



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO
AMAZONAS
CAMPUS MANAUS-DISTRITO INDUSTRIAL
TECNOLOGIA EM MECATRÔNICA INDUSTRIAL**

JOÃO VICTOR PONTES DUTRA

**SISTEMA DE INSPEÇÃO DE BATERIA DE LÍTIO EM PLACAS ELETRÔNICAS:
UMA PROPOSTA A PARTIR DA INTEGRAÇÃO DE CLP E APRENDIZADO
DE MÁQUINA**

**MANAUS-AM
2025**

JOÃO VICTOR PONTES DUTRA

**SISTEMA DE INSPEÇÃO DE BATERIA DE LÍTIO EM PLACAS ELETRÔNICAS:
UMA PROPOSTA A PARTIR DA INTEGRAÇÃO DE CLP E APRENDIZADO
DE MÁQUINA**

Trabalho de Conclusão de Curso apresentada à banca examinadora do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas como requisito parcial para obtenção do título de Tecnólogo em Mecatrônica Industrial.

Orientador (a): Prof. Dr. João Bernardo Aranha Ribeiro

**MANAUS-AM
2025**

Dados Internacionais de Catalogação na Publicação (CIP)

D978s	<p>Dutra, João Victor Pontes.</p> <p>Sistema de inspeção de bateria de lítio em placas eletrônicas: uma proposta a partir da integração de CLP e aprendizado de máquina/ João Victor Pontes Dutra. — Manaus, 2025. 74 f.: il. color.</p> <p>Monografia (Graduação) — Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, <i>Campus</i> Manaus Distrito Industrial, Curso de Tecnologia em Mecatrônica Industrial, 2025. Orientador: Prof. João Bernardo Aranha Ribeiro, Dr.</p> <p>1. Indústria 4.0. 2. Aprendizado de máquina. 3. CLP. I. Ribeiro, João Bernardo Aranha. II. Instituto Federal de Educação, Ciência e Tecnologia do Amazonas. III. Título.</p> <p style="text-align: right;">CDD 629.892</p>
-------	--

Elaborada por Oziane Romualdo de Souza (CRB 11/ nº 734).

JOÃO VICTOR PONTES DUTRA

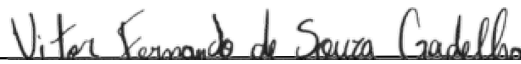
**SISTEMA DE INSPEÇÃO DE BATERIA DE LÍTIO EM PLACAS ELETRÔNICAS:
Uma proposta a partir da integração de CLP e Aprendizado de Máquina**

Monografia apresentada à banca examinadora do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas como requisito parcial para obtenção do título de Tecnólogo em Mecatrônica Industrial.

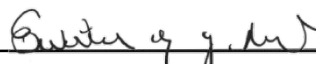
Orientador: Prof. Dr. João Bernardo Aranha Ribeiro

Aprovado em 16 de setembro de 2025.

BANCA EXAMINADORA



Prof. Esp. VITOR FERNANDO DE SOUZA GADELHA
Instituto Federal de Educação, Ciência e Tecnologia do Amazonas



Prof. Esp. EWERTON ANDREY GODINHO RIBEIRO
Instituto Federal de Educação, Ciência e Tecnologia do Amazonas

MANAUS-AM
2025

AGRADECIMENTOS

Primeiramente gostaria de agradecer a Deus pelo fôlego de vida, saúde e inteligência, à minha mãe Esmeralda Pontes pelo apoio e suporte em todos os momentos, aos meus amigos e professores do IFAM Campus Manaus Distrito Industrial, em particular aos senhores João Bernardo (orientador), Daniel Fonseca (coordenador) e ao time de Engenharia de Automação do Instituto Calcomp de Tecnologia que sempre me apoiaram e me encorajaram a nunca desistir mesmo perante as dificuldades. Sem eles não seria possível alcançar tal feito.

RESUMO:

O presente trabalho tem como objetivo o desenvolvimento de um sistema de inspeção de baterias de lítio a partir da integração do Controlador Lógico Programável (CLP) e Aprendizado de Máquina (AM), identificando vantagens, desvantagens e desafios associados ao atendimento das necessidades das instalações industriais. O projeto busca utilizar materiais e tecnologias de baixo custo, garantindo sua viabilidade econômica e facilidade de implementação. A inovação do trabalho está na adaptação dos critérios organizacionais, utilizando ferramentas da indústria 4.0, proporcionando uma solução prática, eficiente e assertiva. Dessa forma, o sistema proposto visa atender a esses aspectos, oferecendo uma alternativa acessível para detecção de falhas de montagem, otimização e qualidade. A integração foi feita usando redes industriais como Modbus/TCP, bastante utilizadas no âmbito industrial. A inspeção do sistema baseia-se em um aprendizado supervisionado, realizando a detecção da bateria via webcam e, em seguida, compara a imagem capturada com o modelo criado a partir do Dataset OK e NG, e, portanto, a comparação entre ambas implicará em um resultado aprovado ou reprovado da peça. O sistema desenvolvido foi capaz de obter um bom desempenho de detecção e classificação durante os testes e validações. Ademais, o trabalho designou a observação dessa integração nos demais segmentos, instigando oportunidades e melhorias para efetivo avanço. Portanto, o estudo concluiu que, apesar das adversidades para a implementação de AM aos CLP's serem significativas, a integração demonstra um rumo promissor para otimizar a eficiência operacional, a flexibilidade, a transmissão de dados, a redução de custos e o auxílio nas tomadas de decisão em empresas que possuem recursos limitados e carência de modernização.

Palavras-chave: Aprendizado de Máquina. CLP. Integração. Indústria 4.0. Inspeção.

ABSTRACT:

The objective of this work is to develop a lithium battery inspection system based on the integration of Programmable Logic Controllers and Machine Learning, identifying advantages, disadvantages, and challenges associated with meeting the needs of industrial facilities. The project seeks to use low-cost materials and technologies, ensuring its economic viability and ease of construction. The innovation of the work lies in the adaptation of organizational needs, using Industry 4.0 tools, providing a practical, efficient, and assertive solution. Thus, the proposed system aims to address these aspects, offering an accessible solution for detecting assembly, optimization, and quality failures. The system inspection is based on a supervised learning that detects the battery via webcam and then compares the captured image with the model created from the OK and NG datasets. Therefore, the difference between the two results in an approved or rejected result for the part. The developed system performed well in its inspections during testing and validation. In addition, the work designated the observation of this integration in other segments, instigating opportunities and improvements for effective advancement. Therefore, the study concluded that, despite the significant challenges to implementing AM in PLCs, the integration shows promise for optimizing operational efficiency, flexibility, data transmission, cost reduction, and decision-making support in companies with limited resources and a need for modernization.

Keywords: Machine Learning. PLC. Integration. Industry 4.0. Inspection.

LISTA DE ILUSTRAÇÕES

- Figura 1: Criação do primeiro CLP por Dick Morley.
- Figura 2: Fluxograma de funcionamento do CLP.
- Figura 3: Estrutura física do autor linear.
- Figura 4: Softwares e ferramentas projetados para utilização.
- Figura 5: Sistema de visão computacional em parques de estacionamento.
- Figura 6: Exemplos de sistemas e câmeras disponíveis no mercado.
- Figura 7: Equipamentos utilizados na construção do projeto.
- Figura 8: Fluxograma sequencial de funcionamento do sistema.
- Figura 9: Componentes do painel elétrico.
- Figura 10: Demais componentes.
- Figura 11: Layout do painel elétrico.
- Figura 12: Montagem final do painel.
- Figura 13: Descrição de pinos e conexões do painel.
- Figura 14: Modelo do atuador linear utilizado no projeto.
- Figura 15: Equipamentos utilizados na montagem do atuador.
- Figura 16: Montagem final e conexões do atuador.
- Figura 17: Configuração e definição de variáveis do servo.
- Figura 18: Parametrização de funções do servo.
- Figura 19: Demais parametrizações do servo.
- Figura 20: Definição de endereço IP e sub-máscara do servo.
- Figura 21: Criação do projeto e alteração do modelo CPU.
- Figura 22: Configuração de parâmetros Ethernet no CLP via GX Works3.
- Figura 23: Ativação da função CC Link Basic no software GX.
- Figura 24: Identificação e comunicação do servoamplificador via CC Link.
- Figura 25: Tabela de entradas digitais do CLP.
- Figura 26: Tabela de saídas digitais do CLP.
- Figura 27: Tabela de memórias do CLP.
- Figura 28: Demais comandos.
- Figura 29: Tabela de conversão para comunicação Modbus.
- Figura 30: Parâmetros de referência para comunicação Modbus.
- Figura 31: Inclusão da biblioteca FB do servo no software GX.
- Figura 32: Importação do arquivo para o software GX.

Figura 33: Definição de variáveis globais.

Figura 34: Dataset para peças OK.

Figura 35: Dataset para peças NG.

Figura 36: Posicionamento da placa para definição da ROI.

Figura 37: Área definida da ROI padrão.

Figura 38: Definição das coordenadas X1 e Y1 da ROI.

Figura 39: Definição das coordenadas X2 e Y2 da ROI.

Figura 40: Modificação do endereço IP e máscara do PC.

Figura 41: Ativação da configuração Modbus/TCP no software GX.

Figura 42: Adição da comunicação Modbus/TCP ao software GX.

Figura 43: Criação do modelo para inspeção.

Figura 44: Funcionamento integral do sistema

Figura 45: Teste do funcionamento prático do sistema.

Figura 46: Resultados de inspeção da bateria gerado pelo sistema

Figura 47: Gráfico de resultados gerais após testes.

Figura 48: Gráfico de representatividade de rejeitos.

Figura 49: Gráfico de eficácia geral do sistema.

Figura 50: Tabela de custo total do projeto.

Figura 51: Gráfico comparativo entre sistemas e câmeras comerciais

LISTA DE ABREVIATURAS E SIGLAS

AM	Aprendizado de Máquina
CLP	Controle Lógico Programável
CAN	<i>Controller Area Network</i>
FBD	<i>Function Block Diagram</i>
IA	Inteligência Artificial
IHM	Interface Homem-Máquina
IL	<i>Instruction List</i>
IP	<i>Internet Protocol</i>
IED	Instrumento Eletrônico Inteligente
LD	Linguagem Ladder
ML	Machine Learning
MPL	<i>Multi Layer Perceptron</i>
PDCA	<i>Plan, Do, Check and Act</i>
PLN	Processamento de Linguagem Natural
PWM	Pulse Width Modulation
PC	Personal Computer
RNA	Rede Neural Artificial
ROI	<i>Region Of Interest</i>
ST	<i>Structured Text</i>
SFC	<i>Sequential Function Chart</i>
SCADA	<i>Supervisory Control And Data Acquisition</i>
VS CODE	<i>Visual Studio Code</i>

SUMÁRIO

1 INTRODUÇÃO	13
1.1 Estrutura do Trabalho	13
1.2 Justificativa	14
1.3 Motivação	14
1.4 Objetivos	15
1.4.1 Objetivo geral	15
1.4.2 Objetivos específicos	15
2 REFERENCIAL TEÓRICO	16
2.1 Hardwares utilizados	16
2.1.1 Controle lógico programável	16
2.1.2 Atuador linear	19
2.1.3 Webcam	20
2.1.4 Outros hardwares	21
2.2 Software utilizados	22
2.3 Inteligência Artificial e Aprendizado de Máquina	23
2.4 Protocolos de rede	24
2.5 Vantagens e desafios da integração	26
2.6 Inspeção na indústria	27
2.6.1 Tipos de inspeções industriais	27
2.6.2 Tecnologia de inspeção na indústria	28
2.6.3 Soluções disponíveis no mercado	29
3 O SISTEMA	31
3.1 Projetando o sistema	31
3.2 Hardware do sistema	32
3.3 Ferramentas computacionais	33
3.4 Princípios de funcionamento do sistema	34
3.5 Montagem do painel elétrico	36
3.5.1 Componentes selecionados do painel elétrico	36
3.5.2 Fabricação do painel elétrico	38
3.6 Análise do atuador linear	42
3.6.1 Componentes selecionados para montagem	43
3.6.2 Configuração do servomotor	45

3.7 Parametrização do CLP no software GX	47
3.7.1 Ajustes iniciais	47
3.7.2 Definição de variáveis e comando no CLP	50
3.7.3 Adição de bloco de funções ao projeto	53
3.7.4 Programação	55
3.8 Aprendizado de máquina para inspeção	56
3.8.1 Configuração	56
3.8.2 Implementação da integração sistema-CLP	60
3.8.3 Programação do modelo e processo de inspeção	62
3.8.3.1 Criação do modelo de aprendizado de máquina	62
3.8.3.2 Processo de integração e inspeção	64
4 RESULTADOS E DISCUSSÕES	66
4.1 Teste e avaliação de desempenho	66
4.2 Análise de custo e comparação com demais sistemas e câmeras	70
4.3 Vantagens e desvantagens e desafios do projeto	71
4.3.1 Vantagens	71
4.3.2 Desvantagens	72
4.3.3 Desafios	72
5 APLICAÇÕES FUTURAS	73
6 CONCLUSÃO	74
7 REFERÊNCIAS	75
8 ANEXOS	80
ANEXO A – FLUXOGRAMA DE ATIVIDADES PARA REALIZAÇÃO DO PROJETO	80
ANEXO B – ESQUEMA ELÉTRICO DO PAINEL	81
ANEXO C – ESTRUTURAS LÓGICAS DAS LINHAS 1 A 05 EM LADDER	82
ANEXO D – ESTRUTURAS LÓGICAS DAS LINHAS 06 A 10 EM LADDER	82
ANEXO E – ESTRUTURAS LÓGICAS DAS LINHAS 11 A 15 EM LADDER	83
ANEXO F – ESTRUTURAS LÓGICAS DAS LINHAS 16 A 20 EM LADDER	83
ANEXO G – CÓDIGO-FONTE DO MODELO DE APRENDIZADO DE MÁQUINA	84
ANEXO H – CÓDIGO-FONTE DO SISTEMA DE INSPEÇÃO EM PYTHON	85

1 INTRODUÇÃO

A modernização do setor industrial foi impulsionada pelas grandes revoluções industriais que contribuíram para o surgimento de tecnologias emergentes e desenvolvimento de sistemas capazes de realizar o monitoramento e controle de processos produtivos em tempo real, melhorando a eficiência, reduzindo custos e tornando-os mais estáveis e inteligentes.

Segundo Brandão (2024), 52% das empresas brasileiras já adquiriram ferramentas necessárias para automação em processos e têm experiência em usá-las, destacando a utilização da Inteligência Artificial (IA) e Aprendizado de Máquina como parte dos fluxos de trabalho e gerenciamento de tarefas. Dessa forma, o uso crescente da Inteligência Artificial e automação de atividades vem desempenhando um papel significativo dentro das empresas, tornando-as robustas diante de um mercado competitivo.

A partir disso, a integração dos algoritmos de Aprendizado de Máquina e Controle Lógico Programável se tornaram primordiais para tomada de decisões estratégicas, detecção de falhas e antecipação de manutenções nos processos industriais, uma vez que contribuem para a otimização de resultados empresariais e criando oportunidades para a intensificação de melhorias com ênfase em sustentabilidade, redução de gastos operacionais e impacto ambiental.

Entretanto, apesar do avanço tecnológico da informática, a combinação dessas tecnologias enfrenta desafios significativos como a segurança cibernética, padronização, impactos éticos e econômicos que podem dificultar a implementação nas organizações de pequeno e médio porte.

1.1 ESTRUTURA DO TRABALHO

Este trabalho está estruturado da seguinte forma: o próximo Capítulo apresenta o contexto da justificativa, motivação, questão de pesquisa, objetivos e resultados esperados. O Capítulo 2 traz o referencial teórico com conceitos, definições em hardware e software utilizados no projeto. No Capítulo 3 é descrito o sistema proposto, mostrando como ele foi desenvolvido e o que foi feito para atingir o objetivo geral. No Capítulo 4 são apresentados os resultados e discussões e, por fim, no Capítulo 5 são apresentadas as sugestões de trabalhos futuros, considerações finais e anexos pertinentes ao trabalho.

1.2 JUSTIFICATIVA

A criação de sistemas inteligentes criados a partir de Aprendizado de Máquina tem ganhado popularidade ao longo dos últimos anos, sendo amplamente utilizada em segmentos que vão desde aplicações industriais até soluções medicinais. Alinhado com o avanço exponencial da indústria 4.0, considera-se que muitos dos sistemas e dispositivos presentes no setor industrial estão conectados à internet por meio de redes como wi-fi, Bluetooth, 5G e outras ferramentas que permitem a coleta de informações, gerenciamento de operações e integração de forma eficiente. Sob essa perspectiva, o CLP está inteiramente ligado a esse aspecto, já que possui a capacidade de coletar e registrar dados de diversas fontes como sensores, atuadores, dispositivos IoT e sistemas que possibilitam identificações de defeitos.

Contudo, a adoção desses recursos pode apresentar adversidades como custo de implementação e manutenção, que necessitam de infraestruturas compactas e alto investimento, o que pode ser um obstáculo para empresas com orçamento limitado. Por exemplo, isso é um impasse nas otimizações de atividades de identificação e prevenção de falhas em processos de montagem. Considerando esse cenário, o desenvolvimento de um sistema de inspeção a partir da integração de CLP e AM torna-se relevante para resolver essa adversidade.

Portanto, o sistema proposto contribuirá diretamente para a otimização desses processos e atendimento às metas empresariais, aspectos cruciais para a sustentabilidade, viabilidade econômica e vantagem competitiva diante do comércio industrial. Dessa forma, o projeto elaborado atende a uma necessidade real e urgente, especialmente em atividades industriais que necessitam de modernização, redução de falhas e custos operacionais.

1.3 MOTIVAÇÃO

Como admirador da inovação, aliado às minhas experiências em Engenharia de Manufatura, sempre busquei compreender a profundidade como funcionavam os processos industriais com foco na melhoria contínua e no meu desenvolvimento profissional.

Em 2023, trabalhando como assistente de processos na Venttos Eletronics, vivenciei de perto os desafios da organização em automatizar as inspeções finais na montagem de placas eletrônicas, pois elas eram feitas manualmente. Para atender às exigências dos clientes e contribuir para o cumprimento das metas, fui encarregado a buscar soluções disponíveis no mercado. Embora existam ferramentas avançadas de inspeção e câmeras com inteligência artificial embarcada, o preço elevado dificultou a implementação naquela ocasião.

Já em 2024, juntamente com o time de automação do Instituto Calcomp de Tecnologia, participei da entrega de uma estação robótica de inspeção final para máquinas de cartão, que integrava um algoritmo de inteligência artificial a um CLP, realizando inspeções cosméticas por meio de uma câmera industrial robusta. Após várias pesquisas, vi que é possível utilizar webcams convencionais com a mesma finalidade, dessa maneira, despertou em mim uma curiosidade ainda maior por estudar e desenvolver sistemas acessíveis com montagens sem complicações.

Diante desse cenário e com base nos conhecimentos adquiridos ao longo da minha formação no curso de Tecnologia em Mecatrônica Industrial, este trabalho propõe o desenvolvimento de uma solução tecnológica por meio da criação de um sistema inteligente de inspeção, integrando estruturas de AM a um CLP, utilizando um protocolo de comunicação industrial e componentes amplamente comercializados.

Utilizando itens como webcam, atuador linear, computador e materiais elétricos comuns no segmento industrial, decidi aplicar minha experiência técnica e acadêmica na criação de um recurso manuseável e eficiente, frente as soluções disponíveis no mercado em nível global.

Essa iniciativa busca contribuir para a modernização de processos em organizações que enfrentam dificuldades para se otimizar. Portanto, meu objetivo é ajudá-las a se modernizar, alcançar as metas estipuladas, manter boas relações comerciais com os clientes e tornar o sistema de inspeção mais viável e confiável para todos que enfrentam os mesmos desafios.

1.4 OBJETIVOS

1.4.1 Objetivo geral

Desenvolver um sistema de inspeção automatizada para detecção de ausência e deslocamento de baterias de lítio em placas eletrônicas a partir da integração do Controlador Lógico Programável e Aprendizado de Máquina, tornando prioritário a eficiência, baixo custo e otimização de processos.

1.4.2 Objetivos específicos

- Integrar Aprendizado de Máquina e CLP para tomada de decisões e acionamentos de equipamentos.
- Configurar e avaliar a comunicação sistema-CLP via Modbus/TCP.

- Determinar como serão utilizadas as lógicas de programação que atendam aos requisitos de integração do projeto.
- Selecionar os materiais e dispositivos que serão utilizados no projeto com base em funcionalidade, custos e disponibilidade de peças de reposição no mercado.
- Comprovar a eficácia do uso do sistema apresentando testes e dados.
- Identificar melhorias para aplicações futuras, mantendo a ideia inicial proposta.

2 REFERENCIAL TEÓRICO

Neste capítulo será apresentada uma revisão teórica com o objetivo de estudar e fundamentar a implementação de um sistema de inspeção de baterias de lítio utilizando como pilar a associação de CLP e Aprendizado de Máquina.

2.1 HARDWARES UTILIZADOS

Para a idealização e construção do sistema de inspeção, diversos dispositivos e tecnologias podem ser usados, porém o funcionamento do sistema depende de hardware adequado. A seguir, serão exibidos os conceitos dos dispositivos utilizados no projeto.

2.1.1 Controlador Lógico Programável

O Controlador Lógico Programável (CLP) ou Controle Lógico Programável pode ser descrito como um computador industrial com a capacidade de armazenar instruções para implementação de funções de controle (estrutura sequencial, temporização ou contagem), além de realizar operações lógicas aritméticas, manipulação de dados e comunicação via protocolos de redes, sendo utilizado no controle de processos e máquinas industriais. A projeção dos CLP's se consolidou devido à necessidade da indústria automobilística da empresa GM na década de 70, em suas linhas de produção. Em paralelo, Dick Morley, nomeado em diversas literaturas como o “pai do CLP”, criou esse dispositivo a fim de substituir os painéis elétricos contendo 22 relés de comando, abrindo espaço para uma revolução na área de controle e automação industrial, como mostra a Figura 1.

Com o resultado assertivo, o CLP rapidamente adentrou não só nas linhas de produção das indústrias automobilísticas, como também em indústrias têxteis e de base.

Figura 1- Criação do primeiro CLP por Dick Morley



Fonte: materialpublic.imd.ufrn.br, 2015.

Segundo Zancan (2016), o avanço dos controladores programáveis é fragmentado em cinco gerações:

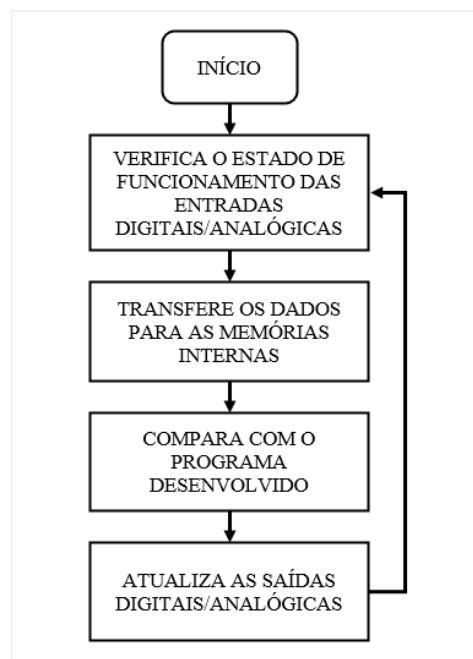
- Na 1ª geração, a programação era feita em assembly, ou seja, o programador deveria obter um conhecimento avançado do hardware do equipamento e seus componentes eletrônicos.
- Na 2ª geração, originaram-se as linguagens de nível médio, permitindo a conversão e adaptação das lógicas desenvolvidas pelo programador em linguagem de máquina.
- Na 3ª geração, o CLP obteve entradas de programação, permitindo sua conexão a teclados ou programadores portáteis.
- Na 4ª geração, os CLP's traziam entradas de comunicação serial, possibilitando sua programação diretamente de um PC. O software de programação instalado no computador permitia, além da programação e transferência do programa ao CLP, testar e modificar o funcionamento da aplicação.
- Já na 5ª geração, os CLP's são capazes de transmitir dados a partir dos protocolos de comunicação (Modbus, Profibus, Profinet), facilitando a interface com outros dispositivos e com sistemas supervisórios, como o SCADA.

Atualmente, o CLP possui programação em diferentes tipos de linguagens, são elas:

- Ladder (LL) – Linguagem gráfica semelhante aos diagramas de contato.
- Texto Estruturado (TE) – Semelhante às linguagens de alto nível.
- Diagrama de Blocos (DB) – Utilizado para lógicas combinatórias e controle contínuo.
- Lista de Instruções (LI) – Linguagem textual baseada em instruções simples.
- Diagrama Sequencial (DS) – Usado para representar etapas de processo.

De acordo com Sturaro (2009), o CLP funciona de forma sequencial, fazendo um ciclo de varredura de acordo com as etapas da Figura 2. Quando uma etapa é executada, as demais ficam inativas e o tempo total é nomeado CLOCK, necessitando, em casos específicos, de processadores com alto desempenho.

Figura 2 – Fluxograma de funcionamento do CLP



Fonte: O próprio autor, 2025.

Início: Verifica a operação da CPU, memórias, circuitos auxiliares, existência de um programa de usuário, valida as comunicações presentes entre dispositivos, emite aviso de erro em caso de falhas e desativa todas as saídas estabelecidas.

Verifica o estado das entradas: Averigua cada uma das entradas, certificando-se de que houve acionamento.

Compara com o programa desenvolvido: Através das instruções do usuário sobre qual ação tomar em caso de acionamento das entradas, o CLP habilita as memórias das saídas.

Atualiza as saídas: As saídas são acionadas ou desativadas conforme a determinação do CPU e um novo ciclo é iniciado.

Na indústria moderna, os CLP's é um equipamento essencial na instalação do *Supervisory Control and Data Acquisition (SCADA)*, uma vez que esse sistema tem a possibilidade de coletar dados em tempo real para armazenamento, utilização e análise orientada ao indivíduo, garantindo um funcionamento eficiente, flexível e seguro dentro das instalações industriais. Assim permitem que diversas ações sejam executadas, como o acionamento de válvulas, controle manual de uma atividade, detecção de problemas, diagnósticos e reconfiguração (CRAVO, 2024).

2.1.2 Atuador linear

Os atuadores planares ou lineares são dispositivos eletromagnéticos, denominados também conversor eletromecânico de energia, já que executam movimentos unidirecionais sobre um plano ao longo de uma trajetória estabelecida e possuem aplicações principalmente na área de automação e periféricos para microprocessadores.

Segundo Silva (2022), os atuadores lineares apresentam uma variedade de aplicações, sejam elas no setor primário ou secundário. Desse modo, são empregados em sistemas que necessitam inclinar, levantar, puxar ou empurrar produtos em uma trajetória unidirecional. Um dos mecanismos essenciais para a montagem do atuador linear é o motor de corrente contínua (CC) devido à sua facilidade de controle e robustez, operando em ambos os sentidos de rotação e constância à variação de carga em seu eixo.

Além disso, o servoamplificador ou driver torna-se importante na aplicação aos motores, já que gerencia o funcionamento do equipamento, fornecendo ao motor a potência necessária com a forma e intensidade corretas, utilizando a Modulação por Largura de Pulso (do inglês *Pulse Width Modulation – PWM*).

O autor relata que esta técnica é amplamente vista em microcontroladores comerciais e utilizada de forma habitual em sistemas de acionamentos elétricos.

Conforme Castrucci (2018), a implementação dos atuadores lineares em sistemas de movimentação, além da seleção correta do motor, a variável de posição em que se encontra o mecanismo também é importante para determinar o seu controle.

Tratando-se de sua estrutura, os atuadores utilizam-se de uma barra roscada conectada a um sistema de polias e correia ou um fuso de esfera, capaz de transformar o movimento rotacional do motor em um movimento linear adequado mostrado na Figura 3. A partir disso, as aplicações de movimentos lineares de maneira categórica podem ser implantadas em processos industriais que necessitam mover cargas sem intervenção humana, incorporadas em muitos casos, com robôs industriais.

Figura 3 - Estrutura física do atuador linear



Fonte: kalatec.com.br, 2025.

2.1.3 Webcam

Segundo K&F Concept (2024), a webcam é um dispositivo de entrada de vídeo que permite capturar imagens e transmiti-las em tempo real através da internet. Ela é geralmente conectada a um computador ou laptop através de uma porta USB e pode ser usada para videoconferências, streaming e gravação de vídeo em tempo real.

As principais características de uma webcam incluem resolução de imagem, taxa de quadros por segundo, foco automático, zoom digital, microfone embutido e compatibilidade com diferentes sistemas operacionais. É importante escolher uma webcam com boa qualidade de imagem e áudio para obter resultados satisfatórios, principalmente em aplicações que envolvem captura de imagens e vídeos.

A webcam pode ser de qualquer tipo, pois serve apenas para visualizar a imagem no software e a qualidade dela pode interferir no programa de visão computacional.

2.1.4 Outros hardwares

Assim como os equipamentos mostrados anteriormente são essenciais na elaboração do projeto, deve-se também levar em consideração a contextualização do computador e do painel elétrico para a efetiva integração do sistema com CLP.

De modo geral, um computador ou *Personal Computer* (PC) é uma máquina capaz de coletar, gerenciar e fornecer os resultados da utilização de dados para um ou mais objetivos.

Segundo Amaral (2010), a aplicação de um esquema básico de um processamento de dados, seja ele automático ou manual, resulta em uma informação. Logo, os dados precisam ser processados para que algum resultado tenha significado para alguém ou para o próprio computador.

No sistema desenvolvido, o PC possui um papel importante para processamento de dados e tomada de decisão, pois ele é responsável por executar os algoritmos de visão computacional e aprendizado de máquina, processando as imagens capturadas pela webcam e classificando-as como aprovadas ou reprovadas. Ademais, o PC estabelece a comunicação com o CLP via protocolo Modbus/TCP, enviando os resultados da inspeção e recebendo sinais de controle para sincronizar o processo.

Para receber e enviar os sinais de controle do CLP, utiliza-se um painel elétrico industrial, também denominado quadro elétrico ou quadro de comando, que constitui um conjunto de dispositivos fundamentais para o gerenciamento e distribuição de energia elétrica em ambientes industriais. No âmbito da automação, este painel integra componentes essenciais, como disjuntores, fonte de alimentação de 24 VCC, bornes de ligação e *switch* industrial, que asseguram a proteção, a alimentação e a interconexão entre os diferentes elementos do sistema.

Conforme Santos (2023), ao distribuir a energia de forma equilibrada, prevenindo sobrecargas e curto-circuito, o painel elétrico industrial contribui significativamente para a continuidade operacional, a eficiência energética e a segurança dos colaboradores da empresa. Portanto, a infraestrutura elétrica efetua um papel crucial na otimização dos processos produtivos, assegurando a alimentação elétrica segura e confiável de todos os equipamentos e máquinas.

2.2 SOFTWARES UTILIZADOS

Tendo em vista a efetivação da integração do sistema de inspeção com o CLP, foi utilizado software disponibilizado de forma gratuita e interativa. Dessa forma, a seguir, será apresentada a definição dos principais softwares e ferramentas utilizados no projeto.

- **Visual Studio Code:** Ambiente virtual gratuito (Figura 4a) e de código aberto para desenvolvimento de aplicações em diversas linguagens de programação como Python, JavaScript/TypeScript, C, Java e entre outras. O editor permite importar bibliotecas, sendo que, em soluções que envolvem a interatividade entre dispositivos de controle e visão computacional, as mais utilizadas são:
 1. **OpenCV:** Permite que programas de computador capturem e interpretem imagens e vídeos.
 2. **Pymodbus:** Concede a comunicação Modbus (um protocolo de comunicação industrial muito usado com CLP's, sensores e outros dispositivos).
 3. **NumPy:** Permite trabalhar com vetores, matrizes e operações matemáticas de forma rápida e eficiente.
 4. **Time:** Recurso padrão para trabalhar com tempo e pausas no programa.
 5. **Joblib:** Utilizado para treinar e salvar modelos de Aprendizado de Máquina, garantindo a eficiência do classificador.
- **GX Works3:** Software de programação e manutenção ofertado pela fabricante Mitsubishi Electric, destinado principalmente para os sistemas de controle MELSEC MX Controller, MELSEC iQ-R e MELSEC iQ-F Series (Figura 4b).
- **MR Configurator2:** Software de configuração da Mitsubishi Electric para regulagem, monitoramento, diagnóstico, leitura ou escrita de parâmetros do servo de acionamento. Ele possui monitoramento em tempo real do funcionamento do servomotor e usado para configurar os modelos de servo acionamento (servoamplificador) como MR-JE e MR-J4 (Figura 4c).
- **Paint:** Ferramenta padrão do sistema operacional Windows, a qual permite recorte, modelação e sobreposição de imagens (Figura 4d).

Figura 4 - Softwares e ferramentas projetados para utilização



Fonte: O próprio autor, 2025.

2.3 INTELIGÊNCIA ARTIFICIAL E APRENDIZADO DE MÁQUINA

A Inteligência Artificial (IA) pode ser definida como ferramenta com a habilidade de aparentar comportamentos análogos aos de um ser humano, capaz de realizar tarefas cognitivas como aprendizagem, raciocínio, ação e reconhecimento de fala. Relacionada, em muitos casos, com Aprendizado de Máquina (do inglês Machine Learning) e Big Data, a IA é usada para análise de dados, previsões, categorização de objetos, processamento de linguagem natural, indicações e recuperação inteligente de dados (LIMUX, s.d.).

Conforme Krischanski et al. (2020), a inteligência artificial possui ramificações como representação do conhecimento, movimentação robótica, percepção e gerenciamento de objetos. Dessa maneira, os sistemas de IA aprendem e melhoram por meio da interação com grandes quantidades de dados, reconhecendo padrões e relações estabelecidos que muitas vezes são imperceptíveis à visão humana.

Segundo Lobo (2017), a Inteligência Artificial tem diversas etapas ou competências como reconhecimento dos padrões e imagens, entendimento da linguagem aberta, observação e reconhecimento de relações, nexos e compreensão de algoritmos de decisão apresentados por especialistas da área.

Com o desenvolvimento exponencial de aplicações de Inteligência Artificial, pode-se nomear suas seguintes subáreas: *Machine Learning*, Processamento de Linguagem Natural (PLN), Visão Computacional, Sistemas Especialistas, Aprendizado Federado e IA Distribuída, AI Fuzzy (Lógica Nebulosa), Planejamento e Otimização, Agentes Inteligentes, Autoconsciência (*Self-Aware*), Análise de dados e Aprendizado Profundo (CAROLA, 2024).

O Aprendizado de Máquina pode ser estabelecido como uma subárea da Inteligência Artificial que permite que os artifícios aprendam e melhorem automaticamente a partir de experiências (dados), de fato, usando algoritmos, as matrizes são treinadas em grandes volumes

de dados para realizar previsões ou classificar informações com precisão, sendo amplamente aplicado em recomendações, detecção de fraudes e diagnósticos médicos (MITCHELL, 1997).

De acordo com Garcia (2023), o emprego de técnicas de IA possibilita que a máquina possa ainda planejar sequências de atividades para alcançar metas, ou seja, dado um grande conjunto de dados para treinamento, um algoritmo de Aprendizado de Máquina gera um modelo capaz de identificar entradas e saídas.

Segundo Cavallari (2022), com base nos algoritmos de aprendizado, pode-se categorizar os tipos de detecção em três divisões: aprendizado supervisionado, não-supervisionado e semi-supervisionado.

O aprendizado supervisionado é quando o modelo aprende a partir de dados de entrada e apresenta uma resposta desejada. Têm-se como exemplos a criação de sistemas de visão para classificar produtos não conformes e conformes de acordo com o padrão estabelecido.

O aprendizado não supervisionado é relacionado quando o modelo apenas recebe dados de entrada e não possui uma resposta padrão, ou seja, o sistema tenta identificar padrões de respostas de forma independente. A ideia é que o algoritmo consiga ponderar os dados somente baseado em características intrínsecas do *Dataset* (ANDRADE, 2024).

Já o aprendizado semi-supervisionado atua de forma intermediária entre supervisionado e não supervisionado, uma vez que busca identificar respostas padrão e de forma independente. A ideia de utilizar essa técnica é para melhorar o desempenho de modelos com baixa precisão, obtendo mais informações para classificação e análise de objetos.

Portanto, Segundo Da Silva et al. (2023), a IA e suas subáreas podem ser caracterizadas como tecnologia emergente assim como Internet das Coisas (IoT) e realidade aumentada, as quais trouxeram transformações positivas, permitindo avanços em áreas como assistência médica, automação de processos e até mesmo na tomada de decisões, podendo aumentar a eficiência e produtividade nas empresas.

2.4 PROTOCOLOS DE REDE

Com o desenvolvimento significativo da informática alinhado com o surgimento da microeletrônica, diferentes dispositivos microprocessados para automação no setor industrial foram criados, e partir disso, surgiu-se a necessidade de se comunicar aos CLP's, atuadores e computadores de maneira sinérgica.

Desse modo, a interligação desses equipamentos possibilitou que fossem compartilhados recursos de dados, os quais passaram a ser únicos, não necessitando serem replicados, trazendo mais segurança aos usuários da informação (ROSARIO, 2005).

Logo, os protocolos de comunicação industrial tratam-se de um conjunto de regras e padrões técnicos, caracterizado em muitas literaturas como “linguagem comum”, que define como dispositivos e sistemas de diferentes fabricantes devem trocar informações de forma eficiente e padronizada, desconsiderando especificações individuais.

Eles estabelecem parâmetros essenciais, como formatos de dados, tempos de resposta e métodos de controle de erros, assegurando uma comunicação integrada e confiável, que é desenvolvida através do padrão ISO/OSI, sendo os mais utilizados o Ethernet, CANopen, Foundation Fieldbus, Modbus, Profibus e entre outros.

Os principais tipos de redes e protocolos de comunicação são:

- **Protocolo CAN:** É um protocolo de comunicação serial que descreve os serviços da camada de enlace de dados, de acordo com o padrão ISO/OSI. Nela, são definidos os diferentes tipos de frames, validação, monitoramento de mensagens e a forma de detecção de erros.
- **CANopen:** É uma rede baseada em CAN, ou seja, utiliza telegramas CAN para troca de dados na rede. Foi desenvolvida como uma rede embarcada padronizada com capacidade de configuração flexível.
- **Profibus:** É um protocolo utilizado para designar um sistema de comunicação digital que pode ser implementado em diversas áreas de aplicação. Trata-se de um sistema aberto e padronizado pelas normas IEC 61158 e IEC 61784, que engloba desde o meio físico até perfis de dados para determinados conjuntos de equipamentos.
- **DeviceNet:** É um protocolo baseado em CAN, que possui as mesmas características de frame de dados, acesso à rede, controle de erros do protocolo CANopen.
- **Modbus:** É um protocolo no qual os bits de dados são transmitidos sequencialmente através de um barramento ou canal de comunicação. Várias tecnologias utilizam comunicação serial, entre elas, as interfaces RS-232 e RS-485 (NOGUEIRA, 2009).

Segundo Cavanlli (2016), o protocolo Modbus possui ramificações como:

- **Modbus TCP/IP:** Utilizado para comunicação entre sistemas de supervisão e CLP's. Esse protocolo é embutido no protocolo TCP/IP, e transmitido através de redes padrão Ethernet, com controle de acesso por *Carrier Sense Multiple Access/ Colision Detection* (CSMA/CD).
- **Modbus Plus:** Utilizado para comunicação entre CLP's, módulos de I/O, inversores e Interface Homem-Máquina (IHM), a qual usa interface de comunicação RS-485, taxa de transmissão em Mbps e controle de acesso por *High Level Data Link Control* (HDLC).
- **Modbus Padrão:** É usado para comunicar CLP's, com dispositivos de I/O de dados, Instrumentos Eletrônicos Inteligentes (IED's), onde usa interface RS-232 ou RS-485, em conjunto com o protocolo mestre-escravo.

2.5 VANTAGENS E DESAFIOS DA INTEGRAÇÃO

Com o incremento dos requisitos de qualidade, meio ambiente, segurança e aumento do importe de matéria-prima, grande parte das organizações globais investe em soluções tecnológicas da indústria 4.0 como controladores, inteligência artificial e aprendizado de máquina, de modo a atingir métricas organizacionais pré-estabelecidas como redução de custos e aumento de eficiência produtiva.

Segundo Mochuan (2024), a integração entre as tecnologias da indústria 4.0 podem gerar vantagens como:

- **Detecção de defeitos:** Ao implementar recursos de IA, os CLP's podem identificar informações em tempo real, auxiliando na detecção de objetos e falhas nas linhas de produção.
- **Adaptabilidade a mudanças:** Os algoritmos de Aprendizado de Máquina têm a capacidade de otimizar estratégias de controle a qual inclui correções de parâmetros para maximizar a eficiência e minimizar o desperdício, desse modo, os sistemas podem se adaptar as mudanças nas condições do processo de forma mais eficiente do que sistemas tradicionais, aumentando a flexibilidade e a capacidade de resposta.
- **Segurança:** Os algoritmos de IA em aplicação com CLP em sistemas de visão, podem ser capazes de detectar situações perigosas como acesso restrito a salas

seguras, mau funcionamento de maquinários, evitando acidentes e danos operacionais e materiais.

Apesar das vantagens, a integração da IA e suas subáreas (como Aprendizado de Máquina) com CLP's, enfrentam desafios significativos que podem limitar sua aplicação em ambientes industriais.

Um dos grandes desafios que vêm afetando não só ambas as tecnologias, como também as demais inovações alinhadas à indústria 4.0, é a crescente sofisticação das ameaças cibernéticas e os riscos potenciais relacionados ao acesso e controle remoto dos dispositivos conectados às redes de computadores.

Outro desafio presente nas organizações é a necessidade de padronização, ou seja, em razão da tendência de investimento em automação industrial das corporações em nível global, o CLP possui diversos tipos de soluções proprietárias, o que acaba gerando uma complexidade para a integração e a interoperabilidade entre as tecnologias emergentes.

Outra adversidade enfrentada pelas organizações é o custo de implementação, ou seja, segundo Soori et al. (2020), a dependência de infraestrutura robusta e tecnologias avançadas pode ser um obstáculo para empresas com orçamento limitado. Dessa forma, as tecnologias requerem hardware especializado, como GPUs de alto desempenho, assim como CLP's de última geração e sistemas SCADA que representam um investimento operacional considerável.

De acordo com Lugli e Santos (2015), a necessidade de calibração constante dos sensores e a atualização contínua de softwares tornam os sistemas mais onerosos ao longo de seu ciclo de vida.

2.6 INSPEÇÃO NA INDÚSTRIA

2.6.1 Tipos de inspeções industriais

O método de inspeção trata-se de observar e determinar se o produto está conforme as especificações funcionais e cosméticas em comparação ao projeto inicial a fim de atender às necessidades do cliente. Esse procedimento pode ser dividido em: inspeção visual manual, inspeção óptica automática e raio-x.

Além do mais, os fatores externos como mão de obra, capacidade e setup das linhas de produção, tornam-se tópicos importantes para obter uma inspeção assertiva. Logo, a otimização desse processo é essencial, pois através das ferramentas avançadas de inspeção, é capaz de

garantir a confiabilidade e padronização do processo produtivo, facilita o rastreamento de falhas e implementação de ações corretivas.

Tratando-se de Placas de Circuito Impresso (PCI), as inspeções podem ser fundamentadas em:

- **Referência:** É utilizada para comparação de uma placa-teste a uma placa padrão de forma que a diferença entre ambas possa ser classificada como não conforme.
- **Projeto:** É usado quando há necessidade de comparação com uma placa piloto, em outras palavras, é o modelo que foi projetado, incluindo as especificações técnicas iniciais.
- **Híbrida:** É utilizada tanto uma imagem de referência como o projeto da placa.

Conforme Lameira (2020), as tecnologias embasadas em imagem são projetadas a fim de realizar a verificação do componente por meio de comparação.

Desse modo, as imagens capturadas do produto são analisadas a partir do banco de dados de imagens adquiridas anteriormente, onde é possível julgar imagens de acerto e erro dos componentes.

Segundo Peallat (2018), as diversas técnicas de comparação variam desde algoritmos básicos até redes neurais, porém o fundamento de inspeção não é alterado, em que é possível comparar a imagem atual com as imagens armazenadas no banco de dados.

O autor também relata que através de padrões matemáticos e geometria, a inspeção baseada em algoritmo, utiliza técnicas de reconhecimento de padrões para encontrar os componentes nas imagens, ou seja, ao invés de comparar uma imagem com outra, a tecnologia usa um padrão definido (geometria ou esqueleto do objeto) para inspecionar os componentes, garantindo a confiabilidade do sistema empregados e qualidade.

2.6.2 Tecnologia de inspeção na indústria

Para otimizar o processo de inspeção de placas de circuito impresso, a criação de um sistema é uma solução eficaz. O processo de inspeção pode ser realizado por diferentes métodos, sendo os mais práticos o uso de câmeras compactas integradas a uma visão computacional elaborada a partir de código-fonte para processamento de imagens, e um controlador lógico para monitoramento dos equipamentos industriais e tomada de decisão advindo de uma programação.

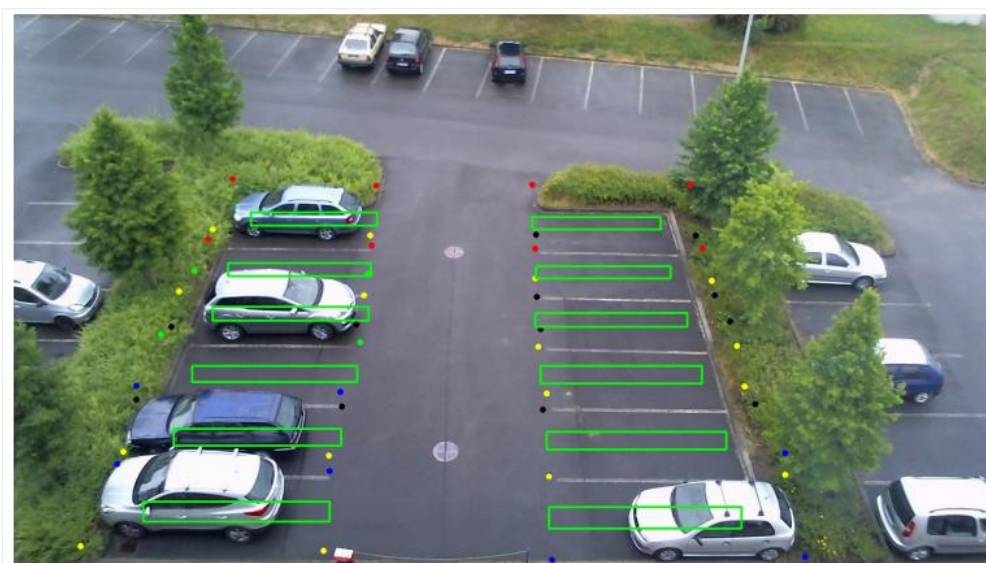
Com integração entre essas tecnologias é capaz de detectar falhas visuais, anomalias ou desvios de padrão com base em um banco de dados de imagens adquiridas, logo, os resultados

programados estabelecem uma orientação para o CLP atuar de forma imediata como por exemplo, em caso de excesso de rejeitos, o CLP envia uma informação a uma saída, permitindo que ele interrompa o processo imediatamente e só retorne após uma determinada ação (manutenção).

As vantagens de sistemas como estes, quando comparados à inspeção convencional, evidenciam-se ainda mais quando a linha de produção deve ser monitorada em alta velocidade e com precisão (STIVANELLO,2004).

O trabalho de Saraiva (2015) demonstra na Figura 5 uma aplicação prática desses sistemas, pois propôs um sistema de visão computacional de baixo custo baseado no Raspberry Pi 2 e numa webcam com resolução HD para a detecção de lugares livres em parques de estacionamento. O algoritmo de visão foi baseado num detetor de arestas e num limite dinâmico atribuído a cada lugar para inferir sobre a condição do lugar inspecionado.

Figura 5 - Sistema de visão computacional em parques de estacionamento



Fonte: sapientia. ualg.pt, 2015.

2.6.3 Soluções disponíveis no mercado

Analisando o mercado atual, vi que há vários sistemas inteligentes e câmeras industriais embarcadas, que possuem tecnologia própria para inspeção de produtos e comunicação estável entre dispositivos via rede, todavia um limitante é o alto custo, logística, manutenção e peças de reposição no mercado nacional.

- **Sistema de Visão Série CV-X:** É um sistema inovador da Keyence (Figura 6a) com configuração e programação robusta, com combinação de câmeras industriais, iluminação de alta intensidade e processamento avançado de ferramentas. É utilizada em inspeções automatizadas em todos os setores comerciais e industriais. Seu valor custa em média R\$ 98.000,00.
- **Sistema de Visão Série FH:** É um sistema da Omron (Figura 6b) com IA embutida, com combinação de câmeras compactas com alta velocidade e resolução, utilizado em âmbitos industriais no que se refere à detecção e rotulação de peças. Seu valor custa em média R\$ 40.000,00.
- **Sistema de Visão Computacional Weg:** É uma solução de inspeção inteligente da fabricante Weg (figura 6c), possui câmeras industriais, unidade de processamento (CVU), software inteligente dedicado e IA embutida. Seu valor custa em média R\$ 20.000,00.
- **Câmera Intel D495I:** É uma câmera inteligente da marca Intel (Figura 6d), possui longo alcance, resolução de alta profundidade, sensores de imagens dedicados e algoritmo avançado. Ele é recomendado para aplicações como navegação robótica e reconhecimento de objetos. Seu valor custa em média R\$ 1.700,00.
- **Câmera Inteligente Série VE:** É uma câmera robusta da fabricante Banner (Figura 6e), possui software gratuito fornecido para o usuário. Ele é aplicado em soluções de detecção, posicionamento de parte e análise de falhas. Seu valor no mercado é, em média, R\$ 32.000,00.

Figura 6 - Exemplos de sistemas e câmeras disponíveis no mercado



Fonte: O próprio autor, 2025.

Uma das principais vantagens desses dispositivos inclui: inspeção de mais de uma peça em um determinado ciclo, alto desempenho de processamento, algoritmos de IA embutida, geração de relatório de qualidade e suporte a ambientes industriais com ruídos, variação de iluminação e poeira. Ademais, consideram-se como desvantagens a presença de sistemas fechados, dependência de suporte técnico e infraestrutura robusta.

Diante destas condições, entendo que essas tecnologias se tornam inviáveis para empresas com orçamento restrito. Então acredito ser mais viável produzir um equipamento que seja acessível, sofisticado e que atenda aos requisitos das organizações.

3 O SISTEMA

Neste capítulo será discorrido acerca da preparação do sistema proposto, destacando principais aspectos de montagem, funcionamento, parametrização e integração dos mecanismos utilizados.

A concepção de um sistema de inspeção otimizado e de baixo custo deve levar em consideração não apenas os aspectos técnicos relacionados aos componentes, mas também a facilidade de uso, a segurança e a durabilidade do equipamento. Além disso, é importante garantir que o projeto seja escalável e modular, permitindo que ele seja adaptado para diferentes tipos de CLP's, webcams e critérios específicos do usuário.

3.1 PROJETANDO O SISTEMA

Para a execução do projeto e a construção de um sistema eficaz e de baixo custo, foram adotadas atividades de acordo com o ciclo PDCA, o qual trata-se de um método de gerenciamento com foco na melhoria contínua que tem como objetivo controlar e melhorar os processos e produtos. Dessa forma, consideram-se as seguintes etapas: Planejar (*Plan*), Fazer (*Do*), Checar (*Check*) e Agir (*Act*).

- **Planejar:** Definição e conceito do projeto, análise de dispositivos disponíveis no mercado, avaliação de vantagens e desvantagens, capacitação profissional, definição de layout do painel elétrico para o CLP e determinação da linguagem de programação para desenvolvimento do sistema.
- **Fazer:** Montagem física do painel elétrico do CLP, montagem do atuador linear, desenvolvimento da programação CLP em Ladder e programação em Python,

parametrização do servo, criação do *Dataset* de imagens, integração do sistema desenvolvido com CLP e atuador linear via Modbus IP/TCP.

- **Checar:** Validação integral do sistema e verificação de bugs e falhas de comunicação entre os mecanismos.
- **Agir:** Treinamento de imagens para expansão do *Dataset*, instalação de um LED para simulação de uma torre luminosa em casos de excessos de falhas limite definidos na programação CLP, correção de bugs e falhas de comunicação e padronização do processo.

Conforme ilustração disponível no anexo A deste trabalho, foi criado um gráfico de Gantt tomando como base o ciclo PDCA mencionado anteriormente. As fases iniciais, realizadas entre abril e início de maio, foram direcionadas ao levantamento de requisitos, estudo de viabilidade e criação de relatórios. Essa fase garantiu a definição clara dos objetivos, escopo e recursos necessários para o desenvolvimento do sistema.

Na sequência, ainda no mês de maio, foram executadas as atividades de modelagem do sistema e início da implementação, incluindo a programação dos parâmetros principais. Parte dessas tarefas ocorreu de maneira simultânea, sem comprometer a qualidade.

No mês de junho, o foco foi direcionado para a finalização da implementação da integração dos componentes e realização de testes preliminares. Esta etapa foi importante para identificar falhas e promover ajustes antes da entrega final.

Em julho, concentraram-se as atividades de testes finais, validação do sistema, ajustes de desempenho e preparação para a apresentação. Dessa forma, o cronograma encerra com a integração final do projeto, contemplando tanto a parte técnica quanto a documentação para defesa do trabalho.

3.2 HARDWARE DO SISTEMA

A Figura 7 mostra os equipamentos implementados no desenvolvimento do sistema, levando em consideração que eles foram disponibilizados pelo Instituto Calcomp para fins acadêmicos.

- **Painel elétrico:** Montado para distribuição de energia elétrica dos componentes, proteção dos circuitos e comando dos equipamentos do projeto.
- **Atuador linear OBR:** Operado para movimentação do produto a partir do ponto inicial (*home*) até o ponto determinado para início de inspeção, considerando

variáveis de velocidade, aceleração e direção previamente configuradas. O conjunto (servomotor e servoamplificador) foi anexado na montagem do mecanismo.

- **Notebook Book Samsung:** Usado para programação do sistema e CLP, armazenamento de dados de imagens OK e NG, ponte de comunicação via Modbus entre as tecnologias mencionadas, treinamento, ajuste de parâmetros e atualizações do modelo. Diferente de alguns processos que utilizam IA avançada e computadores de alto desempenho (com 16 a 32 GB de RAM e placa de vídeo dedicada), neste projeto não há essa limitação. Pode-se aplicar um computador comum, encontrado na faixa de R\$ 1.500 a R\$ 2.500, o que reduz o investimento.
- **Webcam Brio 500:** Utilizado para captura de imagens de análise, entrada do modelo de AM para classificar se a peça é aprovada ou com defeito (reprovada). É um modelo simples e acessível, custando entre R\$ 450 e R\$ 700, suficiente para a resolução necessária e dispensando câmeras industriais de alto valor.

Figura 7 - Equipamentos utilizados na construção do projeto.



Fonte: O próprio autor, 2025.

3.3 FERRAMENTAS COMPUTACIONAIS

No desenvolvimento do sistema, foram utilizadas ferramentas computacionais conceituadas na seção 2.3 para atender aos aspectos de programação, configuração e modelagem do projeto. A seguir, são descritas as finalidades de cada uma delas.

O MR Configurator2 foi empregado para parametrizar o servoamplificador, permitindo a otimização e monitoramento em tempo real do servo.

O GX Works foi utilizado para ambientação de programação do CLP, permitindo a criação, simulação e monitoramento da sequência lógica, garantindo o funcionamento e integração do controlador com os demais dispositivos em campo.

O Paint foi útil para designação e dimensionamento da Região de Interesse (do inglês *Region of Interest* - ROI), ou seja, área específica dentro de uma imagem para comparação do frame capturado com o modelo padrão.

Por fim, o Visual Code foi usado para o desenvolvimento e organização dos algoritmos de processamento de imagens e integração do Aprendizado de Máquina com o CLP, proporcionando a execução de testes e treinamentos do modelo.

3.4 PRINCÍPIOS DE FUNCIONAMENTO DO SISTEMA

Após análise das soluções usualmente utilizadas, percebe-se que os sistemas de inspeções industriais são compostos por um sistema embarcado responsável por coletar e transmitir o resultado do inspecionamento a partir de dados já coletados anteriormente.

Mas para isso, é necessário ter um sistema mecânico e elétrico para movimentação do produto até a área determinada de inspeção, acionamento e controle dos equipamentos industriais para possíveis correções e interrupções dentro do processo produtivo.

Conforme a Figura 8, foi criado um fluxograma para demonstrar o funcionamento do sistema de modo geral.

O processo inicia-se com a etapa “início”, onde o usuário (operador) liga o painel elétrico, prepara os softwares e os comandos do CLP para a execução de inspeção.

Em seguida, o operador insere a placa no berço do atuador, certificando que esteja posicionada corretamente para o início do ciclo.

Após o posicionamento, o operador deverá apertar o botão de inicialização, que aciona o ciclo programado no CLP. Nesta etapa, a memória responsável pela movimentação da base do atuador é ativada, direcionando a placa para a área de inspeção.

A primeira verificação é a detecção do metal acoplado à base do atuador, caso o sensor detecte a presença do metal, ele envia um sinal para o CLP, fazendo com que interrompa a movimentação e inicie o processo de inspeção. Caso contrário, o sistema ficará no aguardo do sinal do CLP, ou seja, aguardando o produto a ser inspecionado.

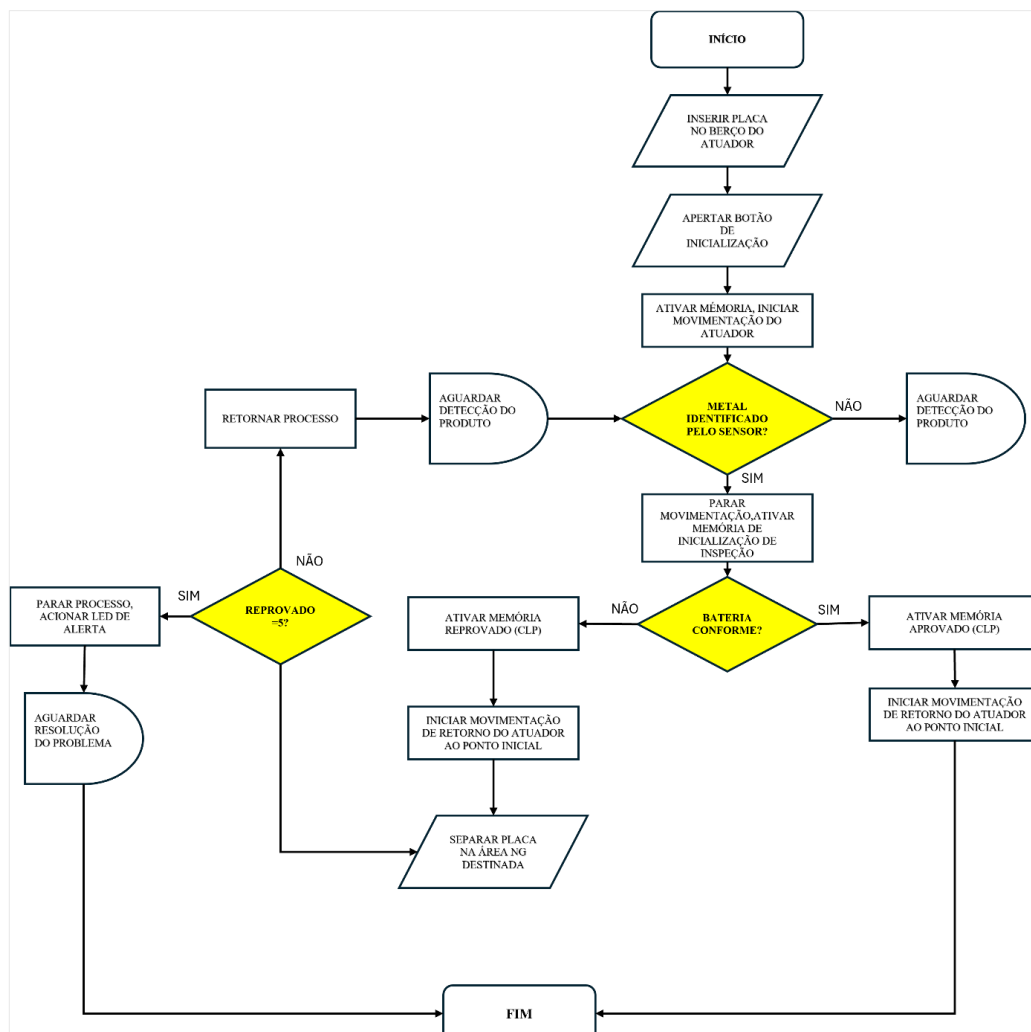
A segunda verificação consiste na análise das características da bateria, validando e comparando as questões de posicionamento e presença dela de acordo com o modelo treinado.

Se aprovado, o sistema enviará um comando via Modbus para o CLP, ativando a memória de movimentação de retorno do atuador. Nesse estágio, o operador deverá retirar a placa do berço do atuador e direcioná-la para o próximo processo ou área OK estabelecida.

Se reprovado, o sistema também enviará um comando via Modbus para o CLP, ativando a memória de movimentação de retorno do atuador. Nessa etapa, o operador deverá retirar a placa do berço do atuador e direcioná-la à área de NG estabelecida.

Na programação CLP foi incluído um contador de rejeitos para monitorar a quantidade de peças reprovadas durante o processo. Quando o número de rejeições atingir o limite de cinco peças, o sistema aciona o LED de alerta a fim de avisar que há falhas de montagem da bateria ou instabilidade do sistema. Nesse caso, o operador deverá averiguar o presente problema, realizar as correções e, somente após a solução, reiniciar a contagem de rejeitos e retomar a operação do sistema.

Figura 8 – Fluxograma sequencial de funcionamento do sistema



Fonte: O próprio autor, 2025.

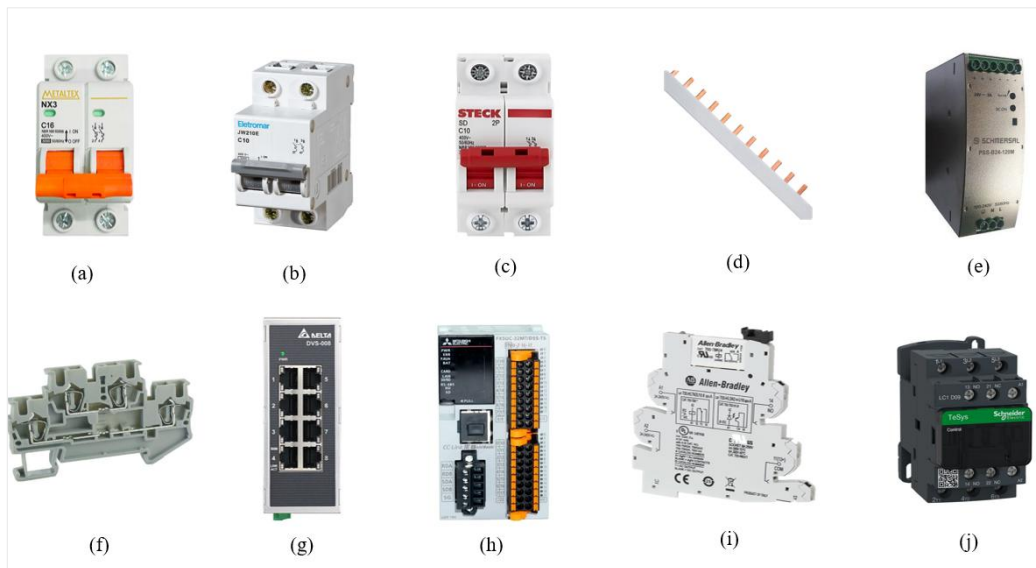
3.5 MONTAGEM DO PAINEL ELÉTRICO

3.5.1 Componentes selecionados do painel elétrico

Como mostra a Figura 9, foram selecionados os seguintes componentes do painel elétrico:

- **(a) Disjuntor bipolar 16A:** Modelo estabelecido foi NX3 da marca Metaltex, é responsável pela proteção e alimentação geral do painel elétrico.
- **(b) Disjuntor bipolar 10A:** Modelo estabelecido foi JW2 da marca Eletromar, é responsável pela proteção e alimentação geral do servo amplificador.
- **(c) Disjuntor bipolar 10A:** Modelo estabelecido foi SD da marca Steck, é responsável pela proteção e alimentação da fonte de alimentação 24 VCC.
- **(d) Pente bipolar 80A:** O exemplar escolhido foi XH3 da marca Legrand. Foi utilizada para “jumper” a distribuição de energia do disjuntor geral para os demais disjuntores do circuito.
- **(e) Fonte de alimentação:** O modelo escolhido foi PSS-B24 da marca Schmersal. Foi utilizado para conversão de energia 110V/220V em 24 VCC para alimentação do CLP, sensores e dispositivos de comando.
- **(f) Borne duplo:** O modelo designado foi PTT2,5 da fabricante Phoenix, utilizado para interligação e organização de cabeamentos elétricos.
- **(g) Switch industrial:** O modelo escolhido foi DVS-008 da fabricante Delta. Foi utilizado para transmissão de dados entre os dispositivos conectados via Ethernet/IP.
- **(e) CLP:** O modelo escolhido foi FX5U da marca Mitsubishi, foi implementado devido à compatibilidade com o protocolo Modbus e à integração interativa com os demais equipamentos da própria fabricante. Esse CLP é do tipo NPN e possui 16 entradas e saídas digitais, o que facilita a ligação de vários dispositivos em uma mesma expansão.
- **(f) Relé interface:** O modelo escolhido foi 700-TBR24 da marca Allen Bradley, foi utilizado para proteção e conversão de sinais para conexão dos dispositivos de campo nas entradas e saídas do CLP.
- **(g) Contactora tripolar:** O modelo adotado foi o da fabricante Schneider. Foi usado para acionamento do servoamplificador sob comando do CLP em 24 VCC.

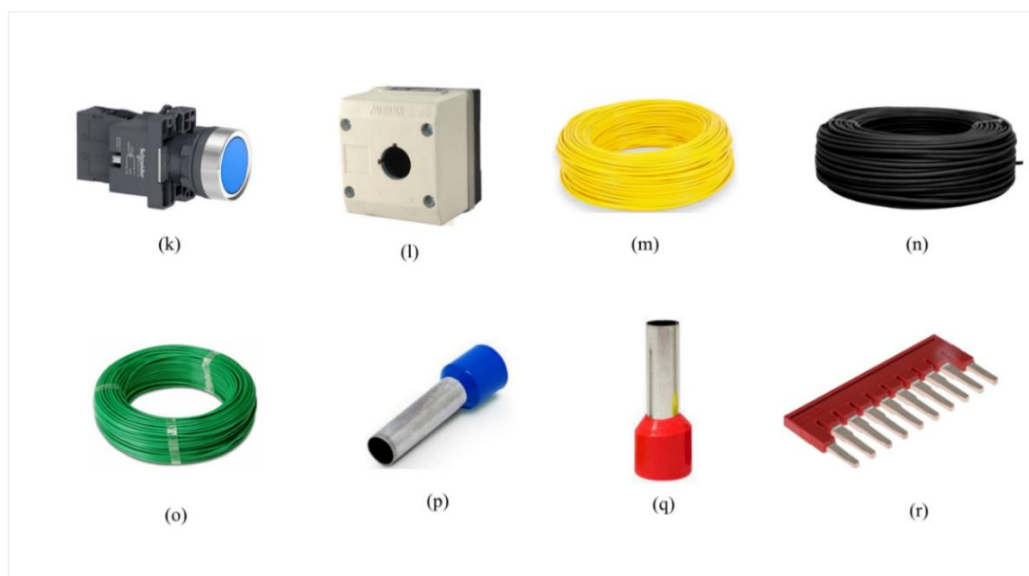
Figura 9 – Componentes do painel elétrico



Fonte: O próprio autor, 2025.

A Figura 10 também expõe os demais componentes utilizados na montagem do painel.

- **(k) Botão Push 24 VCC:** O modelo escolhido foi XB5 da marca Schneider. Foi usado para acionamento manual de movimento da base do atuador.
- **(l) Caixa para botão:** Utilizada para proteção do botão de acionamento.
- **(m) Cabo 1,5 mm:** Utilizado para interligação, passagem de energia e distribuição de sinais dos componentes elétricos como botão Push, led e relés interface.
- **(n) Cabo 1,0 mm:** Foi utilizado para alimentação 24 VCC para os dispositivos eletrônicos como Switch e CLP.
- **(o) Cabo 2,5 mm:** Utilizado para interligação e passagem de tensão do disjuntor 10A para o servo amplificador.
- **(p) Terminal ilhós 2,5 mm:** Utilizado para os cabos de 2,5 mm.
- **(q) Terminal ilhós 1,5 mm:** Utilizado para os cabos de 1,5 mm.
- **(r) Ponte de interligação:** Usada para “jumper” as tensões positivas (P24) e negativas (N24) nos bornes para ligação dos sensores fotoelétricos, botão Push, led e demais dispositivos eletrônicos.

Figura 10 - Demais componentes

Fonte: O próprio autor, 2025.

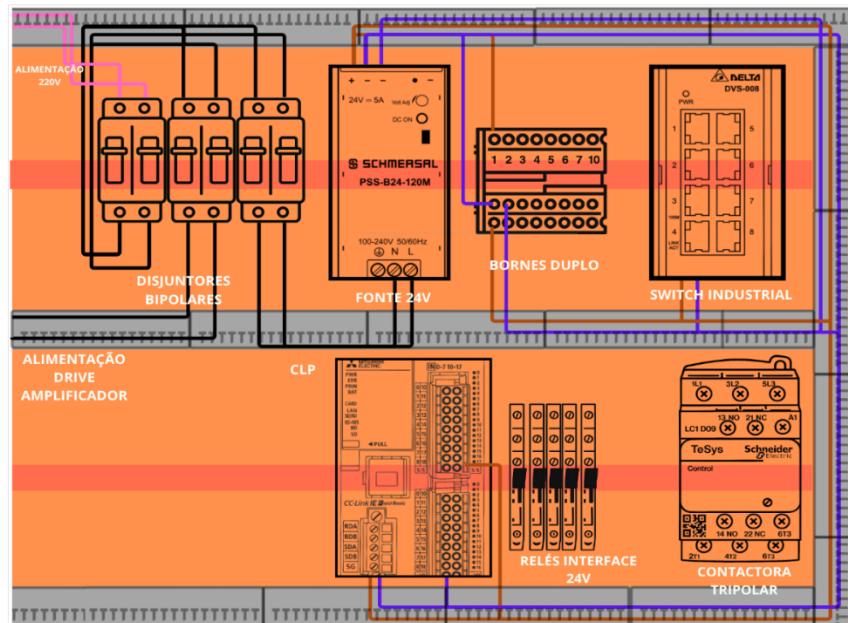
3.5.2 Fabricação do painel elétrico

Conforme mostra a Figura 11, foi criado um layout gráfico para servir de referência na montagem física do painel elétrico do sistema. Esse desenho não apenas apresenta a posição dos dispositivos, mas organiza visualmente sua modelação e conexões elétricas, contribuindo para o entendimento técnico e futuras manutenções.

O painel possui 50 cm de largura por 40 cm de comprimento, em que utiliza dois trilhos DIN alocados nas posições superiores e inferiores do painel para sustentação e organização dos dispositivos. Também foram utilizadas quatro canaletas plásticas de 40 cm de largura por 60 cm de comprimento para distribuição e passagem dos fios elétricos.

Ademais, os componentes selecionados do painel foram disponibilizados pelo Instituto Calcomp, mas podem ser substituídos com modelos mais econômicos conforme necessidade do usuário. A montagem física do painel foi realizada sob orientação técnica do time de Engenharia de Automação, assegurando a correta instalação e boas práticas de montagem de painéis elétricos industriais.

Figura 11 - Layout do painel elétrico



Fonte: O próprio autor, 2025.

A Figura 12 apresenta o painel físico finalizado do sistema. Na ilustração, percebe-se a organização dos dispositivos e cabeios de acordo com as características e polarização de cada componente utilizado.

Na parte superior esquerda, encontram-se os disjuntores bipolares e suas alimentações. Ao lado, está instalada a fonte de 24 VCC e suas ligações “jumpeadas”.

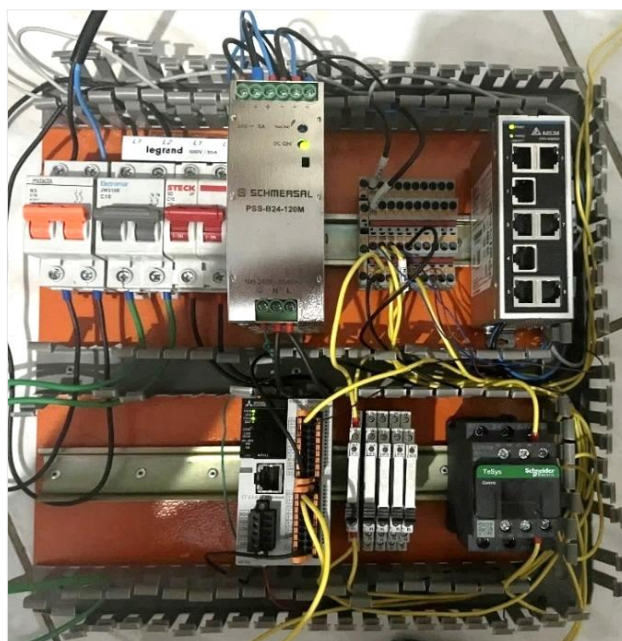
No centro superior localizam-se os bornes duplos. No total, foram usados 10 bornes, considerando as suas conexões superiores e inferiores destinadas a alimentação P24 (24V) e as centrais destinadas às conexões N24 (GND ou 0V).

No canto superior direito nota-se o *switch* industrial e suas ligações. O dispositivo possui 8 portas Ethernet, porém no projeto foram utilizadas apenas 3 portas levando em consideração as conexões do PC, CLP e servoamplificador.

Abaixo da fonte, encontra-se o CLP FX5 e suas conexões. No total foram utilizadas apenas 3 entradas digitais e 2 saídas digitais.

Ao lado do CLP, situam-se os relés interface que isolam eletricamente o circuito de comando, potência e conversão de sinais.

Por fim, no canto inferior direito, acha-se a contactora tripolar e suas ligações.

Figura 12 - Montagem final do painel

Fonte: O próprio autor, 2025.

O anexo B deste trabalho apresenta o diagrama elétrico do painel, destacando a energização dos dispositivos de comando e potência, incluindo o servoamplificador e fonte de alimentação. Já a Figura 13, mostra os pinos e conexões realizadas conforme especificação de cada componente utilizado.

Inicialmente, é possível observar que, para o funcionamento integral do painel, é necessária uma alimentação externa de 220V, a qual alimenta o disjuntor geral (NX3) de 16A, responsável pela distribuição de energia para os demais disjuntores auxiliares de 10A, ambos sob proteção adicional do barramento (jumpers).

A partir dessa distribuição, o disjuntor JW alimenta o servoamplificador através dos cabos de energização ligados às entradas L1 e L3 do equipamento.

Já o disjuntor SD alimenta a fonte de 24 VCC por meio dos cabos conectados às entradas N e L da mesma.

O servoamplificador possui conexões dedicadas para o servomotor, identificadas como U, V e W (fio vermelho, branco e preto), além do fio de aterramento (fio verde). Os cabos de encoder (CN2), cabo Ethernet (CN1) e bateria (CN4) também são interligados ao servo para comunicação, parametrização e operação do atuador linear.

A fonte possui 3 saídas positivas e negativas, desse modo, para viabilizar a distribuição da alimentação aos dispositivos que necessitam de alimentação de 24 VCC, foi necessário realizar jumpers entre as conexões.

A primeira saída positiva é destinada à alimentação dos bornes P24. A segunda saída é destinada à alimentação positiva (V1+) do *switch* e pino (+) do CLP. Já a terceira saída é atribuída a outra alimentação positiva (V2+) do *switch*.

A primeira saída negativa é destinada à alimentação dos bornes N24. A segunda saída é atribuída à alimentação negativa (V0) do *switch*. Por consequência, a terceira saída negativa é destinada a outra alimentação negativa (V0) do *switch* e pino (-) do CLP.

Nos bornes com alimentação P24 estão conectados: entrada A1 (+) da contactora tripolar, sinal comum do CLP (S/S), entrada (11) do relé interface, alimentação positiva do sensor de inspeção e *home* e alimentação positiva do LED de alerta.

Já nos bornes com alimentação N24 estão conectados: entrada A2 (-) da contactora tripolar, sinal comum do CLP (COM0), alimentação negativa do sensor de inspeção e *home* e alimentação negativa do botão de inicialização.

O CLP é projetado com as entradas digitais (X0, X1 e X2) conectadas aos cabeamentos de retorno de sinal do botão de inicialização, sensor *home* e de inspeção, respectivamente. As saídas digitais (Y0 e Y1) são destinadas aos acionamentos do relé interface e do LED de alerta, respectivamente, com base na lógica de programação definida pelo usuário.

Durante o funcionamento integral do painel, o uso dos relés interface não foi necessário, pois os dispositivos de campo (sensores e botões) operam no mesmo sinal lógico do CLP, no caso, NPN. Assim como os relés, optou-se por não utilizar a contactora haja vista que a alimentação do servoamplificador foi realizada diretamente pelo disjuntor JW2 (10 A), e, portanto, o diagrama elétrico não contempla as ligações dos equipamentos supracitados, trazendo mais simplicidade funcional e econômica ao usuário.

Dessa forma, o diagrama elétrico juntamente com as indicações de pinagem e conexões, sumariza a topologia elétrica do sistema, mostrando a distribuição de energia, interconexão lógica entre dispositivos e proteção, garantindo o funcionamento de forma efetiva, prática e de fácil compreensão.

Figura 13 - Descrição de pinos e conexões do painel

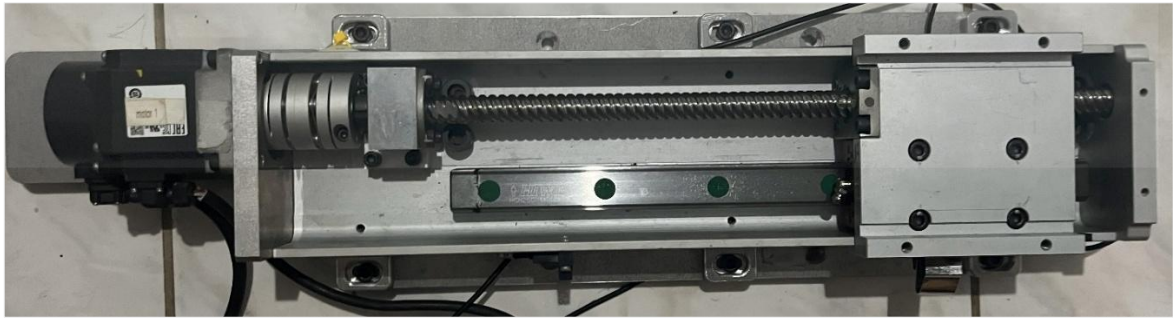
ALIMENTAÇÃO EXTERNA 220V	DISJUNTOR BIPOLAR 16A
DISJUNTOR BIPOLAR 16A	DISJUNTOR BIPOLAR JW 10A
DISJUNTOR BIPOLAR 16A	DISJUNTOR BIPOLAR SD 10A
	SERVO AMPLIFICADOR MR-JE-20C
	FONTE DE ALIMENTAÇÃO 24VCC
SERVO AMPLIFICADOR MR-JE-20C	
DISJUNTOR BIPOLAR JW 10A FASE 1	L1
DISJUNTOR BIPOLAR JW 10A FASE 2	L3
U	FIO VERMELHO (SERVOMOTOR)
V	FIO BRANCO (SERVOMOTOR)
W	FIO PRETO (SERVOMOTOR)
ATERRAMENTO	FIO VERDE (SERVOMOTOR)
CN2	CABO ENCODER (SERVOMOTOR)
CN1	CABO ETHERNET
CN4	BATERIA
FONTE DE ALIMENTAÇÃO 24VCC	
DISJUNTOR BIPOLAR SD 10A FASE 1	N
DISJUNTOR BIPOLAR SD 10A FASE 2	L
SAÍDA 1 (+)	BORNES P24 (+)
SAÍDA 2 (+)	V1+ (SWTICH) / PINO + (CLP FX5U)
SAÍDA 3 (+)	V2+ (SWTICH)
SAÍDA 4 (-)	BORNES N24 (-)
SAÍDA 5 (-)	0V (SWITCH)
SAÍDA 6 (-)	0V (SWITCH) / PINO - (CLP FX5U)
BORNES P24 (-)	
BORNE 1 (+)	A1 RELÉ
BORNE 2 (+)	S/S (CLP)
BORNE 3 (+)	I1 RELÉ
BORNE 4 (+)	FIO POSITIVO (+) SENSOR HOME
BORNE 5 (+)	SINAC (+) LED
BORNE 6 (+)	FIO POSITIVO (+) SENSOR INSPEC
BORNES N24(-)	
BORNE 1 (-)	FIO NEGATIVO (-) SENSOR HOME
BORNE 2 (-)	A2 CONTACTORA
BORNE 3 (-)	SINAL (-) BOTÃO
BORNE 4 (-)	FIO NEGATIVO (-) SENSOR INSPEC
BORNE 5 (-)	COM0 (CLP)
CLP FXSU	
X0	BOTÃO
X1	FIO DE SINAL SENSOR HOME
X2	FIO DE SINAL SENSOR INSPEC
Y0	A2 RELÉ 01
Y1	SINAL (-) LED

Fonte: O próprio autor, 2025.

3.6 ANÁLISE DO ATUADOR LINEAR

Os atuadores lineares possuem diversas extensões disponíveis no mercado, logo para realizar os testes de movimentação do produto até a zona de inspeção do sistema, utilizou-se o atuador de 500 mm da marca OBR conforme a Figura 14. Diante dos atuadores disponibilizados pelo Instituto, esse mecanismo foi escolhido devido à precisão em seus movimentos, velocidade, estrutura simples e custo acessível.

Figura 14 – Modelo do atuador linear utilizado no projeto



Fonte: O próprio autor, 2025.

3.6.1 Componentes selecionados para montagem

A Figura 15 apresenta os seguintes componentes elementares durante a montagem do atuador:

- **(a) Servoamplificador:** O modelo usado foi o MR-JE-20C da Mitsubishi. Foi inserido para controle e monitoramento do servomotor.
- **(b) Servomotor:** O modelo escolhido foi o HG-23KN da Mitsubishi. Foi utilizado para movimentação da placa até o ponto de inspeção da webcam. Esse servo possui rápida velocidade de resposta às alterações de tensão e corrente provenientes do servo amplificador.
- **(c) Bateria:** Utilizada para sustentação e armazenamento de energia do servoamplificador.
- **(d) Acoplamento:** Foi montado para unir dois eixos rotativos, ou seja, o eixo do fuso do atuador com diâmetro de 10 mm com o eixo do servomotor com 14 mm a fim de evitar o desalinhamento axial.
- **(e) Cabo de alimentação:** Usado para alimentação de tensão do servo amplificador para o servomotor.
- **(f) Cabo encoder:** Utilizado para transmissão de dados de velocidade, posição e aceleração do servomotor para o servo amplificador.
- **(e) Sensor forquilha:** O modelo escolhido foi EE-SX 671 da Omron, utilizado para detecção da peça metálica acoplada à base do atuador para definição do ponto inicial (*home*) e ponto de inspeção, essencial para o funcionamento do ciclo automático do sistema.

Figura 15 - Equipamentos utilizados na montagem do atuador



Fonte: O próprio autor, 2025.

A Figura 16 evidencia a montagem física do conjunto mecânico e eletrônico do atuador linear, integrando os elementos de transmissão de movimento e controle.

Na parte superior, observa-se a base fixa ao fuso e sua peça metálica na lateral. O movimento é transmitido por meio do acoplamento ligado ao eixo do motor, a fim de reduzir vibrações mecânicas. Ao longo da estrutura do atuador, nota-se a instalação de dois sensores forquilha, posicionados terminantemente para a detecção de fim de curso (posição de inspeção) e posição inicial (*home*), primordiais para emissão de sinais de referência para o CLP, no que se refere à posição em que o produto se encontra.

Nas imagens abaixo, observa-se a região mecânica do atuador e o servomotor fixado à estrutura com cabeios de potência conectados. Logo em seguida, as últimas ilustrações mostram a ligação do servoamplificador com conexões de rede e alimentação elétrica, seguindo as conexões indicadas na Figura 13.

Figura 16 - Montagem final e conexões do atuador



Fonte: O próprio autor, 2025.

3.6.2 Configuração do servomotor

Em razão da facilidade de configuração presente nos servos amplificadores da série MR-JE, foram alterados parâmetros de acordo com as instruções indicadas pela própria fabricante em seus manuais, evidenciados na tabela (Figura 17).

Figura 17 - Configuração e definição de variáveis do servo

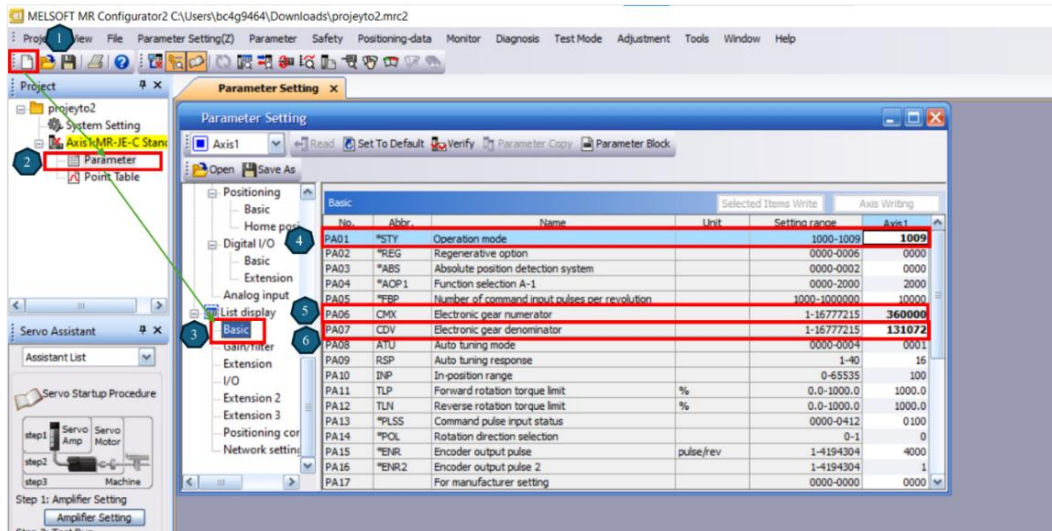
PARÂMETROS	VALOR	DESCRIÇÃO
PA01	1009	AJUSTE DO MODO DE OPERAÇÃO PARA PROFILE MODE
PA06	360000	MOVIMENTO DE GIRO COM UMA VOLTA (360,000°)
PA07	131072	PULSOS DO ENCODER COM UMA VOLTA
PD01	1C00	DESABILITAR FIM DE CURSO E SINAL DE EMERGÊNCIA

Fonte: O próprio autor, 2025.

A Figura 18 apresenta o passo a passo da configuração dos parâmetros do servoamplificador no MRConfigurator2:

1. Foi criado um projeto.
2. Foi acessado o menu “*Parameter*”.
3. Foi acessada a aba “*Basic*”, onde se concentram os parâmetros fundamentais do servo.
4. **PA01 - Operation Mode:** Foi ajustado para **1009** (*Profile Mode*), haja vista que esse modo de operação permite executar movimentos pré-configurados dentro do próprio amplificador, interpretando os comandos (velocidade, aceleração e torque) vindo do CLP.
5. **PA06 - Electronic gear numerator:** Foi definido para **360000**, pois especifica a quantidade de pulsos necessários para completar uma rotação (volta) no eixo do atuador.
6. **PA07 – Electronic gear denominator:** Foi ajustado para **131072**, já que esse modo corresponde à relação de movimentação do servo, que permite adequar essa relação entre pulsos de entrada e o deslocamento real do atuador.

Figura 18 - Parametrização de funções do servo

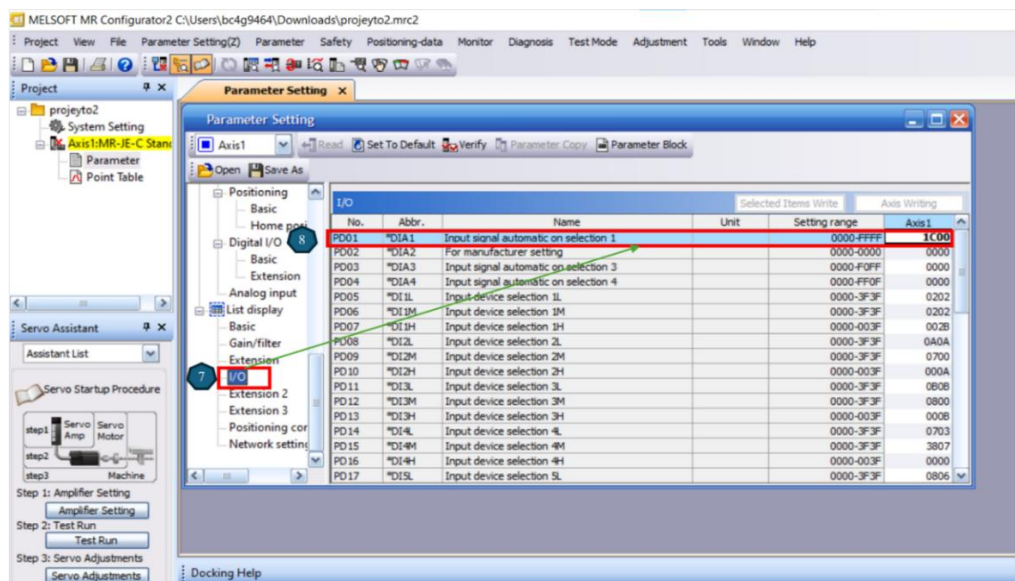


Fonte: O próprio autor, 2025.

A Figura 19 complementa o processo de parametrização do servo.

7. Foi acessado o menu “I/O” para exibição das funções de entrada do servo.
8. **PD01** - *Input signal automatic on selection 1*: Refere-se à primeira entrada digital. Foi ajustado para o valor **1C00**, pois corresponde a habilitação automática de movimento (Servo ON) com retorno ao ponto de referência (*home*) e reset do alarme.

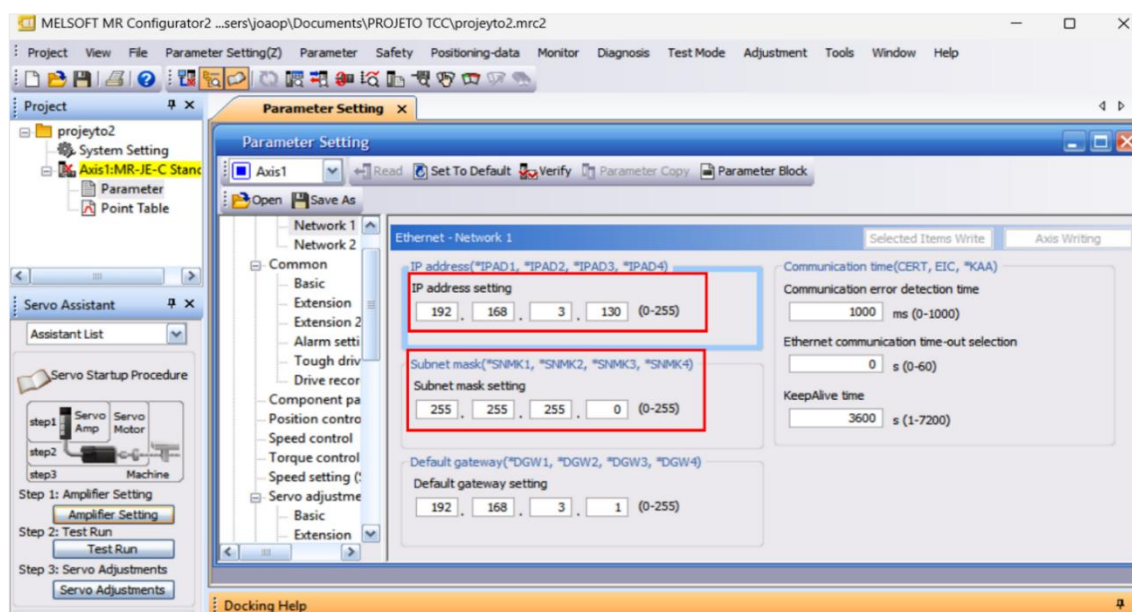
Figura 19 - Demais parametrizações do servo



Fonte: O próprio autor, 2025.

A Figura 20 mostra a interface de configuração de rede utilizada para comunicação entre o servoamplificador e os demais dispositivos. Nesta etapa foi definido o endereço IP (192.168.3.130) e máscara sub-rede (255.255.255.0), parâmetros que permitem identificar e integrar o servo ao modo Ethernet, garantindo a troca de dados com o CLP.

Figura 20 - Definição de endereço IP e sub-máscara do servo



Fonte: O próprio autor, 2025.

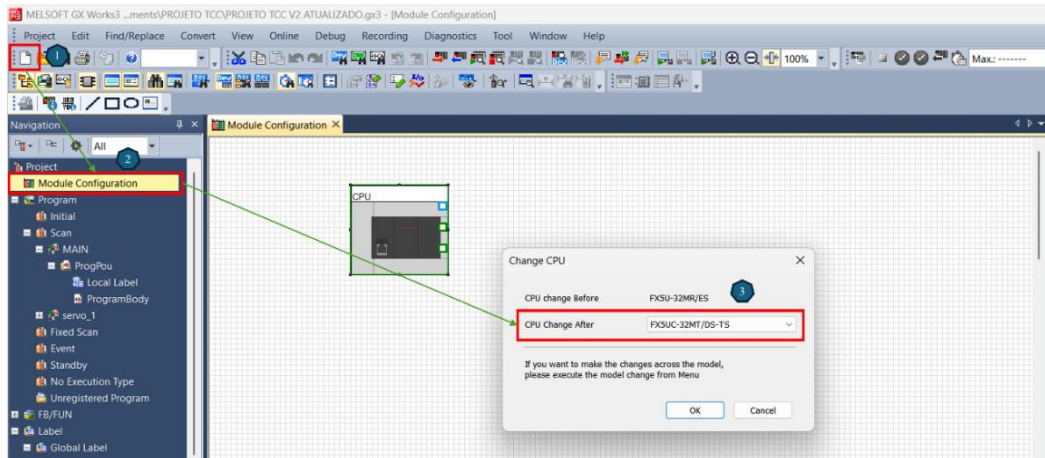
3.7 PARAMETRIZAÇÃO DO CLP NO SOFTWARE GX

3.7.1 Ajustes iniciais

Após montagem física e parametrização do servo, iniciou-se a configuração do CLP, seguindo as seguintes etapas mostradas na Figura 21, que incluem a definição do modelo e a preparação do ambiente de programação.

1. Foi criado um projeto no software GX Works3.
2. Na tela principal do software, observa-se o menu “*Module Configuration*”, local destinado à configuração da CPU (CLP).
3. A janela de opções exibe os campos “*CPU Change Before*” e “*CPU Change after*”, permitindo modificar o modelo da CPU do projeto. Nessa ocasião, o modelo foi alterado para FX5U-32MT/DSS-TS, haja vista que é o modelo de CLP escolhido e montado fisicamente.

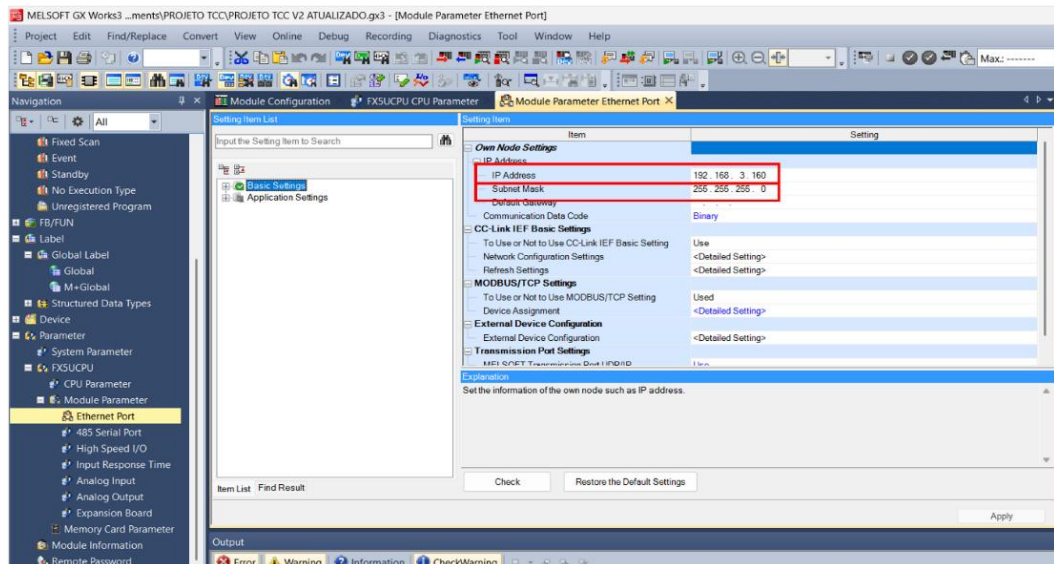
Figura 21 – Criação do projeto e alteração do modelo CPU



Fonte: O próprio autor, 2025.

A Figura 22 exibe a tela de parametrização da porta Ethernet do CLP no software, etapa necessária para estabelecer a comunicação entre o controlador, o servoamplificador e o sistema. Na seção “*Ethernet Port*”, foi definido o endereço IP (192.168.3.160) e máscara sub-rede (255.255.255.0), com a mesma configuração de rede do servo amplificador.

Figura 22 - Configuração de parâmetros Ethernet no CLP via GX Works3

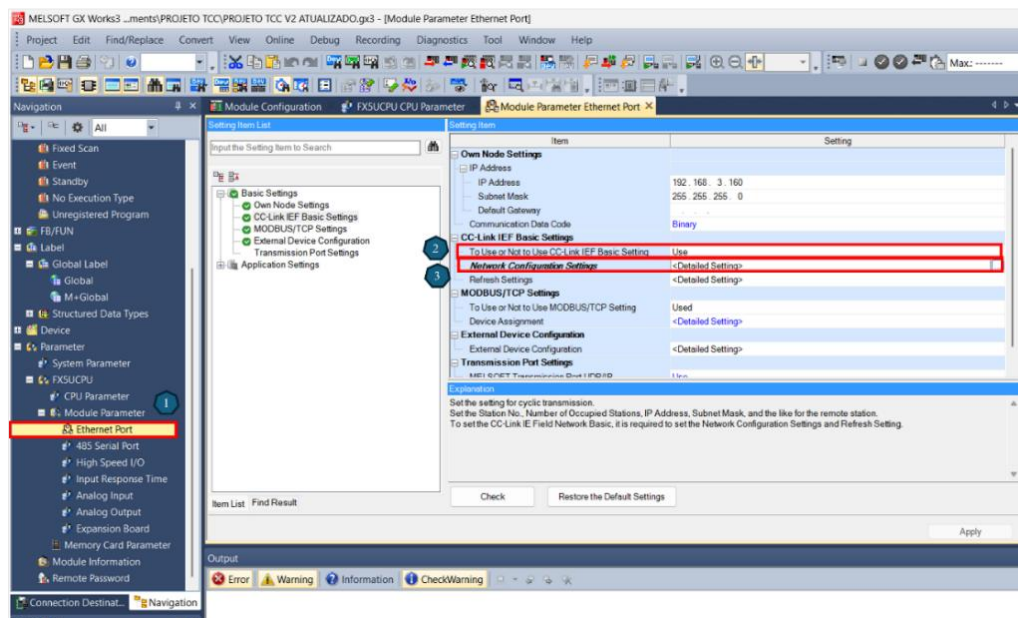


Fonte: O próprio autor, 2025.

A imagem a seguir (Figura 23) refere-se à habilitação do protocolo CC Link IE Field Basic Settings, considerando os seguintes passos:

1. Na aba “*Module Parameter*”, foi acessada a opção “*Ethernet Port*”, onde é possível definir as demais configurações de rede do CLP.
2. Na parte central, foi necessário habilitar o modo “*CC Link IEF Basic*”, já que esse modo permite compartilhar dados em alta velocidade entre os equipamentos da própria Mitsubishi como CLP’s, IHM’s e servos amplificadores.
3. Após habilitação, foi necessário acessar a seção “*Network Configuration Settings*” para visualização do detalhamento geral de rede, como número de estações utilizadas, mapeamento de dispositivos na mesma rede e atualizações de ciclos.

Figura 23 - Ativação da função CC Link Basic no software GX



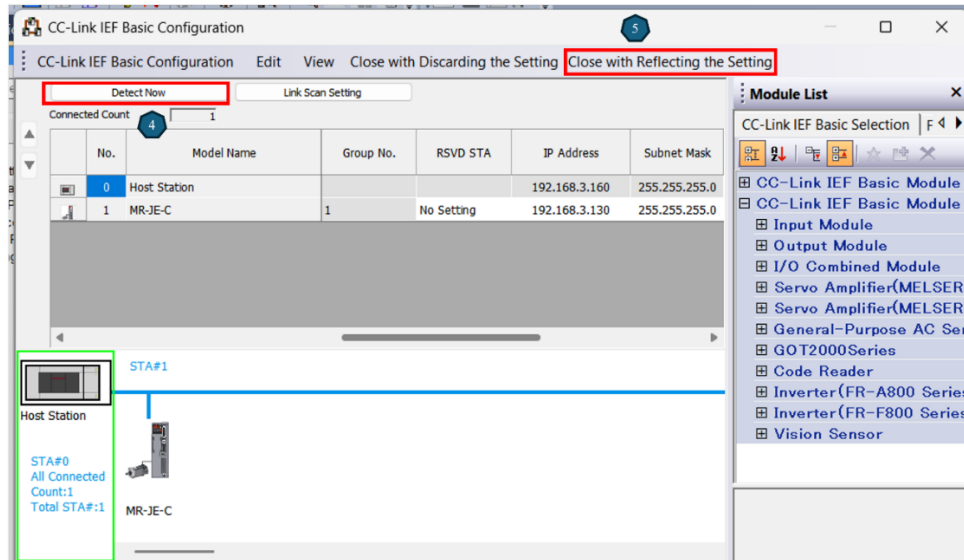
Fonte: O próprio autor, 2025.

A Figura 24 prossegue com as demais etapas de configuração:

4. Na parte superior, a opção “*Detect Now*” permite realizar a varredura (*scan*) automática da rede, identificando os dispositivos compatíveis e conectados à rede Ethernet do CLP. Nesse caso, observa-se na parte central do ambiente, a detecção do servoamplificador com suas respectivas informações de rede estabelecidas.
5. Após isso, o botão “*Close with Reflecting the Setting*” finaliza o procedimento, salvando as configurações para que sejam utilizadas no projeto. Portanto, esse estágio certifica que o CLP identifique e comunique corretamente com os

dispositivos que utilizam a função CC Link, garantindo o compartilhamento de dados de rede com o sistema.

Figura 24 – Identificação e comunicação do servoamplificador via CC Link



Fonte: O próprio autor, 2025.

3.7.2 Definição de variáveis e comando no CLP

No sentido de desenvolver uma programação organizada e interativa, foram utilizadas as tabelas a seguir para referenciar as memórias, entradas e saídas digitais do CLP.

A tabela (Figura 25) apresenta o endereçamento das entradas digitais utilizadas com suas respectivas nomenclaturas e descrições.

Figura 25 – Tabela de entradas digitais do CLP

ENTRADA	DESCRIÇÃO
X0	BOTÃO INICIAR
X1	SENSOR HOME
X2	SENSOR DE INSPEÇÃO

Fonte: O próprio autor, 2025.

A tabela a seguir (Figura 26) mostra o endereçamento das saídas digitais utilizadas com suas respectivas nomenclaturas e descrições.

Figura 26 – Tabela de saídas digitais do CLP

SAÍDA	DESCRIÇÃO
Y0	RELÉ INTERFACE
Y1	LED

Fonte: O próprio autor, 2025.

A tabela (Figura 27) mostra o endereçamento interno das memórias do CLP, priorizando os acionamentos importantes durante o funcionamento integral do sistema.

Figura 27 – Tabela de memórias do CLP

MEMÓRIA	DESCRIÇÃO
M0	COMANDO BOTÃO INICIAR
M1	COMANDO SENSOR HOME
M2	COMANDO SENSOR INSPEÇÃO
M4	MOV FRENTE
M6	COMANDO APROVADO
M8	COMANDO REPROVADO
M9	COMANDO RETORNO
M50	SERVO ON
M100	PY APROVADO
M200	INICIAR INSPEÇÃO
M300	PY REPROVADO

Fonte: O próprio autor, 2025.

Por consequência, a tabela (Figura 28) apresenta os comandos complementares na programação para controle e monitoramento do servomotor junto ao atuador linear durante sua operação.

- **MOTOR_LINEAR.SERVOON**: refere-se ao comando de habilitar (1) ou desabilitar (0) servomotor.
- **DMOV (MOTOR_LINEAR.PV SPEED)**: relaciona-se ao comando para definir a velocidade de operação, movimentando o atuador no sentido de avanço (frente), com referencial de (-10000) pulsos.
- **DMOV (MOTOR_LINEAR.PV SPEED)**: Semelhante ao comando anterior, contudo invertendo o sentido de movimento do atuador, retornando para a posição inicial com referencial de (10000) pulsos.
- **C0**: Trata-se do contador de peças reprovadas, configurado para acionar ações de alerta ao atingir o limite de 5 unidades.

- **RST C0:** Refere-se ao comando para zerar o contador, reestabelecendo o controle após realizar ações corretivas do processo.

Figura 28 – Demais comandos

OUTROS COMANDOS	VALOR	DESCRIÇÃO
MOTOR_LINEAR.SERVOON	1	BIT SERVO ON
DMOV (MOTOR_LINEAR.PV SPEED	-10000	VELOCIDADE EM MODO VEL (MOVER PARA FRENTE)
DMOV (MOTOR_LINEAR.PV SPEED	10000	VELOCIDADE EM MODO VEL (RETORNO)
C0	5	CONTADOR REPROVADO
RST C0	1	RESET CONTADOR REPROVADO

Fonte: O próprio autor, 2025.

A tabela (Figura 29) apresenta a relação entre as memórias do CLP e os endereços padrão do protocolo Modbus/TCP, utilizada para viabilizar a comunicação entre o controlador e outros dispositivos externos, como o sistema de inspeção desenvolvido. Nesse caso, considera-se as seguintes informações das memórias de integração:

- **M0** – Corresponde a memória do botão de inicialização (comando manual) para movimentação da base;
- **M100** – Corresponde a memória de status de aprovação do produto pelo sistema;
- **M200** – Corresponde a memória de espera do sinal do CLP para inicialização do clique automático de inspeção;
- **M300** – Corresponde a memória de status de reprovação do produto pelo sistema;

Figura 29 – Tabela de conversão para comunicação Modbus

CLP	MODBUS DEVICE N.
M0	8192
M100	8292
M200	8392
M300	8492

Fonte: O próprio autor, 2025.

Com base na tabela citada anteriormente, a Figura 30 exibe a configuração do parâmetro “*Modbus Device Allocation Parameter*”, que define essa correspondência no software, garantindo a integração e interoperabilidade dentro da arquitetura do projeto.

Figura 30 – Parâmetros de referência para comunicação Modbus

Item	Coil	Input	Input Registers	Holding Registers
MODBUS Device Allocation Parameter Set the parameter so that the MODBUS device can be related to the device memory of PLC CPU as a slave.				
Allocation 1				
Device	Y0	X0		D0
Start MODBUS Device No.	0	0	0	0
Allocation Points	1024	1024	0	8000
Allocation 2				
Device	M0	MEMÓRIA REFERÊNCIA		SDO
Start MODBUS Device No.	8192	0	0	20480
Allocation Points	7680	0	0	10000

Fonte: O próprio autor, 2025.

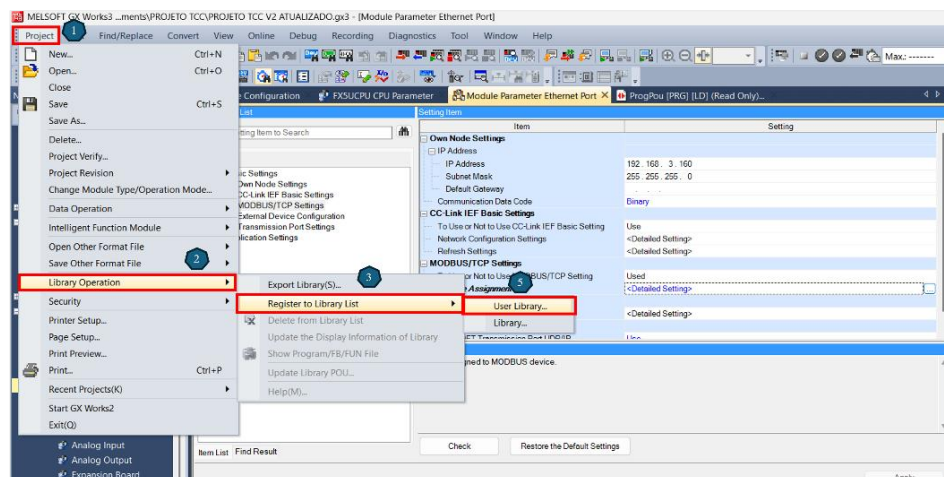
3.7.3 Adição de bloco de função ao projeto

Para que o CLP enviase os comandos corretos para o servoamplificador e compartilhasse as informações de cinemática, foi necessário realizar o download do bloco de funções da série MR-JE-C no site da Mitsubishi e posteriormente adicioná-lo ao software. Para isso, foram seguidas as etapas demonstradas na Figura 31.

1. No menu principal, foi acessado a aba “*Project*”;
2. Em seguida, foi acessado a janela “*Library Operation*”;
3. É acessado a opção “*Register to Library List*”
4. A opção “*User Library*” é exibida e selecionada posteriormente;

Esse estágio foi importante para habilitar o controle do servoamplificador com o CLP, uma vez que as alterações de velocidade, quantidade de rotações e habitação do servo (servo ON) podem ser feitas dentro do ambiente de programação.

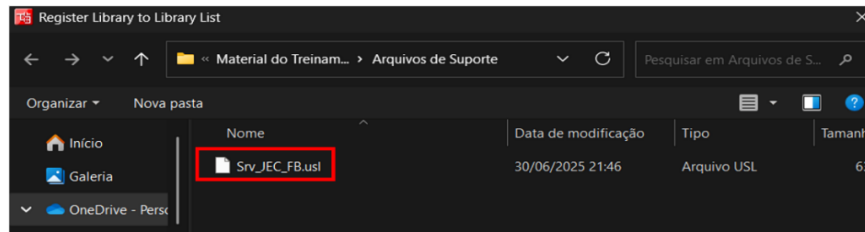
Figura 31 – Inclusão da biblioteca FB do servo no software GX



Fonte: O próprio autor, 2025.

A Figura 32 ilustra o processo de importação do arquivo do bloco de funções para o software, onde ele é selecionado para inclusão no ambiente de programação. O arquivo também está disponível no material de treinamento da plataforma de cursos da fabricante.

Figura 32 - Importação do arquivo para o software GX

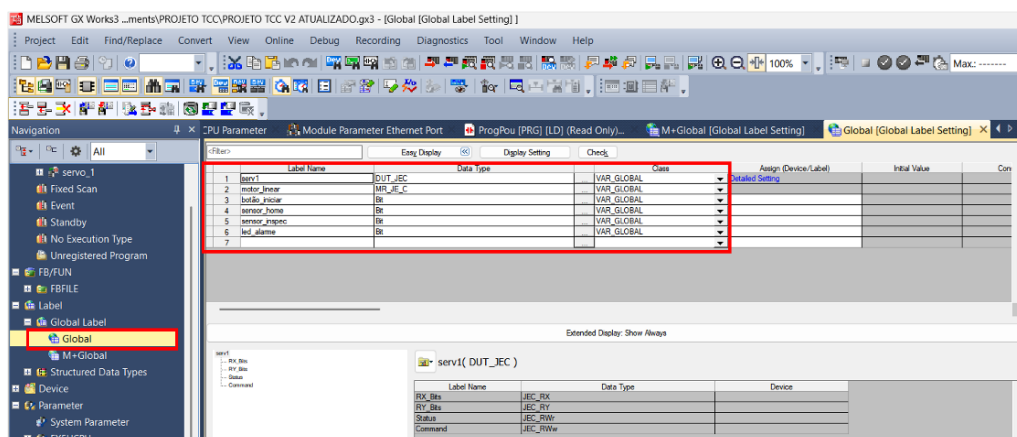


Fonte: O próprio autor, 2025.

Após importação, é necessário incluir as variáveis globais presentes no arquivo, ou seja, variáveis que podem ser acessadas por qualquer parte do programa. Para isso, seguiram-se as seguintes etapas, evidenciadas na Figura 33.

1. Na aba “Label”, foi acessada a janela “Global”, o que permitiu definir as variáveis que serão utilizadas durante o programa.
2. Na região central do ambiente de configuração, observa-se que cada variável nomeada foi correlacionada com o tipo de dados como DUT_JEC (dados específicos do servomotor), MR_JE_C (bloco de função padrão do servomotor) e Bit (Habilitar e desabilitar comando).

Figura 33 – Definição de variáveis globais



Fonte: O próprio autor, 2025.

3.7.4 Programação

Após configuração do ambiente de programação no software, deu início ao desenvolvimento da lógica do CLP visando atender às necessidades descritas no fluxograma (Figura 8).

A linguagem utilizada no projeto foi a Ladder, pois trata-se de uma linguagem com estrutura dinâmica, intuitiva e rápida adaptação, representada através de linhas horizontais e símbolos gráficos como bobinas, contatos e temporizadores.

O uso da linguagem Ladder foi essencial neste projeto por várias situações:

- **Visualização acessível:** O Ladder colabora com o entendimento do processo lógico de forma gráfica, assemelhando-se ao comportamento de circuitos elétricos com relés.
- **Ajuste simples:** as alterações de parâmetros definidos na programação, podem ser feitas em tempo real, adequando-se às necessidades do usuário.
- **Rápida análise:** Em casos de falhas ou erros na sequência lógica, o Ladder permite detectar imediatamente qual condição não está sendo implementada, auxiliando os usuários nas correções na programação.
- **Segurança sequencial:** O Ladder permite adicionar os intertravamentos, sendo assim, as memórias dos mecanismos só são executadas quando as condições de segurança forem cumpridas.

Desse modo, a lógica foi planejada de maneira adaptável, ou seja, cada etapa do processo desde a inicialização até a pós-inspeção está associada a uma condição bem definida, viabilizando manutenções, análises e futuras melhorias, evidenciadas nos anexos C à F deste trabalho.

A lógica por trás da programação desenvolvida seguiu os seguintes fundamentos:

- **Inicialização do sistema:** Nas primeiras linhas do programa (1 a 4) garantem que o processo só inicie após um comando manual (botão), permitindo a movimentação da base para a zona de inspeção. Isso evita acionamentos indesejados e assegura que o processo inicie após o consentimento do operador.
- **Intertravamento:** Nas linhas 5 a 7, são incluídas as condições de intertravamento a fim de garantir que a inspeção só comece quando o sensor de inspeção detectar a peça metálica da base. Além disso, a lógica “selo” mantém essa condição até que o movimento do atuador seja concluído.

- **Controle de movimento:** Nas linhas 8, 13, 14, 15 e 16 são utilizados os registradores (DMOV), ativação do bit de velocidade e lógica “selo” do próprio bloco de função do servo, pois permite que o deslocamento do atuador com valores de pulsos pré-determinados, garantindo repetibilidade e clique automático do atuador.
- **Controle de resultados de inspeção:** Nas linhas 9 a 12 foram definidos os comandos para peças aprovadas e reprovadas de acordo com as memórias dedicadas para cada resultado, evitando que um comando envie um resultado incorreto. As memórias (M6, M8 e M9) trabalham como memórias de estado, auxiliando o CLP e indicando em qual etapa do ciclo de programação está em execução.
- **Controle de rejeitos e alarmes:** Nas linhas 17 a 19, foi inserida a função de contagem de rejeitos e alertas para excesso de peças não conformes. Isso gera um monitoramento em tempo real, pois ao atingir o limite pré-definido (5 rejeitos), um dispositivo, no caso, um LED, é ligado, interrompendo o processo até que sejam feitas as devidas correções e retornando à rotina do sistema.

Portanto, o raciocínio proposto buscou garantir que a integração e o funcionamento do sistema fossem eficientes, rastreáveis e de fácil manutenção, critérios indispensáveis dentro das empresas.

3.8 APRENDIZADO DE MÁQUINA PARA INSPEÇÃO

3.8.1 Configuração

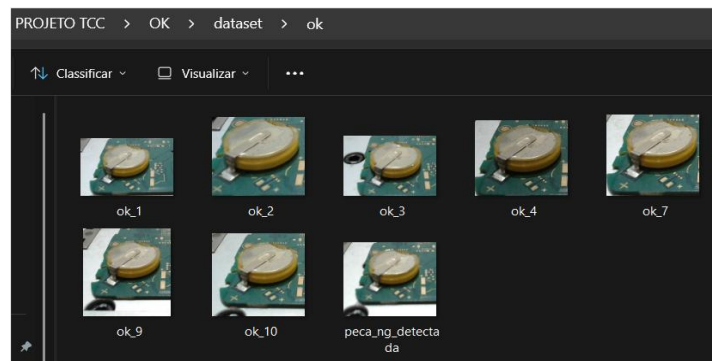
A projeção inicial da implementação do Aprendizado de Máquina para classificação de peças, a partir da utilização da webcam Brio 500 (Figura 7d), por obter características técnicas para inspeção visual, como qualidade de imagem, correção automática de iluminação, enquadramento dinâmico e custo acessível no mercado da informática. Essas características garantem maior precisão na captura de imagens, contribuindo para a eficácia do modelo de inspeção e viabilidade econômica do projeto.

Todas as imagens iniciais obtidas por meio da webcam compuseram o *Dataset* do sistema, em outras palavras, trata-se de um conjunto estruturado de dados visuais (imagens) usados para treinar e testar o modelo de AM, uma vez que influenciam diretamente no comportamento do sistema em reconhecer padrões, distinguir as baterias conformes e não

conformes e enviar esses resultados para o CLP de maneira integrada, transformando-o como classificador automático.

A Figura 34 mostra a primeira parte da estrutura do *Dataset* utilizado no desenvolvimento do sistema. Nessa conjunção, são exibidas as imagens na classificação “OK” que está relacionada com as peças em conformidade com os critérios de qualidade. Cada imagem representa uma amostra visual, destacando variações de ângulos, iluminação e posição da bateria. Dessa forma, para aumentar a confiabilidade do modelo de classificação, é necessário obter um maior grupo de amostras visuais.

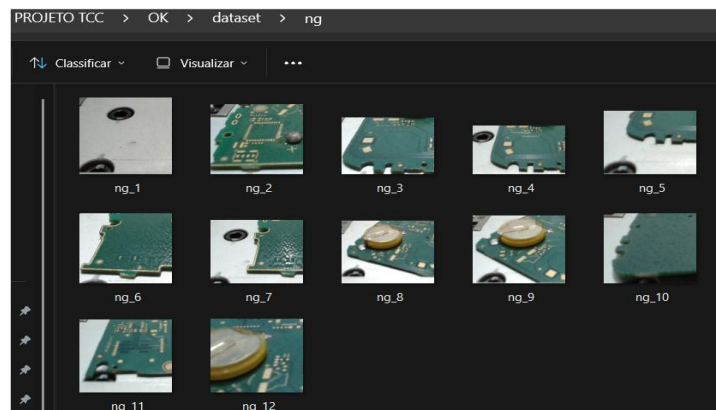
Figura 34 – Dataset para peças OK.



Fonte: O próprio autor, 2025.

Da mesma forma, a Figura 35 apresenta a segunda parte da estrutura do *Dataset*, onde as imagens na categoria “NG” estão relacionadas com peças consideradas não conformes (reprovadas) pela qualidade. É essencial incluir estas imagens, pois ajudam o classificador a distinguir corretamente os padrões visuais de baterias aprovadas e reprovadas.

Figura 35 – Dataset para peças NG.



Fonte: O próprio autor, 2025.

Esse procedimento foi essencial para que o sistema de inspeção pudesse avaliar e classificar as condições das baterias de forma assertiva, comparando a ROI extraída a partir das imagens capturadas pela webcam com a ROI padrão definida. Desse modo, a Figura 36 evidencia o posicionamento correto da bateria na base junto à webcam para definição das informações essenciais que auxiliaram na classificação do sistema.

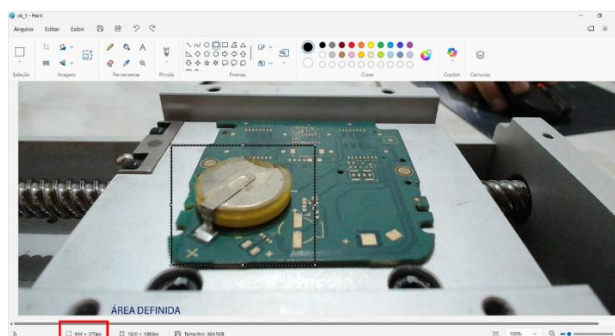
Figura 36 – Posicionamento da placa para definição da ROI.



Fonte: O próprio autor, 2025.

Na Figura 37, observa-se a placa com a bateria devidamente montada e soldada dentro da área de inspeção, enquanto o quadrado em destaque representa a zona delimitada que corresponde à região de interesse com sua dimensão indicada no canto inferior esquerdo (444 x 319 pixels). Essa zona trata-se da área pertinente da placa, ou seja, a região onde a bateria está montada. Esse método permitiu isolar apenas a parte de interesse, descartando os demais elementos externos que não contribuem para a classificação, possibilitando assim que o sistema concentrasse seu processamento apenas na região crítica para detecção de não conformidades como deslocamentos e ausência da bateria, aumentando a precisão e assertividade do modelo criado.

Figura 37 – Área definida da ROI padrão.

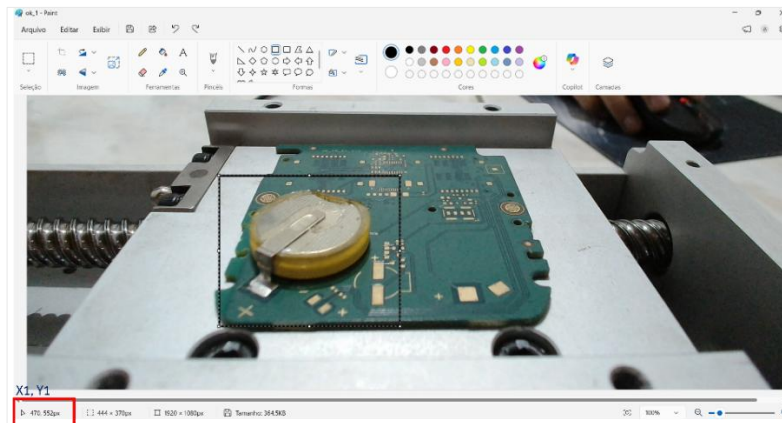


Fonte: O próprio autor, 2025.

A Figura 38 destaca as informações das coordenadas da ROI (X1 e Y1) exibidas no canto inferior esquerdo (470;552 pixels).

Essas informações foram importantes, pois determinaram exatamente onde a ROI começa dentro da imagem completa, desse modo, se o ponto inicial da ROI não tivesse sido bem definido, o sistema poderia analisar partes incorretas da placa, gerando falsas detecções e eliminando até pontos críticos de comparação da bateria.

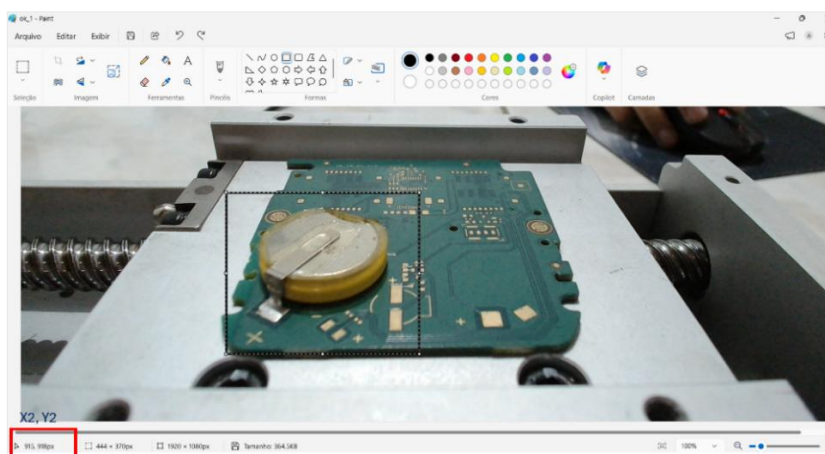
Figura 38 – Definição das coordenadas X1 e Y1 da ROI.



Fonte: O próprio autor, 2025.

De forma análoga, a Figura 39 resalta as informações das coordenadas da ROI (X2 e Y2) evidenciadas no canto inferior esquerdo (915; 918 pixels).

Figura 39 – Definição das coordenadas X2 e Y2 da ROI.



Fonte: O próprio autor, 2025.

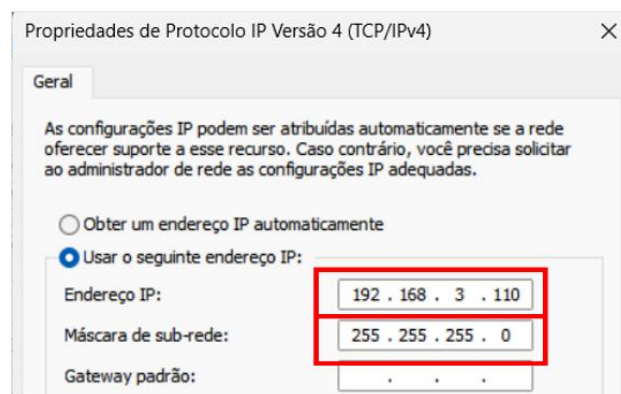
Portanto, as coordenadas da ROI padrão (X1, Y1, X2, Y2) tiveram como principal finalidade orientar o sistema, sendo inseridas nos conjuntos de instruções (*script*) apenas para assegurar que o sistema sempre irá analisar a mesma região da placa, ou seja, a área onde se encontra a bateria, independentemente de pequenas variações de posicionamento no enquadramento e captura da webcam. Desse modo, elas não foram um fator crítico para o funcionamento da integração, mas sim um recurso de padronização que garantiu a repetibilidade no processo de inspeção.

3.8.2 Implementação da integração sistema-CLP

Para viabilizar uma integração eficiente entre o sistema de inspeção e o CLP, foi necessário ajustar o endereço IP do computador (192.168.3.110) e sub-rede (255.255.255.0) na mesma faixa de rede do controlador e do servoamplificador, conforme demonstrado na Figura 40.

Essa configuração é fundamental para a comunicação do computador com os demais softwares e hardware, garantindo a praticidade e o funcionamento do sistema.

Figura 40 – Modificação do endereço IP e máscara do PC



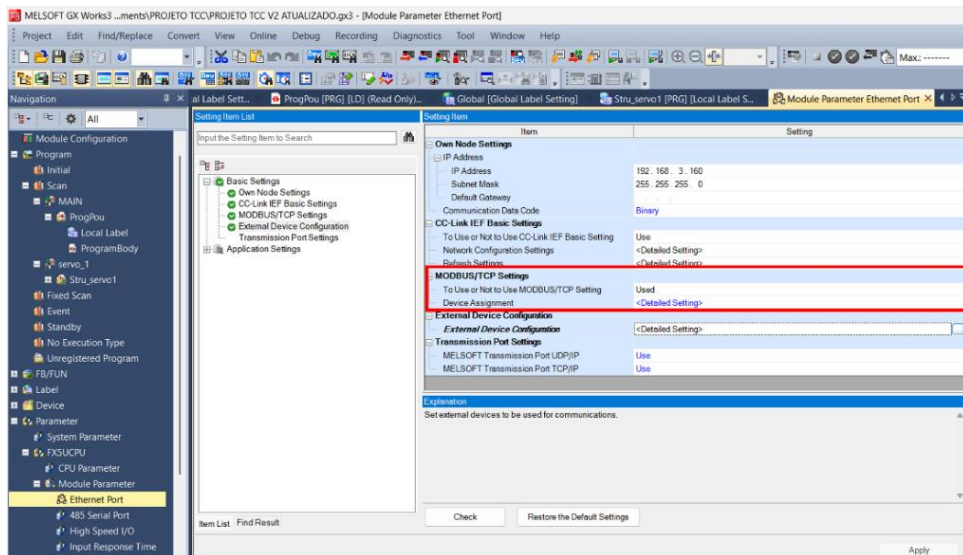
Fonte: O próprio autor, 2025.

A Figura 41 demonstra a tela de configuração para habilitar a comunicação Modbus/TCP no software, seguindo as orientações abaixo:

1. A comunicação foi habilitada através da opção “Used”
2. Foi acessada a opção “Device Assignment”, a qual deu acesso à janela de mapeamento automático dos dispositivos ligados à rede.

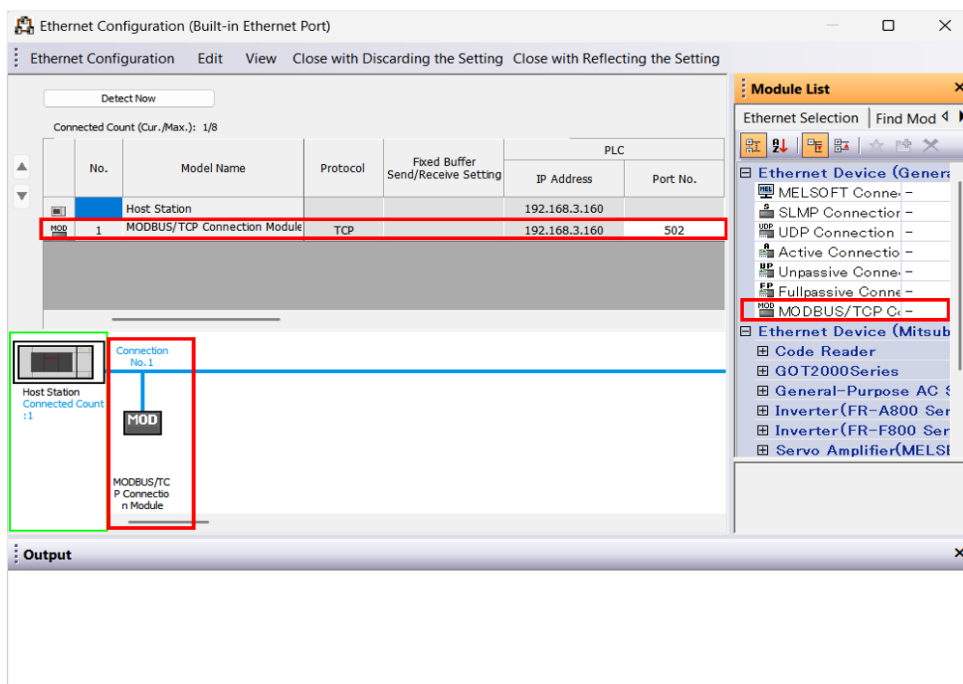
3. Na janela (Figura 42), foi necessário “arrastar” a função “MODBUS/TCP” para a região central do ambiente. Nesse caso, observa-se a detecção do modo Modbus com suas respectivas informações de rede, que serão inseridas na programação do sistema.

Figura 41 – Ativação da configuração Modbus/TCP no software GX



Fonte: O próprio autor, 2025.

Figura 42 – Adição de comunicação Modbus/TCP ao software GX



Fonte: O próprio autor, 2025.

3.8.3 Programação do modelo e processo de inspeção

O *script* foi desenvolvido sob orientação técnica do time de Software do Instituto Calcomp visando atender às necessidades descritas no fluxograma (Figura 8).

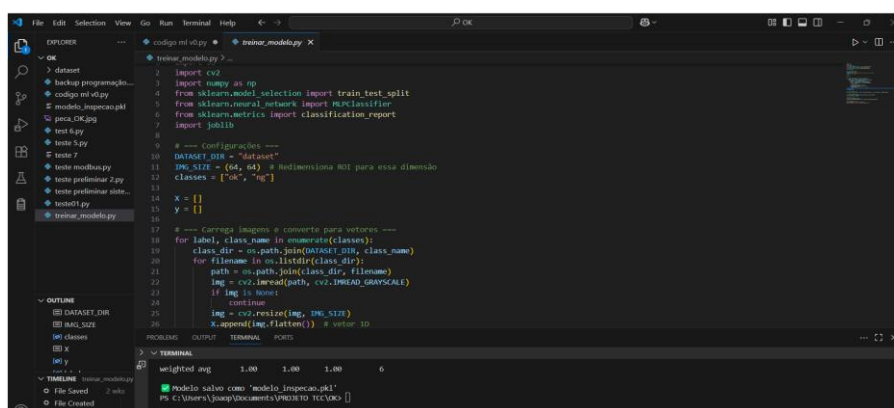
Logo, para melhor entendimento, a programação foi organizada em etapas, iniciando-se pela criação do modelo de inspeção, e posteriormente, o desenvolvimento da programação principal que inclui as rotinas de processamento de inspeção, captura de frame pela webcam e todos os requisitos de comunicação com CLP, de acordo código-fonte disponível no anexo G e H deste trabalho.

3.8.3.1 Criação do modelo de aprendizado de máquina

1. **Preparação de imagens do modelo:** As imagens do *Dataset* foram extraídas das suas respectivas pastas mostradas nas Figuras 34 e 35, em seguida, foram convertidas para escala de cinza e redimensionadas para 64x64 pixels. Segundo Gonzales e Woods (2018), a conversão para escala de cinzas em imagens é um passo essencial em aplicações de visão computacional, uma vez que elimina os três canais de cores (verde, vermelho e azul), mantendo apenas um, o que facilita o processamento dos algoritmos, mantendo as informações estruturais com maior grau de importância. Dessa forma, a conversão supracitada reduz ruídos visuais, limita a utilização de altos recursos computacionais e permite a detecção da estrutura do objeto com maior facilidade. O redimensionamento para 64x64 pixels é um tamanho recomendado pelos desenvolvedores, pois garante a qualidade visual sem prejudicar o processamento, ou seja, trabalhar nessas dimensões garante a regularidade do conjunto de dados, uma vez que os classificadores supervisionados necessitam de entradas com o mesmo número de atributos (dados).
2. **Treino e Teste:** Os atributos são divididos em 80% treino e 20% teste ($test_size=0.2$), ou seja, possibilita que o modelo aprenda a partir de um conjunto e compare com os demais. Esse processamento é necessário para verificar a capacidade de variação de imagens, dessa forma, o classificador funciona com imagens inéditas (aquelas que foram capturadas pela webcam), normaliza e compara com as demais existentes.

3. **Conjunção do modelo:** O *MPLClassifier*, rede neural implementada, trata-se de um conjunto de algoritmos de aprendizagem supervisionados que consegue identificar padrões complexos nos dados de entrada. Logo, o modelo treinado com essa rede ajusta os atributos das imagens capturadas de forma interativa até reduzir os erros de predição. De acordo com Ferreira (2019), o uso do *Multi layer Perceptrons* (MPL) traz capacidade de aproximação universal e sua flexibilidade para formar soluções de qualidade para uma ampla classe de problemas.
4. **Avaliação de Desempenho:** O parâmetro utilizado para avaliação foi o **classification_report** que apresenta indicadores estatísticos (*precision*, *recall* e *f1-score*) essenciais para avaliação de performance do classificador em cada seção. Esses indicadores permitem analisar não só as taxas de acertos do classificador, como também o desempenho da detecção da bateria e seus resultados (aprovados ou reprovados), garantindo maior confiabilidade do sistema de inspeção.
5. **Registro do modelo:** Após o treinamento, o modelo foi salvo no diretório do computador do usuário (**modelo_inspeção.pkl**). Esse arquivo é importante, pois evita treinar e criar um modelo sempre que o sistema for inicializado. Porém, a cada imagem que necessita ser inserida nas pastas do *Dataset*, é necessário atualizar o arquivo. Por fim, a Figura 43 apresenta a finalização do modelo criado no ambiente VS Code, certificando que as etapas anteriores foram bem-sucedidas.

Figura 43 – Criação do modelo para inspeção



```

1 import cv2
2 import numpy as np
3 from sklearn.model_selection import train_test_split
4 from sklearn.neural_network import MLPClassifier
5 from sklearn.metrics import classification_report
6 import joblib
7
8 # --- Configurações ---
9
10 DATASET_DIR = "dataset"
11 IMG_SIZE = (64, 64) # Dimensao ROI para essa dimensao
12 classes = ["ok", "ng"]
13
14 x = []
15 y = []
16
17 # --- Carrega imagens e converte para vetores ---
18 for label, class_name in enumerate(classes):
19     class_dir = os.path.join(DATASET_DIR, class_name)
20     for filename in os.listdir(class_dir):
21         path = os.path.join(class_dir, filename)
22         img = cv2.imread(path, cv2.IMREAD_GRAYSCALE)
23         if img is None:
24             continue
25         img = cv2.resize(img, IMG_SIZE)
26         x.append(img.flatten()) # vetor ID

```

Terminal Output:

```

weighted avg: 1.00 1.00 1.00 0
Modelo salvo com 'modelo_inspecao.pkl'

```

Fonte: O próprio autor, 2025.

3.8.3.2 Processo de integração e inspeção

1. **Configuração inicial do sistema:** O programa principal necessitou da importação das bibliotecas (*OpenCV*, *Time*, *NumPy*, *Joblib* e *pymodbus*) para garantir que todos os recursos do hardware (webcam e CLP) e software (modelo treinado e Ladder) fossem disponíveis antes da execução. Além disso, foi necessário estabelecer a conexão Modbus/TCP com o CLP, utilizando as informações de rede (IP e Port) e endereços de memória responsáveis por receber resultados da inspeção (aprovado ou reprovado). As coordenadas da ROI (citadas na seção 3.8.3) e a definição da resolução da webcam também foram incluídas na programação a fim de manter a padronização das variáveis estabelecidas no sistema.
2. **Aquisição de imagens:** A webcam foi ajustada para capturar as imagens na resolução 1920x1080 pixels (resolução padrão do dispositivo). Essa etapa é realizada a partir do momento em que o CLP envia o sinal de inspeção (bit de memória 200) e, portanto, o sistema realiza a captura de um frame (imagem) válido para comparação. O objetivo dessa etapa foi garantir que a inspeção fosse realizada somente quando houvesse um comando do CLP, sincronizando a inicialização do sistema e o processo de capturas de imagens.
3. **Pré-processamento de imagens:** Nessa etapa, a imagem capturada passa por uma transformação de padronização como aplicação de nitidez para realce de bordas, conversão para escala de cinza, recorte da ROI e redimensionamento para 64x64 pixels, compatível com o modelo de classificação criado. Essa etapa é importante, pois busca reduzir a variabilidade de informações das imagens, otimiza a capacidade computacional e padroniza a entrada do modelo.
4. **Preparação dos dados:** a ROI tratada é convertida em um vetor unidimensional (*flatten*), uma vez que o MPL trabalha com entrada em formato vetorial. A imagem digital pode ser representada como matriz bidimensional de pixels, onde cada pixel tem um valor correspondente a uma intensidade de cor. Logo, o *flatten* é responsável por reorganizar esses pixels em um vetor linear para realizar as operações aritméticas de classificação pelo sistema. Segundo Bishop (2006), a transformação de imagens em vetores é uma etapa primordial na preparação de dados para aprendizado supervisionado, permitindo que operações matemáticas extraiam padrões relevantes. Portanto, essa etapa busca

converter a informação visual em dados numéricos adequados ao modelo criado, permitindo a classificação da bateria.

5. **Análise da peça:** O modelo é carregado no ambiente onde realiza a predição com base no vetor da ROI, ou seja, considera como peça aprovada (classe 0) conforme critérios, e peça reprovada (classe 1), fora do padrão estabelecido.
6. **Comunicação com CLP:** De acordo com o resultado da classificação, se a peça estiver **aprovada**, o bit M100 é ativado por 6 segundos. Se a peça tiver sido **reprovada**, o bit M300 é ativado por 6 segundos. Logo, após esse tempo, o sinal é resetado, aguardando a próxima inspeção. Essa etapa apresenta, de fato, a cooperação das ferramentas, já que o resultado do sistema intervém nas ações do CLP.
7. **Encerramento do processo:** O sistema permanece em loop infinito aguardando novas peças até que o usuário encerre a execução. Quando finalizado, a webcam é desligada, a conexão com o CLP é interrompida e os recursos de software são encerrados.

Portanto, a Figura 44 mostra a aplicação prática das etapas supracitadas, exibindo o funcionamento do sistema em tempo real, de acordo com o código-fonte desenvolvido e disponível no anexo F deste trabalho.

Figura 44 – Funcionamento integral do sistema

```

1  import numpy as np
2  import cv2
3  import time
4  import joblib
5  from pymodbus.client import ModbusTcpClient
6
7  # === Configuração Modbus ===
8  client = ModbusTcpClient('192.168.3.168', port=502)
9  m100_endereco = 8292 # memória resultado peça OK
10 m200_endereco = 8392 # memória de início de inspeção
11 m300_endereco = 8492 # memória resultado peça NG
12
13 # === Resolução da câmera e ROI ===
14 CAM_LARGURA = 1920
15 CAM_ALTURA = 1080
16 ROI_coodernadas_padrao = (471, 552, 915, 922) # x1, y1, x2, y2
17 ML_Tamanho = (64, 64)
18
19 # === Função de nitidez ===
20 def aplicação_nitidez(imagem):
21     kernel = np.array([[0, -1, 0],
22                       [-1, 5, -1],
23                       [0, -1, 0]])
24     return cv2.filter2D(imagem, -1, kernel)
25

```

Terminal Output:

```

PS C:\Users\joaop\Documents\PROJETO TCC\OK> & C:\Users\joaop\AppData\Local\Microsoft\WindowsApps\python3.11.exe "c:\Users\joaop\Documents\PROJETO TCC\OK\codig
o ml v0.py"
Modelo carregado com sucesso.
Sistema de inspeção iniciado
Aguardando M200 do CLP...

```

Fonte: O próprio autor, 2025.

4 RESULTADOS E DISCUSSÕES

Neste capítulo será apresentado o desempenho funcional do sistema, incluindo dados sobre sua operação e uma análise de custo com outros sistemas disponíveis no mercado. O objetivo é evidenciar a real proposta do sistema, demonstrando sua vantagem, desvantagem, desafio, eficiência operacional aliada a um investimento acessível, destacando assim sua viabilidade técnica e econômica.

4.1 TESTES E AVALIAÇÃO DE DESEMPENHO

Foram conduzidos testes em placas para inspeção de baterias de lítio, onde elas foram disponibilizadas pelo time de Automação do Instituto Calcomp. O ciclo de experimento foi estruturado seguindo o fluxograma (Figura 8), assegurando que cada etapa fosse realizada de maneira controlada e padronizada.

Os testes foram executados em ambiente doméstico e laboratorial, selecionados intencionalmente para simular, de maneira prática, a rotina do conjunto de instruções do sistema em um processo produtivo, permitindo analisar seu desempenho antes de uma aplicação real na indústria. Essa tratativa possibilitou mapear eventuais correções e validar a eficiência do sistema em cenários próximos à realidade operacional, sem atingir a segurança ou a integridade física das baterias testadas.

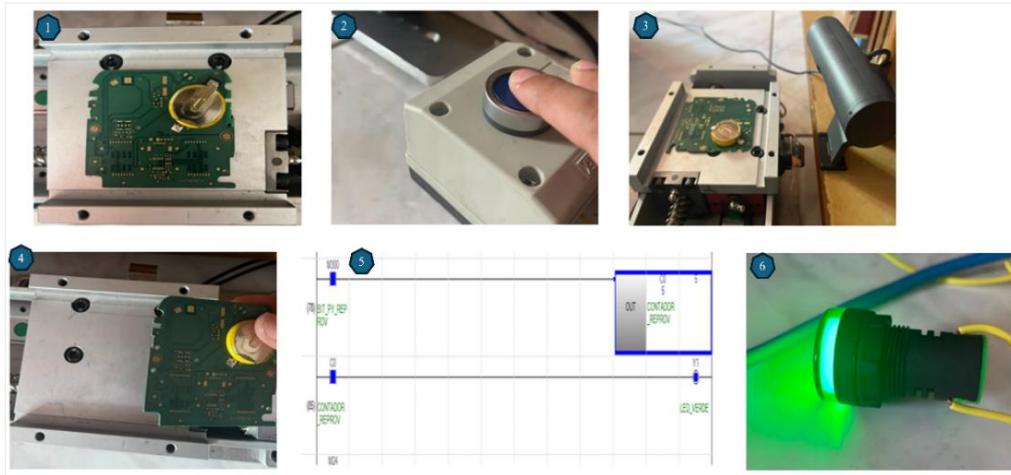
A Figura 45 apresenta as etapas de teste do sistema:

1. A placa foi posicionada na base do atuador linear. Observa-se que a placa foi devidamente alinhada à base para garantir a captura da imagem.
2. O botão de inicialização foi ativado, fazendo com que a base do atuador linear se movimentasse até a área de inspeção.
3. O sensor de inspeção enviou um sinal a partir da detecção da peça metálica da base, interrompendo sua movimentação e iniciando o processo de inspeção.
4. Resultado aprovado - A base retornou ao ponto inicial após 6 segundos, portanto, o operador retirou a placa da base e a direcionou para a próxima etapa enquanto o sistema aguardava o posicionamento de uma nova peça.
5. Resultado reprovado - A base retornou ao ponto inicial após 6 segundos, o operador retirou a placa da base e a direcionou para a área de rejeito enquanto o

sistema aguardava o novo comando. Nessa etapa, o contador de rejeito é inicializado.

6. Quando o contador atingiu o limite de 5 peças rejeitadas, o LED de alerta foi acionado, necessitando de intervenções técnicas no processo.

Figura 45 – Teste do funcionamento prático do sistema



Fonte: O próprio autor, 2025.

A Figura 46 apresenta, de maneira clara, os resultados da inspeção da bateria ao usuário, nas condições de APROVADO (à esquerda) e REPROVADO (à direita), destacando a região de montagem da bateria e evidenciando a capacidade do sistema em classificar corretamente as peças analisadas. Tais resultados demonstram a eficácia do projeto na diferenciação entre condições conformes e não conformes, reforçando sua aplicabilidade em processos de inspeção de placas eletrônicas e no controle de qualidade.

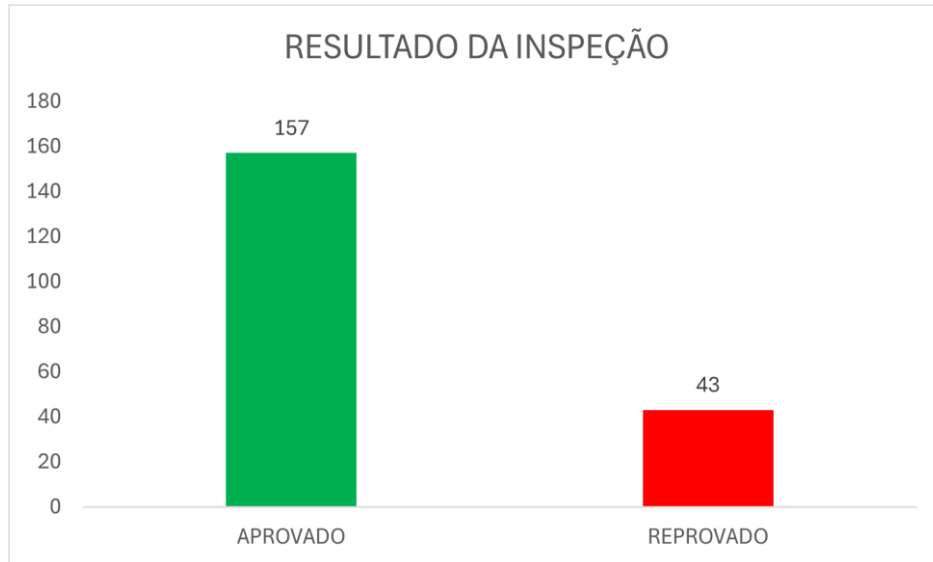
Figura 46 – Resultados de inspeção da bateria gerado pelo sistema



Fonte: O próprio autor, 2025.

No total, foram realizadas 200 inspeções de baterias, das quais 157 foram aprovadas e 43 foram reprovadas pelo sistema de inspeção conforme mostra a Figura 47.

Figura 47 – Gráfico de resultados gerais após testes



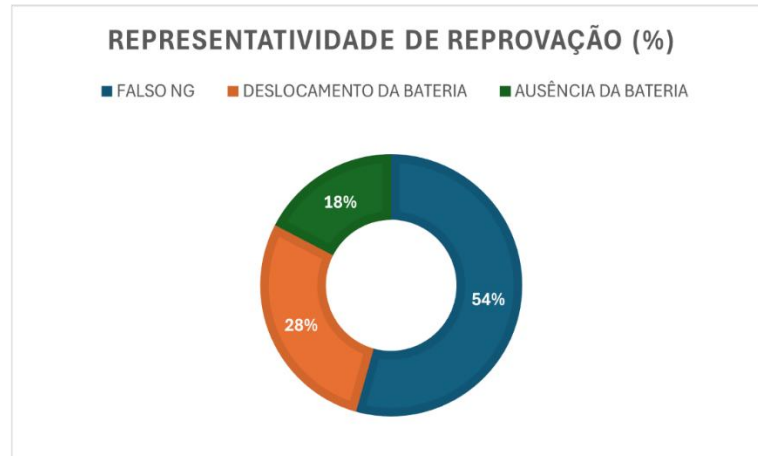
Fonte: O próprio autor, 2025.

Dentre as peças reprovadas, identificou-se que:

- **25 resultados** foram “falso NG”, ou seja, a bateria estava presente e em condição aceitável, mas o sistema classificou incorretamente como reprovado.
- **18 resultados** foram corretos, pois correspondiam a falhas reais, sendo 5 casos de ausência de bateria e 13 casos de deslocamento (mau posicionamento) durante montagem da bateria.

Observa-se que o sistema ainda apresenta adversidades, principalmente em função das 25 falsas reprovações ocasionadas por fatores externos, como variações de iluminação e a adaptação a diferentes locais de instalação. No entanto, tais fatores não comprometeram diretamente a eficiência geral do sistema proposto. Nesse contexto, o gráfico da Figura 48 apresenta a representatividade, em porcentagem, dos dados de rejeição pelo sistema.

Figura 48 – Gráfico de representatividade de rejeitos

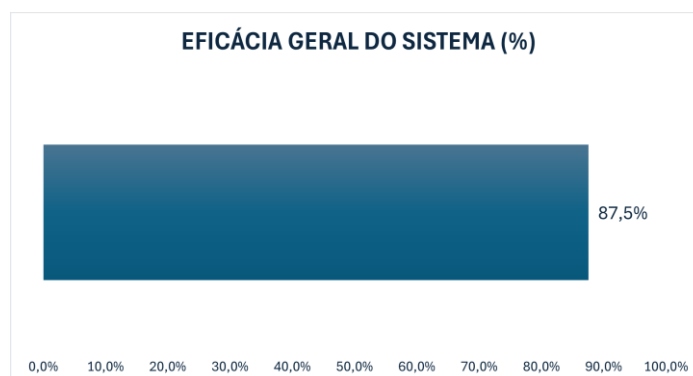


Fonte: O próprio autor, 2025.

Tendo em vista os resultados obtidos pelo sistema (157 aprovações verdadeiras e 18 reprovações verdadeiras), o desempenho do modelo de Aprendizado de Máquina foi de 87,5% em relação ao total de inspeções realizadas durante os testes. Embora o CLP e o servomotor não tenham falhado, as decisões incorretas do modelo (25 falsas reprovações) afetaram o rendimento final do processo e, portanto, foi considerado e definido a eficácia do sistema. Quanto ao desempenho dos hardwares, estes apresentaram funcionamento correto em 100% dos casos, garantindo robustez e confiabilidade operacional.

Dessa forma, ainda que as falsas rejeições representem uma limitação do modelo, a solução como um todo demonstrou elevada confiabilidade, integrando corretamente hardware e software e assegurando sua aplicabilidade em processos de inspeção de placas eletrônicas. O gráfico da Figura 49 apresenta a eficácia real do sistema, considerando a integração com os demais dispositivos do projeto.

Figura 49 -Gráfico de eficácia geral do sistema



Fonte: O próprio autor, 2025.

4.2 ANÁLISE DE CUSTO E COMPARAÇÃO COM DEMAIS SISTEMAS E CÂMERAS

A Figura 50 apresenta os valores e a quantidade de cada componente utilizado no projeto, totalizando um investimento de R\$ 14.826,90, com base nos preços de 2024. Esse valor, entretanto, pode ser reduzido com a substituição por componentes elétricos mais econômicos que desempenhem a mesma função, assim servomotor e servoamplificador de outra fabricante mais acessível. Além disso, considerando que alguns equipamentos, como relés de interface e a contactora não foram utilizados aliado com a disponibilização dos demais componentes pelo instituto que podem ser substituídos, o investimento pode ser ajustado para aproximadamente R\$ 13.780,00.

Figura 50 – Tabela de custo total do projeto

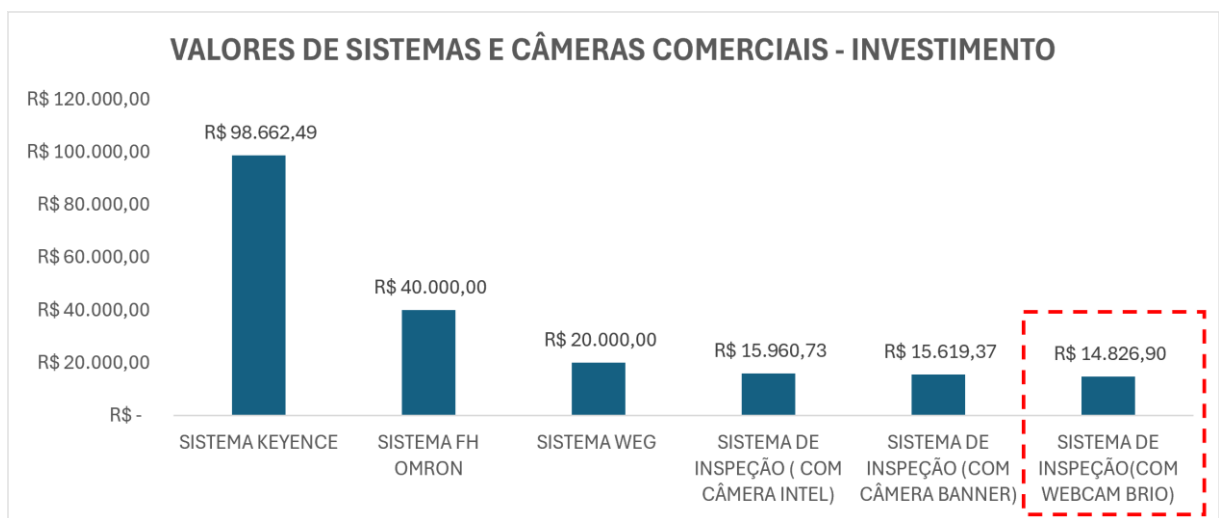
ITEM	CUSTO UNIT	QTS USO	UNID	TOTAL
WEBCAM LOGITECH BRIO 500	R\$ 629,53	1	PÇ	R\$ 629,53
SENSOR FOTOELÉTRICO 24VDC	R\$ 87,91	2	PÇ	R\$ 175,82
CLP FX5U MITSUBISHI	R\$ 3.852,50	1	PÇ	R\$ 3.852,50
SWITCH DVS DELTA	R\$ 1.528,81	1	PÇ	R\$ 1.528,81
PENTE DE ALIMENTAÇÃO 80A	R\$ 38,15	1	PÇ	R\$ 38,15
DISJUNTOR BIPOLAR 16A	R\$ 28,42	1	PÇ	R\$ 28,42
DISJUNTOR BIPOLAR 10A	R\$ 21,71	2	PÇ	R\$ 43,42
FONTE 24 VDC	R\$ 545,00	1	PÇ	R\$ 545,00
RELÉ INTERFACE	R\$ 20,23	5	PÇ	R\$ 101,15
ATUADOR DE COMPRIMENTO 500 MM	R\$ 1.117,75	1	PÇ	R\$ 1.117,75
SERVO AMPLIFICADOR MR-JE-20C	R\$ 2.780,75	1	PÇ	R\$ 2.780,75
SERVOMOTOR HG-KN23	R\$ 2.167,98	1	PÇ	R\$ 2.167,98
CONTACTORA TRIPOLAR 220V C24V	R\$ 202,47	1	PÇ	R\$ 202,47
BORNE PARAFUSO PARAFUSO	R\$ 11,10	10	PÇ	R\$ 111,00
CABO ENCODER	R\$ 499,00	1	PÇ	R\$ 499,00
BOTÃO PUSH + CAIXA PARA BOTÃO	R\$ 42,98	1	CJ	R\$ 42,98
CABO DE ALIMENTAÇÃO	R\$ 278,07	1	PÇ	R\$ 278,07
PONTE DE INTERLIGAÇÃO	R\$ 38,50	1	PÇ	R\$ 38,50
BATERIA	R\$ 275,50	1	PÇ	R\$ 275,50
ACOPLAMENTO 14 MM P/ 10MM	R\$ 118,90	1	PÇ	R\$ 118,90
CABO 2,5 MM	R\$ 1,56	50	MT	R\$ 78,00
CABO 1,0 MM	R\$ 0,82	50	MT	R\$ 41,00
CABO 1,5 MM	R\$ 0,82	50	MT	R\$ 41,00
TERMINAL 2.5 MM	R\$ 0,04	50	PÇ	R\$ 2,00
TERMINAL 1.5 MM	R\$ 0,05	50	PÇ	R\$ 2,50
CABO ETHERNET 2 M	R\$ 28,90	3	PÇ	R\$ 86,70
TOTAL				R\$ 14.826,90

Fonte: O próprio autor, 2025.

A Figura 51 apresenta a comparação de valores de investimento entre diferentes sistemas e câmeras industriais comerciais voltados à inspeção industrial. É possível observar que os sistemas consolidados no mercado, como o da Keyence e o da Omron, possuem custos elevados, o que apresenta um impedimento na criação de projetos de menor escala. Logo em seguida, outros sistemas de inspeção construídos com câmeras industriais como Weg, Intel e Banner apresentam valores intermediários, porém acima do valor investido no projeto.

Por outro ângulo, o sistema desenvolvido neste trabalho, utilizando uma webcam Logitech, apresentou um valor de investimento acessível entre todas as opções apresentadas. Esse diferencial representa uma economia de até 85% em relação ao sistema Keyence e cerca de 26% em relação aos demais sistemas de inspeção. Logo, o investimento reduzido do projeto não apenas demonstra a viabilidade econômica, como também ressalta seu potencial de inclusão e difusão tecnológica, permitindo assim que pequenas e médias empresas possam implementar um sistema de inspeção automatizada sem a necessidade de um investimento de alto valor.

Figura 51 – Gráfico comparativo entre sistemas e câmeras comerciais



Fonte: O próprio autor, 2025.

4.3 VANTAGENS E DESVANTAGENS E DESAFIOS DO PROJETO

Em comparação com as opções mencionadas anteriormente, as vantagens, desvantagens e desafios do projeto são:

4.3.1 Vantagens

1. **Flexibilidade e Personalização:** Como as programações foram realizadas em Python e Ladder, é possível adicionar novas funções com facilidade e alterar critérios de decisão lógica conforme necessidade orientada pelo corpo técnico da produção e qualidade. Como também o CLP, tem a capacidade de gerenciar mais equipamentos, garantindo uma automação mais robusta.

2. **Armazenamento de dados:** Após o processo de inspeção, é possível armazenar imagens e resultados das inspeções (OK/NG) automaticamente, contribuindo para análises de causas e soluções.
3. **Softwares gratuitos:** Com a utilização de softwares gratuitos para programação de acordo com equipamentos selecionados, portanto, não é necessário realizar pagamento de plataforma da fabricante ou acesso premium para download de aplicativos.
4. **Custo:** O sistema utilizou webcam convencional, em vez de câmeras industriais caras, o que é possível conectar a um PC, sem necessidade de sistemas dedicados como o da fabricante Keyence.

4.3.2 Desvantagens

1. **Inspeção individual:** Por se tratar de uma webcam comercial, só há possibilidade de inspecionar uma bateria por vez, o que pode restringir a capacidade de produção.
2. **Sensibilidade a fatores externos:** Dependendo da aplicabilidade, a webcam pode apresentar ruído visual e instabilidade na captura contínua, que envolve variações de iluminação e vibração, fazendo com que o sistema possa gerar falsos resultados.
3. **Treinamento manual:** Devida a presença de fatores externos e alteração física da bateria, pode haver necessidade de realizar o treinamento do modelo necessitando do suporte técnico para manter a eficácia do sistema.

4.3.3 Desafios

1. **Capacitação Profissional:** Dependendo do nível de conhecimento técnico do programador alinhado com os equipamentos e seus fabricantes, pode haver necessidade de realizar cursos e treinamentos para o desenvolvimento da programação e integração.
2. **Confiabilidade e repetibilidade:** A diferença entre baterias OK e NG pode ser mínima, dessa forma pode haver a necessidade de alcançar alta taxa de acerto (acurácia) com baixa taxa de falsos positivos/negativos.

3. **Comunicação e integração:** A integração entre as tecnologias do projeto pode ser feita através dos protocolos de rede, logo pode apresentar instabilidade na rede, travamentos no sistema operacional, sobrecarga de processamento ou falhas momentâneas no serviço.
4. **Artefatos mecânicos:** Tratando-se de mecanismos de movimentação como esteira e atuadores lineares, a adoção de um acoplamento mecânico para garantir o funcionamento adequado do motor torna-se importante para a transmissão de movimento. Como também, para inibir a movimentação da peça na inspeção, pode haver necessidade de uma base fixa sobre o mecanismo.

5 APLICAÇÕES FUTURAS

Para continuidade deste trabalho, propõe-se as seguintes aplicações futuras:

- **Desenvolvimento de Modelos de Aprendizado de Máquina para ambientes críticos:** Analisar o desempenho de sistemas de inspeção em aplicações críticas como separação de mercadorias, venda de cosméticos, logística e alimentação, considerando requisitos rigorosos de segurança e confiabilidade.
- **Inclusão de robôs industriais:** Explorar a implementação de robôs industriais na manipulação de peças, otimizando o desempenho em aspectos onde a repetibilidade e a estabilidade precisam coabitar.
- **Integração com sistemas supervisórios:** Avaliar a incorporação de sistemas supervisórios, explorando recursos como análise de falhas, indicadores visuais (IHM's), rastreabilidade e relatórios.
- **Desenvolvimento de interface gráfica:** Criar e implementar uma interface gráfica interativa a partir das bibliotecas Python voltadas para navegação e interação de resultados com o usuário, apoiando tomadas de decisão por meio de indicadores de desempenho.

6 CONCLUSÃO

Assim, a partir dos resultados obtidos através da composição do projeto e testes práticos, é possível afirmar que o sistema de inspeção baseado em Aprendizado de Máquina integrado ao CLP, atingiu com êxito seus objetivos previstos na elaboração da solução, a qual obteve uma eficiência de 87,5%. As decorrências demonstraram desempenho transparente que atendessem aos requisitos básicos de inspeção, confiabilidade, controle do CLP com a capacidade de análise e tomada de decisão, destacando-se pela velocidade de captura, comparação, tempo de resposta e construção com componentes convencionais disponíveis no mercado nacional. O processo de integração mediante a utilização de redes industriais como Modbus, permite a execução de tarefas de forma simples e padronizada, o que classifica a solução desenvolvida como acessível, econômica e dinâmica. Além disso, os softwares e ferramentas utilizadas estão disponibilizados gratuitamente nas plataformas das fabricantes, o que torna o sistema aberto a futuras aplicações e otimizações, permitindo personalizações e avanços conforme a necessidade do usuário. A principal limitação observada está nos 25 casos de falso NG (12,5%) ocorridos durante os testes, o que impactou a produtividade por gerar rejeição indevida, porém o projeto apresenta potencial para melhorias e ajustes de parâmetros e ampliação, seja por meio da inclusão de robôs industriais para manipulação de peças, transmissão de resultados para o IHM, torre de iluminação para alertas, automação mais precisa ou adaptação para diferentes escalas de uso, o que o torna uma solução versátil e promissora no contexto da indústria 4.0. Alguns aperfeiçoamentos podem ser aplicados ao projeto visando melhorar ainda mais sua performance, como o desenvolvimento de AM mais preciso e compacto. Desse modo, o sistema criado se apresenta como uma alternativa estratégica e eficaz para pequenas e médias indústrias, que buscam modernizar seus processos sem a necessidade de grandes investimentos em tecnologia de ponta ou infraestrutura, garantindo o alcance das metas operacionais com custo reduzido.

7 REFERÊNCIAS

AMARAL, Allan Francisco Forzza. **Arquitetura de computadores: curso técnico em informática**. Colatina, ES: CEAD/IFES, 2010. 100 p. Disponível em:

https://www.ufsm.br/app/uploads/sites/413/2018/12/02_arquitetura_computadores.pdf.

Acesso em: 18 ago. 2025.

ANDRADE, Willian Jeferson. **Deteção de anomalias de posicionamento em robôs industriais utilizando aprendizado de máquina**. 2024. 74 f. Dissertação (Mestrado em Engenharia Elétrica e Informática Industrial) – Universidade Tecnológica Federal do Paraná. Curitiba, 2024.

BISHOP, Christopher M. **Pattern recognition and machine learning**. New York: Springer, 2006.

BRANDÃO, Marcelo. **A importância da gestão do conhecimento para o futuro com novas tecnologias**. Consumidor Moderno, 11 dez. 2024. Disponível em:

<https://consumidormoderno.com.br/conhecimento-tecnologia-gestao/>. Acesso em: 17 ago. 2025.

CAROLA, Gabriel Alves. **Aplicações de inteligência artificial em sistemas de controle inteligente**. 2024. 50 f. Trabalho de Conclusão de Curso (Bacharelado em Engenharia da Computação) – Universidade de Goiás, Goiânia, 2024.

CASTRUCCI, P. L.; BITTAR, A.; SALES, R. M. **Controle automático**. Rio de Janeiro: GEN/LTC, 2011.

CAVALLIN, Fernando. **Estudo sobre redes de comunicação para automação industrial**. 2016. 58 f. Monografia (Especialização em Automação Industrial) – Universidade Tecnológica Federal do Paraná, Curitiba, 2016.

CAVALLARI, G. B. **Estudo de representações de imagens de múltiplos domínios a partir de aprendizado profundo não supervisionado e semi-supervisionado**. 2022. 65 p.

Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2022.

CRAVO, Edilson. **SCADA: o que é, importância, como funciona e mais**. 2024. Disponível em: <https://blog.kalatec.com.br/scada/>. Acesso em: 18 ago. 2025.

DA SILVA, Elcio Brito et al. **Automação & Sociedade Volume 2: Tecnologias emergentes associadas à Quarta Revolução Industrial**. Brasport, 2019.

FERREIRA, Carlos A. **MLP Classifier**. 2019. Disponível em: <https://medium.com/@carlosalbertoff/mlp-classifier-526978d1c638>. Acesso em: 18 ago. 2025.

GARCIA, Gabriel Lino. **Detecção de Fake News Utilizando Aprendizado de Máquina**. 2023. 79 f. Dissertação (Mestrado em Ciências) – Universidade Estadual Paulista (UNESP), Faculdade de Ciências, Bauru, 2023.

GOOGLE CLOUD. **What is artificial intelligence (AI)?** [S.l.]: Google Cloud, 2025. Disponível em: <https://cloud.google.com/learn/what-is-artificial-intelligence?hl=pt-BR>. Acesso em: 18 ago. 2025.

GONZALEZ, Rafael C.; WOODS, Richard E. **Processamento digital de imagens**. 4. ed. São Paulo: Pearson, 2018.

K&F CONCEPT. **O que é a webcam?** [S.l.]: K&F Concept, 2025. Disponível em: https://br.kentfaith.com/article_o-que-%C3%89-a-webcam_2204?srsItd=AfmBOopEBiOhTTPyQcEnSAfBsOyWR96SLg0eD3WwsZlnZPjm-YIDBV4. Acesso em: 18 ago. 2025.

KRISCHANSKI, Adilson et al. **Ferramentas para controle supervisor e aplicações na robótica móvel: comportamentos emergentes em sistemas multirroboóticos através de**

técnicas de inteligência artificial. In: SEMINÁRIO DE INICIAÇÃO CIENTÍFICA DA UNIVERSIDADE DO ESTADO DE SANTA CATARINA, 33., 2023, Joinville. Anais [...]. Joinville: UDESC, 2023. Disponível em:

http://udesc.br/arquivos/udesc/id_cpmenu/16925/FERRAMENTAS_PARA_CONTROLAR_SU_PORVIAO_E_APLICAR_NA_ROBOTICA_MELHORES_COMPORTAMENTOS_EMERGENCIAIS_EM_16950446100827_16925.pdf. Acesso em: 18 ago. 2025.

LAMEIRA, Lucas Meurer. **Sistema de inspeção visual de placas de circuito impresso.** 2020. Trabalho de Conclusão de Curso (Graduação em Engenharia Elétrica) – Universidade Federal do Rio Grande do Sul, Escola de Engenharia, Departamento de Engenharia Elétrica, Porto Alegre, 2020.

LOBO, L. C. **Inteligência artificial e medicina.** Revista Brasileira de Educação Médica, v. 41, n. 2, 2017. Disponível em: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0100-55022017000200185&lng=pt&nrm=iso&tlng=pt. Acesso em: 18 ago. 2025.

LUGLI, Alexandre Baratella; SANTOS, Max Mauro Dias. **Redes industriais para automação industrial: AS-I, Profibus e Profinet.** São Paulo: Érica, 2015.

MICHEL, N. et al. **Sistema inteligente de supervisão e controle de capacidade em processos industriais: integração de SCADA, IA e aprendizado de máquina.** Aracê, v. 6, n. 4, p. 16090–16112, 18 dez. 2024.

MOCHUAN DRIVES. **Integrando recursos de aprendizado de máquina e IA em aplicações de controladores PLC industriais.** [S.l.]: Mochuan Drives, 2025. Disponível em: <https://www.mochuan-drives.com/pt/a-news-integrating-machine-learning-and-ai-capabilities-in-industrial-plc-controller-applications>. Acesso em: 18 ago. 2025.

MONTERE ELÉTRICA. **Quadro de distribuição de energia: como garantir segurança e eficiência elétrica.** [S.l.]: Montere Elétrica, 2025. Disponível em: <https://www.montereletrica.com.br/blog/categorias/artigos/quadro-de-distribuicao-de-energia-como-garantir-seguranca-e-eficiencia-eletrica>. Acesso em: 18 ago. 2025.

NOGUEIRA, Thiago A. **Redes de Comunicação para Sistemas de Automação Industrial**. Monografia de conclusão de curso. Ouro Preto, 2009. Disponível em: www.em.ufop.br/cecau/monografias/2009/THIAGO%20AUGUSTO.pdf. Acesso em: 18 ago. 2025.

OPENCV. Flags for video I/O — **OpenCV 4.0.0**. [S.l.]: OpenCV.org, 2025. Disponível em: https://docs.opencv.org/4.0.0/d4/d15/group__videoio__flags__base.html#gaeb8dd9c89c10a5c63c139bf7c4f5704d. Acesso em: 18 ago. 2025.

OPERACIONAL, U. **Controlador lógico programável**. [S.l.]: [s.n.], [20--]. Disponível em: https://professorcesarcosta.com.br/upload/imagens_upload/Apostila_do_Curso_Clp-1.pdf. Acesso em: 17 ago. 2025.

PEALLAT, J.-M. et al. **What is really inside your AOI?** 2018. Disponível em: <https://lume.ufrgs.br/bitstream/handle/10183/217436/001120788.pdf?sequence=1&isAllowed=y>. Acesso em: 18 ago. 2025.

ROSÁRIO, João M. **Princípios de mecatrônica**. 1. ed., 4. reimpr. São Paulo: Prentice Hall, 2005.

SANTOS, Abel. **Por que otimizar o espaço no painel elétrico industrial?** *Blog +Perto*, ABB Eletrificação, 23 nov. 2023. Disponível em: <https://loja.br.abb.com/blog/post/por-que-otimizar-painel-eletrico-industrial?srsltid=AfmBOorql87VXgIN5tSuTG-3Kswf9w8jll---jX0wuHhmop1xwXCULns>. Acesso em: 19 ago. 2025.

SARAIVA, David Braz. **Sistema de visão inteligente de baixo custo para parque de estacionamento**. 2015. 80 f. Dissertação (Mestrado em Engenharia Elétrica e Eletrônica) – Universidade do Algarve, Gambelas, 2015.

SCIKIT-LEARN. MLPClassifier. [S.l.]: Scikit-learn, 2025. Disponível em: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html. Acesso em: 18 ago. 2025.

SILVA, Bruno Gabriel da. **Atuador linear embarcado de baixo custo**. 2022. Trabalho de Conclusão de Curso (Bacharelado em Engenharia Elétrica) – Universidade Tecnológica Federal do Paraná, Apucarana, 2022.

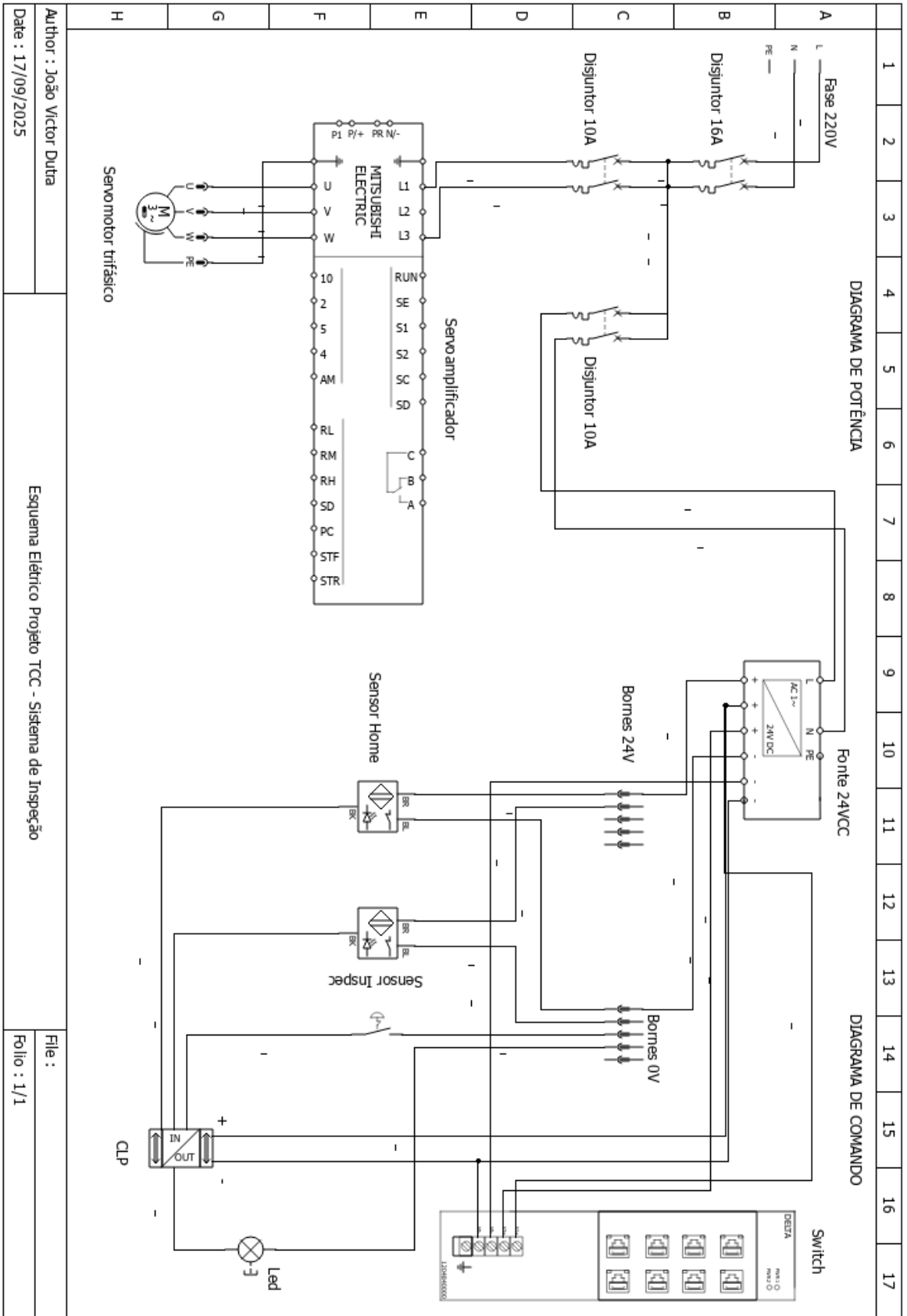
SOORI, Mohsen; AREZOO, Behrooz; DASTRES, Roza. **Artificial intelligence, machine learning and deep learning in advanced robotics: a review**. Cognitive Robotics, v. 3, p. 54–70, 2023. Disponível em: <https://doi.org/10.1016/j.cogr.2023.04.001>. Acesso em: 18 ago. 2025.

STIVANELLO, Maurício Edgar. **Inspeção industrial através de visão computacional**. 2004. 62 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Universidade Regional de Blumenau, Blumenau, 2004.

STURARO, André Zelioli. **Automação de bombas dosadoras com auxílio de controlador lógico programável**. 2009. Monografia (Graduação em Engenharia Mecânica – Automação e Sistemas) – Universidade São Francisco, 2009.

ZANCAN, Marcos Daniel. **Controladores programáveis**. [S.l.]: [s.n.], 2016.

Anexo B – Esquema elétrico do painel



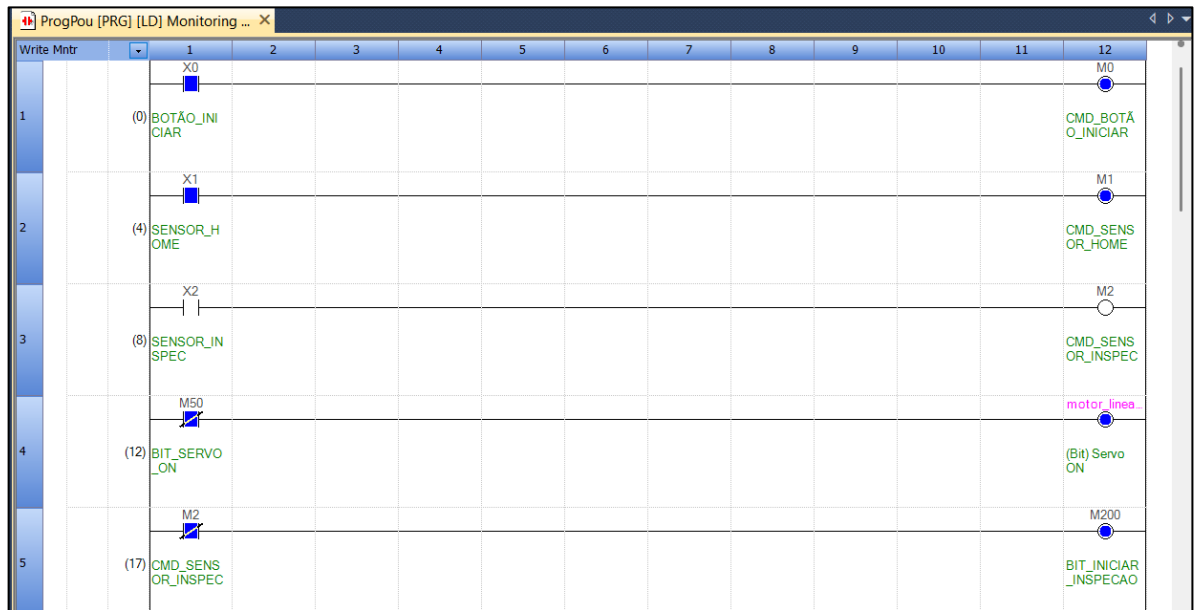
Autor : João Victor Dutra
Date : 17/09/2025

Esquema Elétrico Projeto TCC - Sistema de Inspeção

File :
Folio : 1/1

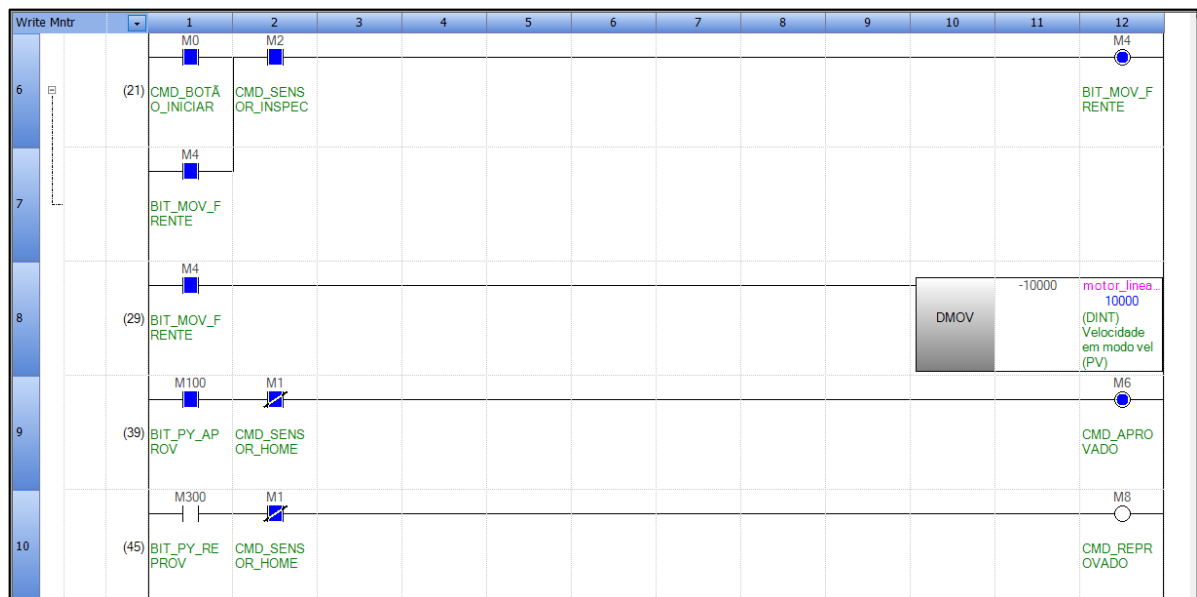
Fonte: O próprio autor

ANEXO C - Estrutura lógica das linhas (1 a 5) em Ladder



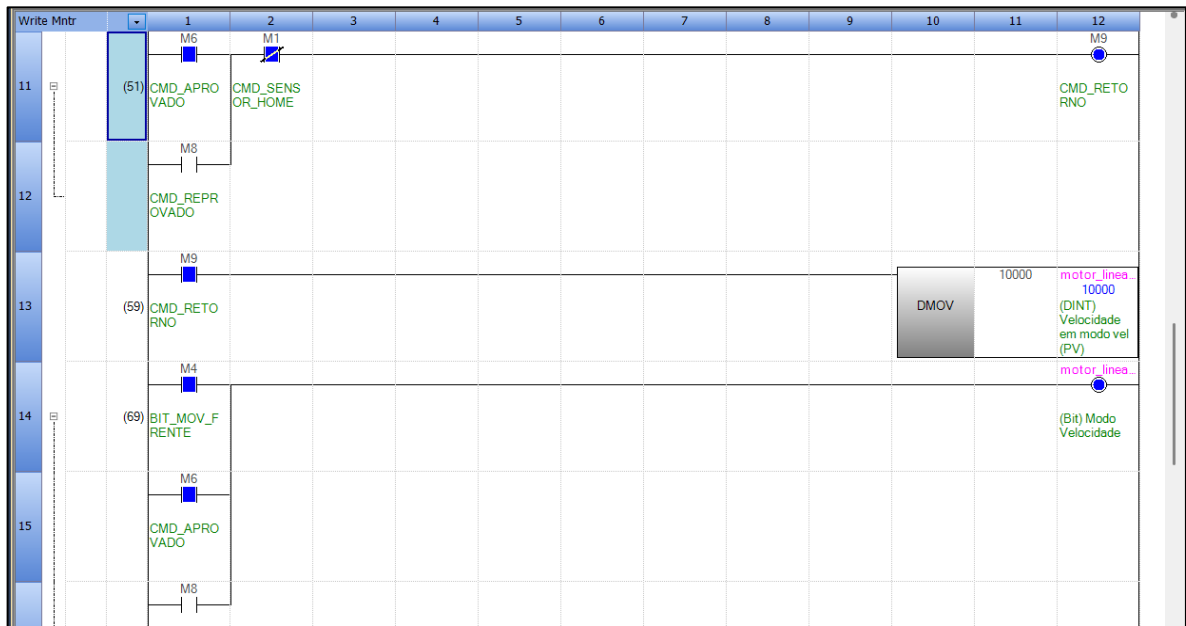
Fonte: O próprio autor

ANEXO D - Estrutura lógica das linhas (6 a 10) em Ladder



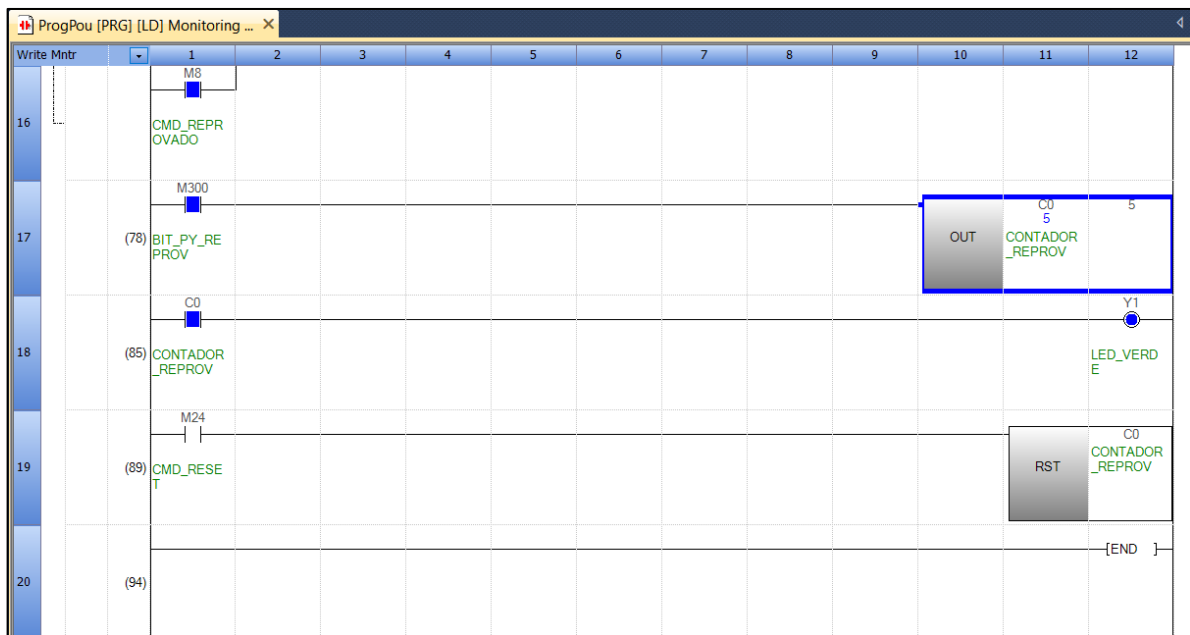
Fonte: O próprio autor

ANEXO E - Estrutura lógica das linhas (11 a 15) em Ladder



Fonte: O próprio autor

ANEXO F- Estrutura lógica das linhas (16 a 20) em Ladder



Fonte: O próprio autor

ANEXO G – Código- fonte do Modelo de Aprendizado de Máquina em Python

```

import os
import cv2
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import classification_report
import joblib

DATASET_DIR = "dataset"
IMG_SIZE = (64, 64)
classes = ["ok", "ng"]

X = []
y = []

for label, class_name in enumerate(classes):
    class_dir = os.path.join(DATASET_DIR, class_name)
    for filename in os.listdir(class_dir):
        path = os.path.join(class_dir, filename)
        img = cv2.imread(path, cv2.IMREAD_GRAYSCALE)
        if img is None:
            continue
        img = cv2.resize(img, IMG_SIZE)
        X.append(img.flatten())
        y.append(label)

X = np.array(X)
y = np.array(y)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

clf = MLPClassifier(hidden_layer_sizes=(100,), max_iter=500)
clf.fit(X_train, y_train)

y_pred = clf.predict(X_test)

print (classification_report(y_test, y_pred, target_names=classes))

joblib.dump(clf, "modelo_inspecao.pkl")

print ("✅ Modelo salvo como 'modelo_inspecao.pkl'")

```

ANEXO H – Código-fonte do processamento de inspeção e integração em Python

```

import numpy as np
import cv2
import time
import joblib
from pymodbus.client import ModbusTcpClient

client = ModbusTcpClient('192.168.3.160', port=502)
m100_endereço = 8292
m200_endereço = 8392
m300_endereço = 8492

CAM_LARGURA = 1920
CAM_ALTURA = 1080
ROI_coodernadas_padrão = (505, 476, 927, 870)
ML_Tamanho = (64, 64)

def aplicação_nitidez(imagem):
    kernel = np.array([[0, -1, 0],
                       [-1, 5, -1],
                       [0, -1, 0]])
    return cv2.filter2D(imagem, -1, kernel)

try:
    modelo = joblib.load("modelo_inspecao.pkl")
    print("✅ Modelo carregado com sucesso.")
except Exception as e:
    print(f"❌ Erro ao carregar modelo: {e}")
    exit()

if not client.connect():
    print("❌ Erro de conexão com o CLP")
    exit()

cap = cv2.VideoCapture(0)
cap.set(cv2.CAP_PROP_FRAME_WIDTH, CAM_LARGURA)
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, CAM_ALTURA)
if not cap.isOpened():
    print("❌ Câmera não disponível.")
    client.close()
    exit()

```

```

print("📧 Sistema de inspeção iniciado")
while True:
    print("🕒 Aguardando M200 do CLP...")
    while True:
        resp = client.read_coils(address=m200_endereço, count=1)
        if not resp.isError() and resp.bits[0]:
            print("✅ Sinal M200 recebido – Iniciando inspeção")
            break
        time.sleep(0.9)

    for _ in range(5):
        cap.read()
    ret, frame = cap.read()
    if not ret:
        print("❌ Erro na captura. Reiniciando ciclo.")
        continue

    frame_nitido = aplicação_nitidez(frame)
    img_gray = cv2.cvtColor(frame_nitido, cv2.COLOR_BGR2GRAY)

    x1, y1, x2, y2 = ROI_coordenadas_padrao
    ROI = img_gray[y1:y2, x1:x2]
    ROI_capturada = cv2.resize(ROI, ML_Tamanho).flatten().reshape(1, -1)

    classe_predita = modelo.predict(ROI_capturada)[0]
    aprovado = classe_predita == 0
    texto = f'{'PEÇA APROVADA' if aprovado else 'PEÇA REPROVADA'}'
    cor = (0, 255, 0) if aprovado else (0, 0, 255)

    frame_resultado = frame.copy()
    cv2.putText(frame_resultado, texto, (50, 80), cv2.FONT_HERSHEY_SIMPLEX, 1.2, cor,
                3)
    cv2.imshow("RESULTADO", frame_resultado)
    cv2.waitKey(10000)
    if cv2.getWindowProperty("RESULTADO", cv2.WND_PROP_VISIBLE) >= 1:
        cv2.destroyWindow("RESULTADO")

    if not aprovado:
        print("❌ PEÇA REPROVADA")
        cv2.imwrite("peca_ng_detectada.jpg", frame_resultado)
        res = client.write_coil(address=m300_endereço, value=True)
        if not res.isError():
            print("🔴 M300 ligada (NG)")

```

```
    time.sleep(6)
    client.write_coil(address=m300_endereço, value=False)
    print("🔴 M300 desligada")
else:
    print("❌ Erro ao acionar M300")
else:
    print("✅ PEÇA APROVADA")
    cv2.imwrite("peca_OK.jpg", frame_resultado)
    res = client.write_coil(address=m100_endereço, value=True)
    if not res.isError():
        print("🔴 M100 ligada (OK)")
        time.sleep(6)
        client.write_coil(address=m100_endereço, value=False)
        print("🔴 M100 desligada")
    else:
        print("❌ Erro ao acionar M100")

key = cv2.waitKey(1)
if key == ord('q'):
    break

cap.release()
cv2.destroyAllWindows()
```