das superfícies de contato durante a operação. Duas funções que podemos citar no controle de nível são:

- **Redução do Atrito e Desgaste:** A lubrificação contínua reduz o atrito entre a ferramenta e a peça, minimizando o desgaste e prolongando a vida útil das ferramentas.
- **Qualidade do Acabamento:** A lubrificação adequada contribui para uma melhor qualidade do acabamento superficial da peça usinada, evitando arranhões e imperfeições que podem comprometer a funcionalidade e a estética do produto final.

2.2.1.1 Métodos de controle

O controle de nível em reservatórios é uma prática essencial para garantir a estabilidade e eficiência de sistemas de armazenamento de líquidos. Um dos métodos mais simples e amplamente utilizados é o controle ON-OFF, que opera uma válvula ou bomba para manter o nível de líquido dentro de limites predefinidos. Quando o nível do líquido cai abaixo de um ponto de ajuste mínimo, a bomba é ativada, e quando atinge o ponto de ajuste máximo, a bomba é desligada. Este método é vantajoso por sua fácil implementação, baixo custo e manutenção mínima (OGATA, 2004).

Além disso, métodos mais avançados como o Controle Proporcional Integral Derivativo (PID) são frequentemente utilizados para melhorar a precisão e a resposta dinâmica do sistema, ajustando continuamente a saída de controle com base no erro de nível e suas variações ao longo do tempo (NISE, 2023).

2.2.1.2 Controle de consumo e desperdício

O controle do consumo de óleo solúvel e a prevenção de desperdício, especialmente devido ao transbordamento do reservatório, são essenciais para a eficiência operacional e a sustentabilidade econômica e ambiental das operações.

- Redução de Desperdício: Controlar o consumo de óleo solúvel ajuda a minimizar o desperdício, reduzindo os custos operacionais associados à compra excessiva de fluido.
- Manutenção da Qualidade do Fluido: Um consumo bem regulado assegura que a qualidade do óleo solúvel seja mantida por mais tempo, evitando a necessidade de substituições frequentes e dispendiosas.

2.3 SENSORES

Os sensores são dispositivos que medem uma propriedade física do ambiente, como temperatura, luminosidade, distância, presença ou rotação, convertendo essas medições em tensão ou corrente elétrica equivalente (BALBINOT, 2011). Para escolher o sensor adequado, é importante levar em conta diversos fatores, como sensibilidade, estabilidade, precisão, velocidade de resposta, resolução, faixa de operação, histerese, vida útil, custo, entre outros (JUNIOR, 2019). A determinação do nível de um reservatório pode ser feita por meio de medições diretas e indiretas, utilizando sensores de pressão, sensores de capacitância, ultra-som ou laser (BALBINOT, 2011).

Na Tabela 1 e Tabela 2 estão listados alguns dispositivos para estes tipos de medições.

Tabela 1 – Classificação das medidas de nível (direta)

Tecnologia aplicada	Líquido	Sólido
Medição por visor de nível	X	
Medição por boias e flutuantes	X	
Medição por contatos de eletrodos	X	
Medição por sensor por contato	X	
Medição por unidade de grade		X
Medição por capacitância	X	X

Fonte: Adaptado de Fialho (2004)

Tabela 2 – Classificação das medidas de nível (indireta)

Tecnologia aplicada	Líquido	Sólido
Medição por capacitância	X	X
Medição por empuxo	X	
Medição por célula d/p CELL	X	
Medição por caixa de diafragma	X	
Medição por tubo em U	X	
Medição por borbulhamento	X	
Medição por radioatividade	X	X
Medição por ultra-som	X	X
Medição por vibração	X	X
Medição por pesagem	X	X
Medição por laser	X	X

Fonte: Adaptado de Fialho (2004)

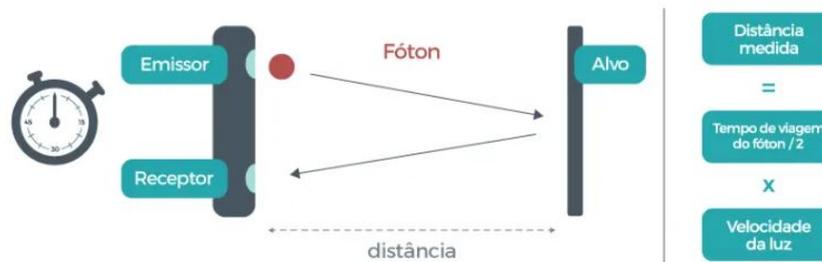
Considerando as configurações físicas habituais dos tanques em uso e a possibilidade de implementação em locais já em funcionamento, é conveniente minimizar as alterações na ins-

talação existente ao máximo possível ao optar por adicionar sensores. Assim, uma opção ideal seria instalar o sensor na parte superior do tanque, garantindo uma instalação sem a necessidade de esvaziamento. Em casos em que o acesso à parte inferior ou lateral do tanque seja viável com o mínimo de interferência na instalação atual, outra alternativa seria medir a pressão da água dentro do tanque utilizando uma tubulação já existente ou fazendo uma pequena adaptação no sensor. Destacam-se os sensores que melhor atendem a essas considerações: o sensor de laser para instalação na parte superior do tanque, permitindo sua colocação sem esvaziar o reservatório, e o sensor de pressão para a parte inferior, requerendo apenas uma conexão simples compatível com as características do sensor e permitindo acesso interno ao tanque.

2.3.1 Sensor laser

Os sensores laser são dispositivos que utilizam feixes de laser para medir distâncias, detectar objetos, ou monitorar níveis de líquidos e sólidos (BALBINOT, 2011). Eles são amplamente utilizados em diversas indústrias devido à sua alta precisão, velocidade de resposta e capacidade de operar em condições adversas (LEMOS, 2020). Os sensores laser funcionam emitindo um feixe de luz laser que é refletido pelo objeto alvo como mostra a Figura 2 .

Figura 2 – Sensor laser



Fonte: Lemos (2020)

O sensor então mede o tempo que o feixe leva para retornar, calculando a distância com base na velocidade da luz. Este método é chamado de *Time-of-Flight* (ToF) uma técnica de medição de distância que calcula o tempo que um sinal leva para viajar de um emissor a um receptor após ser refletido por um objeto. Essa técnica é amplamente utilizada em sensores de distância, como sensores laser e *Light Detection and Ranging* (LiDAR), bem como em tecnologias de imagem Tridimensional (3D) (LEMOS, 2020). O sensor mede o tempo que o feixe leva para retornar e, com base na velocidade da luz, calcula a distância até o objeto.

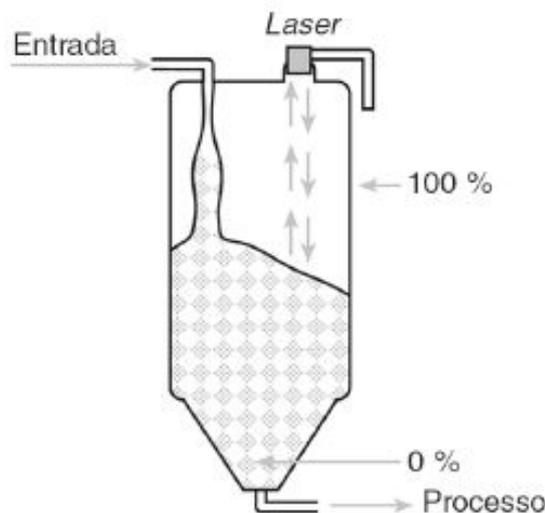
A fórmula básica para calcular a distância (d) usando o tempo de voo é:

$$d = \frac{ct}{2} \quad (2.1)$$

- (d) = distância até o objeto (em metros)
- (c) = velocidade da luz (aproximadamente $3 \cdot 10^8$ metros por segundo)
- (t) = tempo de voo (em segundos)

Para medir o nível de um reservatório, um sensor é instalado acima do líquido, conforme ilustrado na Figura 3. Este sensor é posicionado a uma altura conhecida em relação à base do reservatório. A determinação do nível do líquido é feita pela diferença entre a medida obtida pelo sensor e a altura conhecida. Essa diferença é então convertida em litros, permitindo a quantificação precisa do volume de líquido presente no reservatório (BALBINOT, 2011).

Figura 3 – Esquema de um medidor de nível por laser



Fonte: Adaptado de Balbinot (2011)

2.3.1.1 Sensor pressão

O medidor de pressão, como o próprio nome sugere, é um dispositivo utilizado para medir a força exercida sobre uma área específica do sensor, resultando na determinação da pressão (BALBINOT, 2011). No sensor de pressão diferencial, a medição é feita pela diferença de pressão entre dois pontos distintos. Para medir o nível de um líquido, o sensor é submerso no líquido em questão, criando uma coluna de líquido. A altura dessa coluna é determinada pela relação entre a pressão exercida e a densidade do líquido (FIALHO, 2004).

2.3.1.2 Comparação entre sensores

Os sensores a laser apresentam diversas vantagens em comparação aos sensores de pressão, especialmente em termos de precisão e confiabilidade. Enquanto os sensores de pressão podem ser influenciados por variações na densidade do fluido e desgaste mecânico, os sensores

a laser oferecem medições extremamente precisas sem contato físico, eliminando problemas de desgaste e necessidade de manutenção frequente (MOREIRA, 2019). Além disso, os sensores a laser são mais robustos em condições ambientais adversas, como altas vibrações e temperaturas extremas, onde os sensores de pressão podem ter seu desempenho comprometido.

2.4 INTERNET DAS COISAS

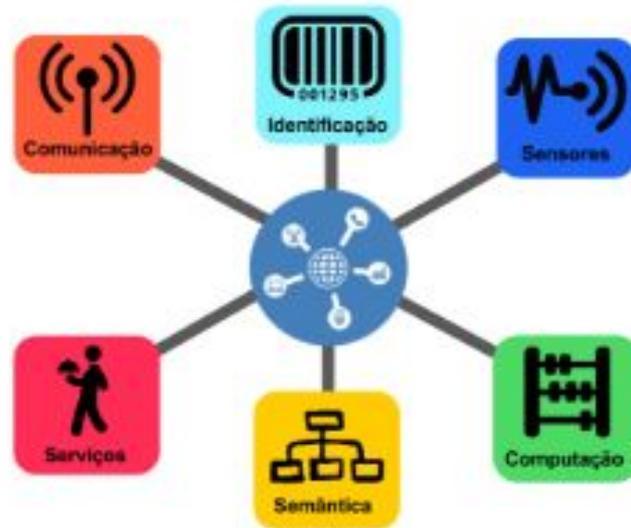
A Internet das Coisas IoT, do inglês *Internet of Things*, é uma revolução tecnológica que está transformando a maneira como interagir com o mundo ao nosso redor, conectando objetos do cotidiano à internet para coletar e compartilhar dados (MAGRANI, 2018). Essa interação envolve máquinas, veículos, aparelhos eletrodomésticos, sensores e atuadores sendo acessados remotamente por computadores, *tablets* e *smartphones* através da internet (SACOMANO, 2018). A caracterização da IoT abrange diversas tecnologias, a Figura 4 apresenta os blocos básicos que podem ser classificados em seis grupos distintos:

- Identificação: Para reconhecer o objeto e conectá-lo à internet, utilizando tecnologias como *Radio Frequency Identification* (RFID), *Near Field Communication* (NFC) e endereçamento IP.
- Sensores/Atuadores: Dispositivos que interagem com o ambiente, onde os sensores coletam dados do meio e os atuadores modificam o ambiente com base nesses dados.
- Comunicação: Para conectar objetos inteligentes utilizando tecnologias como *Wireless Fidelity* (Wi-Fi), *Bluetooth*, IEEE 802.15.4 e RFID.
- Computação: Composta por microcontroladores, processadores e FPGAs, responsáveis pela execução do processamento.
- Serviços: Divididos em identificação, agregação de dados, colaboração e inteligência, e ubiquidade.
- Semântica: Refere-se à extração de conhecimento e uso eficiente dos recursos na IoT, visando fornecer serviços específicos. Utiliza técnicas como *Resource Description Framework* (RDF), *Web Ontology Language* (OWL) e *Efficient XML Interchange* (EXI) (SANTOS, 2016).

2.5 MICROCONTROLADORES

Os microcontroladores (MCUs) são circuitos integrados que incorporam diversas funcionalidades em um único chip. Entre essas funcionalidades, destacam-se a memória *Random Access Memory* (RAM) e *Read Only Memory* (ROM), temporizadores, contadores, interfaces de

Figura 4 – Bloco básico de IoT



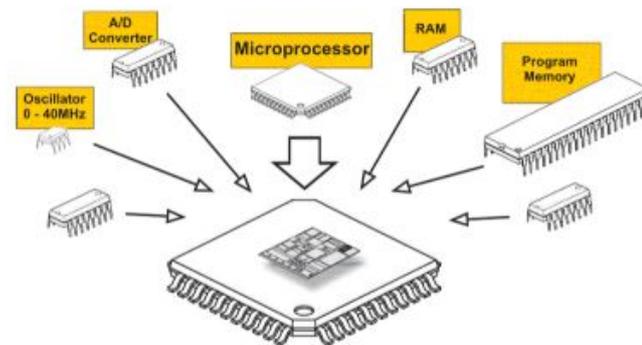
Fonte: Adaptado de Santos (2016)

entrada e saída de dados, conversores analógico-digital (ADC) e digital-analógico (DAC), além de interfaces para comunicação serial, conforme ilustrado na Figura 5. Para que o microcontrolador execute tarefas específicas, é necessário programá-lo com um conjunto de instruções denominado *firmware*. Esse *firmware* é geralmente desenvolvido em linguagem C e C++, utilizando um *Integrated Development Environment* (IDE) que oferece ferramentas para edição, compilação e depuração do código (GOMES, 2013)(MAGRANI, 2018).

As características de interconexão dos componentes encapsulados nos Microcontroladores são definidas por suas arquiteturas. Exemplos notáveis de modelos de arquitetura incluem ARM, ATMEL, PIC e PowerPC. Cada uma dessas arquiteturas possui especificações próprias que determinam a forma como os componentes internos, como unidades de processamento, memórias e periféricos, são organizados e interagem entre si. Essas arquiteturas influenciam diretamente o desempenho, a eficiência energética e a aplicabilidade dos MCUs em diferentes contextos e aplicações (ABREU, 2012).

Os microcontroladores são componentes essenciais em diversos dispositivos eletrônicos modernos, e alguns dos mais utilizados atualmente incluem o Arduino, o ESP8266, ESP32 e o STM32. O Arduino, especialmente o modelo Uno, é amplamente adotado em projetos de prototipagem e educação devido à sua simplicidade e vasta comunidade de suporte. O ESP8266 e ESP32, por sua vez, são populares em projetos de Internet das Coisas (IoT) graças ao seu baixo custo e capacidade de conexão Wi-Fi integrada conforme ilustrado na Figura 6 (OLIVEIRA, 2019). Já o STM32, da STMicroelectronics, é valorizado em aplicações industriais e de alta performance por sua robustez e variedade de recursos (STMICROELETRONICS, 2022).

Figura 5 – Esquema dos periféricos integrados para microcontrolador



Fonte: Adaptado de Gomes (2013)

Figura 6 – Microcontroladores Arduino UNO, ESP8266 e ESP32



Fonte: O Autor (2024)

2.6 SISTEMAS DE COMUNICAÇÃO

Para que dispositivos eletrônicos embarcados possam se comunicar, é essencial o uso de protocolos de comunicação que padronizem o envio e a recepção de informações. No contexto IoT, destacam-se protocolos que combinam baixo consumo energético com alta confiabilidade, como o *Message Queue Telemetry Transport* (MQTT), *Constrained Application Protocol* (CoAP) e o *Hypertext Transfer Protocol* (HTTP) (TORRES ATSLANDS R. ROCHA, 2020). Um aspecto notável dos protocolos é a simplicidade de implementação nos dispositivos, o que assegura uma maior adoção e, como resultado, uma comunidade de pesquisadores e desenvolvedores mais engajada.

2.6.1 I2C

A comunicação I2C *Inter-Integrated Circuit* desempenha um papel crucial na Internet das Coisas (IoT), proporcionando uma maneira eficiente e simples de conectar sensores, atuadores e outros dispositivos periféricos a microcontroladores e sistemas de processamento. Em um ambiente IoT, onde a coleta e a troca de dados são essenciais, o protocolo I2C permite a integração de múltiplos dispositivos em um único barramento, facilitando a comunicação entre componentes de diferentes fabricantes e com diferentes funcionalidades. Por exemplo, sensores

de temperatura, umidade, pressão e movimento podem ser conectados a um microcontrolador central utilizando o I2C, permitindo a coleta de dados em tempo real e a tomada de decisões automatizadas com base nesses dados (IDEALI, 2021).

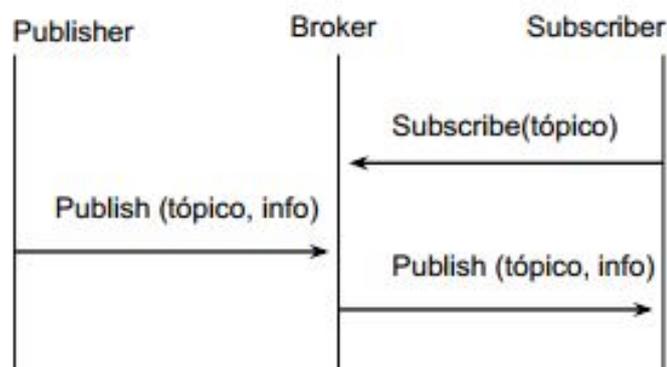
A utilização do I2C na IoT oferece várias vantagens, incluindo a redução da complexidade do sistema e a economia de espaço e custos, uma vez que apenas duas linhas de comunicação são necessárias para conectar múltiplos dispositivos. Além disso, o I2C suporta a detecção e correção de erros, aumentando a confiabilidade da comunicação em ambientes onde a integridade dos dados é crítica. A flexibilidade do protocolo também permite a fácil expansão do sistema, adicionando novos dispositivos ao barramento sem a necessidade de reconfigurações complexas. Em resumo, a comunicação I2C é uma ferramenta essencial para o desenvolvimento de soluções IoT eficientes, escaláveis e confiáveis, facilitando a interconexão de dispositivos e a criação de redes inteligentes que melhoram a automação e a eficiência em diversas aplicações, desde casas inteligentes até indústrias conectadas (IDEALI, 2021).

2.6.2 MQTT

O MQTT é um protocolo de comunicação leve e eficiente, amplamente utilizado em aplicações de IoT devido à sua capacidade de operar em redes com largura de banda limitada e dispositivos com recursos restritos (LIMA, 2018). Este protocolo emprega a estratégia de publicação/assinatura (*publish/subscribe*) para a transferência de mensagens. Além disso, o MQTT oferece mecanismos que garantem a entrega das mensagens. Este protocolo utiliza as camadas de transporte e rede da arquitetura TCP/IP (SANTOS, 2016).

O MQTT consiste de três componentes básicos: o *subscribe*, o *publish* e o *broker*. A Figura 7 ilustra a sequência de operações do MQTT.

Figura 7 – *Publish/subscribe*



Fonte: Adaptado de Santos (2016)

Inicialmente, os dispositivos se registram *subscribe* em um *broker* para receber informações sobre dados específicos, permitindo que o *broker* os notifique sempre que publicadores

publish divulgam os dados de interesse. Os dispositivos inteligentes *publish* enviam informações para os assinantes *subscribe* por meio do *broker* (MAGRANI, 2018).

2.6.3 HTTP

O HTTP é um protocolo de comunicação utilizado para a transferência de dados na World Wide Web (FOROUZAN, 2010). Ele opera como uma camada de aplicação sobre a arquitetura TCP/IP, permitindo a troca de informações entre clientes, como navegadores web, e servidores. HTTP é baseado em um modelo de requisição-resposta, onde o cliente envia uma requisição ao servidor, que por sua vez processa e retorna uma resposta correspondente.

A comunicação cliente-servidor ocorre por meio de uma mensagem estruturada, que pode ser composta por até três elementos distintos:

- A linha inicial, que indica a requisição ou o status;
- Os metadados, conhecidos como cabeçalhos;
- Opcionalmente, o conteúdo principal, referido como corpo da mensagem.

Uma requisição HTTP é uma mensagem enviada por um cliente (geralmente um navegador web) a um servidor, solicitando a realização de uma ação específica (SEIXAS, 2023). Essa comunicação é uma parte fundamental do protocolo (HTTP), que é usado para a troca de informações na web. Na Tabela 3 são mostradas as principais requisições:

Tabela 3 – Métodos HTTP

Método	Ação
GET	Solicita a representação de um recurso específico
HEAD	Solicita da mesma forma que o GET porém sem o corpo da resposta
POST	Submete uma entidade a um recurso específico
PUT	Atualiza as representações do recurso pela carga da requisição
DELETE	Remove um recurso específico
CONNECT	Estabelece um túnel para o servidor identificado pelo recurso de destino
OPTIONS	Descreve as opções de comunicação com o recurso de destino
PATCH	Aplica modificações parciais em um recurso

Fonte: Adaptado de Mozilla (2023)

Uma resposta HTTP é a mensagem que o servidor envia de volta ao cliente após processar uma requisição. Ela fornece as informações solicitadas ou notifica sobre o sucesso ou falha da operação. Aqui estão os principais tipos de uma resposta HTTP Tabela 4:

Tabela 4 – Principais códigos de status HTTP

Código	Frase
100	<i>Continue</i>
102	<i>Processing</i>
200	<i>OK</i>
201	<i>Created</i>
204	<i>No Content</i>
301	<i>Moved Permanently</i>
303	<i>See Other</i>
400	<i>Bad Request</i>
401	<i>Unauthorized</i>
403	<i>Forbidden</i>
404	<i>Not Found</i>
500	<i>Internal Server Error</i>
502	<i>Bad Gateway</i>
503	<i>Service Unavailable</i>

Fonte: Adaptado de Mozilla (2023)

2.6.4 JSON

O *JavaScript Object Notation* é um formato de intercâmbio de dados leve e amplamente utilizado para a transmissão de informações entre sistemas. Ele é baseado em um subconjunto da linguagem de programação *JavaScript*, mas é independente de linguagem, o que significa que pode ser utilizado por diversas linguagens de programação, como *Python*, *Java*, *C#*, entre outras. JSON é composto por pares de chave-valor e listas ordenadas de valores, o que facilita a leitura e a escrita tanto por humanos quanto por máquinas (FROZZA, 2018).

2.6.5 BLUETOOTH

Bluetooth é um protocolo de comunicação sem fio usado em redes pessoais *Bluetooth* (PAM). Ele permite a troca de informações entre dispositivos como computadores, impressoras, câmeras digitais, celulares, *notebooks* e consoles de videogame, utilizando uma frequência de curto alcance. O *Bluetooth* opera na frequência de 2,4 GHz, na mesma banda ISM não licenciada usada por outros protocolos de RF, como ZigBee e Wi-Fi. Ele possui um conjunto padronizado de regras e especificações que o diferencia desses outros protocolos (IDEALI, 2021).

2.7 TECNOLOGIA DE COMUNICAÇÃO

2.7.1 Wi-Fi

A tecnologia de comunicação Wi-Fi é uma solução de Rede Local (LAN) sem fio, desenvolvida para oferecer conectividade de banda larga em ambientes internos, seguindo a *Institute of Electrical and Electronic Engineers* (IEEE) especificação 802.11. Wi-Fi, que sig-

nifica Fidelidade Sem Fio, suporta uma taxa de dados de camada física de até 100 Mbps em ambientes internos, com alcance de aproximadamente 40 metros. Esta tecnologia oferece taxas de dados de pico superiores às dos sistemas 3G, pois opera em uma largura de banda maior, de 20 MHz. Uma das principais vantagens do Wi-Fi sobre o 3G é a ampla disponibilidade de dispositivos compatíveis. Todas as redes Wi-Fi operam em modo *half-duplex*. Os padrões Wi-Fi especificam uma largura de banda de canal fixa de 25 MHz para 802.11b e 20 MHz para redes 802.11a (LIMA, 2018).

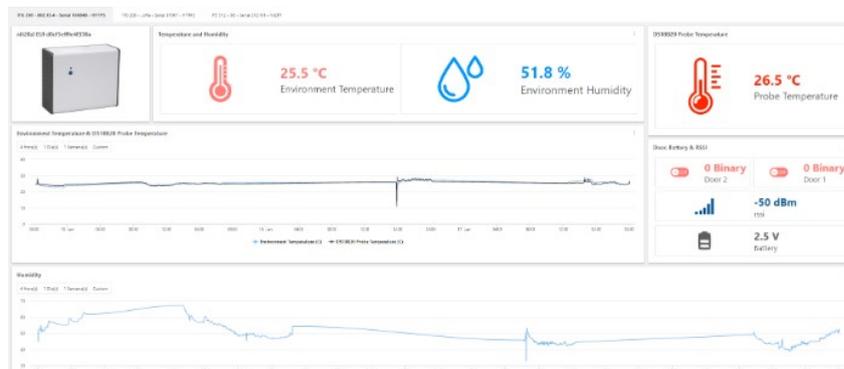
Os sinais Wi-Fi são recebidos por dispositivos como computadores e telefones celulares que possuem essa tecnologia integrada. Esses dispositivos interpretam os sinais e estabelecem uma conexão de internet sem a necessidade de cabos. Os pontos de acesso, que consistem em antenas e roteadores, são responsáveis pela transmissão e recepção das ondas de rádio. As antenas têm um alcance de transmissão de rádio que varia entre 300 e 500 metros (IDEALI, 2021).

2.8 PLATAFORMA TAGO.IO

A Tago.IO é uma plataforma de Internet das Coisas (IoT) que oferece uma solução completa para conectar, armazenar, analisar e visualizar dados de dispositivos conectados. Desenvolvida para ser flexível e escalável, a Tago.IO permite que empresas e desenvolvedores criem aplicações IoT de forma rápida e eficiente, sem a necessidade de infraestrutura complexa (TAGO, 2024). A plataforma suporta uma ampla gama de protocolos de comunicação, incluindo MQTT, HTTP e CoAP, facilitando a integração com diversos tipos de dispositivos e sensores .

Uma das características mais notáveis da Tago.IO é sua interface intuitiva e personalizável, que permite aos usuários criar dashboards interativos para monitoramento em tempo real e análise de dados. Além disso, a plataforma oferece recursos avançados como gerenciamento de dispositivos, análise de dados, alertas personalizados e integração com serviços de terceiros através de APIs (XAVIER, 2019).

Figura 8 – Exemplo de Dashboard Tago



Fonte: Adaptado de Xavier (2019)

3 METODOLOGIA

O presente capítulo apresenta a proposta de trabalho, com a descrição detalhada das etapas a serem seguidas para a obtenção do sucesso dos objetivos propostos, com a utilização dos conceitos estudados no capítulo 2. Também é descrito o cenário atual do processo de abastecimento de óleo solúvel para máquinas CNC na empresa Aço Peças Demore, destacando o método atual de abastecimento, e apresentar as possíveis melhorias com esta implementação.

3.1 CENÁRIO ATUAL

O propósito deste trabalho surge devido a uma demanda de uma empresa especializada em usinagem de precisão, localizada no município de Caxias do Sul-RS. A planta da empresa possui cerca de 120 máquinas (CNC) e, conta com um quadro de 120 funcionários. Devido seu número de máquinas existe um setor específico responsável pela emulsão do óleo de corte para as máquinas, como ilustrado na Figura 9.

Figura 9 – Setor de emulsão do óleo.



Fonte: O Autor (2024)

A empresa possui um sistema de tubulações que liga este setor à fábrica, onde por meio de um bloco *manifold* realiza-se a derivação através de mangueiras para cada máquina. A partir desta etapa o abastecimento fica a critério do operador. Os principais problemas no cenário atual são:

- Controle do nível manual: Não há qualquer controle automático de quando for necessário fazer o abastecimento do reservatório.
- Transbordamento: O operador realiza o abastecimento pode esquecer de fechar o registro, ocorrendo o transbordamento, conforme ilustrado na Figura 10.

- Indicadores de Consumo: Não há qualquer dado apresentando o consumo individual de fluido de corte para cada equipamento.

Figura 10 – Transbordamento do óleo por super abastecimento.

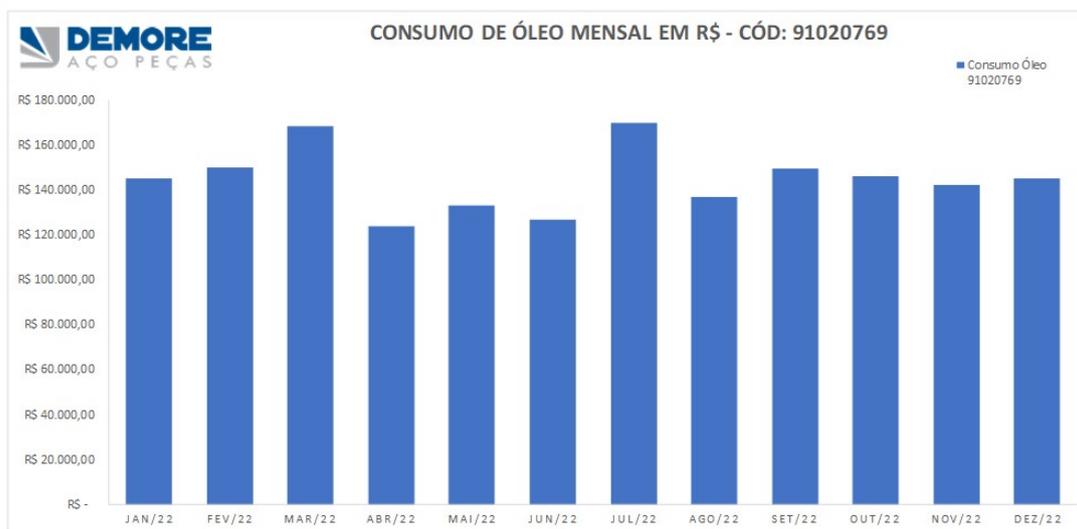


Fonte: O Autor (2024)

3.1.1 Impacto Financeiro

O controle do nível de óleo é essencial para a reposição rápida, garantindo a performance e a qualidade das peças usinadas. Por exemplo, em 2023, apenas um dos modelos de óleo utilizado resultou em um gasto de R\$ 1.737.760,23. Foram consumidos mais de 160 tambores de 208 litros de óleo puro, que, ao serem diluídos na proporção de 2% na central, geraram mais de 1.664.000 litros de óleo de corte emulsionados.

Figura 11 – Gráfico de Indicadores.



Fonte: Setor de TI Aço Peças Demore (2024)

Nos últimos seis meses, a produção em um único equipamento dedicado à usinagem de peças para implementos agrícolas consumiu um total de 1200 litros de óleo de corte. Esse consumo expressivo de óleo reflete a capacidade operacional intensiva da máquina, responsável por fabricar componentes que exigem alta precisão e qualidade para o mercado agrícola.

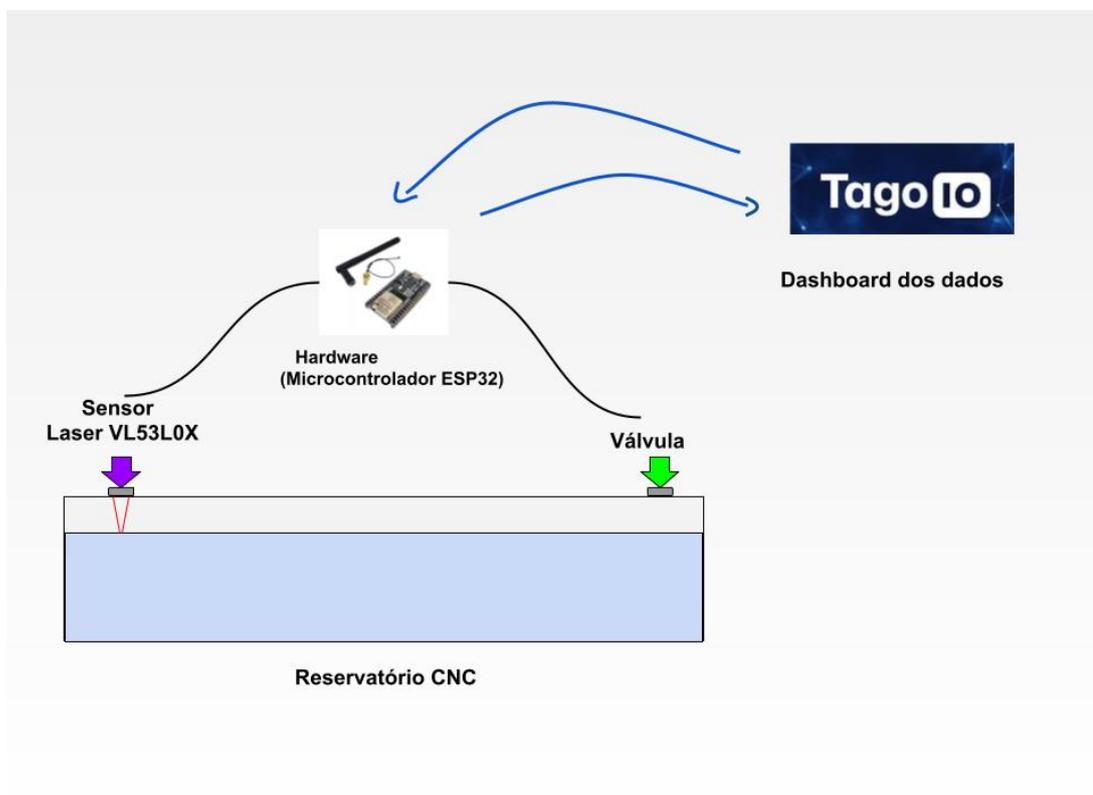
O uso do óleo de corte é crucial, pois ele desempenha um papel fundamental na redução de atrito e dissipação de calor durante o processo de usinagem, o que não apenas melhora a qualidade final das peças, mas também prolonga a vida útil das ferramentas de corte, garantindo operações suaves e eficientes.

A quantidade de óleo utilizada é um indicador da produtividade da máquina e fornece dados valiosos para o planejamento de suprimento e manutenção, assegurando que o equipamento continue a operar de forma sustentável e econômica.

3.2 PROJETO DO SISTEMA DE MEDIÇÃO DE NÍVEL

Este projeto visa desenvolver um sistema de monitoramento de nível de tanque utilizando um sensor laser VL53L0X, um microcontrolador ESP32, e a plataforma IoT Tago.IO. O sistema é projetado para medir o nível de líquidos em um reservatório de uma máquina (CNC), controlar uma válvula de enchimento e enviar dados para um dashboard online para monitoramento em tempo real, como ilustrado na Figura 12. Os elementos descritos são apresentados no decorrer deste capítulo.

Figura 12 – Representação do sistema.



Fonte: O autor (2024)

3.3 HARDWARE

O *hardware* ESP32 é um microcontrolador versátil e poderoso, desenvolvido pela Espressif Systems. Com um design que integra Wi-Fi e Bluetooth, a ESP32 é ideal para aplicações de Internet das Coisas (IoT), que exigem conectividade sem fio. Este microcontrolador é equipado com um processador dual-core de 32 bits, capaz de operar em frequências de até 240 MHz, o que proporciona um desempenho robusto para execução de tarefas complexas. Além disso, a ESP32 possui uma ampla gama de interfaces periféricas, incluindo *General-Purpose Input/Output* (GPIO), *Analog Digital Converter* (ADC), *Digital-to-Analog Converter* (DAC), *Circuito Inter-integrado* (I2C), *universal asynchronous receiver/transmitter* (UART), e *Pulse Width Modulation* (PWM), tornando-a extremamente adaptável para diferentes requisitos de projetos.

No contexto deste projeto de monitoramento de nível de tanque, a ESP32 desempenha um papel central ao integrar e gerenciar diversos componentes do sistema. Ela é responsável por ler os dados do sensor de distância VL53L0X, processar essas informações e controlar a válvula que regula o fluxo de líquido no tanque. Além disso, um display LCD foi incorporado para fornecer *feedback* visual local.

A capacidade de processamento da ESP32 permite que o sistema execute essas tarefas em tempo real, garantindo precisão e eficiência. Além disso, a conectividade Wi-Fi embutida facilita a comunicação com a plataforma Tago.IO.

3.4 SENSOR LASER

O sensor adotado para o projeto é o sensor laser VL53L0X, desenvolvido pela STMicroelectronics, é um módulo de medição de distância baseado na tecnologia *Time-of-Flight* (ToF). Este sensor é amplamente utilizado em aplicações que exigem medições precisas de distância e detecção de objetos, graças à sua alta precisão e confiabilidade. Este sensor possui algumas características importantes citadas abaixo:

- **Tecnologia *Time-of-Flight* (ToF):** O VL53L0X utiliza pulsos de laser infravermelho para medir o tempo que a luz leva para retornar ao sensor após refletir em um objeto, permitindo medições de distância precisas e rápidas.
- **Faixa de Medição:** O sensor é capaz de medir distâncias de até 2 metros, dependendo das condições ambientais e da refletividade do alvo.
- **Resolução:** Oferece uma resolução de até 1 mm, ideal para aplicações que exigem alta precisão.
- **Tensão de Operação:** Embora o núcleo do sensor opere entre 2.6V e 3.5V, o VL53L0X pode ser alimentado com 5V devido a um regulador de tensão integrado.

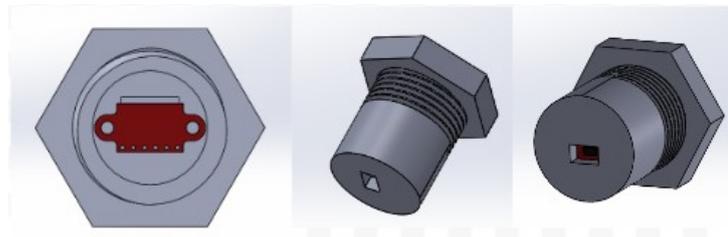
- Interface de Comunicação: Utiliza uma interface I2C para comunicação com microcontroladores, facilitando a integração em sistemas embarcados.

A biblioteca Adafruit VL53L0X desempenha um papel crucial na integração do sensor VL53L0X ao sistema, oferecendo uma interface simplificada para comunicação eficaz com o microcontrolador ESP32. Esta biblioteca fornece funções pré-configuradas para inicialização e leitura de dados do sensor, reduzindo significativamente a complexidade do código necessário para operar o sensor.

Além disso, ela suporta ajustes na configuração do sensor, como a calibração do alcance e da precisão, permitindo uma adaptação flexível às necessidades específicas do projeto. A biblioteca facilita a execução dos cálculos necessários para medir distâncias, retornando esses valores precisamente em milímetros, o que é essencial para aplicações que demandam alta precisão em medições de distância.

Para a instalação deste sensor em ambiente industrial foi projetado invólucro de proteção, onde foi confeccionado em um processo de impressão 3D, ilustrado na Figura 13 e também seu desenho para fabricação anexado no Apêndice A.

Figura 13 – Invólucro de proteção do sensor.



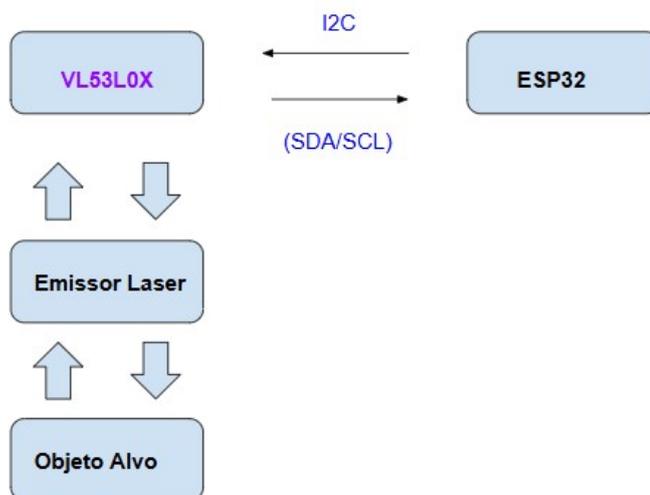
Fonte: Elaborado pelo autor(2024)

O fluxograma a seguir apresentado pela Figura 14 mostra as etapas que serão realizadas pelo sensor.

O VL53L0X, um sensor de tempo de voo (ToF) laser, realiza internamente cálculos para determinar a distância até o objeto alvo. O sensor emite um pulso laser, mede o tempo de retorno e, calcula a distância considerando a velocidade da luz.

A comunicação entre o sensor VL53L0X e o ESP32 é realizada através do protocolo I2C, que é uma escolha comum para conectar dispositivos de baixa velocidade em curtas distâncias devido à sua simplicidade e eficiência. No projeto, o cabo utilizado possui um comprimento de 800 mm. Essa distância está dentro do limite máximo recomendado para aplicações I2C, que é em torno de 1 metro. Ao operar dentro dessas especificações, o projeto mantém a integridade e a confiabilidade da comunicação, minimizando o risco de interferências e perda de dados que podem ocorrer em distâncias maiores.

Figura 14 – Etapas de funcionamento do sensor.



Fonte: Elaborado pelo autor(2024)

A escolha do sensor laser VL53L0X em vez de um sensor ultrassônico para o projeto pode ser justificada por várias razões que destacam as vantagens do sensor laser no contextos específicos:

- **Precisão e Resolução:** O sensor laser VL53L0X oferece uma precisão superior, capaz de medir distâncias com resolução milimétrica, o que é crucial para aplicações que exigem alta precisão, como o monitoramento de níveis de fluido em tanques CNC.
- **Tamanho e Consumo de Energia:** O VL53L0X é compacto e possui baixo consumo de energia, tornando-o ideal para aplicações em que o espaço é limitado e a eficiência energética é uma preocupação.
- **Rapidez de Resposta:** Sensores laser operam rapidamente, fornecendo medições quase instantâneas. Isso é vantajoso quando rápidas atualizações de dados são necessárias, permitindo respostas em tempo real a mudanças nos níveis de fluido.
- **Direcionalidade:** Sensores laser têm feixes altamente direcionais, permitindo medições precisas mesmo em espaços confinados ou em aplicações onde se requer foco em um ponto específico, evitando leituras erradas causadas por obstáculos próximos que poderiam interferir nos sensores ultrassônicos.

Essas vantagens fazem do sensor laser VL53L0X uma escolha mais adequada para o projeto, garantindo medições precisas, rápidas e confiáveis em ambientes industriais exigentes.

3.5 ELETROVÁLVULA E CONTROLE

O sistema utiliza uma eletroválvula solenoide de 24 V para controlar o fluxo de abastecimento do tanque, conforme ilustrado na Figura 15. Esta eletroválvula foi escolhida devido à sua robustez e compatibilidade com sistemas industriais, onde fontes de 24 V são comuns. O controle da válvula é implementado através de um circuito de interface, que permite ao ESP32, operando em 3,3 V, acionar a válvula de 24 V de forma segura.

Figura 15 – Válvula solenoide para abastecimento.



Fonte: Primaserra Automação e Inovação(2024)

Foi utilizado um módulo rele para adequar o sinal de tensão da saída, uma vez que ESP32 opera em nível lógico de 3,3 V e a eletroválvula opera em 24 V.

A lógica de controle emprega um método ON-OFF, também conhecido como controle de dois pontos ou histerese, implementado diretamente no *firmware* do ESP32. Este método utiliza as leituras do sensor VL53L0X para gerenciar efetivamente o nível do fluido no tanque. Quando o volume do tanque cai abaixo de 50% de sua capacidade total, o sistema ativa a válvula para iniciar o abastecimento. O monitoramento contínuo garante que, ao atingir 90% da capacidade, a eletroválvula seja automaticamente fechada, evitando transbordamentos.

Este controle simples, mas eficaz, mantém o tanque em níveis operacionais adequados, reduzindo a necessidade de intervenções manuais. Além disso, o sistema registra o volume no início e no fim de cada ciclo de abastecimento, permitindo o cálculo do volume adicionado. Todos esses dados, juntamente com o status atual da válvula, são transmitidos para a plataforma Tago.IO.

Isso possibilita o monitoramento remoto e a análise de dados de uso a longo prazo, oferecendo análises valiosos para a gestão do sistema e potencializando a eficiência operacional.

A máquina utiliza um sistema de refrigeração que liga a bomba para resfriar as ferramentas de corte, retornando o óleo ao reservatório. Este retorno pode aumentar rapidamente o nível de fluido, e os 10% de margem servem como uma segurança adicional para acomodar esse influxo, evitando transbordamentos.

Com este método, o sistema não só mantém os níveis operacionais adequados, mas também minimiza a necessidade de intervenções manuais. Adicionalmente, o volume é registrado no início e no fim de cada ciclo de abastecimento para calcular o volume adicionado.

3.6 DISPLAY LCD

A interface do usuário do sistema é implementada através de um display LCD I2C e um botão de interação, proporcionando uma experiência intuitiva e informativa. O firmware utiliza a biblioteca *LiquidCrystalI2C* para controlar um display de 16 colunas e 2 linhas, onde são exibidas informações cruciais do sistema. Um menu rotativo apresenta três telas principais:

- Total de litros abastecidos
- Nível atual do tanque
- Status da válvula

O sistema alterna automaticamente entre essas telas a cada 3 segundos, oferecendo uma visão geral constante das condições do tanque, como ilustrado na Figura 16.

Figura 16 – Menu apresentado no display.



Fonte: O autor(2024)

A interação do usuário é facilitada por um botão conectado ao pino 23 do ESP32. Uma pressão curta no botão permite a navegação manual entre as telas, enquanto uma pressão longa (superior a 5 segundos) ativa funções específicas, como zerar o contador de litros abastecidos ou alternar manualmente o estado da válvula. Este design de interface combina visualização passiva de informações com controle ativo do sistema, adaptando-se às necessidades do operador e garantindo fácil acesso às funcionalidades críticas do equipamento.

3.7 DASHBOARD TAGO.IO

A integração com a plataforma Tago.io desempenha um papel crucial no sistema, permitindo o armazenamento em nuvem, visualização em tempo real e análise de dados a longo

prazo. O ESP32 utiliza a biblioteca *HTTPClient* para estabelecer comunicação com a API do Tago.io através de requisições HTTP POST.

Cada variável relevante, como o nível do tanque, status da válvula e total de litros abastecidos, é enviada como um objeto JSON estruturado, incluindo metadados como limites superior e inferior. A autenticação é realizada através de um *token*, Figura 17, único garantindo a segurança da transmissão de dados.

Figura 17 – Token gerado na pagina da Tago.io.

Token information



Token

88a8f05e-68a7-40ed-84e8-a45e32e53c7d

Fonte: Tago.io (2024)

A função `sendToTagoIOVariable()` encapsula este processo, simplificando o envio de dados e tratando as respostas do servidor. Além do envio, o sistema também recupera configurações armazenadas no Tago.io, como dimensões do tanque através da função `getToTagoIOVariable()`, permitindo atualizações remotas de parâmetros críticos.

A aba de configuração permite ajustar as dimensões específicas do tanque das máquinas (CNC), que são cruciais para garantir a precisão do monitoramento e controle de volume de óleo solúvel. As seguintes configurações estão disponíveis:

- **Distancia sensor:** Insira a distância entre o sensor e a borda superior do tanque. Este valor ajuda a ajustar a leitura do sensor para considerar a posição relativa de instalação, garantindo leituras precisas do nível de fluido.
- **Largura Tanque:** Insira a largura interna do tanque em centímetros (mm). Este valor é essencial para calcular a área da superfície do tanque.
- **Comprimento Tanque:** Insira o comprimento interno do tanque em centímetros (mm). Com a largura, este valor permite determinar a capacidade volumétrica potencial do tanque.

- **Altura Tanque:** Defina a altura total do tanque em centímetros (mm). Este parâmetro é necessário para calcular o volume total disponível e para ajustar os níveis de alarme do fluido.

Ao detalhar essas configurações apresentadas na Figura 18, o sistema não apenas garante a funcionalidade e a precisão do monitoramento, mas também oferece uma interface amigável para o usuário final, facilitando a gestão operacional do fluido de corte.

Figura 18 – Tabela de configuração do tanque da máquina

The screenshot displays the Tago IO web interface. At the top left is the Tago IO logo. Below it are two tabs: 'Teste' and 'Setup'. The main content area is titled 'Setup Tanque MQ01'. On the left, there is a section labeled 'Section #1' with a sub-label 'My section description'. To the right of this section are four configuration items, each with a pencil icon and a text input field: 'Distancia Sensor', 'Largura Tanque', 'Comprimento Tanque', and 'Altura Tanque'. A blue 'Submit' button is positioned at the bottom right of the configuration area.

Fonte: O Autor (2024)

Esta abordagem bidirecional facilita não apenas o monitoramento, mas também o ajuste remoto do sistema.

Na plataforma Tago.IO, os dados recebidos podem ser visualizados em *dashboards* personalizados, com gráficos, indicadores e alertas, oferecendo uma interface intuitiva para operadores e gestores.

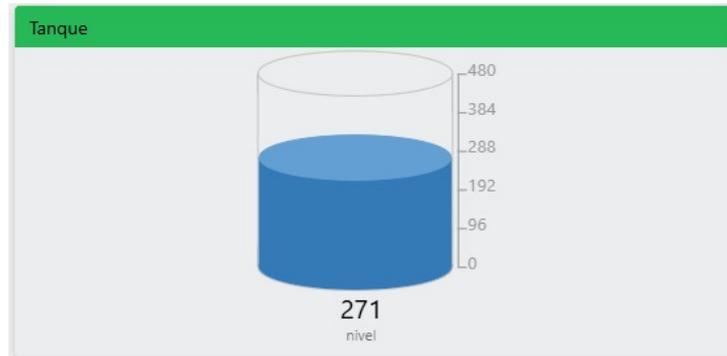
Na tela principal do controle de nível do sistema são apresentados os indicadores de:

Indicador do nível atual do tanque, tamanho do tanque e total de litros existente no tanque conforme ilustrado na Figura 19.

Indicador de status da válvula, o vermelho representa a eletroválvula fechada e verde eletroválvula aberta conforme mostra na Figura 20.

O indicador de total de litros, apresenta o total de litros que foi abastecido pelo sistema conforme a Figura 21.

Figura 19 – Indicador do nível do tanque.



Fonte: O Autor (2024)

Figura 20 – Indicador de status da eletroválvula.



Fonte: O Autor (2024)

Figura 21 – Indicador total litros.



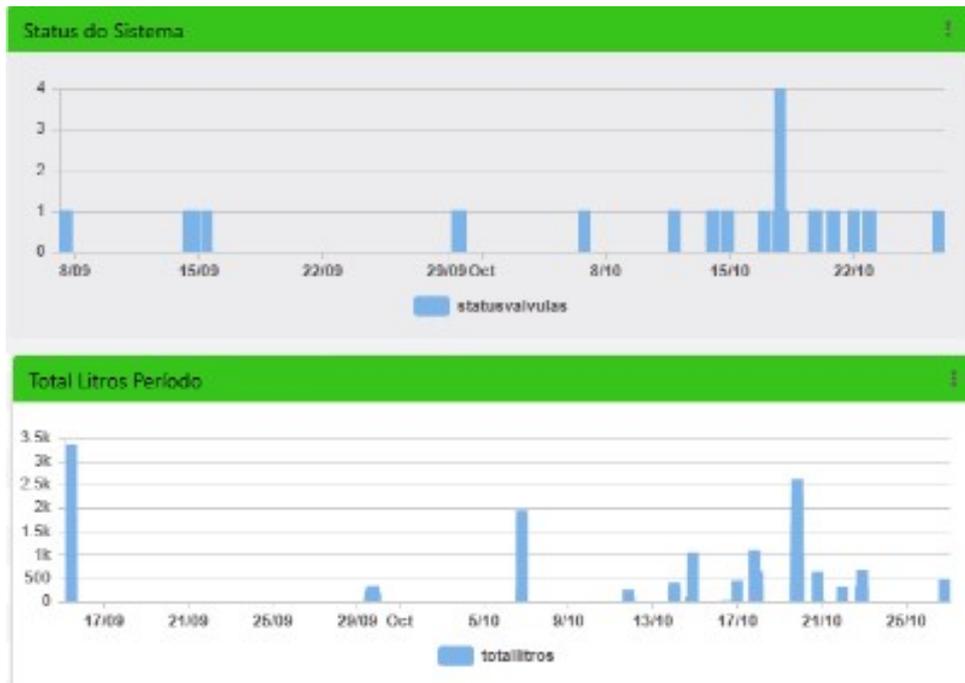
Fonte: O Autor (2024)

Os gráficos implementados, conforme ilustrado na Figura 22, fornecem valiosos insights sobre o desempenho do sistema. O primeiro gráfico mostra a quantidade de vezes que o sistema foi ativado, enquanto o segundo gráfico detalha o volume total abastecido em litros diariamente.

Essas visualizações permitem monitoramento contínuo e avaliação da eficiência operacional ao longo do tempo. Além disso, a capacidade de exportar esses dados em formatos XLSX e CSV facilita análises mais aprofundadas, possibilitando o uso de ferramentas de análise de dados para identificar tendências, otimizar operações e planejar manutenção preventiva, garantindo que o sistema opere de forma eficaz e eficiente.

Esta integração transforma o sistema local em uma solução IoT completa, permitindo tomada de decisões baseada em dados concretos para ajudar na tomada de decisões. O Apêndice B apresenta as etapas para a configuração dos dashboard na plataforma.

Figura 22 – Indicador Status do Sistema e Total Litros.



Fonte: O Autor (2024)

3.8 FIRMWARE

O *Firmware* foi desenvolvido utilizando a IDE do Arduino, que oferece um ambiente completo para edição, compilação, gravação do código. Essa plataforma facilita o desenvolvimento, permitindo que o código seja testado e ajustado de maneira eficiente durante o processo de criação do protótipo o *Firmware* está no Apêndice D.

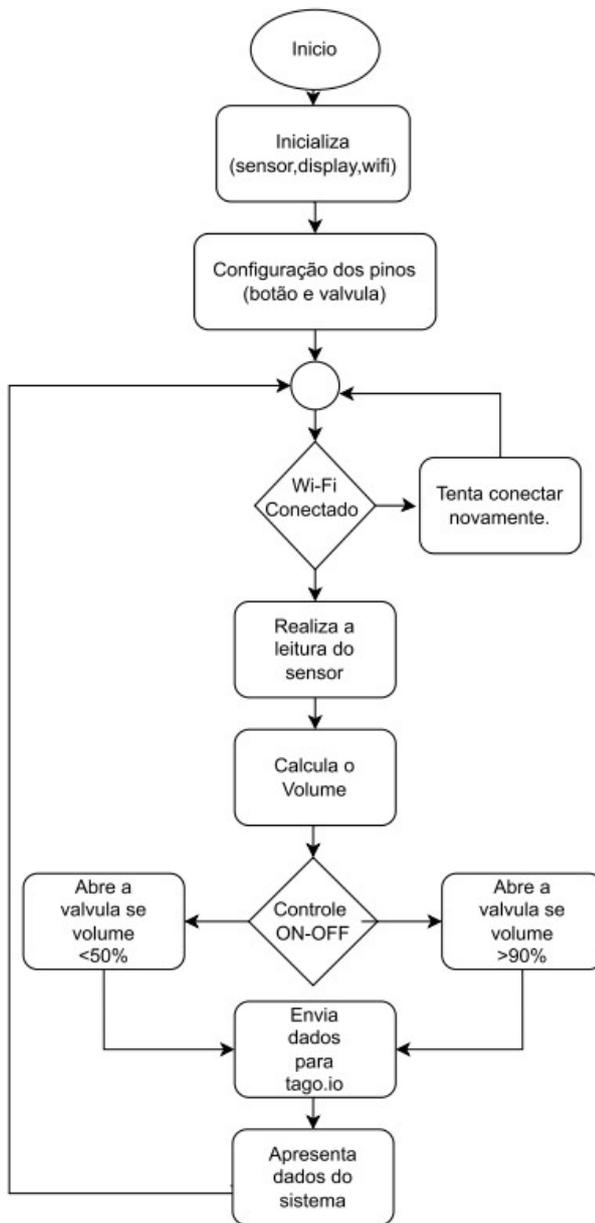
Para o controle de nível de óleo solúvel, o *Firmware* utiliza um ESP32, que integra funcionalidades de medição, controle e comunicação. Inicialmente, o sistema conecta-se a uma rede Wi-Fi especificada, possibilitando a comunicação com a plataforma Tago.IO. Essa conexão permite obter e atualizar configurações do tanque, como suas dimensões e a distância do sensor.

A gestão da conexão Wi-Fi é feita de maneira resiliente, com tentativas automáticas de reconexão a cada 5 minutos, caso a conexão inicial falhe. Além disso, o uso da EEPROM assegura que as configurações críticas do sistema sejam armazenadas de forma permanente, permitindo que o dispositivo recupere as informações necessárias mesmo após reinicializações.

O *Firmware* também incorpora um sistema de controle automático para a válvula de enchimento do tanque. Utilizando o sensor de distância VL53L0X, o firmware realiza medições periódicas para determinar o nível de óleo no tanque. Com base nas leituras, o sistema calcula o volume atual e decide automaticamente quando abrir ou fechar a válvula. Se o volume do tanque cair abaixo de 50% de sua capacidade total, a válvula é ativada para iniciar o enchimento. Quando o nível atinge 90%, a válvula é fechada para evitar transbordamentos, garantindo assim um controle eficiente e seguro do processo de abastecimento, um fluxograma ilustrado na

Figura 23 a lógica de funcionamento do *Firmware*.

Figura 23 – Fluxograma de funcionamento.



Fonte: O Autor (2024)

Além do controle automático, o *Firmware* oferece uma interface de usuário intuitiva através de um display LCD e um botão. O display exibe informações críticas, como o total de litros abastecidos, o nível atual de óleo e o status da válvula, permitindo que os usuários monitorem o sistema em tempo real. O botão permite a navegação entre diferentes menus no display, oferecendo uma maneira simples de interagir com o sistema e verificar seu estado operacional sem a necessidade de um computador ou dispositivo externo.

A comunicação com a plataforma Tago.IO é uma característica central do *Firmware*, permitindo o monitoramento remoto e a análise de dados. O *Firmware* envia dados importantes,

como o nível do tanque e o status da válvula, para a nuvem, onde podem ser visualizados e analisados em tempo real. Isso não apenas facilita a gestão do sistema, mas também fornece percepções valiosas para otimizar o processo de controle de nível de óleo. Essa capacidade de comunicação remota é essencial para aplicações em ambientes industriais, onde a supervisão constante e a resposta rápida são cruciais para a eficiência operacional.

4 RESULTADOS

No decorrer deste capítulo, são apresentados e analisados os resultados obtidos a partir das avaliações delineadas no capítulo anterior. Este capítulo foca na análise detalhada dos dados coletados durante os experimentos, destacando o desempenho do sistema de controle de nível de óleo solúvel implementado com a ESP32.

4.1 MÁQUINA E INSTALAÇÃO

O projeto piloto foi instalado em centro de usinagem da Mazak Nexus5000, Figura 24, este equipamento foi escolhido devido seu histórico de transbordamentos e também por se tratar de um equipamento que possui um abastecimento de 3 vezes durante a semana. Este equipamento possui um reservatório com dimensões descritas na Tabela 5.

Tabela 5 – Dimensões do tanque e volume em Litros

Dimensão	Medidas
Comprimento	2000 mm
Largura	1395 mm
Altura	200 mm
Volume	558 litros

Fonte: O Autor (2024)

Figura 24 – Centro de Usinagem Mazak 5000.



Fonte: O Autor(2024)

O hardware composto pelos sistema de controle, sensor laser e eletro-válvula, foi instalado na parte superior do reservatório, como ilustrado na Figura 25.

Figura 25 – Instalação do projeto.



Fonte: O Autor(2024)

4.2 TESTE DE MEDIÇÃO DO SENSOR

A distância entre o objeto e o sensor desempenha um papel crucial na determinação precisa do volume de óleo no reservatório. Durante os testes, identificou-se que havia um erro de leitura na medição do nível, o que levou à necessidade de ajustar um *offset* de 50 mm na posição de instalação do sensor em relação ao topo do tanque. Este ajuste foi essencial para corrigir as discrepâncias nas leituras do sensor, garantindo que as medições refletissem com precisão o nível real de óleo.

Para calibrar o sensor, foi realizado um teste abrangente em que o reservatório foi preenchido até o seu nível máximo, correspondente a um total de 558 litros e uma altura de coluna de óleo de 200 mm. Este procedimento de calibração permitiu verificar a exatidão das leituras do sensor em condições reais de operação. A Figura 26 ilustra o processo de enchimento do tanque e o posicionamento do sensor, destacando a situação de medição ao atingir a capacidade máxima.

Além de corrigir o erro de leitura inicial, o ajuste do offset melhorou a precisão geral do sistema de medição, permitindo uma gestão mais eficaz e eficiente do volume de óleo no reservatório. Este ajuste assegurou que o sistema pudesse apoiar decisões operacionais críticas com confiança, minimizando riscos de transbordamentos ou desabastecimento.

Realizou-se a medição de três distâncias em pontos previamente definidos para avaliar o comportamento do sensor em termos de repetibilidade e precisão. Esses dados estão documentados no Apêndice E. A Tabela 6 apresenta os resultados obtidos para as diferentes distâncias.

Os resultados das medições indicaram que, à medida que a distância aumentava, o erro nas leituras também se elevava, sugerindo uma degradação na precisão das medições de volume. Essa tendência aponta que, em distâncias maiores, as leituras do sensor apresentam uma variação mais significativa, possivelmente devido a limitações na capacidade do sensor de manter a precisão em condições de medição mais distantes.

Figura 26 – Teste de calibração do sensor.



Fonte: O Autor(2024)

Tabela 6 – Resultado das Medições

altura óleo (mm)	Nível calculado	Nível medido	QTD de amostras	Erro mediana
100	279	276	30	1,08%
150	418	415	30	0,72%
200	558	556	30	0,36%

Fonte: O Autor (2024)

4.3 TESTES DE VOLUME

Durante os testes de volume do tanque, foram coletados dados específicos para verificar a confiabilidade e a precisão do sistema de medição. Inicialmente, o nível do tanque foi registrado em 432 litros, correspondente a uma altura de 155 mm de coluna de óleo ilustrado, na Figura 27, estes testes foram realizado com a máquina (CNC) parada.

Para avaliar a precisão do sensor e a correspondência entre o volume teórico e o medido, o reservatório foi abastecido manualmente com quatro baldes de 20 litros cada, totalizando um acréscimo de 80 litros ao volume inicial. Este método manual permitiu um controle preciso do aumento de volume, essencial para validar a calibração do sensor.

Após o teste, o sistema de medição exibiu um volume de 511 litros, correspondente a uma altura de 183 mm, conforme ilustrado na Figura 28. Este resultado apresentou uma di-

Figura 27 – Teste de calibração de Nível.



Fonte: O Autor(2024)

ferença de apenas 1 litro em relação ao cálculo real. Essa pequena discrepância evidencia um percentual de erro mínimo, confirmando a eficácia do sistema de medição em fornecer dados confiáveis e consistentes. A precisão observada sugere que o sistema está bem calibrado e é capaz de suportar a gestão de volume de forma eficiente.

Figura 28 – Nível após o abastecimento manual.



Fonte: O Autor (2024)

4.4 COMPORTAMENTO DA VÁLVULA

Durante os testes, o comportamento da válvula foi cuidadosamente monitorado para avaliar sua eficácia e resposta em condições operacionais típicas. A válvula foi projetada para abrir e fechar automaticamente de acordo com os níveis de óleo determinados pelo sensor, desempenhando um papel crucial na manutenção do equilíbrio do volume do tanque. Observou-se que a válvula respondia de maneira eficiente aos comandos de controle, com tempos de abertura e fechamento adequados, prevenindo tanto o transbordamento quanto o esvaziamento excessivo do reservatório.

A precisão do sistema de controle da válvula foi validada através de testes repetitivos, que confirmaram sua capacidade de operar dentro dos parâmetros esperados. Durante os quatro abastecimentos realizados ao longo do período de duas semanas, a válvula demonstrou uma operação consistente, ajustando-se com precisão às variações no nível de óleo.

5 CONSIDERAÇÕES FINAIS

O protótipo desenvolvido para o controle de nível de óleo solúvel em máquinas CNC demonstrou-se eficaz e promissor, atingindo os objetivos de automação e monitoramento remoto. A implementação do sistema, que utiliza sensores laser para medir o nível de óleo e comunicação via IoT pela plataforma Tago.IO, trouxe importantes melhorias para o processo, como a redução do desperdício e o aumento da eficiência operacional. Essa solução, além de reduzir custos associados ao consumo do óleo de corte, também oferece mais segurança e precisão no gerenciamento dos recursos utilizados nas operações de usinagem.

Durante o desenvolvimento do projeto, diversos desafios técnicos foram enfrentados e superados, como a adaptação dos sensores a um ambiente industrial e a integração entre o microcontrolador ESP32 e os sistemas de comunicação. Esses obstáculos representaram oportunidades de aprendizado significativas, possibilitando um aprofundamento nos conceitos de controle e automação.

Além disso, o projeto reforçou a importância de soluções tecnológicas que integram diferentes sistemas e que podem ser personalizadas para atender às necessidades específicas da indústria 4.0, mostrando-se como uma experiência enriquecedora para futuras aplicações.

Para trabalhos futuros, espera-se que esse protótipo possa ser escalado para diferentes tipos de máquinas e aprimorado com tecnologias mais avançadas de monitoramento e controle. O sucesso inicial do projeto levou o diretor da empresa a aprovar a confecção de cinco unidades adicionais, dando início à aplicação em outras máquinas na linha de produção. A expansão do sistema para capturar dados mais detalhados e sua integração com plataformas de análise de dados podem elevar ainda mais a eficiência do processo e proporcionar uma tomada de decisão mais informada.

Esse trabalho, portanto, contribui para o avanço das tecnologias de IoT na indústria e abre caminho para novas pesquisas e aplicações voltadas para a automação e o uso inteligente de recursos em processos industriais.

REFERÊNCIAS

- ABREU, M. M. de. Sistemas para identificação de áudio linux embarcado em microcontrolador arm. 2012. Disponível em: <<https://bdta.abcd.usp.br/item/003170710>>. Acesso em: 04 Jun. 2024.
- ALMEIDA, P. S. de. **Processos de Usinagem**. 1. ed. 2015. Disponível em: <<https://integrada.minhabiblioteca.com.br/reader/books/9788536520070/pageid/2>>.
- BALBINOT, A. **Instrumentação e Fundamentos de Medida**. 2. ed. [S.l.], 2011.
- CIMM, R. M. **Torno CNC, Fresadora CNC e Centro de Usinagem**. 4. ed. São Paulo: Portal CIMM, 2022. Disponível em: <<https://www.cimm.com.br/portal/>>. Acesso em: 26 mai. 2024.
- DINIZ, A. E. **Aplicação de fluidos de corte em processos de usinagem**. 2001. Disponível em: <<https://www.scielo.br/j/jbsms/a/DfB8pRH7pM6LNHtYn8nVxmJ#>>. Acesso em: 30 mai. 2024.
- FIALHO, A. B. **Instrumentação Industrial - Conceitos, Aplicações e Análises**. 7. ed. [S.l.], 2004. Acesso em: 30 mai. 2024.
- FOROUZAN, B. A. **Comunicação de dados e redes de computadores**. 4. ed. São Paulo: Grupo A, 2010. ISBN 9788563308474. Disponível em: <<https://integrada.minhabiblioteca.com.br/#/books/9788563308474/>>.
- FROZZA, A. A. An approach for schema extraction of json and extended json document collections. 2018. Acesso em: 09 Jun. 2024.
- GOMES, R. M. O. F. Conceção e desenvolvimento de interfaces gráficas baseadas em sistema de microcontroladores microchip. 2013. Disponível em: <<https://repositorio-aberto.up.pt/bitstream/10216/69740/2/27517.pdf>>. Acesso em: 04 Jun. 2024.
- HENRIQUES, M. **Um resgate histórico sobre a usinagem incluindo o CNC e seus processos**. São Paulo: Revista Ferramental, 2021. Disponível em: <<https://www.revistaferramental.com.br/artigo/um-resgate-historico-sobre-usinagem-incluindo-cnc-seus-processos/>>. Acesso em: 26 maio. 2024.
- IDEALI, W. **Conectividade em Automação e IoT: Protocolos I2C,USB,TCP-IP entre outros.Funcionalidade e interligação para automação e ToT**. [s.n.], 2021. Disponível em: <[https://integrada.minhabiblioteca.com.br/reader/books/9786555202564/epubcfi/6/2\[%3Bvnd.vst.idref%3Dcover\]!/4/4/2%4051:3](https://integrada.minhabiblioteca.com.br/reader/books/9786555202564/epubcfi/6/2[%3Bvnd.vst.idref%3Dcover]!/4/4/2%4051:3)>. Acesso em: 08 Jun. 2024.
- JUNIOR, E. A. F. Sistema de monitoramento de nível de reservatório utilizando sensor ultrassônico. **Repositorio Congna**, 2019. Disponível em: <<https://repositorio.pgsscogna.com.br/bitstream/123456789/23644/1/EDMILSON%20ALVES%20FEITOSA%20JUNIOR.pdf>>. Acesso em: 05 Maio. 2024.
- LEMONS, M. F. **Sensor de distância a laser – Time-of-Fligh(ToF)**. São Paulo: Embarcados, 2020. Disponível em: <<https://embarcados.com.br/>>. Acesso em: 01 jun. 2024.

LIMA, J. C. de. Simulação de redes de sensores sem fio utilizando protocolos 6loepam,rpl,mqtt e coap em smart cities. 2018. Disponível em: <<https://tede.upf.br:8080/jspui/handle/tede/1557>>. Acesso em: 07 Jun. 2024.

MACHADO, G. da S. **Análise de fluido de corte orgânico à base de óleo de mamona aplicado à usinagem**. Dezembro. São Paulo: MM máquinas e metais, 2023. Disponível em: <https://www.arandanet.com.br/revista/mm/artigo_tecnico/277-Analise-de-fluido-de-corte-organico-a-base-de-oleo-de-mamona-aplicado-a-usinagem.html>. Acesso em: 27 mai. 2024.

MAGRANI, E. **A internet das Coisas**. 1. ed. [S.l.], 2018.

MARTINS, J. L. . Q. L. . J. . R. Performance evaluation for industrial automation system integration based on enterprise architecture standards and application in cotton textile industry. **ieeexplore-ieee-org**, 2011. Disponível em: <<https://ieeexplore-ieee-org.ez314.periodicos.capes.gov.br/document/6081272>>. Acesso em: 20 abril. 2024.

MAZAK, S. M. The importance of coolants in machining. Mar 2017. Disponível em: <<https://www.firetrace.com/fire-protection-blog/importance-of-coolants>>.

MOREIRA, C. A. da S. Finalidade e aplicações dos sensores industriais. **Repositorio Congna**, 2019. Disponível em: <<https://repositorio.pgsscogna.com.br>>. Acesso em: 02 Jun. 2024.

MOZILLA. Códigos de status de respostas http. 2023. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Status/>>. Acesso em: 13 set. 2024.

NISE, N. S. **Engenharia de sistemas de controle**. 8. ed. [S.l.], 2023.

OGATA, K. **Engenharia de controle moderno**. 5. ed. [S.l.], 2004.

OLIVEIRA, J. **Arduino, ESP32 e ESP8266**. 2019. Disponível em: <<https://xprojetos.net/arduino-esp32-e-esp8266-comparacao/>>. Acesso em: 05 de junho 2024.

PEREIRA, E. d. O. S. A. Indústria 4.0: Conceitos e perspectivas para o brasil. **Revista eletrônica da Universidade da Vale do Rio Verde.**, 2018. Disponível em: <<http://periodicos.unincor.br/index.php/revistaunincor/article/view/4938>>. Acesso em: 04 Maio. 2024.

RAIDEN, K. **Óleo solúvel para centro de usinagem**. São Paulo: [s.n.], 2024. Disponível em: <<https://www.krdobrasil.com.br/oleo-solúvel-centro-usinagem>>. Acesso em: 27 mai. 2024.

SACOMANO, J. B. **Indústria 4.0 : conceitos e fundamentos**. 1. ed. [S.l.], 2018.

SANTOS, B. P. Internet das coisas: da teoria à prática. **Repositorio Universidade Federal de Minas Gerais (UFMG)**, 2016. Disponível em: <<https://homepages.dcc.ufmg.br/~mmvieira/cc/papers/internet-das-coisas>>. Acesso em: 02 Jun. 2024.

SEIXAS. O que é http e como funciona. 2023. Disponível em: <<https://www.azion.com/pt-br/blog/o-que-e-http-e-como-ele-funciona/#:~:text=O%20HTTP%20%C3%A9%20um%20protocolo,para%20a%20troca%20de%20informa%C3%A7%C3%B5es.>>> Acesso em: 15 set. 2024.

SHIRMER, L. Gestão ambiental e as tecnologias 4.0 para a sustentabilidade nas indústrias do vale do taquari. **Contribuciones a las ciencias sociales**, 2024. Disponível em: <<https://ojs.revistacontribuciones.com/ojs/index.php/clcs/article/view/4138>>.

STMICROELETRONICS. **Stm8 8-bit mcus**. 2022. Disponível em: <<https://www.st.com/en/microcontrollers-microprocessors/stm8-8-bit-mcus.html>>. Acesso em: 04 de junho 2024.

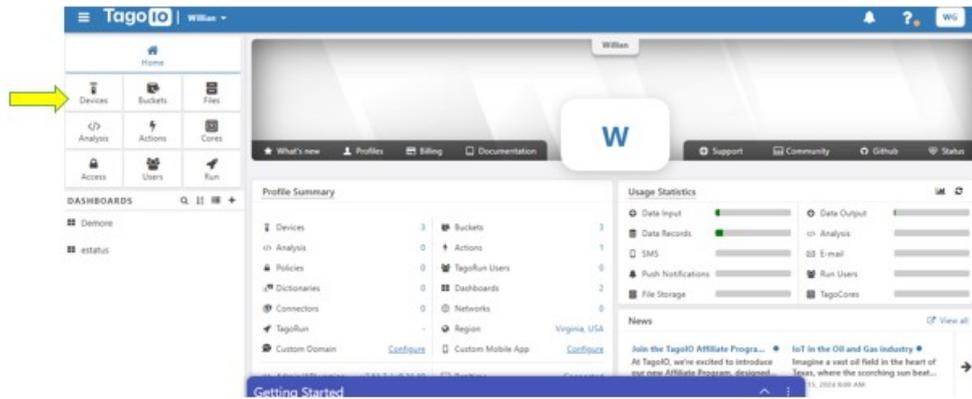
TAGO. **Tago.io**. 2024. Disponível em: <https://tago.io/>.

TORRES ATSLANDS R. ROCHA, J. N. d. S. A. B. B. Analise de desempenho de brokers mqtt em sistema de baixo custo. 2020. Disponível em: <<https://sol.sbc.org.br/index.php/wperformance/article/view/9727/9623>>. Acesso em: 04 Jun. 2024.

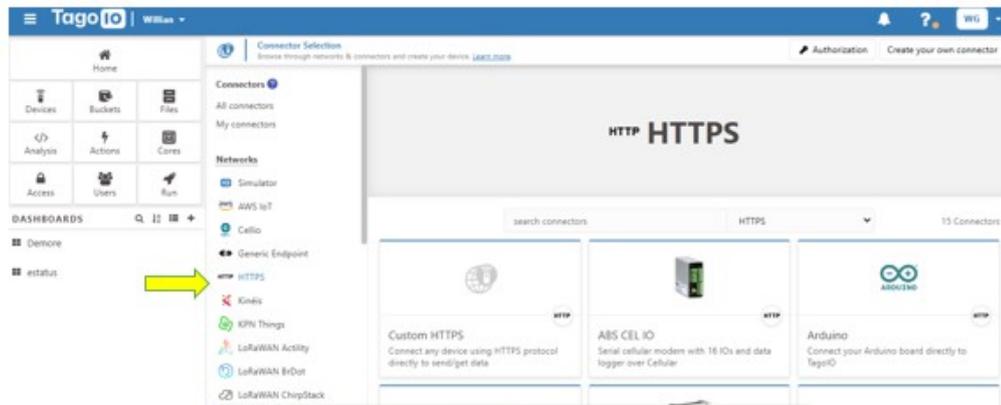
XAVIER, R. C. Conheça a tagoio, a primeira ferramenta cloud para desenvolvimento de solução iot homologada khomp. 2019. Disponível em: <<https://www.khomp.com/pt/tagoio-solucao-iot/>>. Acesso em: 10 set. 2024.

APÊNDICE B – CONFIGURAÇÃO DA PLATAFORMA TAGO.IO

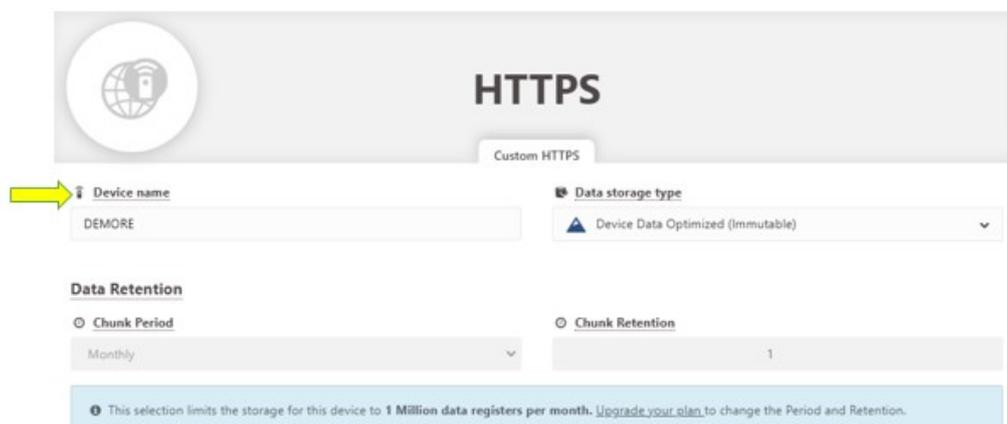
Adicionar Dispositivo.



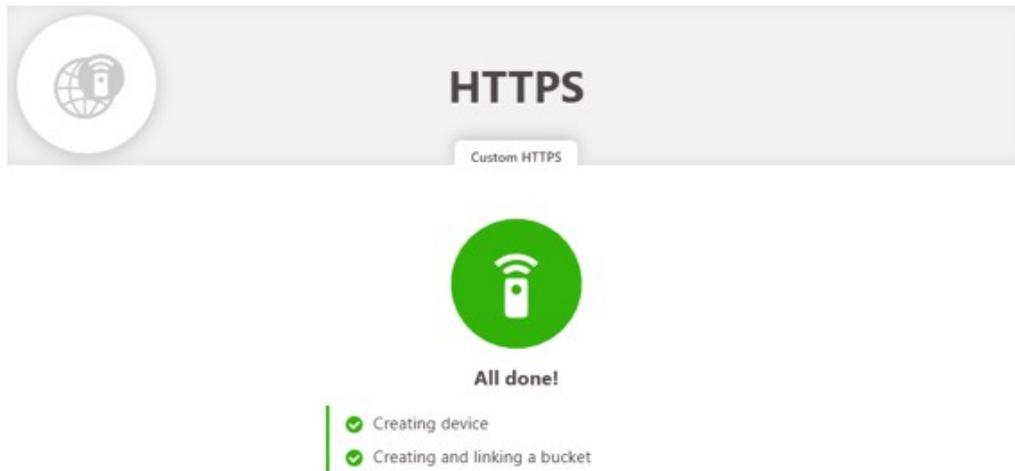
Seleciona HTTPS.



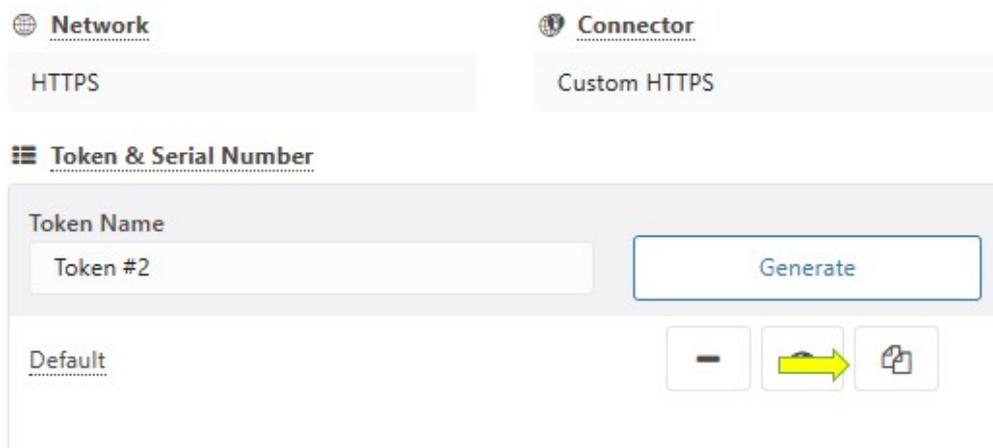
Seleciona custom HTTPS e define um nome para o dispositivo.



Finalizar.



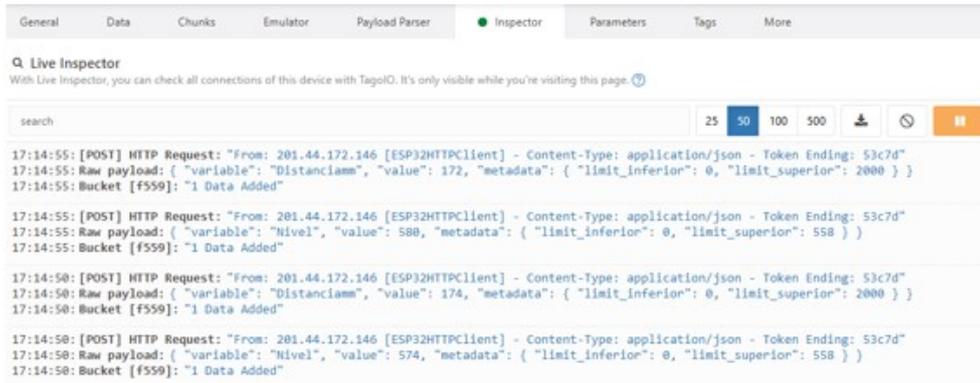
Copiar o token Default.



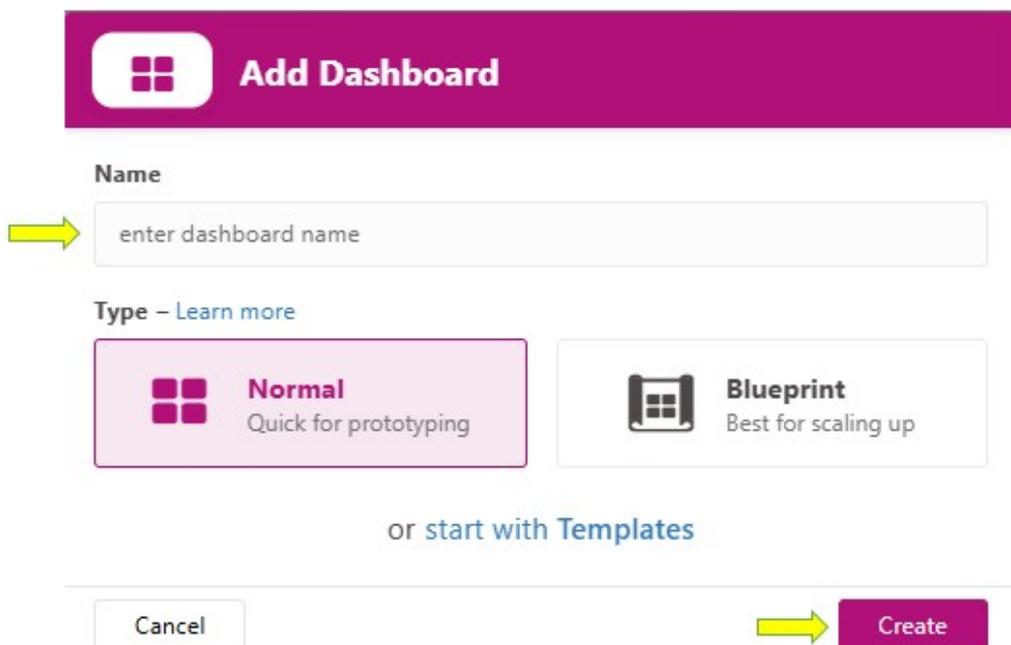
Este token deve ser colocado no firmware da ESP32.

```
// Configurações de rede e API
const char* ssid = "XXXXXXXXXX";           // NOME DA REDE WIFI LOCAL
const char* password = "XXXXXXXXXXXX";    // SENHA DA REDE WIFI LOCAL
const char* server = "api.tago.io";
const char* token = "XXXXXXXXXXXXXXXXXXXX"; // TOKEN DISPONIBILIZADO NA TAGO
```

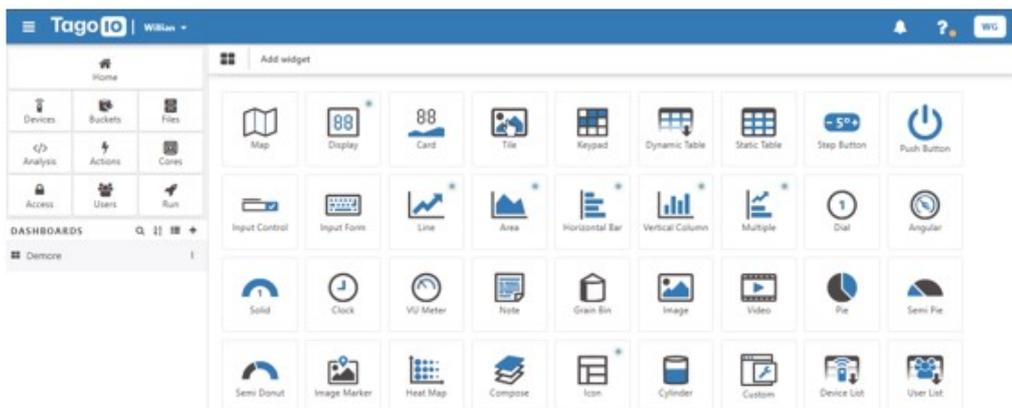
Confirma o recebimento das informações na plataforma TAGO.IO.



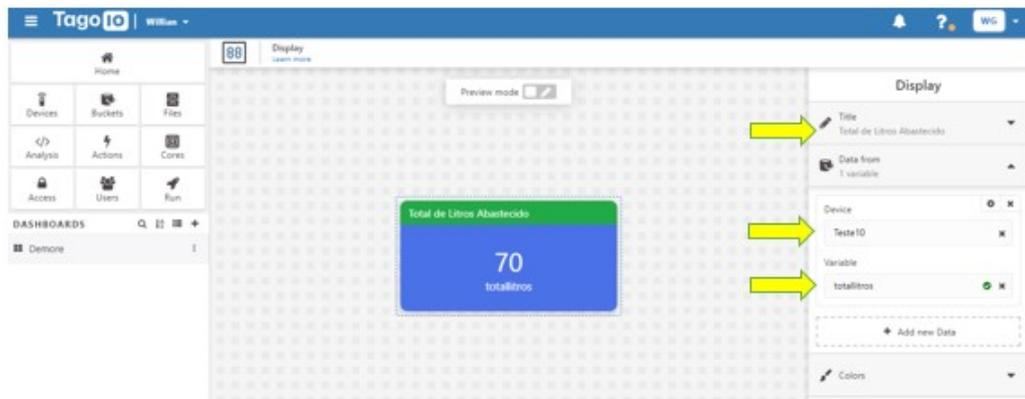
Define um nome para o Dashboard e cria o Dashboard.



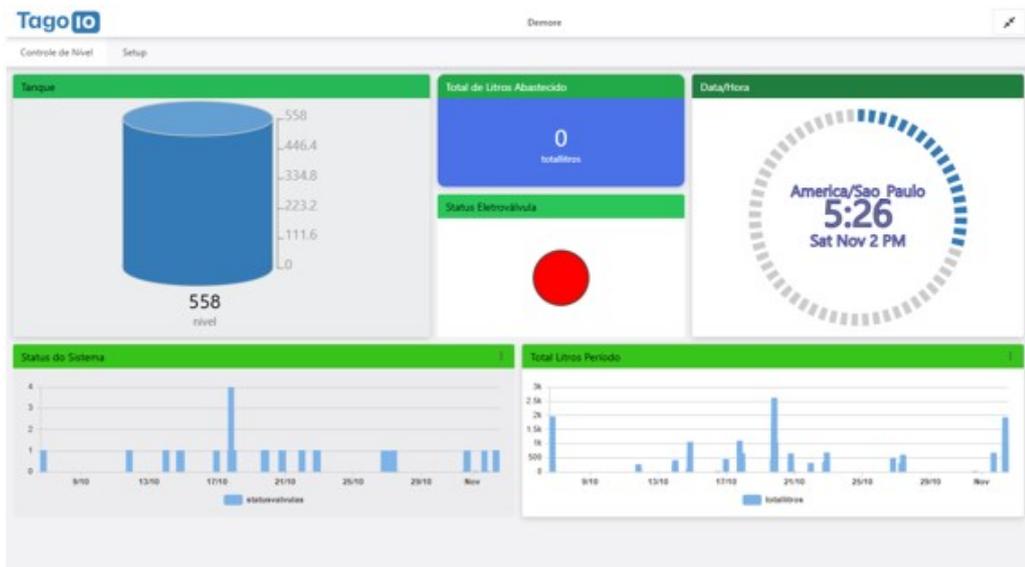
Escolhe uma opção de widget.



Define um nome para widget, seleciona o dispositivo, variável escolhida.



Dashboard criado para o controle de nível.



Setup Tanque MQ01

Section #1
My section description

✎ Distancia Sensor

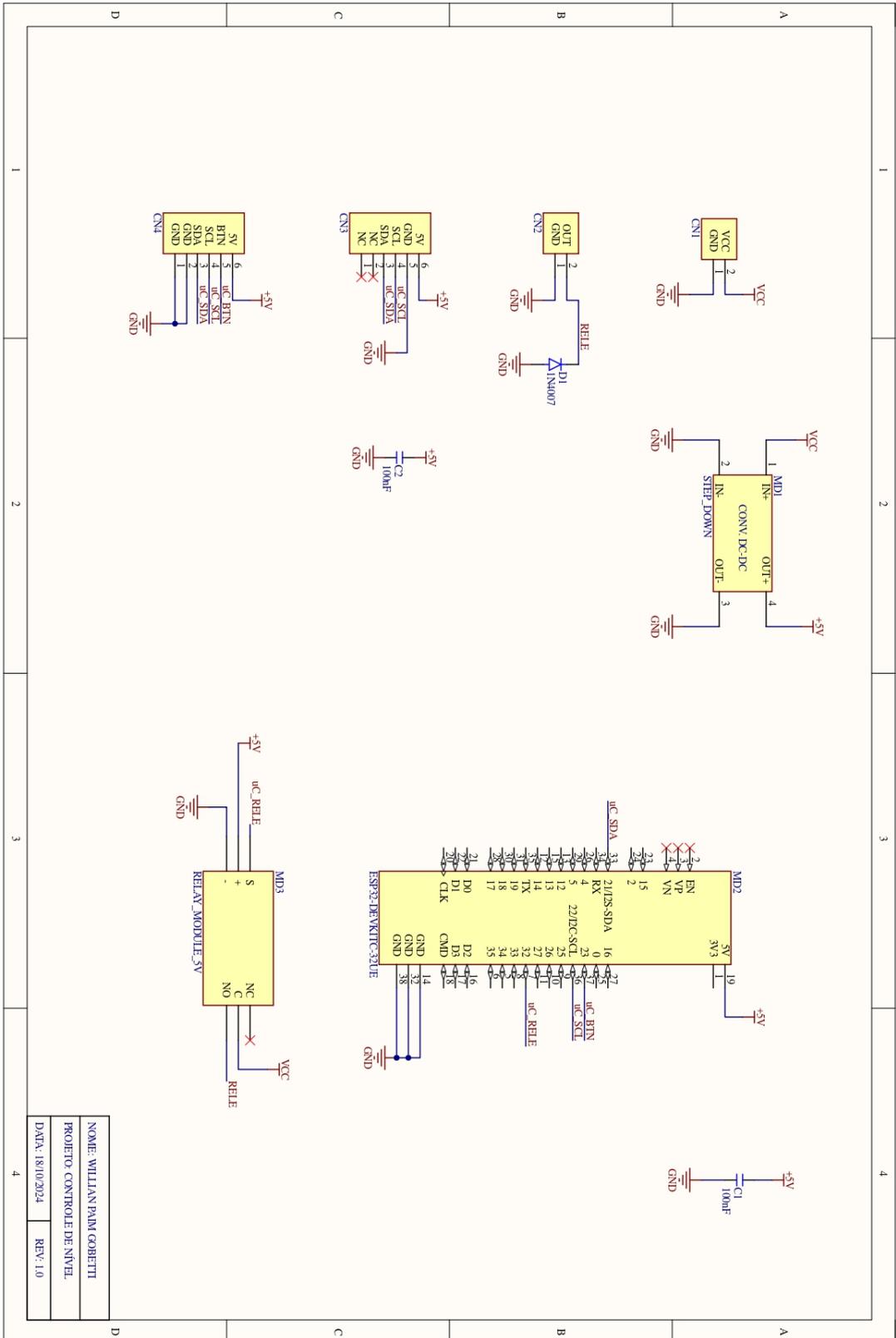
✎ Largura Tanque

✎ Comprimento Tanque

✎ Altura Tanque

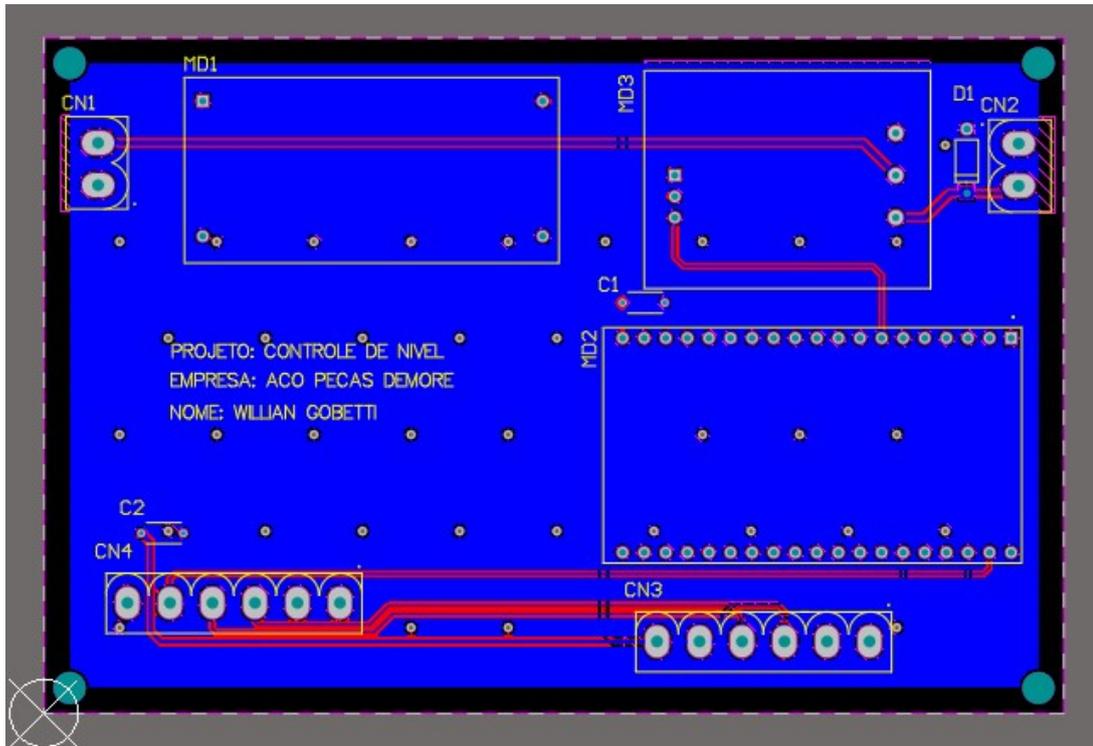
Submit

APÊNDICE C – ESQUEMÁTICO

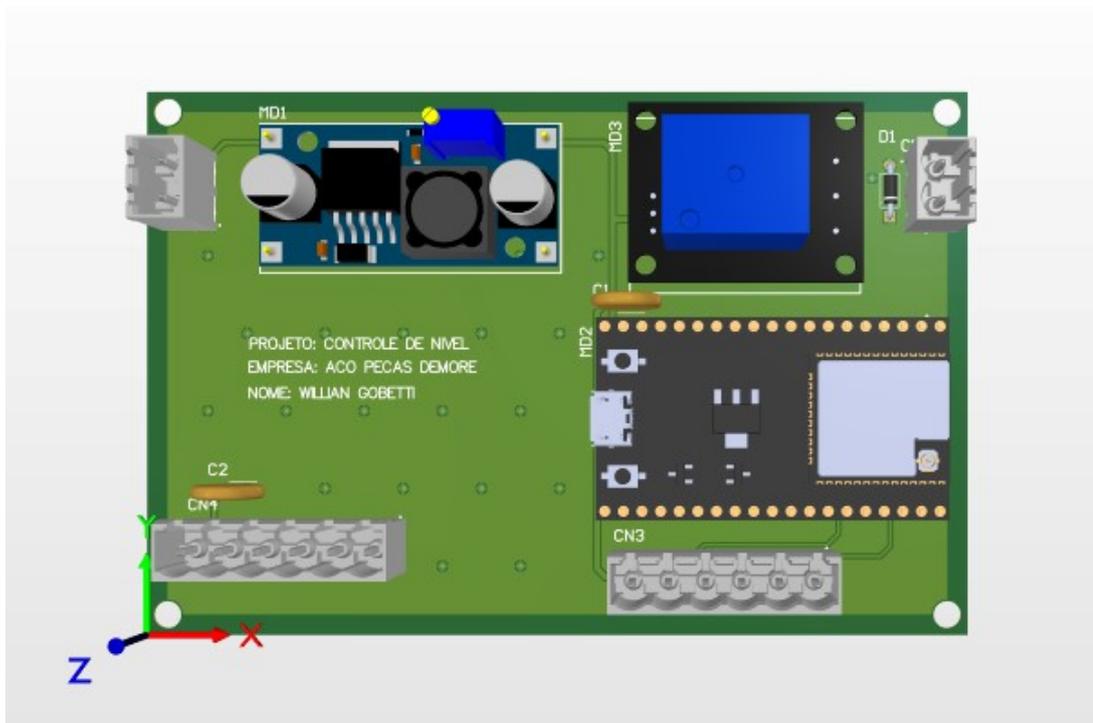


NOME: WILLIAN PAMI GOBETTI
 PROJETO: CONTROLE DE NÍVEL
 DATA: 18/10/2024 REV: 1.0

Placa PCB.



Placa 3D.



APÊNDICE D – FIRMWARE

```

1
2
3 #include <WiFi.h>           // Biblioteca para conexão WiFi
4 #include <HTTPClient.h>     // Biblioteca para requisições HTTP
5 #include <ArduinoJson.h>    // Biblioteca para manipulação de JSON
6 #include "Adafruit_VL53L0X.h" // Biblioteca do sensor de distância
7 #include <EEPROM.h>         // Biblioteca para armazenamento permanente
8 #include <LiquidCrystal_I2C.h> // Biblioteca do display LCD
9
10
11 // Configurações de rede e API
12 const char* ssid = "Demore_VIP";
13 const char* password = "VIP#APDnet!@#$380";
14 const char* server = "api.tago.io";
15 const char* token = "88a8f05e-68a7-40ed-84e8-a45e32e53c7d";
16
17 // Estrutura para armazenamento na EEPROM
18 struct ConfigData {
19     long totalLitros;
20     int distSensor;
21     int widthTank;
22     int lengthTank;
23     int heightTank;
24 };
25
26 // Inicializa o sensor e o display
27 Adafruit_VL53L0X lox = Adafruit_VL53L0X();
28 LiquidCrystal_I2C lcd(0x3F, 16, 2);
29
30 // Variáveis de controle do WiFi
31 bool wifiConnected = false;
32 unsigned long lastWifiAttempt = 0;
33 const unsigned long WIFI_RETRY_INTERVAL = 300000; // 5 minutos
34
35 // Variáveis do tanque
36 int distSensor = 180;           // Distância do sensor ao topo do tanque (mm) 150
37 int widthTank = 2000;           // Largura do tanque (mm) 2000
38 int lengthTank = 1395;         // Comprimento do tanque (mm) 1300
39 int heightTank = 200;          // Altura do tanque (mm)
40 float volumeTotal = 0;         // Volume total calculado
41 int volume = 0;                // Volume atual
42 int totalLitros = 0;           // Total de litros abastecidos
43 int totalLitroInicioAbastecimento = 0; // Volume no início do abastecimento
44 int address = 0;              // Endereço atual da EEPROM
45 int index_menu = 0;           // Índice do menu atual
46
47 // Variáveis de controle de tempo
48 int time_display = 3000;       // Tempo de exibição do display
49 unsigned long backup_time_display = 0; // Último momento de atualização do display
50 unsigned long backup_time_mensure_range = 0; // Último momento de medição
51 int statusVal = 0;             // Estado da válvula (0 = fechada, 1 = aberta)
52
53 // Definições do sistema
54 #define EEPROM_SIZE sizeof(ConfigData)
55 #define valvula 32             // Pino da válvula
56 #define BOTAO_PIN 23          // Pino do botão
57
58 /**

```

```

59  * Carrega as configura es salvas na EEPROM
60  * Esta fun o chamada quando o sistema inicia sem WiFi
61  */
62  void loadConfigFromEEPROM() {
63      ConfigData data;
64      EEPROM.get(0, data);
65
66      Serial.println("Lendo_dados_da_EEPROM:");
67      Serial.print("Total_Litros:_");
68      Serial.println(data.totalLitros);
69
70      // Verifica se os dados s o v lidos
71      if (data.totalLitros >= 0 && data.totalLitros < 1000000) {
72          totalLitros = data.totalLitros;
73          distSensor = data.distSensor;
74          widthTank = data.widthTank;
75          lenghtTank = data.lenghtTank;
76          heightTank = data.heightTank;
77      } else {
78          // Usa valores padr o se os dados n o forem v lidos
79          totalLitros = 0;
80          distSensor = 180;
81          widthTank = 2000;
82          lenghtTank = 1395;
83          heightTank = 200;
84          saveConfigToEEPROM();
85      }
86
87      volumeTotal = calculateVolume(distSensor);
88  }
89
90  /**
91  * Salva as configura es atuais na EEPROM
92  * Esta fun o chamada sempre que houver altera es nas configura es
93  */
94  void saveConfigToEEPROM() {
95      ConfigData data;
96      data.totalLitros = totalLitros;
97      data.distSensor = distSensor;
98      data.widthTank = widthTank;
99      data.lenghtTank = lenghtTank;
100     data.heightTank = heightTank;
101
102     Serial.println("Salvando_dados_na_EEPROM:");
103     Serial.print("Total_Litros:_");
104     Serial.println(data.totalLitros);
105
106     EEPROM.put(0, data);
107     EEPROM.commit();
108  }
109  /**
110  * Configura e tenta estabelecer conex o WiFi
111  * Inclui timeout e atualiza o do display com status
112  */
113  void setupWifi() {
114      unsigned long startAttempt = millis();
115      const unsigned long WIFI_TIMEOUT = 10000; // 10 segundos timeout
116
117      WiFi.begin(ssid, password);
118
119      lcd.clear();
120      lcd.print("Tentando_WiFi...");
121

```

```

122   while (WiFi.status() != WL_CONNECTED && millis() - startAttempt < WIFI_TIMEOUT) {
123       delay(500);
124   }
125
126   if (WiFi.status() == WL_CONNECTED) {
127       wifiConnected = true;
128       lcd.clear();
129       lcd.print("WiFi_Conectado");
130       delay(1000);
131
132       // Atualiza configura es do TagoIO
133       int newDistSensor = getToTagoIOVariable("distanciaSensor");
134       int newWidthTank = getToTagoIOVariable("larguraTanque");
135       int newLenghtTank = getToTagoIOVariable("comprimentoTanque");
136       int newHeightTank = getToTagoIOVariable("alturaTanque");
137
138       if (newDistSensor != -1 && newWidthTank != -1 &&
139           newLenghtTank != -1 && newHeightTank != -1) {
140           distSensor = newDistSensor;
141           widthTank = newWidthTank;
142           lenghtTank = newLenghtTank;
143           heightTank = newHeightTank;
144           saveConfigToEEPROM();
145       }
146   } else {
147       wifiConnected = false;
148       lcd.clear();
149       lcd.print("Modo_Offline");
150       delay(1000);
151       loadConfigFromEEPROM();
152   }
153 }
154
155 void setup() {
156     Serial.begin(115200);
157
158     lcd.init();
159     lcd.backlight();
160     lcd.print("Iniciando ...");
161
162     EEPROM.begin(EEPROM_SIZE);
163     loadConfigFromEEPROM();
164
165     setupWifi();
166
167     pinMode(BOTAO_PIN, INPUT_PULLUP);
168     pinMode(valvula, OUTPUT);
169     digitalWrite(valvula, LOW);
170
171     volumeTotal = calculateVolume(distSensor);
172
173     if (wifiConnected) {
174         sendToTagoIOVariable("StatusValvulas", statusVal, 0, 1);
175     }
176
177     if (!lox.begin()) {
178         Serial.println(F("Falha_ao_iniciar_sensor_VL53L0X"));
179         lcd.clear();
180         lcd.print("Erro_no_sensor");
181         while (1);
182     }
183
184     lcd.clear();

```

```

185     lcd.print("Sistema_Pronto");
186 }
187
188 void loop() {
189     if (!wifiConnected && millis() - lastWifiAttempt > WIFI_RETRY_INTERVAL) {
190         setupWifi();
191         lastWifiAttempt = millis();
192     }
193
194     if ((millis() - backup_time_mensure_range) > 5000) {
195         backup_time_mensure_range = millis();
196         VL53L0X_RangingMeasurementData_t measure;
197
198         Serial.print("Reading_a_measurement...");
199         lox.rangingTest(&measure, false);
200
201         if (measure.RangeStatus != 4) {
202             Serial.print("Distance_(mm):_");
203             Serial.println(measure.RangeMilliMeter);
204             volume = calculateVolume(measure.RangeMilliMeter);
205
206             if (volume < (volumeTotal * 0.5) && statusVal == 0) {
207                 digitalWrite(valvula, HIGH);
208                 statusVal = 1;
209                 Serial.println("Valvula_Ligada");
210                 if (wifiConnected) {
211                     sendToTagIOVariable("StatusValvulas", statusVal, 0, 1);
212                 }
213                 totalLitroInicioAbastecimento = volume;
214             }
215
216             if (volume > (volumeTotal * 0.9) && statusVal == 1) {
217                 digitalWrite(valvula, LOW);
218                 statusVal = 0;
219                 Serial.println("Valvula_Desligada");
220                 if (wifiConnected) {
221                     sendToTagIOVariable("StatusValvulas", statusVal, 0, 1);
222                 }
223
224                 long litrosAbastecidos = volume - totalLitroInicioAbastecimento;
225                 if (litrosAbastecidos > 0) {
226                     totalLitros += litrosAbastecidos;
227                     saveConfigToEEPROM();
228
229                     if (wifiConnected) {
230                         sendToTagIOVariable("TotalLitros", totalLitros, 0, 0);
231                     }
232                 }
233             }
234
235             if (wifiConnected) {
236                 sendToTagIOVariable("Nivel", volume, 0, volumeTotal);
237                 sendToTagIOVariable("Distanciamm", measure.RangeMilliMeter, 0, 2000);
238             }
239         } else {
240             Serial.println("out_of_range");
241             changeMenuAndReset(10, 0);
242         }
243     }
244     if ((millis() - backup_time_display) > time_display) {
245         backup_time_display = millis();
246         time_display = 3000;
247     }

```

```

248     index_menu++;
249     index_menu = index_menu % 3;
250     changeMenuAndReset(index_menu, 0);
251 }
252
253 if (digitalRead(BOTAO_PIN) == LOW) {
254     bool flag = 0;
255     backup_time_display = millis();
256     time_display = 10000;
257
258     while(digitalRead(BOTAO_PIN) == LOW) {
259         if (millis() - backup_time_display > 5000) {
260             changeMenuAndReset(index_menu, 1);
261             flag = 1;
262             break;
263         }
264     }
265
266     if (!flag) {
267         index_menu++;
268         index_menu = index_menu % 3;
269     }
270     changeMenuAndReset(index_menu, 0);
271 }
272 }
273
274 void changeMenuAndReset(int index, bool flagReset) {
275     lcd.clear();
276     switch(index) {
277         case 0:
278             lcd.print("Total_Abastecido:");
279             lcd.setCursor(0, 1);
280             lcd.print(totalLitros);
281             lcd.print("_L");
282             if (flagReset) {
283                 totalLitros = 0;
284                 saveConfigToEEPROM();
285                 if (wifiConnected) {
286                     sendToTagIOVariable("TotalLitros", totalLitros, 0, 0);
287                 }
288             }
289             break;
290
291         case 1:
292             lcd.print("Nivel_Atual:");
293             lcd.setCursor(0, 1);
294             lcd.print(volume);
295             lcd.print("_L");
296             break;
297
298         case 2:
299             lcd.print("Status_Valvula:");
300             lcd.setCursor(0, 1);
301             if (statusVal) {
302                 lcd.print("Aberta");
303                 if (flagReset) {
304                     digitalWrite(valvula, LOW);
305                     statusVal = 0;
306                     Serial.println("Valvula_Desligada");
307                     if (wifiConnected) {
308                         sendToTagIOVariable("StatusValvulas", statusVal, 0, 1);
309                     }
310                 }

```

```

311         } else {
312             lcd.print("Fechada");
313             if(flagReset) {
314                 digitalWrite(valvula, HIGH);
315                 statusVal = 1;
316                 Serial.println("Valvula_Ligada");
317                 if (wifiConnected) {
318                     sendToTagoIOVariable("StatusValvulas", statusVal, 0, 1);
319                 }
320             }
321         }
322         break;
323
324     case 10:
325         lcd.print("Fora_de_alcance");
326         lcd.setCursor(0, 1);
327         lcd.print("Sensor");
328         break;
329
330     default:
331         lcd.print("Menu_inexistente");
332         break;
333     }
334 }
335
336 void sendToTagoIOVariable(String VariableName, int value, int limit_inferior, int limit_superior) {
337     if (!wifiConnected) return;
338
339     StaticJsonDocument<200> jsonDocument;
340
341     jsonDocument["variable"] = VariableName;
342     jsonDocument["value"] = value;
343     jsonDocument["metadata"]["limit_inferior"] = limit_inferior;
344     jsonDocument["metadata"]["limit_superior"] = limit_superior;
345
346     String jsonString;
347     serializeJson(jsonDocument, jsonString);
348
349     HTTPClient http;
350
351     http.begin("http://" + String(server) + "/data");
352     http.addHeader("Content-Type", "application/json");
353     http.addHeader("token", token);
354
355     Serial.println(jsonString);
356
357     int httpResponseCode = http.POST(jsonString);
358
359     if (httpResponseCode == 202) {
360         Serial.println("Sucesso_no_envio");
361     } else {
362         Serial.print("Erro:");
363         Serial.println(httpResponseCode);
364         Serial.println(http.getString());
365     }
366
367     http.end();
368 }
369
370 int getToTagoIOVariable(String VariableName) {
371     int valor = -1;
372     HTTPClient http;
373

```

```

374 http.begin("http://" + String(server) + "/data?variable=" + VariableName + "&query=last_item");
375 http.addHeader("Content-Type", "application/json");
376 http.addHeader("token", token);
377
378 int httpResponseCode = http.GET();
379
380 if (httpResponseCode == 200) {
381     Serial.println("Sucesso_em_receber_dado");
382     String payload = http.getString();
383
384     StaticJsonDocument<200> jsonDocument;
385     DeserializationError err = deserializeJson(jsonDocument, payload);
386
387     if (err) {
388         Serial.print(F("deserializeJson()_failed_with_code_"));
389         Serial.println(err.f_str());
390     } else {
391         valor = jsonDocument["result"][0]["value"];
392     }
393 } else {
394     Serial.print("Erro:_");
395     Serial.println(httpResponseCode);
396 }
397 http.end();
398
399 return valor;
400 }
401
402 float calculateVolume(int dist) {
403     float volume = (widthTank * lenghtTank * ((heightTank + distSensor) - dist)) / 1000000.00;
404     return volume;
405 }

```

APÊNDICE E – TABELA DE MEDIDA

Distancia 100		Distancia 150		Distancia 200	
Data e Horário	Nível	Data e Horário	Nível	Data e Horário	Nível
05/11/2024 12:07:41	271	05/11/2024 12:22:27	412	05/11/2024 12:43:03	558
05/11/2024 12:07:00	272	05/11/2024 12:22:27	405	05/11/2024 12:42:58	551
05/11/2024 12:06:19	281	05/11/2024 12:22:27	407	05/11/2024 12:42:53	556
05/11/2024 12:05:38	282	05/11/2024 12:22:27	422	05/11/2024 12:42:48	557
05/11/2024 12:04:57	279	05/11/2024 12:22:27	415	05/11/2024 12:42:43	558
05/11/2024 12:04:16	285	05/11/2024 12:22:27	417	05/11/2024 12:42:38	557
05/11/2024 12:03:35	279	05/11/2024 12:22:27	422	05/11/2024 12:42:33	559
05/11/2024 12:02:54	274	05/11/2024 12:22:27	406	05/11/2024 12:42:28	552
05/11/2024 12:02:13	279	05/11/2024 12:22:27	406	05/11/2024 12:42:23	555
05/11/2024 12:01:32	282	05/11/2024 12:22:27	409	05/11/2024 12:42:18	548
05/11/2024 12:00:51	284	05/11/2024 12:22:27	415	05/11/2024 12:42:13	549
05/11/2024 12:00:10	285	05/11/2024 12:22:27	416	05/11/2024 12:42:08	551
05/11/2024 11:59:29	279	05/11/2024 12:22:27	420	05/11/2024 12:42:03	558
05/11/2024 11:58:48	282	05/11/2024 12:22:27	410	05/11/2024 12:41:58	545
05/11/2024 11:58:07	271	05/11/2024 12:22:27	411	05/11/2024 12:41:53	557
05/11/2024 11:57:26	265	05/11/2024 12:22:27	412	05/11/2024 12:41:48	558
05/11/2024 11:56:45	283	05/11/2024 12:22:27	413	05/11/2024 12:41:43	557
05/11/2024 11:56:04	278	05/11/2024 12:22:27	415	05/11/2024 12:41:38	558
05/11/2024 11:55:23	277	05/11/2024 12:22:27	420	05/11/2024 12:41:33	558
05/11/2024 11:54:42	278	05/11/2024 12:22:27	419	05/11/2024 12:41:28	557
05/11/2024 11:54:01	267	05/11/2024 12:22:27	418	05/11/2024 12:41:23	558
05/11/2024 11:53:20	262	05/11/2024 12:22:27	419	05/11/2024 12:41:18	558
05/11/2024 11:52:39	280	05/11/2024 12:22:27	419	05/11/2024 12:41:13	557
05/11/2024 11:51:58	265	05/11/2024 12:22:27	418	05/11/2024 12:41:08	559
05/11/2024 11:51:17	271	05/11/2024 12:22:27	420	05/11/2024 12:41:03	558
05/11/2024 11:50:36	277	05/11/2024 12:22:27	420	05/11/2024 12:40:58	556
05/11/2024 11:49:55	279	05/11/2024 12:22:27	421	05/11/2024 12:40:53	558
05/11/2024 11:49:14	280	05/11/2024 12:22:27	419	05/11/2024 12:40:48	557
05/11/2024 11:48:33	281	05/11/2024 12:22:27	420	05/11/2024 12:40:43	556
05/11/2024 11:47:52	281	05/11/2024 12:22:27	419	05/11/2024 12:40:38	555

APÊNDICE F – TABELA DE CUSTO

Tabela de componetes e valor usado no projeto.			
Componete	QTD	Valor	Venda
Esp32U Devkit	1,00	R\$ 70,30	Mercado Livre
Modulo rele HW-482 V2.0.1 5V	1,00	R\$ 12,90	Mercado Livre
Conversor DC/DC step dow LM2596	1,00	R\$ 6,72	Eletrus
Sensor VL53L0x	1,00	R\$ 19,99	Casa da Robotica
Display lcd 16X2	1,00	R\$ 23,10	a2Robotios
Conector kk 4 vias	1,00	R\$ 2,00	Eletrus
Conector akz 6 vias	2,00	R\$ 5,00	Eletrus
Conector akz 2 vias	2,00	R\$ 28,00	Eletrus
Placa circuito impresso 7x9cm	1,00	R\$ 17,90	Mercado Livre
Botão	1,00	R\$ 12,00	Eletrus
Eletrovalvula 24V	1,00	R\$ 120,00	Prima-Serra
Prensa cabo 6mm	4,00	R\$ 16,00	Celetro
Caixa plastica patola 150x200x90	1,00	R\$ 39,90	Celetro
Capacitor de 100nF	3,00	R\$ 6,00	Eletrus
Diodo 1N4007	1,00	R\$ 1,00	Eletrus
Cabo flat 30cm	1,00	R\$ 12,00	Eletrus
Adaptador sensor Laser	1,00	R\$ 30,00	Particular
Total	24	R\$ 422,81	

APÊNDICE G – PROJETO INSTALADO





