

**UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL
FACULDADE DE ENGENHARIAS, ARQUITETURA E
URBANISMO E GEOGRAFIA**

FERNANDO HENRIQUE CASTILHO SILVA FLORES

**AUTOMAÇÃO DE UMA CÉLULA LOGÍSTICA UTILIZANDO CLP S7-1200 E
VALIDAÇÃO EM AMBIENTE VIRTUAL**

**CAMPO GRANDE
2025**

FERNANDO HENRIQUE CASTILHO SILVA FLORES

**AUTOMAÇÃO DE UMA CÉLULA LOGÍSTICA UTILIZANDO
CLP S7-1200 E VALIDAÇÃO EM AMBIENTE VIRTUAL**

Trabalho de conclusão de curso
apresentado como exigência para obtenção
do grau de Bacharelado em Engenharia
Elétrica da Universidade Federal de Mato
Grosso do Sul – UFMS

Orientador: Prof. Dr. Valmir
Machado Pereira.

CAMPO GRANDE
2025



Serviço Público Federal
Ministério da Educação
Fundação Universidade Federal de Mato Grosso do Sul



AUTOMAÇÃO DE UMA CÉLULA LOGÍSTICA UTILIZANDO CLP S7-1200 E VALIDAÇÃO EM AMBIENTE VIRTUAL

Fernando Henrique Castilho Silva Flores

Trabalho de Conclusão de Curso apresentado como exigência para obtenção do grau de Bacharelado em Engenharia Elétrica da Universidade Federal de Mato Grosso do Sul – UFMS considerado **aprovado**.

Banca examinadora:

Prof. Dr. Valmir Machado Pereira (Orientador)
Prof. Dr. Paulo Irineu Koltermann
Prof. Dr. Guilherme Augusto Defalque

Campo Grande, 03 de dezembro de 2025.

**NOTA
MÁXIMA
NO MEC**

**UFMS
É 10!!!**



Documento assinado eletronicamente por **Valmir Machado Pereira, Professor do Magisterio Superior**, em 03/12/2025, às 17:38, conforme horário oficial de Mato Grosso do Sul, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).

**NOTA
MÁXIMA
NO MEC**

**UFMS
É 10!!!**



Documento assinado eletronicamente por **Paulo Irineu Koltermann, Professor do Magisterio Superior**, em 03/12/2025, às 17:40, conforme horário oficial de Mato Grosso do Sul, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).

**NOTA
MÁXIMA
NO MEC**

**UFMS
É 10!!!**



Documento assinado eletronicamente por **Guilherme Augusto Defalque, Técnico de Laboratório Área**, em 03/12/2025, às 17:40, conforme horário oficial de Mato Grosso do Sul, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufms.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **6071597** e o código CRC **169B8AC7**.

FACULDADE DE ENGENHARIAS, ARQUITETURA E URBANISMO E GEOGRAFIA

Av Costa e Silva, s/nº - Cidade Universitária

Fone:

CEP 79070-900 - Campo Grande - MS

AGRADECIMENTOS

Agradeço primeiramente a Deus por me conceder saúde, força e resiliência ao longo de toda a jornada acadêmica, e por me guiar até a conclusão deste importante ciclo.

A conclusão deste trabalho é o resultado de muito esforço, mas também do apoio de muita gente. Quero agradecer de coração a todos que me ajudaram a chegar até aqui.

Em primeiro lugar, à minha família, por todo o amor, carinho e apoio que me deram durante todos esses anos. Um agradecimento especial aos meus pais, Osney Castilho Flores e Ana Maria da Silva, por sempre investirem na minha educação, me apoiarem e me incentivarem em todas as minhas escolhas. À minha namorada, Muriel Netto Rodrigues, agradeço muito pelo incentivo e apoio, principalmente nesta reta final. Você foi meu refúgio e minha motivação para sempre buscar o melhor. Aos meus colegas e amigos, tanto dentro quanto fora da faculdade, sou grato pela parceria e pela troca de conhecimento e ajuda que tivemos. A amizade de vocês foi muito importante nesta jornada.

Ao meu orientador, Prof. Dr. Valmir Machado Pereira, expresso minha profunda gratidão pela ajuda e orientação técnica deste trabalho. Sua paciência e disponibilidade foram essenciais para garantir o rigor e a conclusão desse trabalho. Estendo meus agradecimentos ao Prof. Dr. Guilherme Augusto Defalque, por sua contribuição e auxílio com orientações que trouxeram uma nova perspectiva para o desenvolvimento e o fechamento do projeto.

Por fim, agradeço à empresa Real Games, desenvolvedora do software Factory I/O. Sou grato pelo suporte tecnológico concedido, que incluiu o fornecimento de uma chave de licença estendida, um apoio fundamental que possibilitou o teste da lógica de controle e a conclusão do Comissionamento Virtual em tempo hábil. Agradeço também à UFMS e a todos os professores da graduação que, com dedicação e paciência, transmitiram o conhecimento necessário e contribuíram diretamente para a minha formação profissional e pessoal ao longo desses anos.

RESUMO

O presente trabalho aborda a automação de uma célula logística (classificação, paletização e AS/RS), um tema de grande relevância no cenário da Logística 4.0. A validação da lógica de controle para sistemas complexos como este, quando realizada diretamente no ambiente físico, acarreta alto risco operacional e elevados custos. Desta forma, a metodologia de Comissionamento Virtual é fundamental, pois permite a mitigação de falhas e a otimização de parâmetros em um ambiente seguro. O objetivo do estudo é desenvolver e validar esta automação, utilizando uma arquitetura *Hardware-in-the-Loop* (HiL). Para tal, um Controlador Lógico Programável (CLP) Siemens S7-1200 real é integrado a um Gêmeo Digital de alta fidelidade (Factory I/O) para testar a lógica de controle. A classificação de caixas utiliza Controle Analógico com lógica de Faixa Morta para precisão dimensional. O sequenciamento da paletização garante a Rastreabilidade através da sincronização e endereçamento indireto, assegurando a organização dos dados. O subsistema de armazenamento (*Automated Storage and Retrieval System - AS/RS*) é controlado por uma Máquina de Estados Finitos (FSM), que garante o determinismo operacional, e o posicionamento preciso do transelevador é validado por uma rotina de Faixa Morta de 0.001 V no eixo analógico. Os resultados comprovam a eficácia da simulação para otimizar os parâmetros dinâmicos e reduzir riscos na implementação final de sistemas de logística 4.0.

Palavras-chaves: Automação Industrial, Comissionamento Virtual, Gêmeo Digital; Sistema de Armazenagem Automatizado (AS/RS), Máquina de Estados Finitos.

ABSTRACT

This work addresses the automation of a logistics cell (sorting, palletizing, and AS/RS), a topic of great relevance in the logistics 4.0 scenario. Validating the control logic for complex systems like this, when performed directly in the physical environment, entails high operational risk and elevated costs. Therefore, the Virtual Commissioning methodology is fundamental, as it allows for the mitigation of failures and the optimization of parameters in a secure environment. The objective of this study is to develop and validate this automation using a Hardware-in-the-Loop (HiL) architecture. To this end, a real Siemens S7-1200 Programmable Logic Controller (PLC) is integrated with a high-fidelity Digital Twin (Factory I/O) to test the control logic. Box sorting utilizes Analog Control with Deadband logic for dimensional precision. The palletizing sequencing ensures Traceability through synchronization via and indirect addressing, guaranteeing data organization. The storage subsystem (Automated Storage and Retrieval System — AS/RS) is controlled by a Finite State Machine (FSM), which guarantees operational determinism, and the precise positioning of the stacker crane is validated by a Deadband routine of 0.001 V on the analog axis. The results prove the effectiveness of the simulation in optimizing dynamic parameters and reducing risks in the final implementation of Logistics 4.0 systems.

- **Keywords:** Industrial Automation, Virtual Commissioning, Digital Twin, Automated Storage and Retrieval System (AS/RS), Finite State Machine.

LISTA DE FIGURAS

Figura 1 - Comparaçao entre painel de relés e painel com CLP	18
Figura 2 - Arquitetura interna básica de um Controlador Lógico Programável.....	20
Figura 3 - Linguagens de programação do CLP	24
Figura 4 - Exemplos de painéis de (IHM) industriais.	27
Figura 5 - Pilares da Industria 4.0.....	28
Figura 6 - S7- 1200.....	33
Figura 7 - KTP700 Basic	34
Figura 8 - SCALANCE XB005.....	35
Figura 9 – Setup de validação Hardware-in-the-Loop (HiL)	36
Figura 10 - Configuração no TIA Portal	38
Figura 11 - Vista isométrica do Gêmeo Digital da Célula Logística.....	40
Figura 12 - Feixes de detecção do Sensor de Barreira (<i>Light Array</i>)	42
Figura 13- Mecanismo de desvio do Classificador de Rodas (<i>Pop-up Wheel Sorter</i>)	43
Figura 14 - Sensores e atuadores internos do Paletizador	45
Figura 15 - Detalhes dos eixos e atuadores do Transelevador	47
Figura 16 – Fluxograma do processo de separação e classificação de caixas	50
Figura 17 – Fluxograma da lógica de sequenciamento da paletizaçäodo na Figura	52
Figura 18 – Fluxograma do ciclo de armazenamento do sistema AS/RS	54
Figura 19 - Estrutura de organização dos blocos do projeto no software TIA Portal	56
Figura 20 - Lógica Ladder para classificação de caixas por altura	61
Figura 21 – Lógica do desviador utilizando temporizadores TON e TOF	62
Figura 22 – Exemplo de lógica de Controle da abertura da placa e alinhamento	65
Figura 23 – Lógica Ladder para registro de dados RFID	67
Figura 24 - Estrutura do Bloco de Dados "Mapa" (WMS) configurada no TIA Portal	69
Figura 25 – Lógica Ladder para validação de posição analógica com faixa morta.....	74
Figura 26 - IHM exibindo o mapa visual de ocupação do AS/RS em estado inicial	76
Figura 27 - Tela de confirmação da ordem de recuperação de palete específico	77
Figura 28 - Vinculação das tags da IHM aos campos de exibição dos IDs.....	78
Figura 29 – Instabilidade dinâmica observada em velocidades	81
Figura 30 – Lógica Ladder para validação de posicionamento do palete	85
Figura 31 - Tela de consulta do inventário	87
Figura 32 – Lógica Ladder de segurança: Monitoramento e disparo de alarmes	90

Figura 33 – Rotina de Reset e recuperação de falhas na Máquina de Estados.....	93
Figura 34 - Interface HMI exibindo status em tempo real	94
Figura 35 - Estrutura física das 54 vagas.....	94

LISTA DE TABELAS

Tabela 1 - Sensores e Atuadores do Paletizador (<i>Palletizer</i>)	46
Tabela 2 - Sensores e Atuadores do Transelevador (<i>Stacker Crane</i>)	48
Tabela 3 – Mapeamento e Função dos Componentes-Chave do Gêmeo Digital	49
Tabela 4 - Mapeamento de Blocos de Programa no TIA Portal.....	57
Tabela 5 - Estrutura do UDT (User-Defined Type) "ty_Coordenada" no DB "Mapa"	70
Tabela 6 – Função do estados na AR/RS	71
Tabela 7 – Protocolo de busca para cada modo.....	72
Tabela 8 - Comparativo de Comportamento em Função da Velocidade da Esteira.....	81
Tabela 9 - Resultados da Validação do Projeto em Relação aos Objetivos Propostos	96

LISTA DE SÍMBOLOS

Bit Binary Digit
m Metros
Mbps Megabits
mm Milímetros
mV Milivolts
S Segundo(s)
V Volts
VCA Tensão em Corrente Alternada
VCC Tensão em Corrente Contínua

LISTA DE ABREVIATURAS

AS/RS	Automated Storage and Retrieval System (Sistema de Armazenagem e Recuperação Automatizados)
CLP	Controlador Lógico Programável
CNI	Confederação Nacional da Indústria
CPU	Central Processing Unit (Unidade Central de Processamento)
CPS	Sistemas Ciber-Físicos
DB	Data Block (Bloco de Dados)
FBD	Function Block Diagram (Diagrama de Blocos de Função)
FB	Function Block (Bloco de Função)
FC	Function (Função)
FSM	Finite State Machine (Máquina de Estados Finitos)
Hart	Highway Addressable Remote Transducer
HiL	Hardware-in-the-Loop
HMI	Human-Machine Interface (Interface Homem Máquina)
IA	Inteligência Artificial
ID	Identificador
IEC	International Electrotechnical Commission
IHM	Interface Homem Máquina
I/O	Input/Output (Entrada/Saída)
IL	Instruction List (Lista de Instruções)
IP	Internet Protocol
IoT	Internet das Coisas
LCD	Liquid Crystal Display
LD	Ladder Diagram (Diagrama de Contatos)
LED	Light Emitting Diode
MODICON	Modular Digital Controller
NA	Normalmente Aberto
NF	Normalmente Fechado
OB	Organization Block (Bloco de Organização)
PGM	Pulse General Module
PID	Proporcional Integral Derivativo
PLC	Programmable Logic Controller (Controlador Lógico Programável)
P_TRIGGER	Positive Edge Trigger (Detector de Borda de Subida)
SCL	Structured Control Language
SFC	Sequential Function Chart (Diagrama Funcional Sequencial)
ST	Structured Text (Texto Estruturado)
TIA	Totally Integrated Automation
TOF	Timer Off-Delay (Temporizador com Atraso na Desligação)
TON	Timer On-Delay (Temporizador com Atraso na Ligação)
UDT	User-Defined Type (Tipo de Dado de Usuário)

UX	User Experience
VCA	Tensão de Corrente Alternada
VCC	Tensão de Corrente Contínua
WMS	Warehouse Management System (Sistema de Gerenciamento de Armazém)

SUMÁRIO

1.	INTRODUÇÃO.....	15
2.	FUNDAMENTAÇÃO TEÓRICA	18
2.1.	A EVOLUÇÃO DA AUTOMAÇÃO INDUSTRIAL.....	18
2.2.	O CONTROLADOR LÓGICO PROGRAMÁVEL (CLP)	19
2.2.1.	Definição Geral da Arquitetura	19
2.2.2.	Ciclo de Varredura (Scan).....	20
2.2.3.	Módulos de Entrada e Saída.....	21
2.3.	TRANSDUTORES E ATUADORES: A INTERFACE COM O MUNDO FÍSICO	22
2.4.	LINGUAGEM DE PROGRAMAÇÃO.....	23
2.5.	LADDER	24
2.5.1.	Origem e Estrutura.....	24
2.5.2.	Simbologia Básica	25
2.6.	INTERFACE HOMEM-MÁQUINA (IHM).....	26
2.7.	A INDÚSTRIA 4.0 E A LOGÍSTICA INTELIGENTE (LOGÍSTICA 4.0).....	27
2.8.	METODOLOGIAS DE DESENVOLVIMENTO DA INDÚSTRIA 4.0	29
2.8.1.	Gêmeo Digital (<i>Digital Twin</i>)	29
2.8.2.	Comissionamento Virtual (<i>Virtual Commissioning</i>)	30
2.9.	MÁQUINA DE ESTADOS FINITOS	30
3.	METODOLOGIA.....	32
3.1.	ARQUITETURA TECNOLÓGICA E FERRAMENTAL	32
3.1.1.	Hardware	32
3.1.2.	Infraestrutura de Comunicação.....	35
3.2.	MODELAGEM DO GÊMEO DIGITAL DA CÉLULA LOGÍSTICA	39
3.2.1.	Componentes Básicos e de Fluxo	40

3.2.2.	Componentes de Operação e Supervisão.....	41
3.2.3.	Sensores de Detecção e Classificação	41
3.2.4.	Componentes de Processo e Rastreabilidade.....	43
3.2.5.	Sistema de Armazenagem Automatizado (AS/RS)	44
3.3.	DESCRIÇÃO FUNCIONAL DO SISTEMA AUTOMATIZADO	49
3.3.1.	Etapa 1: Separação e Classificação de Caixas	49
3.3.2.	Etapa 2: Paletização e Rastreabilidade	51
3.3.3.	Etapa 3: Armazenamento e Recuperação Automatizado (AS/RS).....	53
3.4.	ESTRUTURA E MODULARIZAÇÃO DO PROGRAMA	55
3.4.1.	Tipos de Blocos	55
3.4.2.	Blocos de Instrução e Funções Usadas no Programa	58
3.4.3.	Lógica de Controle: Etapa 1 – Separação e Classificação.....	60
3.4.4.	Lógica de Controle: Etapa 2 – Paletização e Rastreabilidade RFID	63
3.4.5.	Lógica de Controle: Etapa 3 – Sistema AS/RS e Máquina de Estados (FSM)	68
4.	RESULTADOS EXPERIMENTAIS E DISCUSSÃO TÉCNICA	79
4.1.	SUBSISTEMA DE SEPARAÇÃO E CLASSIFICAÇÃO	79
4.1.1.	Comportamento Físico e Ajuste de Velocidade.....	80
4.1.2.	Precisão da Instrumentação	82
4.1.3.	Lógica de Desvio.....	83
4.2.	SUBSISTEMA DE PALETIZAÇÃO	84
4.2.1.	Precisão de Posicionamento e Controle do Elevador.....	85
4.2.2.	Gestão de Fluxo e Criação de brechas.....	86
4.2.3.	Rastreabilidade RFID e Integridade de Dados.....	86

4.2.4.	Estabilidade de Carga e Lógica de Contadores (Cascata)	88
4.2.5.	Sistemas de Segurança e Alarmes	89
4.3.	SUBSISTEMA AS/RS	90
4.3.1.	Comparativo: Lógica Sequencial vs. Máquina de Estados (FSM).....	91
4.3.2.	Controle de Posição Analógico e a "Faixa Morta" de 0.001V	91
4.3.3.	Algoritmo de Busca de Vagas (WMS).....	92
4.3.4.	Recuperação de Falhas e Rotina de Reset	92
4.4.	INTEGRAÇÃO COM INTERFACE HOMEM-MÁQUINA (IHM).....	93
4.5.	ANALISE DOS RESULTADOS	95
4.5.1.	Objetivos Gerais.....	95
4.5.2.	Objetivos Específicos.....	95
5.	CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS	98
	REFERÊNCIAS	99

1. INTRODUÇÃO

A busca contínua pela otimização de tarefas é uma força motriz do desenvolvimento humano e tecnológico. Desde a aplicação de princípios mecânicos, como alavancas e roldanas, para multiplicar a força humana, até a automação de linhas de produção, o objetivo central sempre foi aumentar a eficiência, a segurança e, consequentemente, a produtividade. No cenário industrial contemporâneo, essa busca atinge um novo patamar com o advento da Quarta Revolução Industrial.

Mais conhecida como Indústria 4.0, um conceito que emergiu por volta de 2011, esta nova era é caracterizada pela fusão entre os mundos físico e digital. Por meio da integração de tecnologias como Inteligência Artificial (IA), Internet das Coisas (IoT), Gêmeos Digitais (*Digital Twins*) e robótica avançada, os processos industriais e logísticos estão sendo radicalmente transformados (ALTUS.[s.d.]). O pilar dessa revolução é a criação de "fábricas inteligentes", onde máquinas, sistemas e processos são interligados para operar de forma autônoma, coletando e analisando dados em tempo real para tomar decisões descentralizadas.

Enquanto os sistemas de armazenamento tradicionais enfrentam gargalos críticos decorrentes de processos manuais como a lentidão na movimentação, a alta incidência de erros humanos e a subutilização do espaço vertical, as soluções da Indústria 4.0 oferecem uma mudança de paradigma. A implementação de Sistemas de Armazenagem e Recuperação Automatizados (AS/RS) contrapõe essas ineficiências com agilidade, precisão no controle de inventário e máxima ocupação espacial. No entanto, a transição para essa tecnologia impõe seus próprios desafios, notadamente os altos custos e riscos operacionais associados ao desenvolvimento e testes em ambiente físico.

Este trabalho propõe o desenvolvimento e a validação da automação de um Sistema de Armazenagem e Recuperação Automatizado (AS/RS) por meio da metodologia de Comissionamento Virtual. Ao utilizar um Gêmeo Digital que é uma réplica virtual fiel do sistema físico, é possível testar e depurar toda a lógica de controle de forma segura e eficiente, reduzindo drasticamente o tempo e os custos do projeto (GRIEVES,2014). A relevância deste estudo reside na aplicação prática de conceitos da Indústria 4.0 para solucionar um problema logístico real, demonstrando como a abordagem de Comissionamento Virtual não apenas acelera o ciclo de desenvolvimento e mitiga riscos, mas também viabiliza a implementação de sistemas de automação integrados e complexos de forma mais acessível e segura.

1.1. OBJETIVOS

1.1.1. O objetivo geral

Desenvolver e validar a automação de uma célula logística (separação, paletização e armazenamento) com CLP Siemens S7-1200 e Comissionamento Virtual via Gêmeo Digital (Factory I/O).

1.1.2. Objetivos Específicos

- Desenvolver a lógica de controle para a separação automática de caixas por tamanho e para o processo sequencial de paletização.
- Desenvolver a lógica de controle para o posicionamento preciso do transelevador do sistema AS/RS, utilizando uma estrutura de Máquina de Estados.
- Implementar um sistema de gestão de inventário em tempo real, utilizando tecnologia RFID para a rastreabilidade das caixas desde a paletização até o armazenamento.
- Projetar e integrar uma Interface Homem-Máquina (IHM) para a supervisão e o monitoramento visual do sistema.
- Integrar um painel de controle físico ao ambiente virtual para permitir a operação e o teste de funcionalidades.
- Validar a funcionalidade integrada de todo o sistema no ambiente de simulação, comprovando a eficácia da metodologia de Comissionamento Virtual para o desenvolvimento de sistemas de automação complexos.

1.2. ORGANIZAÇÃO DO TRABALHO

Este trabalho está estruturado da seguinte maneira:

- O Capítulo 1 apresenta a Introdução, contextualizando o tema da automação logística na Indústria 4.0, a problemática do estudo e os objetivos do trabalho.
- O Capítulo 2 aborda a Fundamentação Teórica, revisando os conceitos essenciais para o projeto, desde a evolução da automação e CLP até as metodologias de Gêmeo Digital

e Comissionamento Virtual.

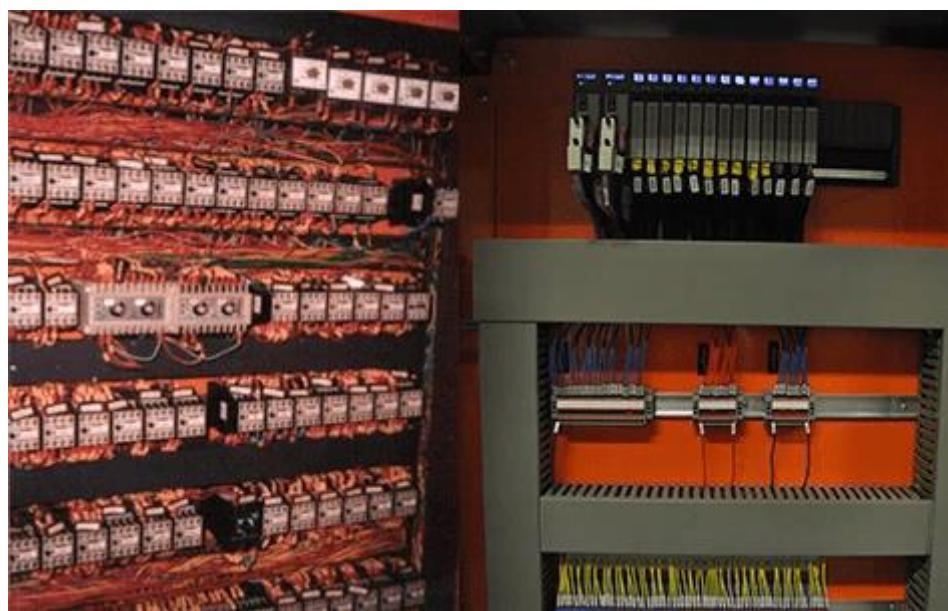
- O Capítulo 3 descreve a Metodologia, detalhando os materiais (hardware e software) e o passo a passo do desenvolvimento do sistema virtual, incluindo a lógica de controle para a separação, paletização e armazenamento.
- O Capítulo 4 apresenta os Resultados e a Discussão, onde são demonstrados os testes de simulação que validam o funcionamento da célula logística automatizada e a eficácia da metodologia de Comissionamento Virtual.
- O Capítulo 5 apresenta as Considerações Finais, resumindo as conclusões do estudo, verificando o atendimento aos objetivos e propondo sugestões para trabalhos futuros.

2. FUNDAMENTAÇÃO TEÓRICA

2.1. A EVOLUÇÃO DA AUTOMAÇÃO INDUSTRIAL

A automação industrial moderna é o resultado de uma evolução tecnológica contínua, marcada por várias transições. No séc. XVIII, a Revolução Industrial começou a promover a substituição da mão de obra manual pela força mecânica, logo a tecnologia avançou e foram continuadas as substituições, sistemas manuais para sistemas mecânicos para sistema de controle eletromecânicos e por fim para os sistemas eletrônicos programáveis. Em meados do século XX, a lógica de controle era predominantemente implementada por meio de painéis a relés (INETEC,[s.d]). Esses sistemas, embora revolucionários para a época e essenciais para as linhas de montagem, como as de automobilismo, eram compostos por centenas ou milhares de relés físicos, interligados por uma complexa malha de fiação para executar uma sequência de operações predeterminada, conforme demonstrado na Figura 1(INETEC, [s.d]).

Figura 1 - Comparaçao entre painel de relés e painel com CLP



Fonte: Inetec (s.d.)

A lógica do processo era, portanto, intrinsecamente definida pelo hardware. Essa abordagem apresenta limitações significativas que se tornaram gargalos para a produtividade, pois os sistemas de relés eram muito inflexíveis, em que para alterar algo no processo exigia

muito tempo para alterar as ligações do relés, além de vários outros problemas. Pensando nisso, no final da década de 1960 a empresa de automobilismo General Motors (GM) lançou um desafio ao mercado. Eles redigiram uma especificação detalhada para um novo tipo de controlador que deveria substituir os painéis de relés (ALTUS, [s.d]). As exigências eram claras: o novo dispositivo precisava ser:

- Programável e reprogramável (a lógica poderia ser alterada via software, não por fiação).
- Robusto para suportar o ambiente hostil do chão de fábrica.
- Modular e de fácil manutenção.

Assim a empresa de Massachusetts chamada Bedford Associates decidiu aceitar o desafio da GM. A equipe era liderada por um engenheiro chamado Richard "Dick" Morley, que hoje é amplamente considerado o "pai do CLP" (MACHADO; FREITAS; JUNIOR, 2021).

Morley e sua equipe na Bedford Associates desenvolveram um dispositivo que atendia a todas as especificações da GM. Eles o chamaram de MODICON 084 (um acrônimo para *Modular Digital Controller*, e 084 era o número do projeto). Este foi o primeiro Controlador Lógico Programável (CLP) ou *Programmable Logic Controller* (PLC) comercialmente viável do mundo (MACHADO; FREITAS; JUNIOR, 2021).

2.2. O CONTROLADOR LÓGICO PROGRAMÁVEL (CLP)

O Controlador Lógico Programável (CLP) é o "cérebro" da vasta maioria dos sistemas de automação industrial modernos, atuando como o elo central entre os dispositivos de campo e a lógica de controle do processo.

2.2.1. Definição Geral da Arquitetura

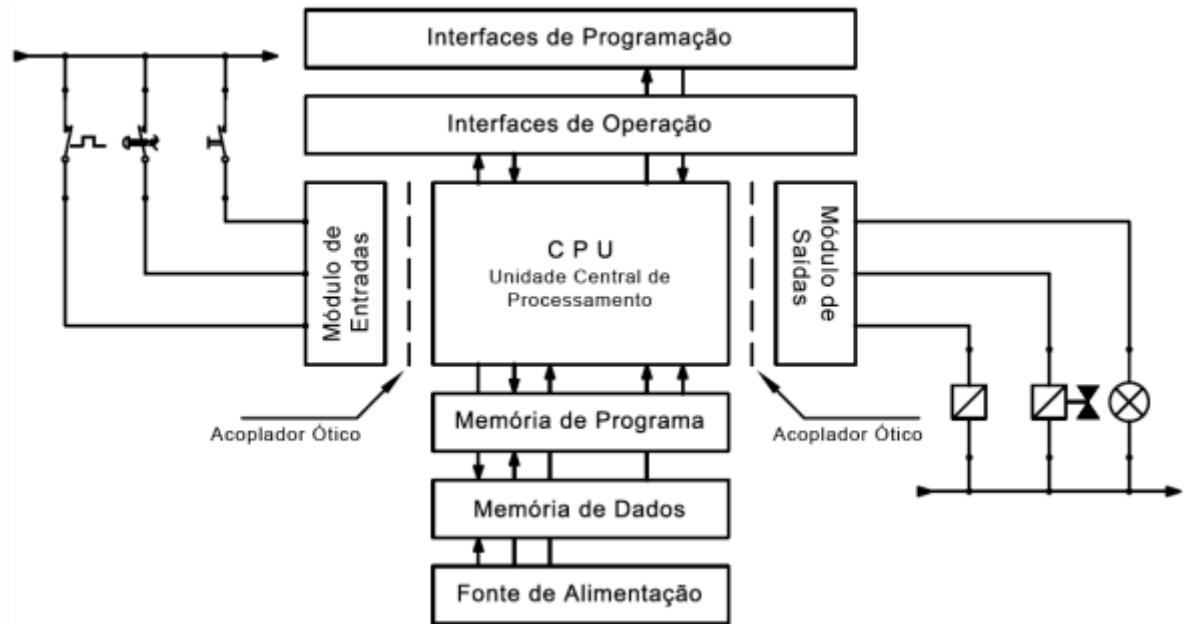
Um CLP pode ser definido como um computador industrial robusto, projetado especificamente para operar em ambientes hostis, suportando vibrações, temperaturas extremas e ruído elétrico, e para executar, de forma confiável e em tempo real, a lógica de controle de máquinas e processos. Sua arquitetura fundamental é composta por quatro elementos principais (GEORGINI, 2009):

- Fonte de Alimentação: Converte a tensão da rede elétrica (110/220 VCA) para as tensões de corrente contínua (24 VCC, 5 VCC) necessárias para alimentar a CPU e os demais circuitos eletrônicos.

- Unidade Central de Processamento (CPU): É o componente de inteligência do CLP. Contém o microprocessador, que executa o programa do usuário, e as memórias.
- Memórias: O CLP utiliza diferentes tipos de memória para suas operações:
 - Memória de Programa: Armazena o programa de aplicação desenvolvida pelo usuário, contendo a lógica de controle.
 - Memória de Dados: Armazena os valores de variáveis, temporizadores, contadores e outros dados que são manipulados durante a execução do programa.
 - Memória Imagem de Entradas e Saídas (*I/O Image Table*): Uma área de memória RAM crucial que armazena uma cópia do estado de todas as entradas e saídas físicas.
 - Módulos de Entrada e Saída (I/O): Formam a interface física entre a CPU e o processo. Eles convertem os sinais elétricos provenientes dos sensores em níveis lógicos que a CPU pode entender e, inversamente, convertem os sinais lógicos da CPU em energia para acionar os atuadores(GEORGINI, 2009).

A Figura 2 ilustra essa arquitetura interna básica e a interligação entre os componentes citados.

Figura 2 - Arquitetura interna básica de um Controlador Lógico Programável.



Fonte: Engelogic (s.d.).

2.2.2. Ciclo de Varredura (Scan)

O funcionamento de um CLP é inherentemente cíclico e determinístico, um princípio

conhecido como ciclo de varredura ou *scan*. Em CLPs de nível industrial esse ciclo é executado repetidamente pela CPU milhares de vezes por segundo e consiste em três etapas fundamentais (FACTORY AUTOMATION, 2023):

- Leitura das Entradas: No início de cada ciclo, a CPU lê o estado de todos os terminais de entrada física (sensores, botões etc.) e armazena essa informação (ligado/desligado, ou valores analógicos) na Memória Imagem de Entrada.
- Execução do Programa: A CPU executa o programa do usuário, instrução por instrução, do início ao fim. Durante essa etapa, a lógica opera exclusivamente sobre os dados contidos na Memória Imagem de Entrada, que funciona como um "instantâneo" (*snapshot*) do estado do processo no início do ciclo. Os resultados da lógica são escritos na Memória Imagem de Saída.
- Atualização das Saídas: Ao final da execução do programa, a CPU transfere o conteúdo da Memória Imagem de Saída para os terminais de saída física, energizando ou desenergizando os atuadores (motores, válvulas, etc.). (FACTORY AUTOMATION, 2023)

A utilização da memória imagem é uma escolha arquitetônica deliberada que garante o comportamento determinístico do sistema. Ao desacoplar a execução da lógica da leitura direta e contínua das entradas físicas, o CLP assegura que o programa opere sobre um conjunto de dados consistente e estável durante todo um ciclo. Se uma entrada mudar de estado no meio da execução do programa, essa mudança só será reconhecida no próximo ciclo de varredura. Isso evita condições de corrida (*race conditions*) e comportamentos imprevisíveis, garantindo a alta confiabilidade exigida em aplicações de controle industrial (FACTORY AUTOMATION, 2023).

2.2.3. Módulos de Entrada e Saída

Os módulos de I/O são os componentes que conectam o CLP ao mundo físico. Eles são classificados em duas categorias principais:

- Módulos Digitais (Discretos): Trabalham com sinais binários, que possuem apenas dois estados: ligado (nível lógico 1) ou desligado (nível lógico 0) (MACHADO; FREITAS; JUNIOR, 2021). Entradas digitais são usadas para conectar dispositivos como botões, chaves de fim de curso e sensores de proximidade. Saídas digitais são usadas para acionar contatores, lâmpadas de sinalização e válvulas solenoides.

- Módulos Analógicos: Trabalham com sinais que variam continuamente dentro de uma faixa, representando grandezas físicas como temperatura, pressão, velocidade ou posição. Os padrões de sinal mais comuns são tensão (0- 10 V ou 0 - 5 V) e corrente (4-20 mA) (MACHADO; FREITAS; JUNIOR, 2021). Entradas analógicas são conectadas a transdutores de pressão, termopares ou potenciômetros, enquanto saídas analógicas são usadas para controlar a velocidade de inversores de frequência ou a abertura de válvulas proporcionais.

2.3. TRANSDUTORES E ATUADORES: A INTERFACE COM O MUNDO FÍSICO

Para que um sistema de automação possa controlar um processo, ele precisa ser capaz de percebê-lo e agir sobre ele. Essa interface bidirecional com o mundo físico é realizada por dois tipos de dispositivos: transdutores e atuadores. Eles podem ser metaforicamente descritos como os "sentidos" e os "músculos" do sistema de automação, respectivamente.

Transdutores são dispositivos que detectam um estímulo ou uma grandeza física do ambiente (como posição, temperatura, presença de um objeto, pressão, luz) e o convertem em um sinal elétrico que pode ser interpretado pelo CLP. Eles são os olhos e ouvidos do sistema, fornecendo os dados brutos sobre o estado atual do processo.

Atuadores, por outro lado, são dispositivos que recebem um sinal de controle elétrico do CLP e o convertem em uma ação física, geralmente movimento ou alguma forma de energia. Eles executam as ordens da lógica de controle. Exemplos comuns incluem motores elétricos, que movimentam esteiras e os eixos do transelevador; cilindros pneumáticos ou hidráulicos, que realizam movimentos lineares; e válvulas solenoides, que controlam o fluxo de fluidos.

A combinação de transdutores, CLP e atuadores forma uma malha de controle. Os transdutores informam o estado do processo ao CLP, a CPU processa essa informação de acordo com o programa e envia comandos aos atuadores, que modificam o estado do processo. Esse ciclo se repete continuamente, permitindo que o sistema mantenha o processo sob controle e o conduza ao estado desejado. A eficácia de um sistema de automação é, portanto, diretamente limitada pela qualidade, precisão e posicionamento de seus transdutores e pela capacidade de resposta e precisão de seus atuadores.

2.4. LINGUAGEM DE PROGRAMAÇÃO

Antes de 1993 havia um grande problema em relação a linguagem de programação em CLPs, pois nessa época cada fabricante possuia sua própria linguagem de programação, isso causava um problema, caso uma fábrica mudasse seu CLP para de outro fabricante haveria necessidade de fazer com que os programadores se especializem nessa nova linguagem, dificultando para troca de CLPs.(AUTOMATION READY PANELS, [s.d.])

Pensando nisso a Comissão Eletrotécnica Internacional (IEC - *International Electrotechnical Commission*), criou a IEC 61131-3 uma norma internacional que padronizava as linguagens de programação usadas nos CLPs que definiu um total de cinco linguagens, sendo duas três gráficas e duas textuais.

2.4.1. Linguagens Gráficas:

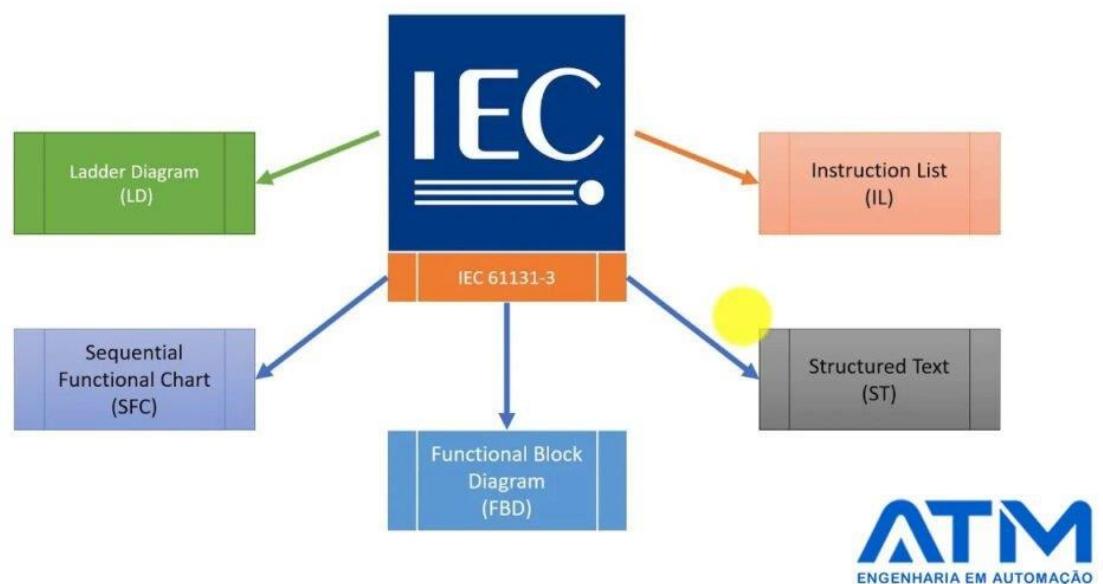
- *Ladder Diagram (LD)* - Diagrama de Contatos: É a linguagem mais popular e tradicional, inspirada nos diagramas elétricos de relés, utilizando contatos e bobinas para construir a lógica de controle. (GEORGINI, 2009)
- *Function Block Diagram (FBD)* - Diagrama de Blocos Funcionais: Utiliza blocos que representam funções prontas (como temporizadores, contadores e operações matemáticas), sendo ideal para controle de processos contínuos e implementação de lógicas mais complexas. (GEORGINI, 2009)
- *Sequential Function Chart (SFC)* - Grafct: Embora seja mais uma ferramenta gráfica para estruturação, ela organiza o programa em passos (etapas) e transições (condições para avançar), perfeita para o controle de máquinas com sequências de operação bem definidas. (GEORGINI, 2009)

2.4.2. Linguagens Textuais:

- *Structured Text (ST)* - Texto Estruturado: É uma linguagem de alto nível com sintaxe similar a Pascal ou C, utilizando comandos como estruturas condicionais (IF...THEN...ELSE, CASE...OF) e laços de repetição (WHILE...DO), excelente para o desenvolvimento de algoritmos complexos, cálculos matemáticos e manipulação de dados. (GEORGINI, 2009)
- *Instruction List (IL)* - Lista de Instruções: É uma linguagem de baixo nível, baseada em

código de máquina, que usa instruções mnemônicas curtas (ex: LD, ST, AND, OR). (GEORGINI, 2009) Atualmente menos utilizada, mas pode ser útil para otimizações de código muito específicas. A Figura 3 ilustra a classificação dessas cinco linguagens padronizadas pela norma IEC.

Figura 3 - Linguagens de programação do CLP



Fonte: ATM Engenharia em Automação (s.d.).

2.5. LADDER

Da norma IEC 61131-3 que padronizou as cinco linguagens para a programação de CLPs, sendo a linguagem mais difundida e tradicionalmente associada aos CLPs, o Diagrama Ladder (LD).

2.5.1. Origem e Estrutura

A linguagem Ladder foi desenvolvida com o objetivo de facilitar a transição de engenheiros e técnicos eletricistas, que estavam acostumados com diagramas de comandos elétricos a relés, para a nova tecnologia dos CLPs. Sua estrutura visual imita intencionalmente esses diagramas. Um programa em Ladder é organizado entre duas barras verticais, que representam os polos de alimentação de um circuito. A lógica é disposta em linhas horizontais,

chamadas de *rungs* (degraus), que se assemelham aos degraus de uma escada, daí o nome "Ladder".(GEORGINI, 2009)

A execução da lógica segue o conceito de um "fluxo de corrente lógico" que percorre cada *rung* da esquerda para a direita. Se houver um caminho contínuo de elementos "fechados" ao longo de um *rung*, a corrente lógica atinge o elemento de saída à direita, ativando-o.

2.5.2. Simbologia Básica

A simbologia do Ladder é baseada em componentes elétricos, conforme detalhado na Tabela 1. Os elementos fundamentais incluem:

- Contatos: Representam condições de entrada ou estados internos. Podem ser:
 - Normalmente Aberto (NA): Simbolizado por --| |--. Permite a passagem da corrente lógica somente quando a variável booleana associada a ele é verdadeira. (FARHAN, 2023)
 - Normalmente Fechado (NF): Simbolizado por --||--. Permite a passagem da corrente lógica quando a variável associada é falsa e a interrompe quando a variável se torna verdadeira. (FARHAN, 2023)
- Bobinas: Representam saídas do sistema ou variáveis de memória internas. São simbolizadas por --()-- e são "energizadas" quando o fluxo de corrente lógica as alcança. (FARHAN, 2023)
- Blocos Funcionais: Para além da lógica booleana simples, a linguagem Ladder incorpora blocos para funções mais complexas, que são indispensáveis em automação. Os mais comuns são:
 - Temporizadores (Timers): Utilizados para criar atrasos ou controlar ações baseadas no tempo, por exemplo, manter uma esteira ligada por 10 segundos após um sensor ser ativado. (FARHAN, 2023)
 - Contadores (Counters): Utilizados para contar eventos, por exemplo, contar o número de caixas que passam por um sensor.(FARHAN, 2023)

2.6. INTERFACE HOMEM-MÁQUINA (IHM)

A Interface Homem-Máquina (IHM), ou *Human-Machine Interface* (HMI), é o painel de controle gráfico que serve como ponto de interação entre o operador humano e o sistema automatizado. Ela traduz os dados complexos e as variáveis do processo, provenientes do CLP, em uma representação visual e intuitiva, facilitando as principais funções de uma IHM na automação industrial são:

- Supervisão e Monitoramento: Exibir o estado do processo em tempo real através de telas gráficas, sinóticos, indicadores de nível, gráficos de tendência e valores numéricos. Isso permite que o operador tenha uma visão clara e imediata do que está acontecendo na planta. (FESTO, 2024)
- Controle e Operação: Fornecer ao operador os meios para interagir com o processo, como botões virtuais para iniciar ou parar ciclos, campos de entrada para ajustar parâmetros (setpoints), e menus para selecionar modos de operação (manual ou automático). (FESTO, 2024)
- Gestão de Alarmes e Diagnósticos: Notificar o operador sobre falhas, paradas ou condições anormais do processo. Um sistema de alarmes eficaz exibe mensagens claras sobre a natureza e a localização da falha, agilizando o diagnóstico e a manutenção(FESTO, 2024). A Figura 4 apresenta exemplos comerciais de IHMs utilizadas na indústria

Figura 4-Exemplos de painéis de (IHM) industriais.



Fonte: Siembra (s.d.).

A concepção de uma IHM eficaz transcende a mera exibição de dados; trata-se de um desafio de design de experiência do usuário (UX) em um contexto industrial. Um sistema como um AS/RS gera uma vasta quantidade de informações, como o estado de dezenas de sensores, a posição dos motores e dados de inventário. Apresentar essa informação de forma bruta seria improdutivo. Uma IHM bem projetada, como a que exibe um "mapa do armazém" em tempo real, abstrai essa complexidade e apresenta ao operador a informação contextualizada e relevante para a tomada de decisão rápida e precisa, aumentando a eficiência e a segurança do sistema como um todo.

2.7. A INDÚSTRIA 4.0 E A LOGÍSTICA INTELIGENTE (LOGÍSTICA 4.0)

O cenário industrial contemporâneo é definido pelo paradigma da Quarta Revolução Industrial, um conceito que transcende a mera digitalização para descrever a integração técnica de Sistemas Ciber-Físicos (CPS) na manufatura e logística. O objetivo central dessa transformação é a criação de "Fábricas Inteligentes", onde a convergência entre o mundo físico

e o digital permite operações autônomas e descentralizadas (KAGERMANN; WAHLSTER; HELBIG, 2013).

Esta revolução distingue-se das anteriores pela velocidade exponencial e pelo impacto sistêmico, sendo impulsionada por um conjunto de tecnologias habilitadoras que transformam fundamentalmente a maneira como os produtos são fabricados (SCHWAB, 2016). A sustentação desse ecossistema baseia-se em pilares fundamentais, validados tanto no cenário internacional quanto no contexto industrial brasileiro. Esses pilares incluem: a Digitalização, Análise de Grandes Volumes de Dados (Big Data e Analytics), a Internet das Coisas (IoT) para coleta contínua de dados, Ciber segurança, Cloud Computing, Manufatura aditiva, Inteligência artificial, Robótica Avançada, Simulação (Gêmeos Digitais) para validação de processos, e a Integração de sistemas (KAGERMANN et al., 2013; CNI, 2016). Esse conjunto de tecnologias é visualizado na Figura 5, que apresenta os pilares fundamentais da Indústria 4.0.

Figura 5 - Pilares da Industria 4.0



Fonte: Altus (s.d.).

A aplicação desses princípios ao gerenciamento da cadeia de suprimentos deu origem à Logística 4.0, que utiliza essas tecnologias habilitadoras para criar fluxos de abastecimento ágeis e rastreáveis, capazes de responder instantaneamente a mudanças no mercado. Nesse

contexto, a implementação de um sistema AS/RS validado virtualmente deixa de ser uma simples automação para se tornar uma aplicação prática de eficiência e previsibilidade, alinhando-se aos requisitos de desempenho exigidos pela nova indústria (CNI, 2016).

2.8. METODOLOGIAS DE DESENVOLVIMENTO DA INDÚSTRIA 4.0

O desenvolvimento de sistemas de automação complexos, como o proposto neste trabalho, é beneficiado por metodologias inovadoras que surgiram com a Indústria 4.0, focadas em virtualização e simulação para otimizar o ciclo de vida do projeto.

2.8.1. Gêmeo Digital (*Digital Twin*)

O conceito de Gêmeo Digital (*Digital Twin*) transcende a simples representação gráfica ou modelagem 3D de um equipamento, definindo-se como um sistema composto por três pilares inseparáveis: a entidade física no espaço real, a entidade virtual no espaço digital e o elo de dados que conecta essas duas esferas. Essa conexão é fundamental, pois permite que o modelo virtual espelhe o comportamento, as características e o estado do ativo físico de forma dinâmica e fiel (GRIEVES, 2014).

Para garantir a robustez técnica da aplicação, distingue-se o Gêmeo Digital de outras formas de virtualização estática pela sua capacidade de integração contínua. A característica determinante dessa tecnologia é o uso de dados em tempo real para criar um ciclo fechado entre o mundo real e o virtual. Diferente de um modelo passivo, o Gêmeo Digital exige um fluxo de informações que habilita não apenas o monitoramento, mas também a simulação de cenários e a predição de comportamentos do sistema antes que eles ocorram no ambiente físico (SIEMENS, [s.d.]).

No contexto deste trabalho, o software de simulação Factory I/O foi utilizado para construir o Gêmeo Digital do sistema AS/RS, aplicando esses conceitos teóricos na prática. Através de uma abordagem *Hardware-in-the-Loop* (HiL), o sistema replica a geometria e as leis físicas (gravidade, atrito, inércia) dos sensores e atuadores, estabelecendo a comunicação bidirecional necessária com o controlador físico (CLP) para validação, previsão e otimização do processo logístico (GRIEVES, 2014; SIEMENS, [s.d.]).

2.8.2. Comissionamento Virtual (*Virtual Commissioning*)

A validação antecipada de sistemas de automação consolidou-se como uma estratégia central para a competitividade na Indústria 4.0. A adoção de tecnologias de simulação e virtualização tornou-se decisiva para aumentar a eficiência operacional, uma vez que o funcionamento de máquinas e equipamentos pode ser mimetizado e virtualizado digitalmente. Essa capacidade permite validar processos produtivos inteiros em ambiente virtual antes mesmo de qualquer implementação física, antecipando gargalos e garantindo a assertividade do projeto (CNI; SEBRAE, 2019).

Na engenharia de controle, essa prática de validação digital é tecnicamente denominada Comissionamento Virtual. Trata-se de uma metodologia que permite identificar e corrigir erros lógicos e de programação antes da montagem da máquina, o que resulta em uma redução significativa no tempo de engenharia estimada em até 30% e evita paradas não planejadas e custos corretivos durante a operação real (SIEMENS, [s.d.]).

Para a aplicação prática destes conceitos, utiliza-se a abordagem conhecida como *Hardware-in-the-Loop* (HiL). Nesta configuração, o controlador físico real (neste caso, o CLP S7-1200) é conectado ao gêmeo digital da célula logística, em vez de ser ligado aos sensores reais. Isso garante que a lógica de controle seja validada em condições dinâmicas seguras, eliminando riscos de danos aos equipamentos e protegendo o investimento, alinhando-se às recomendações de modernização e segurança para a indústria (CNI; SEBRAE, 2019; SIEMENS, [s.d.]).

Em suma, ao desacoplar o desenvolvimento do software da disponibilidade do hardware, o Comissionamento Virtual transforma os projetos de automação, tornando-os previsíveis e aderentes aos prazos exigidos pela Indústria 4.0..

2.9. MÁQUINA DE ESTADOS FINITOS

O controle de processos que possuem uma natureza inherentemente sequencial, como o ciclo de operação de um transelevador, demanda estruturas de programação que garantam a robustez e a facilidade de manutenção do software. Nesse cenário, a utilização da Máquina de Estados Finitos (*Finite State Machine* - FSM) consolidou-se como uma das abordagens mais eficazes para a estruturação lógica de automação (GEORGINI, 2009).

Esse modelo de comportamento é composto por um número finito de estados, transições entre esses estados e ações associadas. Na programação de Controladores Lógicos

Programáveis (CLP), esses elementos traduzem-se em: um estado, que define uma etapa específica e única da operação (como aguardar ou mover); uma transição, que representa a condição lógica necessária para avançar ao próximo passo; e uma ação, que determina as saídas ativadas durante aquele estado específico (GEORGINI, 2009).

A programação estruturada em máquina de estados oferece vantagens significativas sobre a lógica Ladder convencional não estruturada. Enquanto o Ladder simples pode resultar em um emaranhado de intertravamentos complexos, a FSM proporciona clareza e organização ao fluxo do programa. Isso resulta em maior facilidade de depuração (*debugging*), permitindo identificar rapidamente falhas no processo, e aumenta a manutenibilidade do código, simplificando a adição de novas etapas sem comprometer a lógica existente (GEORGINI, 2009).

3. METODOLOGIA

Este capítulo descreve os procedimentos, a arquitetura tecnológica, a metodologia e o fluxo de processo usados para projetar, implementar e validar a célula logística automatizada.. A abordagem metodológica central adotada é o Comissionamento Virtual, um pilar da engenharia moderna alinhada aos preceitos da Indústria 4.0.

Esta metodologia permite que toda a lógica de controle, sequenciamento de processos e interfaces de operação sejam desenvolvidos, testados e depurados em um ambiente de simulação dinâmica, antes da aquisição ou montagem de qualquer componente físico. O núcleo desta abordagem é a criação de um Gêmeo Digital da célula logística, uma réplica virtual funcional que espelha o comportamento de seus correspondentes físicos. Para este fim, o capítulo está estruturado em três seções principais:

- Arquitetura Tecnológica e Ferramental: Detalhamento dos componentes de hardware e das plataformas de software que constituem a solução de automação.
- Modelagem do Gêmeo Digital: Descrição dos componentes virtuais selecionados da biblioteca de simulação para construir a célula logística e sua respectiva função no processo.
- descrição Funcional do Sistema Automatizado: Formalização da lógica de controle e do fluxo operacional implementado no controlador para gerenciar o Gêmeo Digital, detalhando as três etapas do processo: classificação, paletização e armazenamento.

3.1. ARQUITETURA TECNOLÓGICA E FERRAMENTAL

A infraestrutura de hardware e software do projeto foi fundamentada na plataforma Totally Integrated Automation (TIA) da Siemens. Esta escolha estratégica assegura a compatibilidade total entre o controlador lógico, a interface de operação e o ambiente de engenharia, otimizando o fluxo de desenvolvimento e integração.

3.1.1. **Hardware**

O hardware selecionado representa o cérebro e a interface de supervisão do sistema, interligados por uma rede de comunicação industrial.

3.1.1.1. CLP

O componente central de controle do sistema CLP da família SIMATIC S7-1200, especificamente a CPU 1214C DC/DC/DC, conforme ilustrado na Figura 6.

Figura 6 - S7- 1200



Fonte: O autor (2025)

A designação "DC/DC/DC" detalha suas características elétricas: alimentação em corrente contínua (24 VDC), entradas digitais (DI) padrão 24 VDC e saídas digitais (DO) a transistor (DC). Este tipo de saída é ideal para o chaveamento de alta frequência e baixa potência exigido por interfaces de atuadores em um ambiente de automação moderno, em contraste com saídas a relé. As especificações técnicas da CPU 1214C que a qualificam para este projeto incluem:

- Memória de Trabalho (*Work memory*): 150 KB.
- Memória de Carga (*Load memory*): 4 MB.
- Entradas/Saídas (I/O) Integradas:
 - 14 Entradas Digitais (DI) 24 VDC.
 - 10 Saídas Digitais (DO) 24 VDC.
 - 2 Entradas Analógicas (AI) 0-10 VDC.

3.1.1.2. Interface Homem-Máquina (IHM)

Para a supervisão e operação do processo, foi selecionada a IHM SIMATIC HMI KTP700 Basic. Trata-se de um *Basic Panel* que combina operação por toque (*touchscreen*) e teclas físicas. Possui um display de 7 polegadas, oferecendo alta fidelidade visual para a representação gráfica do processo, conforme ilustrado na Figura 7

Figura 7 - KTP700 Basic



Fonte: O autor (2025)

Conforme definido nos objetivos do projeto, a IHM utilizada para o funcionamento inclui:

- Monitoramento Visual: Acompanhamento do status de ocupação das prateleiras do sistema de armazenamento (as estantes).
- Alteração de Modo: Capacidade de mudar o modo de operação do transelevador (por exemplo, de automático para manual ou vice-versa).
- Identificação de Caixas: Verificação dos códigos/identificadores (IDs) das caixas que estão em uma determinada prateleira.

- Seleção de Paleta: Possibilidade de escolher um paleta específico para que o transelevador o retire do sistema.

Sua principal característica de integração é a interface PROFINET embarcada, que permite a comunicação em rede Ethernet industrial diretamente com o CLP S7-1200.

3.1.2. Infraestrutura de Comunicação

Para estabelecer a rede de comunicação entre o CLP e a IHM, foi empregado um *switch* industrial Siemens SCALANCE XB005. Este é um *switch Industrial Ethernet* não gerenciado, projetado especificamente para a robustez do chão de fábrica e para a criação de pequenas topologias de rede em estrela ou linha. Ele dispõe de 5 portas RJ45 com velocidade de 10/100 Mbit/s. conforme ilustrado na Figura 8

Figura 8 - SCALANCE XB005



Fonte: O autor (2025)

No contexto do projeto, o SCALANCE XB005 atua como o *conector* físico da rede PROFINET, garantindo a conectividade entre os dispositivos, como mostrado na Figura 9.

Figura 9 – Setup de validação *Hardware-in-the-Loop* (HiL)



Fontes: O Autor (2025)

3.1.2.1. Software

O Comissionamento Virtual se concretiza através da união de duas plataformas de software distintas: uma dedicada para a criação da lógica, e outra de simulação, essencial para o desenvolvimento do Gêmeo Digital.

3.1.2.2. TIA Portal

O software utilizado para a configuração e programação da solução de automação foi o Siemens TIA Portal V13.

O TIA Portal é uma plataforma que centraliza múltiplas ferramentas de desenvolvimento em um único ambiente. Para este projeto, foram utilizados dois de seus componentes principais:

- STEP 7 Basic V13: O ambiente para configuração do hardware S7-1200 e desenvolvimento de toda a lógica de controle do CLP (em linguagem Ladder, conforme Fundamentação Teórica).
- WinCC Basic V13: O ambiente para o design gráfico das telas de supervisão da IHM KTP700.

A integração nativa destes componentes dentro do TIA Portal simplifica drasticamente o desenvolvimento, pois as *tags* (variáveis) criadas no programa do CLP são instantaneamente acessíveis no editor da IHM.

3.1.2.3. Factory I/O

O Gêmeo Digital da logística foi construído utilizando o software Factory I/O. Trata-se de um ambiente de simulação 3D em tempo real que fornece uma extensa biblioteca de componentes industriais virtuais (sensores, atuadores, esteiras e estações de processo). O Factory I/O foi projetado para se conectar a controladores externos, permitindo que a lógica de um CLP real (ou simulado) opere o Gêmeo Digital.

3.1.2.4. Método de Comissionamento Virtual

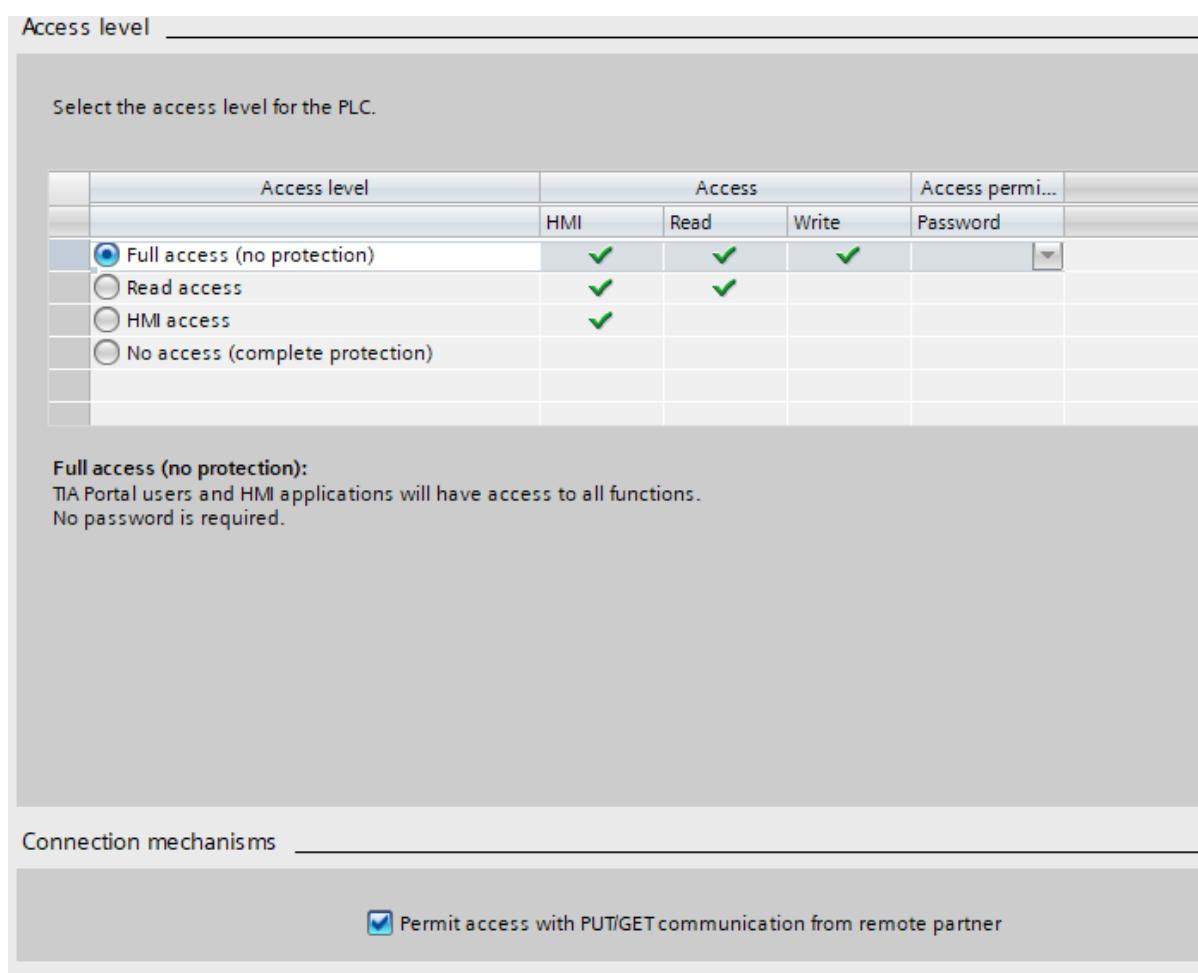
A essência da metodologia de Comissionamento Virtual reside na forma como o TIA Portal e o Factory I/O são integrados. Diferente de uma simulação puramente virtual, este projeto empregou a lógica de controle executada no hardware de automação físico (o CLP S7-1200 e a IHM KTP700), que se comunicou em tempo real com o Gêmeo Digital (a planta virtual no Factory I/O). O fluxo de trabalho de integração seguiu os seguintes passos:

- Configuração no TIA Portal (Lado do CLP):
 - Endereçamento de Rede: O CLP S7-1200 físico recebeu um endereço IP estático em

sua interface PROFINET, permitindo a comunicação em rede.

- Configuração de Proteção: Nas propriedades da CPU, duas configurações de segurança foram essenciais, o "Access level" (Nível de Acesso) foi definido como "HMI access" (ou superior) e, crucialmente, foi habilitada a opção "Permit access with PUT/GET communication from remote partner" (Permitir acesso com comunicação PUT/GET de parceiro remoto). Essa configuração de segurança, crucial para autorizar a IHM a ler e escrever dados no CLP, é ilustrada na Figura 10.

Figura 10 - Configuração no TIA Portal



Fonte: O autor (2025)

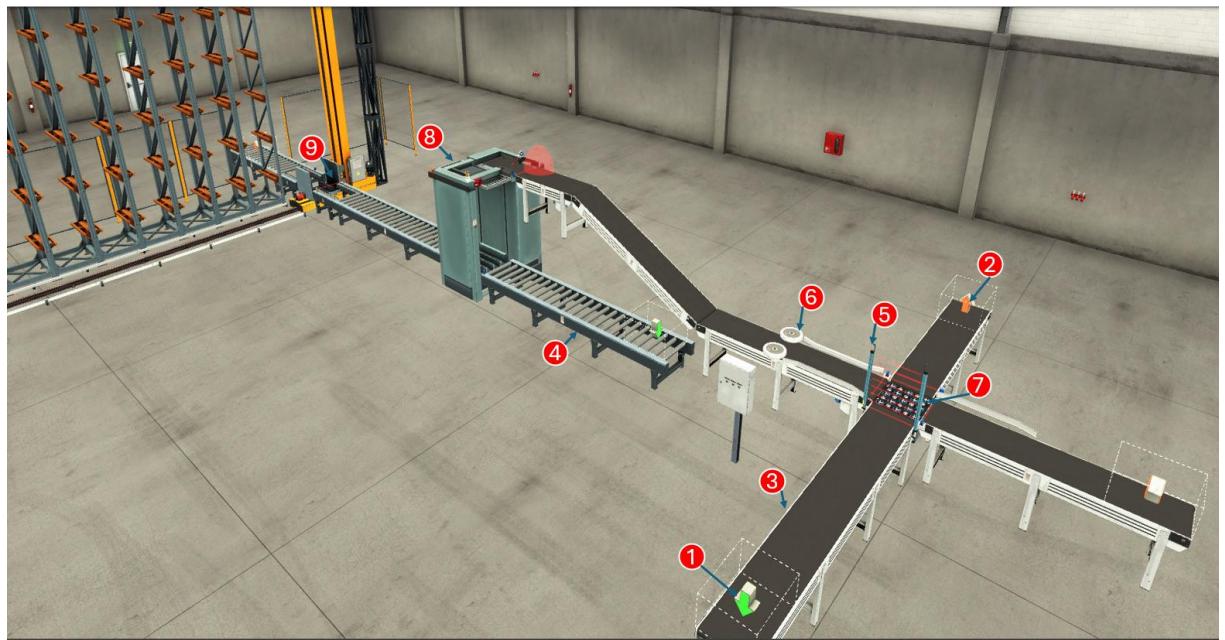
- Mapeamento de I/O: O programa do usuário foi desenvolvido e descarregado (download) no CLP físico. Para evitar conflitos, os endereços de I/O das entradas físicas integradas foram deslocados (ex: para iniciar em %I10.0), liberando os endereços iniciais (%I0.0, %I0.1, etc.) para serem usados exclusivamente pelos sensores virtuais do Factory I/O.

- Configuração no Factory I/O (Lado do Gêmeo Digital):
 - Seleção do Driver: No Factory I/O, o driver de comunicação foi configurado para "Siemens S7-1200/1500".
 - Conexão de Rede: No painel de configuração do driver, no campo "Host", foi inserido o endereço IP físico do CLP S7-1200, estabelecendo a conexão direta pela rede Ethernet (através do switch SCALANCE XB005).
- Loop de Controle Fechado (HIL):
 - Ao estabelecer a conexão, formou-se um ciclo de controle *Hardware-in-the-Loop* (HIL): o Factory I/O monitorava os sinais de comando gerados pelo CLP físico (%Q) e utilizava-os para animar os atuadores do Gêmeo Digital (ex: ligar o motor virtual da esteira). Em contrapartida, os sinais gerados pelos sensores virtuais do simulador eram escritos diretamente na memória de entradas do CLP físico. Este método permitiu validar a lógica de controle executada no hardware real, porém atuando sobre um processo inteiramente virtual e seguro, comprovando a eficácia do Comissionamento Virtual.

3.2. MODELAGEM DO GÊMEO DIGITAL DA CÉLULA LOGÍSTICA

O processo é constituído de três etapas principais: Separação, Paletização e AS/RS. Toda a estrutura foi modelada no software Factory I/O, empregando um conjunto de componentes virtuais que replicam fielmente as funções e o comportamento de equipamentos industriais reais. A seguir, detalha-se o funcionamento real e a aplicação virtual de cada componente-chave utilizado no projeto. A disposição espacial completa da célula logística e a integração destes componentes no ambiente virtual podem ser visualizadas na Figura 11.

Figura 11 - Vista isométrica do Gêmeo Digital da Célula Logística



Fonte: O autor (2025)

Legenda dos componentes: (1) Emissor; (2) Removedor; (3) Esteira Transportadora; (4) Esteira de Rolletes; (5) Sensor de Barreira; (6) Alinhador; (7) Classificador de Rodas; (8) Paletizadora; (9) Transelevador.

3.2.1. Componentes Básicos e de Fluxo

- Emissor (Emitter):
 - *Função Real:* Simula a fonte de produtos em uma linha de produção, como a saída de uma máquina de envase ou um ponto de recebimento de fornecedor.
 - Função no Factory I/O: É um gerador de itens. Foi configurado para criar (emitir) caixas de tamanhos variados para a Etapa 1 e paletes para a Etapa 2. É controlado por uma *tag* de saída digital (DO) do CLP.
- Removedor (Remover):
 - *Função Real:* Representa o fim de um processo, como uma linha de descarte, expedição ou o consumo de um item.
 - Função no Factory I/O: É um destruidor de itens. Quando um item entra em seu volume, é removido da simulação. Foi utilizado no final dos ramais de caixas médias/grandes e no final da esteira de retirada do AS/RS, evitando o acúmulo de itens.
- Esteira Transportadora (Belt Conveyor):

- Função Real: Transportador de correia plana, comumente usado para cargas leves, como caixas de papelão.
- Função no Factory I/O: Utilizado para todo o transporte de caixas nas Etapas 1 e 2. O modelo virtual é controlado por uma *tag* de saída digital (DO) que aciona seu motor (liga/desliga).
- Esteira de Roletes (Roller Conveyor):
 - Função Real: Transportador de rolos tracionados, projetado para cargas pesadas e de base plana, como paletes.
 - Função no Factory I/O: Utilizado para todo o transporte de paletes (alimentação da paletizadora, transporte para o AS/RS e esteira de saída). Também é controlado por uma tag de saída digital (DO).

3.2.2. Componentes de Operação e Supervisão

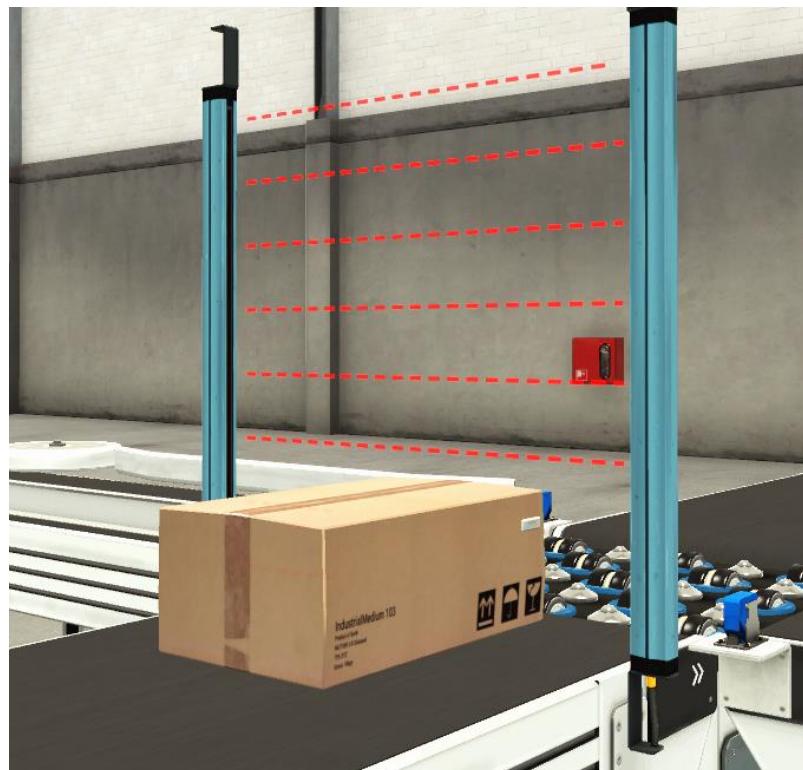
- Painel Elétrico (Electric Switchboard):
 - Função Real: Um gabinete ou painel que centraliza dispositivos de interface humana, como botões, seletores e luzes de sinalização.
 - Função no Factory I/O: Um componente virtual que serve como suporte para a montagem dos botões de comando do sistema.
- Botões (Button):
 - Função Real: Dispositivos de comando momentâneos (de pulso) que, ao serem pressionados, fecham ou abrem um contato elétrico.
 - Função no Factory I/O: Foram utilizados botões virtuais (Ligar, Desligar, Reset) que, quando clicados, ativam uma tag de entrada digital (DI) no CLP.

3.2.3. Sensores de Detecção e Classificação

- Sensor de Barreira (Light Array Emitter/Receiver):
 - Função Real: Uma cortina de luz composta por múltiplos feixes infravermelhos paralelos. É usada tanto para segurança (detecção de invasão) quanto para medição de perfil ou altura.
 - Função no Factory I/O: É o componente-chave da Etapa 1. O sensor virtual pode ser configurado em modo numérico ou analógico. No projeto, ele mede a altura da caixa contando quantos feixes são interrompidos, enviando um valor (ex: 0-10V)

ao CLP, permitindo a lógica de classificação por tamanho. O princípio de medição e a disposição dos feixes infravermelhos são visualizados na Figura 12.

Figura 12 - Feixes de detecção do Sensor de Barreira (Light Array)



Fonte: O Autor (2025)

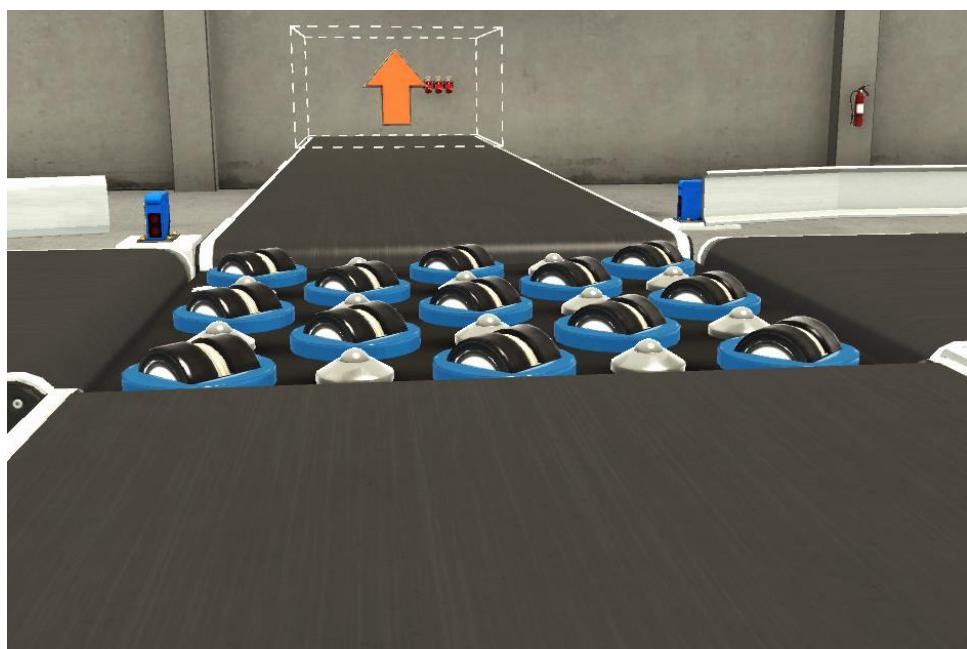
- **Sensor Difuso (Diffuse Sensor):**
 - Função Real: Sensor fotoelétrico onde o emissor e o receptor estão na mesma cápsula. Detecta um objeto quando a luz emitida pelo sensor reflete difusamente na superfície do objeto e retorna ao receptor.
 - Função no Factory I/O: Usado para detecção de presença em pontos específicos onde a posição exata não é crítica. No projeto, detecta a presença de caixas nos ramais de desvio e no ponto de leitura RFID. Mapeado para uma tag de entrada digital (DI).
- **Sensor Retrorreflexivo (Retro-reflective Sensor):**
 - Função Real: Sensor fotoelétrico que requer um espelho prismático (refletor). O sensor detecta um objeto quando este interrompe o feixe de luz que viaja até o refletor e retorna. É mais preciso que o difuso.
 - Função no Factory I/O: Utilizado para detecção precisa de parada. No projeto, é usado no ponto de coleta do palete pronto, sinalizando ao CLP que o palete está na

posição exata para ser pego pelo Transelevador. Mapeado para uma tag de entrada digital (DI).

3.2.4. Componentes de Processo e Rastreabilidade

- Alinhador (*Aligner*):
 - Função Real: Guias laterais, que podem ser passivas (metálicas) ou ativas (com rodas motorizadas, *wheel aligner*), usadas para centralizar produtos em um transportador.
 - Função no Factory I/O: Utilizado na Etapa 2 para garantir que a caixa pequena esteja perfeitamente alinhada antes de entrar na paletizadora, assegurando um empilhamento uniforme.
- Classificador de Rodas (*Pop-Up Wheel Sorter*):
 - Função Real: Um sistema de desvio (*sorter*) que utiliza conjuntos de rodas que se elevam (pop-up) acima dos rolos da esteira principal para desviar produtos em ângulo (ex: 30°, 45°) para linhas secundárias.
 - Função no Factory I/O: É o atuador central da Etapa 1. O componente virtual possui tags de saída digital (DO) para "Girar Esquerda" (*Left*) e "Girar Direita" (*Right*). A lógica do CLP aciona a saída correspondente à classe da caixa (esquerda para pequena, direita para média). A figura 13 demonstra o classificador quando acionado para a esquerda.

Figura 13- Mecanismo de desvio do Classificador de Rodas (*Pop-up Wheel Sorter*)



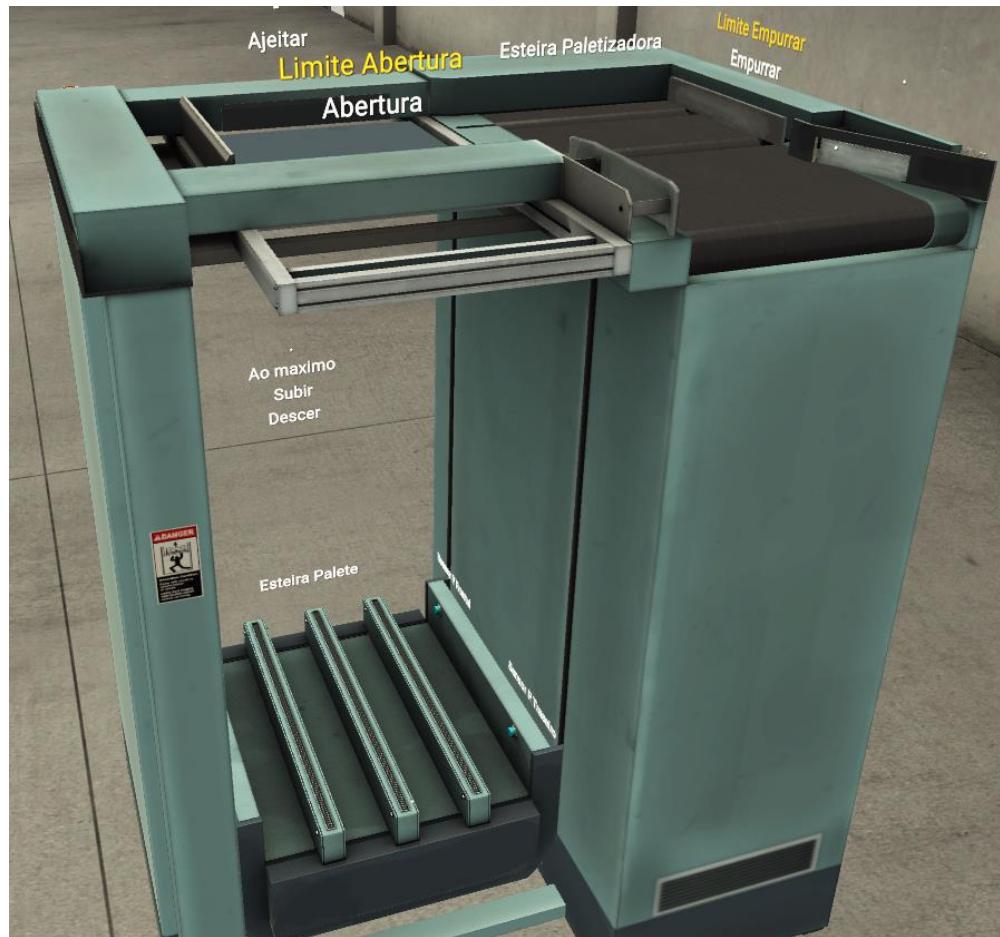
Fonte: O Autor (2025)

- Leitor RFID (RFID Reader):
 - Função Real: Um dispositivo de Identificação por Radiofrequência que lê e escreve dados em *tags* (etiquetas) eletrônicas passivas ou ativas, usado para rastreabilidade.
 - Função no Factory I/O: Um componente de simulação avançado. Não é um simples sensor (DI), mas um dispositivo controlado por comandos. O CLP deve enviar um valor de comando (ex: '2' para Ler, '3' para Escrever) e uma *tag* de execução. O leitor responde com um *status* de sucesso ou erro. No projeto, é usado para gravar o ID único de cada caixa na memória do CLP na Etapa 2.

3.2.5. Sistema de Armazenagem Automatizado (AS/RS)

- Paletizador (*Palletizer*):
 - Função Real: Uma estação de processo complexa, frequentemente robótica, que empilha caixas ou produtos em camadas sobre um palete.
 - Função no Factory I/O: É um componente de alto nível que encapsula múltiplos sensores e atuadores internos (Empurrador (*Pusher*), Garra (*Clamp*), Elevador (*Elevator*), Placa de Depósito (*Open Plate*)). O CLP é responsável por orquestrar a sequência complexa de acionamento desses I/Os virtuais para formar as camadas de caixas. Devido à complexidade mecânica deste equipamento, a Figura 14 identifica a localização interna dos principais atuadores e sensores de limite, facilitando a compreensão das variáveis listadas na Tabela 1 a seguir.

Figura 14 - Sensores e atuadores internos do Paletizador



Fonte: O Autor (2025)

- A Tabela 1 detalha os principais sensores e atuadores internos do Paletizador.

Tabela 1 - Sensores e Atuadores do Paletizador (*Palletizer*)

Nome da Tag(Factory I/O)	Tipo (Sensor/Atuador)	Funcionalidade
Empurrar	Atuador (Saída)	Move o empurrador para posicionar as caixas na placa.
Ajeitar	Atuador (Saída)	Move o grampeador para segurar/centralizar as caixas.
Abertura	Atuador (Saída)	Abre a placa de depósito para liberar a camada de caixas sobre o palete.
Subir	Atuador (Saída)	Move o elevador interno para cima.
Descer	Atuador (Saída)	Move o elevador interno para baixo.
Ao Máximo	Atuador (Saída)	Quando acionado em conjunto com subir ou descer, move o até o limite o elevador, seja para cima ou para baixo.
Esteira Paletizadora	Atuador (Saída)	Gira a esteira de entrada de caixas no sentido de entrada.
Esteira Paleta	Atuador (Saída)	Gira a esteira de roletes (corrente) de paletes no sentido de saída da paletizadora.
Limite Abertura	Sensor (Entrada)	Indica que a escotilha (Abertura) está fechada, totalmente fechada ou aberta.
Limite Empurrar	Sensor (Entrada)	Indica que o empurrador atingiu seu limite (frontal/traseiro).
Sensor P Traseiro	Sensor (Entrada)	Sensor de limite que indica que o paleta está na posição traseira/ entrada da paletizadora (base).
Sensor P Frontal	Sensor (Entrada)	Sensor de limite que indica que o paleta está na posição frontal/ saída da paletizadora.

Fonte: O Autor (2025)

- Estante (Rack):
 - Função Real: A estrutura de aço industrial para armazenamento de paletes em posições (células).
 - Função no Factory I/O: A estrutura passiva (tipo *single-deep*) onde o Transelevador armazena os paletes. O modelo virtual possui 54 posições (9 colunas x 6 linhas), que formam a base para o sistema de gerenciamento de inventário implementado no CLP.
- Transelevador (*Stacker Crane*):
 - Função Real: O robô do sistema AS/RS, que se move em eixos horizontal (X) e vertical (Y) para posicionar seus garfos telescópicos e armazenar ou retirar paletes das estantes.

- Função no Factory I/O: O componente mais complexo do Gêmeo Digital. Foi configurado para o modo de controle Analógico.
- Método de Controle: Diferente de atuadores digitais (liga/desliga), o CLP deve comandar o Transelevador enviando *valores* analógicos (representando 0-10 V) para cada eixo. Um valor específico de tensão no Eixo X corresponde a uma posição horizontal (coluna), e um valor no Eixo Y corresponde a uma posição vertical (linha). A lógica do CLP, portanto, deve conter uma tabela de *look-up* (mapeamento) que converte uma posição lógica da estante (ex: Posição 1) em um par de tensões (ex: X=1.0V, Y=1.2V). Para o controle de posicionamento, é fundamental compreender a orientação dos eixos de movimento. A Figura 15 detalha os eixos Horizontal (X) e Vertical (Y), bem como os pontos de referência para os sensores de posição analógicos.

Figura 15 - Detalhes dos eixos e atuadores do Transelevador



Fonte: O Autor (2025)

- A Tabela 2 detalha os principais sensores e atuadores do Transelevador (configuração Analógica/Digital).

Tabela 2 - Sensores e Atuadores do Transelevador (*Stacker Crane*)

Nome da Tag (Factory I/O)	Tipo (Sensor/Atuador)	Funcionalidade (O que faz)
<i>SetPoint X</i>	Atuador (Saída)	Define a posição alvo (tensão) no eixo horizontal (X).
<i>SetPoint Y</i>	Atuador (Saída)	Define a posição alvo (tensão) no eixo vertical (Y).
Garfo Frontal	Atuador (Saída)	Move os garfos para a frente (para depositar/coletar).
Garfo Traseiro	Atuador (Saída)	Move os garfos para atras (para depositar/coletar).
Posição X	Sensor (Entrada)	Informa a posição atual (tensão) no eixo horizontal (X).
Posição Y	Sensor (Entrada)	Informa a posição atual (tensão) no eixo vertical (Y).
Limite Garfo Frontal	Sensor (Entrada)	Indica que os garfos atingiram o limite Frontal.
No meio	Sensor (Entrada)	Indica que os garfos estão no limite central (recolhidos).
Limite Garfo Traseiro	Sensor (Entrada)	Indica que os garfos atingiram o limite Traseiro.

Fonte: O Autor (2025)

A Tabela 3 sumariza os componentes mais críticos do Gêmeo Digital, conectando-os à sua interface de controle no CLP e à sua função metodológica no processo.

Tabela 3 – Mapeamento e Função dos Componentes-Chave do Gêmeo Digital

Categoría	Componente Virtual (Factory I/O)	Função no Processo (Metodologia)
Classificação	Sensor de Barreira (<i>Light Array</i>)	Mede a altura da caixa para classificação (Pequena, Média, Grande).
Classificação	Classificador (<i>Pop-Up Wheel Sorter</i>)	Desvia os caixas para as esteiras corretos com base na classificação.
Rastreabilidade	Leitor RFID (<i>RFID Reader</i>)	Grava um ID de rastreabilidade único em cada caixa (Etapa 2).
Paletização	Paletizador (<i>Palletizer</i>)	Executa a sequência complexa de empilhamento de caixas em paletes.
Armazenagem	Transelevador (<i>Stacker Crane</i>)	Movimenta-se para posições (X, Y) comandadas pelo CLP para armazenar ou retirar paletes.

Fonte: O Autor (2025)

3.3. DESCRIÇÃO FUNCIONAL DO SISTEMA AUTOMATIZADO

Esta seção descreve o fluxo do processo visual e funcional da célula logística, dividido em três etapas sequenciais, conforme implementado e validado no Gêmeo Digital.

3.3.1. Etapa 1: Separação e Classificação de Caixas

Esta etapa inicial recebe um fluxo misto de produtos e os segregá com base em sua altura.

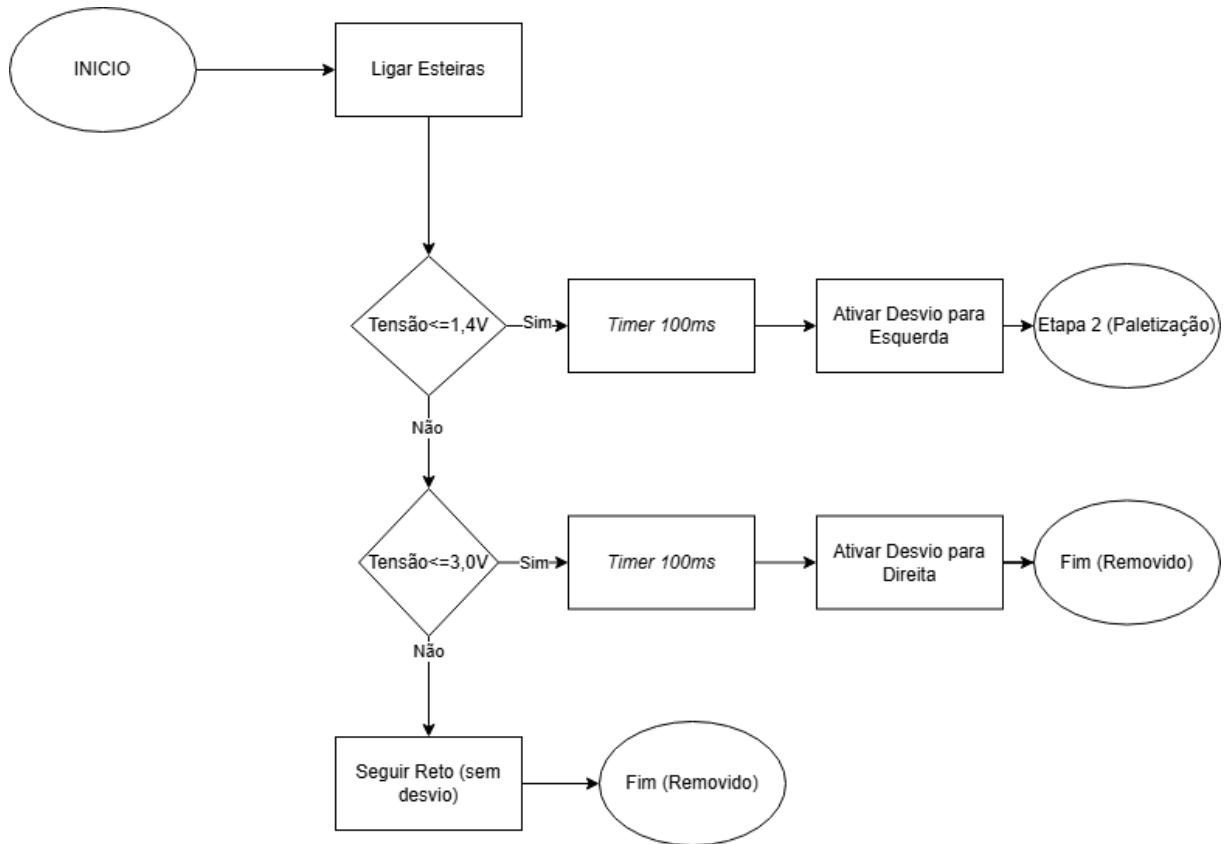
- Geração e Transporte: O processo inicia quando um 'Emissor' (*Emitter*) gera caixas de tamanhos variados (pequena, média ou grande) em uma esteira transportadora principal.
- Medição: A caixa é transportada até passar por um 'Sensor de Barreira' (*Light Array*), que mede sua altura.
- Classificação e Desvio: Com base na altura medida, o 'Classificador de Rodas' (*Pop-Up*

Wheel Sorter) é acionado para direcionar o fluxo:

- Caixas Pequenas: São desviadas para a esteira da esquerda, seguindo para a Etapa 2 (Paletização).
- Caixas Médias: São desviadas para a esteira da direita.
- Caixas Grandes: Passam reto pelo classificador, continuando na esteira principal.
- Descarte (Simplificação): Os ramais das caixas médias e grandes são linhas de descarte para este projeto. As caixas nessas esteiras são transportadas até 'Removedores' (*Removers*), que as retiram da simulação, permitindo que o foco do projeto se mantenha na linha de caixas pequenas.

Para consolidar o entendimento da lógica de decisão baseada nos níveis de tensão do sensor, o fluxo completo de classificação e desvio é detalhado na Figura 16

Figura 16 – Fluxograma do processo de separação e classificação de caixas



Fonte: O Autor (2025)

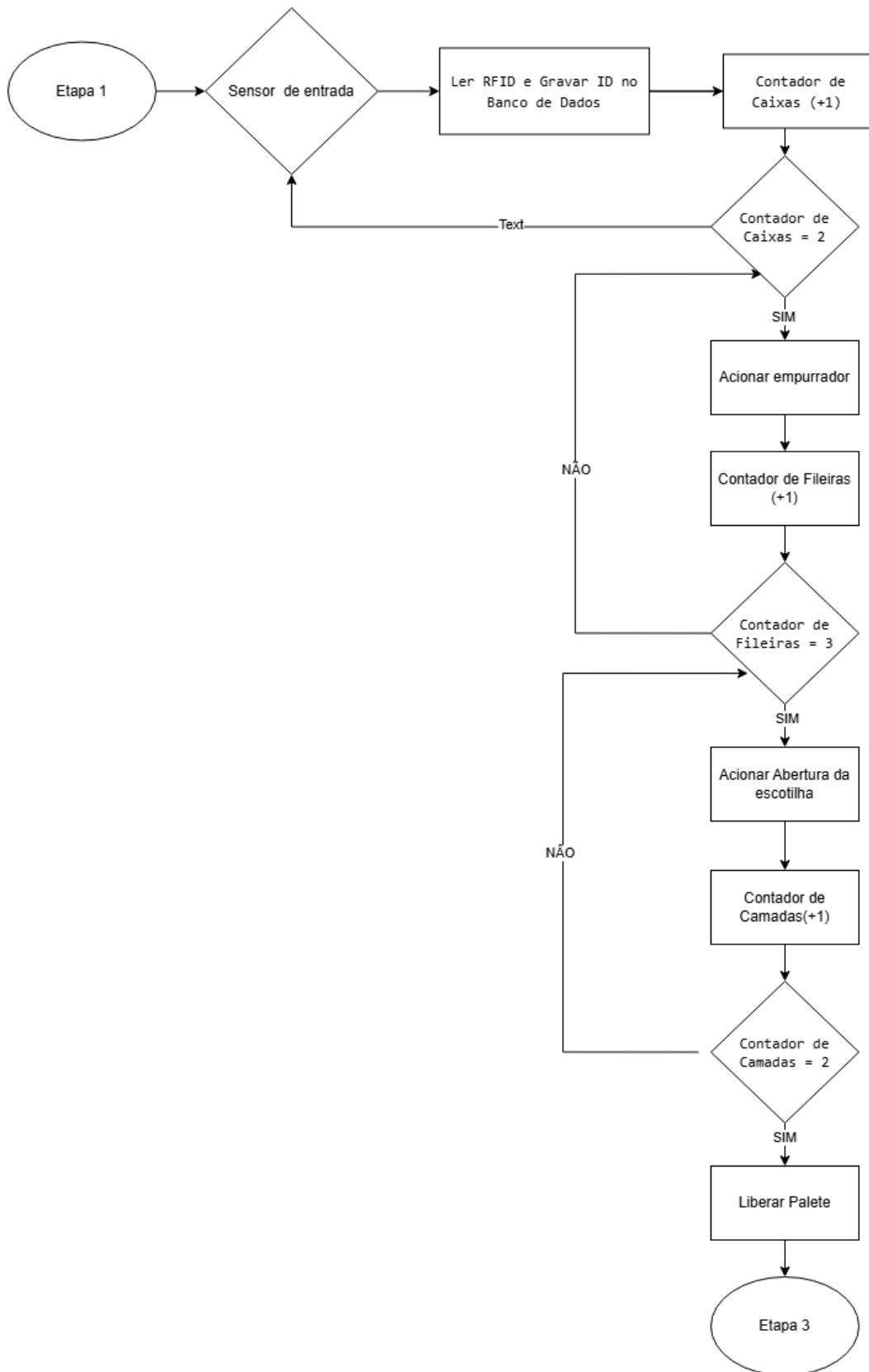
3.3.2. Etapa 2: Paletização e Rastreabilidade

Esta etapa recebe apenas as caixas pequenas da Etapa 1, agrupa-as em um palete e realiza a rastreabilidade.

1. Alinhamento e Rastreio: A caixa pequena passa por 'Alinhadores' (*Aligners*) para garantir o posicionamento correto e, em seguida, passa por um 'Leitor RFID' (*RFID Reader*), onde um ID único de rastreabilidade é lido e enviado para a memória do CLP.
2. Alimentação de Paletes: Simultaneamente, um 'Emissor' (*Emitter*) de paletes alimenta um palete vazio na esteira de roletes, que o transporta até a base do 'Paletizador' (*Palletizer*). Sensores confirmam que o palete está na posição correta.
3. Formação de Camada: O elevador interno do paletizador posiciona o palete. As caixas pequenas entram na estação e são agrupadas (ex: 2x2, formando uma camada de 4 caixas) por um 'Empurrador' (*Pusher*) sobre uma placa de depósito.
4. Empilhamento: Quando a camada está completa, a 'Placa de Depósito' (*Open Plate*) se abre, depositando as 4 caixas sobre o palete. O elevador desce a altura de uma camada e a placa se fecha.
5. Ciclo Completo: O processo de formação de camada e depósito se repete 3 vezes. Ao completar 3 camadas (12 caixas no total), o palete é considerado pronto, baixado ao nível da esteira e liberado para a Etapa 3.

A coordenação entre os sensores, o empurrador e a abertura da escotilha segue uma lógica sequencial rigorosa. O algoritmo de controle, baseado em contadores em cascata, é apresentado na Figura 17.

Figura 17 – Fluxograma da lógica de sequenciamento da paletização do na Figura



Fonte: O Autor (2025)

3.3.3. Etapa 3: Armazenamento e Recuperação Automatizado (AS/RS)

Esta etapa final gerencia o armazenamento e a expedição dos paletes prontos utilizando o sistema AS/RS (*Automated Storage and Retrieval System*).

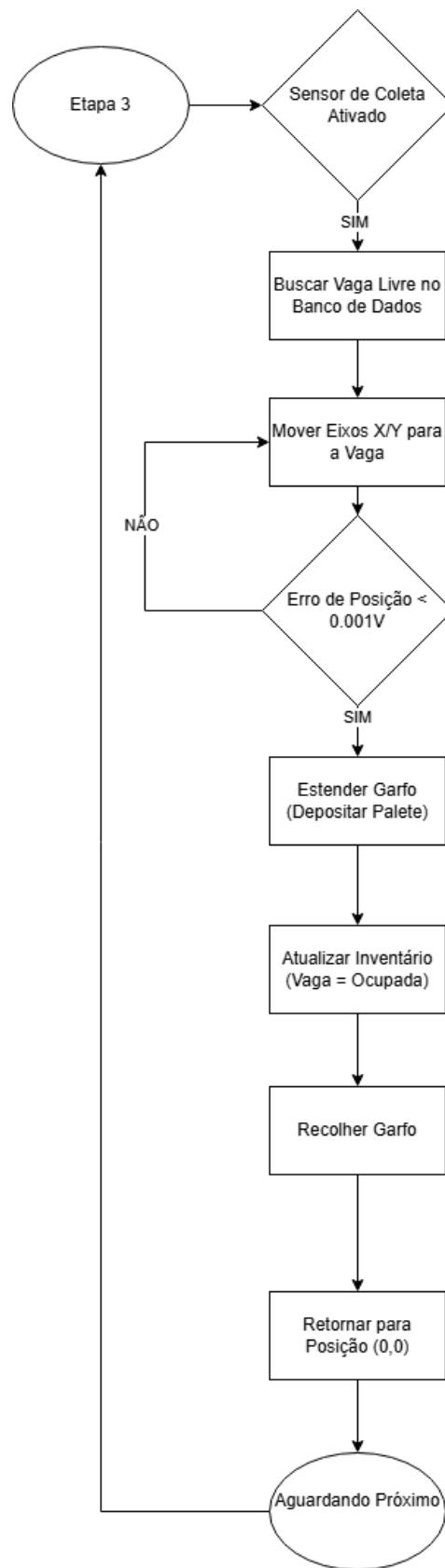
- Sinal de Início: O palete pronto, vindo da Etapa 2, avança pela esteira de roletes e para em um ponto de coleta, detectado por um 'Sensor Retrorreflexivo'.
- Ciclo de Armazenamento:
 - O 'Transelevador' (*Stacker Crane*), o robô central do AS/RS, move-se em seus eixos horizontal (X) e vertical (Y) até a posição de coleta.
 - Ele estende seus garfos, coleta o palete e o retrai.
 - O sistema identifica a próxima “vaga” vazia disponível (dentre as 54 posições).
 - O Transelevador se desloca até a coordenada (X, Y) da vaga livre, estende os garfos, deposita o palete e retrai os garfos.
 - O robô então retorna à sua posição inicial, aguardando o próximo palete.

A sequência detalhada e as condições lógicas que governam o Ciclo de Armazenamento do Transelevador são apresentadas no fluxograma da Figura 18.

- **Ciclo de Retirada (Expedição):**

- Quando o modo de retirada é ativado, o Transelevador identifica o próximo palete a ser retirado.
- Ele se move até a posição (X, Y) daquele palete, estende os garfos, coleta-o e retrai.
- O robô se desloca até uma posição de descarga definida, no lado oposto da estante, e deposita o palete em uma esteira de saída.
- A esteira de saída transporta o palete até um 'Removedor' (*Remover*), que simboliza a expedição e finaliza o processo.

Figura 18 – Fluxograma do ciclo de armazenamento do sistema AS/RS



Fonte: O Autor (2025)

3.4. ESTRUTURA E MODULARIZAÇÃO DO PROGRAMA

A estrutura do programa no TIA Portal é organizada em níveis, usando diferentes tipos de blocos, cada um com uma função específica. A decisão de separar o código em blocos para cada etapa do processo foi intencional. O objetivo foi simplificar o gerenciamento do programa e facilitar qualquer alteração futura.

3.4.1. Tipos de Blocos

3.4.1.1. Blocos de Organização (OBs)

Os Blocos de Organização fazem a ligação entre o sistema operacional do CLP e o programa criado pelo usuário. O bloco principal, **OB1 ("Projeto")**, é executado cicличamente (Ciclo de Varredura) e serve como o ponto de entrada principal do programa. Sua função é organizar a lógica principal, como o estado geral do sistema, e gerenciar a chamada dos Blocos de Função (FBs) que contêm a lógica de processo.

3.4.1.2. Blocos de Função (FBs)

Os Blocos de Função são a base principal da lógica do processo. São blocos de código reutilizáveis que têm a capacidade essencial de guardar seu estado, ou seja, possuem memória interna.

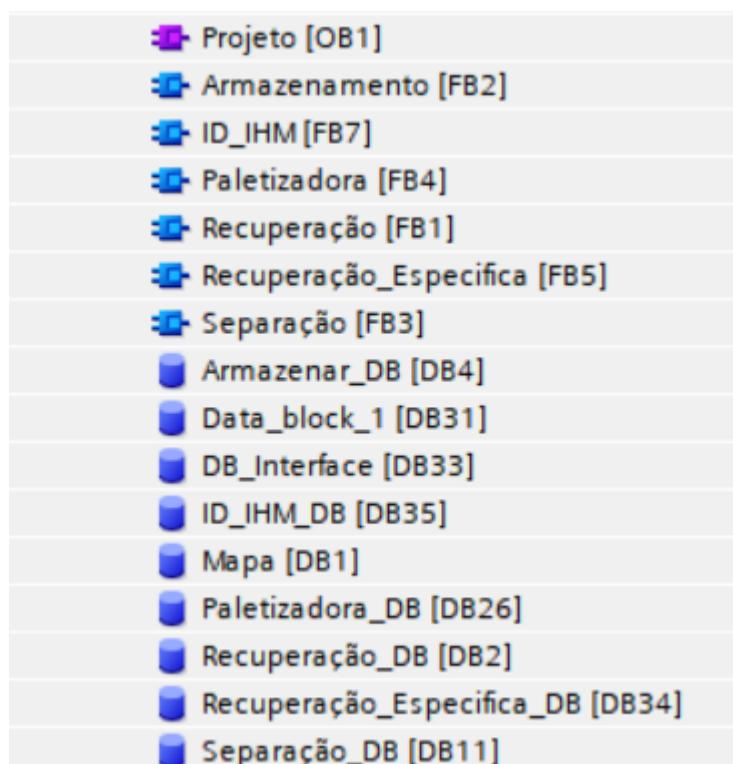
3.4.1.3. Blocos de Dados (DBs)

Os Blocos de Dados são usados apenas para guardar dados e não contêm lógica de programa. Foram usados dois tipos:

- DBs de Instância (iDBs): Bloco de Dados de Instância (iDB) (ex: Armazenar_DB). Este iDB funciona como a "memória particular" do FB, guardando os valores de suas variáveis internas (estáticas) entre os ciclos de *scan*. Esta característica é essencial para processos que seguem uma sequência, como o controle de uma Máquina de Estados Finitos (FSM) no AS/RS ou o sequenciamento da paletizadora, que precisam "lembra" em qual etapa do processo se encontram.

- DBs Globais: Estes são locais centrais para guardar dados, que podem ser acessados por qualquer bloco do programa (OBs, FBs ou FCs). No projeto, eles têm funções importantes para separar (desacoplar) o software :
 - DB "Mapa": Atua como o WMS (Sistema de Gerenciamento de Armazém) do projeto. Este DB armazena o estado (Ocupado), as coordenadas de destino e o conteúdo (IDs das caixas) de todas as 54 vagas da estante.
 - DB "DB_Interface": Funciona como a área de troca de dados entre o CLP e a IHM. A IHM escreve comandos do operador (ex: seleção de modo) neste DB, e os FBs do CLP leem esses comandos para iniciar as ações. A organização hierárquica destes blocos (OBs, FBs e DBs) dentro da árvore de projeto do software é apresentada na Figura 19

Figura 19 - Estrutura de organização dos blocos do projeto no software TIA Portal



Fonte: O autor (2025).

Para detalhar a organização do software, a Tabela 4 apresenta o mapeamento completo dos blocos de programa desenvolvidos, descrevendo seus tipos, associações e funções específicas dentro da lógica de controle.

Tabela 4 - Mapeamento de Blocos de Programa no TIA Portal

Nome do Bloco	Tipo	Bloco de Dados Associado	Função no Projeto
Projeto	OB1	N/A	Bloco principal; executa ciclicamente e chama os FBs de processo.
Separação	FB	Separação_DB (iDB)	Encapsula toda a lógica da Etapa 1: classificação de caixas por tamanho (altura).
Paletizadora	FB	Paletizadora_DB (iDB)	Encapsula toda a lógica da Etapa 2: sequenciamento do paletizador, contagem de caixas/camadas e rastreabilidade RFID.
ID_IHM	FB	ID_IHM_DB	Encapsula a lógica de busca dos Ids para que apareça na IHM
Armazenar	FB	Armazenar_DB (iDB)	Encapsula a lógica da Máquina de Estados Finitos (FSM) para o modo de armazenamento do transelevador (Etapa 3).
Recuperação	FB	Recuperação_DB (iDB)	Encapsula a lógica da FSM para o modo de Recuperação sequencial de paletes (Etapa 3).
Recuperação_Específica	FB	Recuperação_Específica_DB (iDB)	Encapsula a lógica da FSM para o modo de Recuperação de um palete específico, comandado via IHM (Etapa 3).
Mapa	DB (Global)	N/A	Atua como o WMS do sistema. Armazena o estado (ocupado/livre) e o conteúdo (IDs) de todas as 54 vagas da estante.
DB_Interface	DB (Global)	N/A	Serve como intermediário entre a IHM e o CLP para seleção de modo e comandos do operador.

Fonte: O Autor(2025)

3.4.2. Blocos de Instrução e Funções Usadas no Programa

Para criar a lógica de controle nas etapas seguintes, foi usado um conjunto de instruções e blocos de função padrão da linguagem Ladder (conforme norma IEC 61131-3), disponíveis no TIA Portal. Essas funções são as ferramentas básicas que permitem manipular dados, tomar decisões e controlar sequências. As funções mais importantes usadas no projeto são detalhadas abaixo.

3.4.2.1. Instruções de Movimentação de Dados

- MOVE (Mover): Esta é uma das instruções mais básicas, usada para copiar um valor de um lugar para outro. No projeto, ela foi usada em várias situações, como:
 - Definir a velocidade analógica das esteiras (Etapa 1), movendo o valor fixo 2.0 para a memória %MD100 ("Velocidade").
 - Zerar contadores ou listas, movendo o valor 0 para a variável de destino.
 - Transferir o ID lido pelo leitor RFID (%ID46) para a lista de dados do palete.
- MOVE_BLK (Mover Bloco): Uma variação do MOVE usada para mover grande quantidade de dados de uma vez, como *arrays* inteiros. Foi usada na Etapa 2 para transferir a lista completa de 12 IDs de caixas do *array* local do paletizador para o *array* de transferência no DB_Interface ao final do ciclo.

3.4.2.2. Instruções de Comparação

- IN_RANGE (Em Faixa): Usada na Etapa 1 para classificar as caixas. Esta função checa se um valor de entrada (o sinal analógico do %ID30) está dentro de uma faixa definida por um limite mínimo e máximo (ex: entre 0.1 V e 1.4 V para caixas pequenas).
- CMP > (Maior Que) e CMP < (Menor Que): Funções de comparações diretas. O CMP > foi usado na Etapa 1 para identificar caixas grandes (sinal > 3.0 V). O “CMP <” foi

essencial na rotina de validação de posição do AS/RS (Etapa 3), para checar se o erro de posição era menor que o limite aceitável (ex: 0.001 V).

3.4.2.3. Instruções de Temporização (Timers)

- TON (Temporizador com Atraso na Ligação): Esta função atrasa para ligar sua saída (Q). A saída só se torna verdadeira após a entrada (IN) permanecer verdadeira pelo tempo pré-estabelecido (PT). Foi usada para:
 - Sincronizar o desviador na Etapa 1, com um PT de 100 ms para compensar a distância entre o sensor e o atuador.
 - Garantir que atuadores fiquem ligados por um tempo, como o atraso de 150 ms do empurrador na paletizadora.
- T-OFF (Temporizador com Atraso na Desligação): Esta função atrasa para desligar sua saída. A saída (Q) liga imediatamente com a entrada (IN), mas continua ligada por um tempo pré-estabelecido (PT) *depois* que a entrada (IN) desliga. Foi usada para manter as esteiras de desvio ligadas por um tempo após o sensor ser liberado.

3.4.2.4. Instruções de Contagem (Counters)

- CTU (Contador Crescente): Esta função conta pulsos (transições de falso para verdadeiro) na sua entrada (CU). Sua saída (Q) é ativada quando o valor de contagem (CV) atinge o valor pré-estabelecido (PV). Foi a base da lógica de sequência da paletizadora (Etapa 2), com contadores ligados em sequência para contar caixas (PV=2) e camadas (PV=3).

3.4.2.5. Instruções de Detecção de Borda

- P_TRIGGER (Detector de Borda de Subida): Uma função importante para lógica sequencial. Ela dá um pulso de saída que dura só um ciclo de *scan* quando sua entrada detecta uma transição de falso para verdadeiro. Foi essencial na Etapa 2 para garantir que a lógica

de contagem de caixas e o armazenamento do ID do RFID rodasse *só uma vez* por caixa, no momento em que ela é detectada pelo sensor %I12.4.

3.4.2.6. Instruções Matemáticas

- ADD (Adição): Usada para realizar cálculos, como aumentar o índice (a posição) da lista de IDs de caixas na Etapa 2.
- SUB (Subtração) e ABS (Valor Absoluto): Usadas juntas na Etapa 3 (AS/RS) para criar a rotina de "faixa morta". A SUB calcula o erro (diferença) entre a posição desejada e a posição atual. A ABS pega o valor desse erro (sem o sinal de negativo, se tiver), permitindo checar se ele está dentro do limite aceitável.

3.4.3. Lógica de Controle: Etapa 1 – Separação e Classificação

A lógica da primeira etapa do processo, dentro do FB Separação, é responsável por receber caixas misturadas e classificá-las por altura.

3.4.3.1. Partida e Controle de Velocidade Analógico

O início do processo começa com um botão de "Ligar" (%I10.0), que ativa uma memória interna "Sistema Ligado" (%M10.0). Esta memória funciona como uma condição principal para a operação de toda a célula.

Uma decisão de projeto importante nesta etapa foi usar controle de velocidade analógico para as esteiras de caixas. Em vez de uma saída digital (liga/desliga), a lógica usa uma instrução MOVE para escrever um valor fixo do tipo REAL (2.0) em uma variável de memória (%MD100 "Velocidade"). Esta variável é então ligada nas saídas analógicas (%QD30, %QD34, etc.) que controlam a velocidade das esteiras.

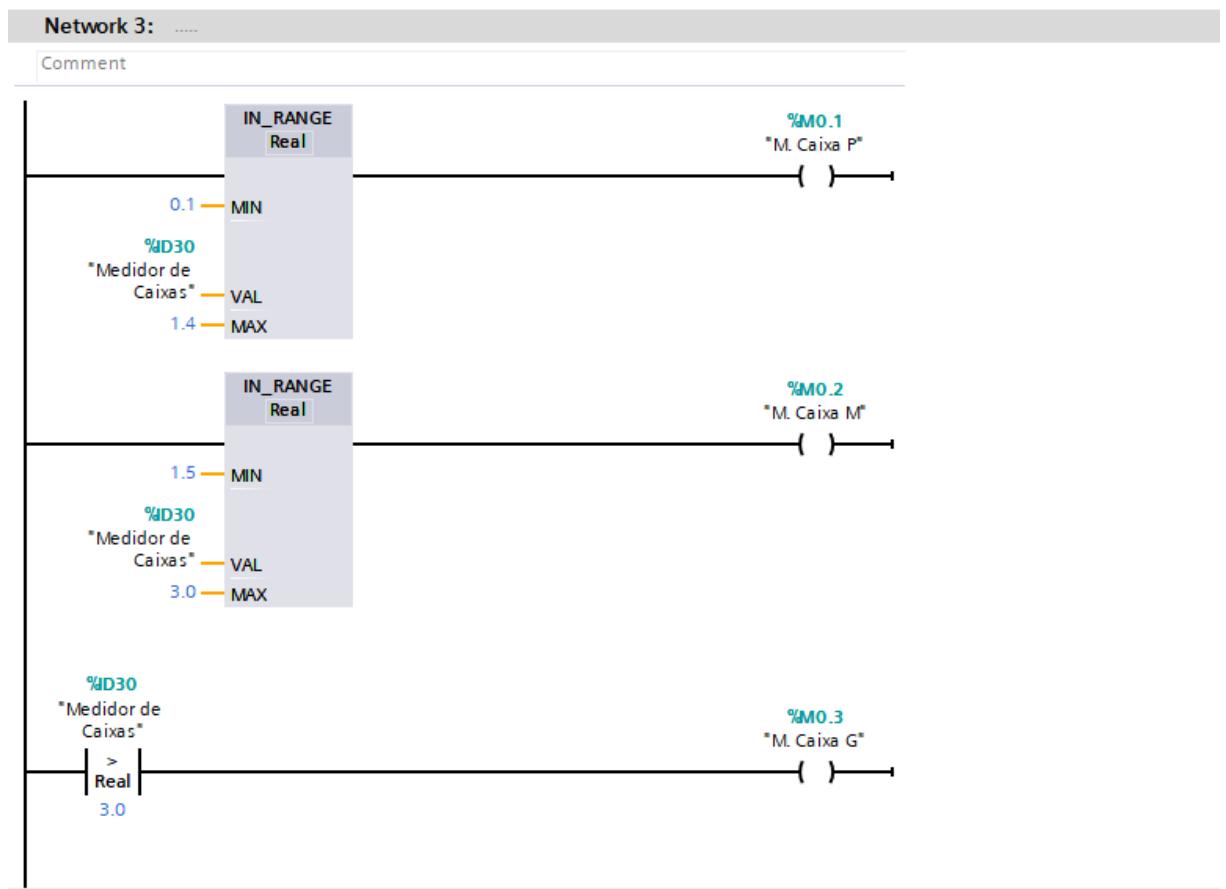
3.4.3.2. Lógica de Classificação Analógica

A classificação é realizada através da leitura do sinal analógico de tensão proveniente da entrada %ID30 ("Medidor de Caixas"). O programa converte este sinal em valores reais para determinar o tamanho da caixa através de comparadores lógicos:

- Caixa Pequena e Média: Utiliza-se a instrução IN_RANGE para verificar se o valor de tensão está dentro das faixas específicas (0.1 a 1.4 para Pequena; 1.5 a 3.0 para Média), ativando as memórias auxiliares correspondentes (%M0.1 e %M0.2).
- Caixa Grande: Utiliza-se a instrução de comparação CMP > (Maior que 3.0) para identificar caixas grandes.

A implementação dessa lógica de comparação analógica em linguagem Ladder, utilizando os blocos IN_RANGE para definir as faixas de tensão, é demonstrada na Figura 20.

Figura 20 - Lógica Ladder para classificação de caixas por altura



Fonte: O Autor (2025).

3.4.3.3. Lógica de Intertravamento e Atuação

Após a classificação, o sistema define a rota. A estrutura de programação para o desvio segue um padrão rigoroso de intertravamento:

- Lógica (Pequena e Média): A lógica de atuação para a Caixa Média (%M0.5 "A.S Media") é idêntica à da Caixa Pequena (%M0.4 "A.S Pequena"). Ambas utilizam contatos

Normalmente Abertos (NA) da sua respectiva memória de classificação em série com contatos Normalmente Fechados (NF) das outras opções.

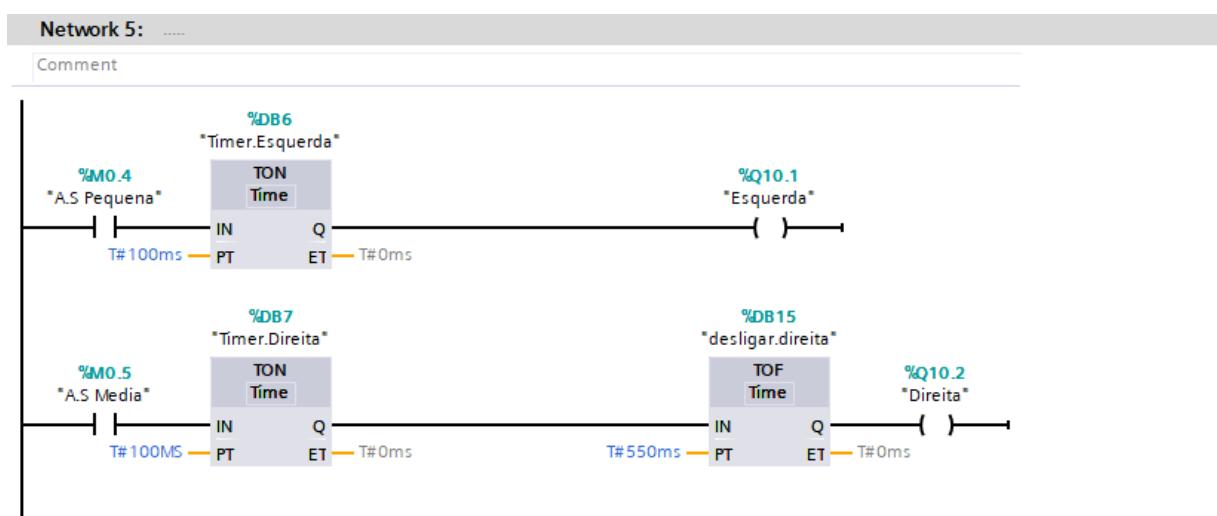
- Fluxo da Caixa Grande: Para a caixa grande, observou-se que não é necessário uma lógica ativa de atuação (como uma memória "A.S Grande" acionando uma saída física).

3.4.3.4. Sincronização Temporal (Atuação dos Desviadores)

A ativação física dos desviadores (*Pop-Up Wheel Sorter*) exige compensação de atraso devido à distância física entre o sensor e o atuador, bem como ajustes baseados nas dimensões da carga.

- TON (*Timer On-Delay*): Tanto para a saída Esquerda (%Q10.1) quanto para a Direita (%Q10.2), a ativação não é imediata. A lógica aciona um temporizador TON com um tempo pré-ajustado (PT) de 100 ms para compensar o deslocamento inicial.
- TOF (*Timer Off-Delay*): Especificamente para o desvio à Direita (Caixas Médias), foi adicionado um temporizador TOF com tempo de 500 ms. Devido à maior extensão longitudinal da caixa média, essa compensação garante que o atuador permaneça ativo por tempo suficiente para permitir a passagem completa do volume antes do desligamento ou acionamento de um novo sensor. A implementação lógica destes temporizadores de compensação para o controle preciso dos desviadores é apresentada na Figura 21.

Figura 21 – Lógica do desviador utilizando temporizadores TON e TOF



Fonte: O Autor (2025)

3.4.3.5. Controle das Esteiras de Saída

Para as esteiras secundárias (Pequena, Média e Grande), utiliza-se sensores nas pontas (ex: Sensor E.M %I10.4) ligados a temporizadores TOF de 5 segundos. Isso mantém os motores acionados tempo suficiente para garantir a entrega completa da caixa antes do desligamento automático.

3.4.4. Lógica de Controle: Etapa 2 – Paletização e Rastreabilidade RFID

A lógica da segunda etapa, dentro do FB Paletizadora, é um processo sequencial complexo que controla o posicionamento de paletes, a contagem de caixas e camadas, e a gravação de dados de rastreabilidade.

3.4.4.1. Sequência de Alimentação de Paletes

O ciclo da paletizadora só pode começar quando um palete vazio está na posição certa. Esta condição é checada por dois sensores na base do elevador: %I11.1 ("Sensor P Traseiro") e %I11.2 ("Sensor P Frontal"). A lógica exige que os dois estejam ativos (%I11.1 E %I11.2) para ligar a memória %M11.2 ("Posicao_Correta"). Usar dois sensores garante que o palete está não apenas presente, mas também alinhado.

A esteira de alimentação de paletes (%Q10.7) possui uma trava de segurança: ela só é ligada se a memória %M11.1 ("Sem_Paleta") estiver ativa (ou seja, %M11.2 for falsa), para evitar que um novo palete seja chamado enquanto a estação estiver ocupada.

3.4.4.2. Sequenciamento de Camadas e Controle do Elevador

A automação é controlada por uma lógica de contadores aninhados (CTU) e um controle do movimento vertical. O movimento é definido pela combinação de %Q11.2 (Subir), %Q11.3 (Descer) e %Q11.4 ("Ao Máximo"), sendo esta última o seletor de amplitude:

- Subida Inicial: A combinação %Q11.2 + %Q11.4 é acionada quando %M11.2 ("Posição_Correta") está ativa. O elevador sobe no curso total. A entrada %I11.3 ("Sensor P Baixo") É um sensor de confirmação que indica a chegada do elevador à base. e junto com um temporizador TOF de 500ms garantem a estabilização na altura de trabalho.

- Descida Gradual: Após a deposição de cada camada, apenas %Q11.3 (Descer) é acionada por um tempo controlado (sem o "Ao Máximo"). Isso faz o elevador descer apenas a altura de uma camada, liberando espaço para a próxima.
- Descida Final: Quando o palete está pronto, a combinação %Q11.3 + %Q11.4 é acionada, forçando o curso total até que o sensor %I11.3 seja ativado novamente na base.

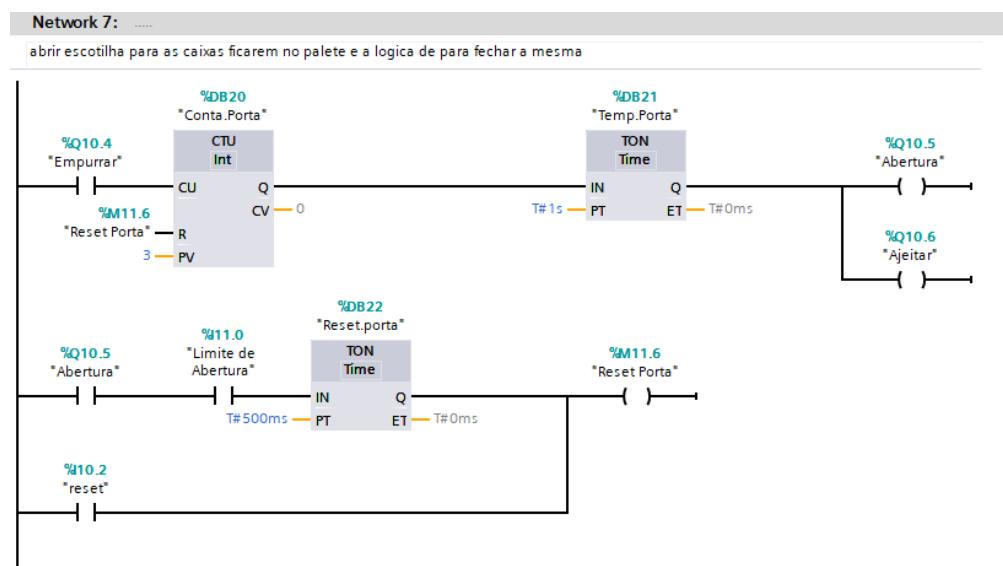
A automação do paletizador é controlada por uma lógica de Contadores Crescentes (CTU) ligados em sequência, garantindo a formação precisa de fileiras e camadas, sendo os seguintes contadores:

- Contador de Caixas (PV=2)
 - Função: É o contador primário, responsável por agrupar as caixas em fileiras de duas unidades.
 - Contagem: Utiliza os pulsos do sensor %I10.6 ("Contador Paletizadora") como entrada. Quando sua contagem acumulada (CV) atinge o valor pré-estabelecido (PV=2), sua saída (Q) é ativada.
 - Atuação: Esta saída Q aciona diretamente o atuador %Q10.4 ("Empurrar"), que move as duas caixas para a área de deposição.
 - Ciclo de Reset: Para garantir que o braço empurrador complete sua ação de ida e volta, um temporizador TON de 150 ms é utilizado após o sensor de limite do empurrador (%I10.7). Somente após este atraso é que a memória de reset %M11.5 é acionada, zerando o contador e permitindo a entrada das próximas duas caixas.
- Contador de Camadas (PV=2)
 - Função: Conta as fileiras formadas para determinar o momento exato de depositar uma camada completa.
 - Aninhamento: Este CTU é ligado em série com o primeiro. Ele usa o evento de "Empurrar" (a saída Q do primeiro contador ou o sinal %Q10.4) como sua entrada de contagem.
 - Contagem: Ao chegar em PV=3, o sistema identifica que 3 fileiras foram formadas (totalizando 6 caixas), ou seja, uma camada completa está pronta.
 - Atuação: A saída deste contador aciona duas saídas cruciais para a deposição:

- %Q10.6 ("Ajeitar"): Um atuador lateral de compactação é ligado para garantir o alinhamento das caixas.
- %Q10.5 ("Abertura"): A escotilha é aberta para depositar a camada de 6 caixas sobre o palete.

A implementação em Ladder dessa lógica de sequenciamento, que coordena o contador de camadas com o acionamento das saídas de abertura e alinhamento, é apresentada na Figura 22.

Figura 22 – Exemplo de lógica de Controle da abertura da placa e alinhamento



Fonte: O Autor (2025).

- Contador de Paleta Completo (PV=2)
 - Função: Monitora a finalização do processo de empilhamento.
 - Contagem: Este CTU conta os eventos de "Abertura" (%Q10.5).
 - Finalização: Ao chegar em PV=2 (indicando 2 camadas depositadas), ele ativa a memória %M11.3 ("M.Descer"). Esta memória sinaliza o fim do ciclo de paletização e dispara o comando para a descida total do elevador.

3.4.4.3. Lógica de Rastreabilidade e Agregação de Dados (RFID)

A função crítica desta etapa é a rastreabilidade: cada caixa que entra na paletizadora possui um ID único que deve ser rastreado e vinculado digitalmente ao palete que está sendo formado. Este processo garante que o sistema saiba exatamente o conteúdo de cada paleta.

armazenado. A lógica foi construída em uma sequência sincronizada de detecção, indexação e armazenamento.

3.4.4.3.1. Detecção e Sincronização (P_TRIG)

O processo inicia com a detecção física da caixa através do sensor difuso **%I12.4** ("Sensor Leitor Tag") localizado na entrada da paletizadora.

Captura de Borda de Subida: O sinal deste sensor não é utilizado diretamente. Ele é conectado a um bloco P_TRIG (*Positive Edge Trigger*). Esta escolha de programação é fundamental: o P_TRIG gera um pulso de apenas um ciclo de scan no exato momento em que a caixa toca o sensor. Isso impede que o sistema conte a mesma caixa múltiplas vezes enquanto ela desliza pela frente do sensor, garantindo que a lógica de registro rode apenas uma vez por objeto.

3.4.4.3.2. Lógica de Indexação (O Contador como indexador)

O pulso gerado pelo P_TRIG aciona simultaneamente uma instrução ADD (Adição). Esta instrução soma 1 à variável interna Contador_de_Caixas. Nesta lógica, o contador não serve apenas para totalizar a produção. Ele atua como um ponteiro para a organização dos dados na memória, indicando em qual "linha" da lista o próximo ID deve ser gravado.

3.4.4.3.3. Armazenamento Dinâmico em Array (Endereçamento Indireto)

No mesmo ciclo de scan, o ID numérico da caixa é lido pela entrada %ID46 ("Leitor ID") proveniente do sistema RFID. A lógica utiliza a instrução MOVE para transferir este ID para o Banco de Dados da paletizadora, utilizando a técnica de endereçamento indireto.

O código implementado é estruturado da seguinte forma:

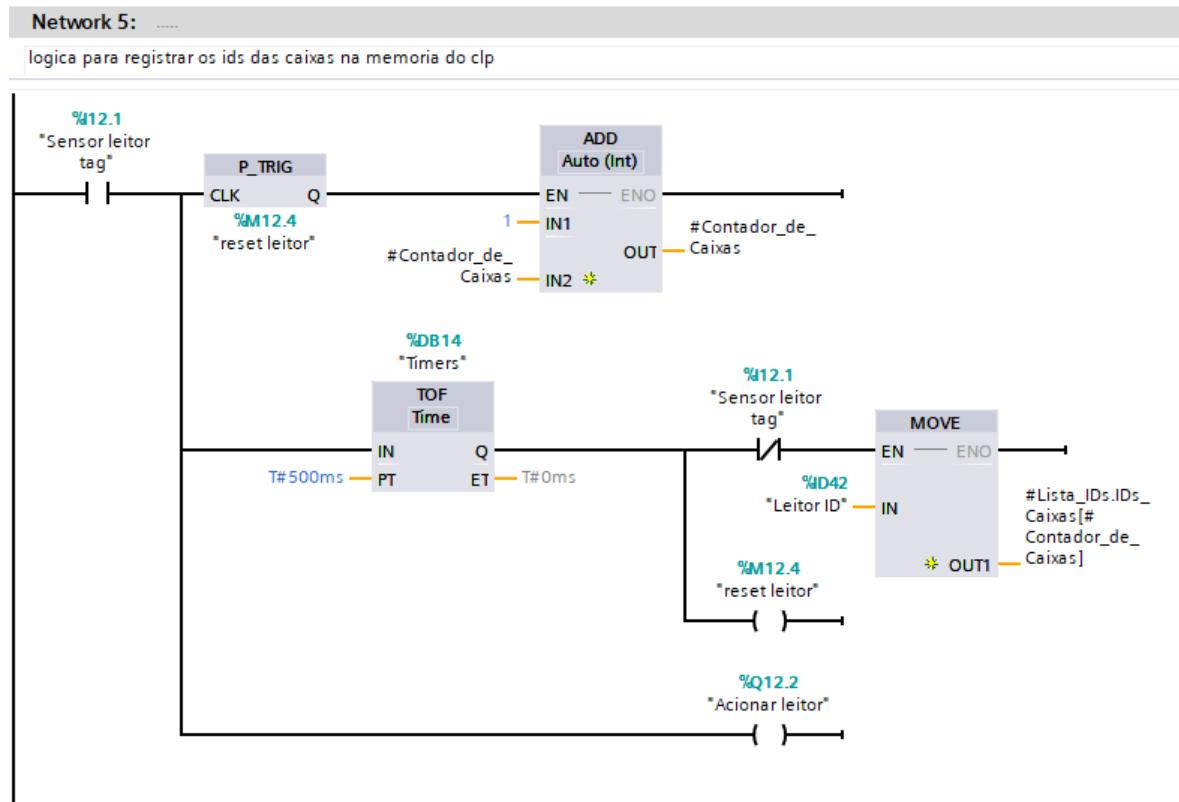
Código: *Lista_IDs#[Contador_de_Caixas]*

Esta estrutura com pontos e colchetes orienta o CLP a navegar hierarquicamente pela memória para salvar o dado no local correto:

- **Lista_IDs (O Array de Dados):** É a variável de destino, estruturada como um Array (vetor). Imagine uma lista ou planilha com 12 linhas numeradas, reservada exclusivamente para guardar os IDs deste palete.
- **# [Contador_de_Caixas]:** Esta é a parte variável do comando. O valor atual do contador (que está entre colchetes) define a posição exata de gravação.

A implementação prática dessa técnica no código Ladder, utilizando a instrução MOVE com indexação variável para gravar os dados no Array, é apresentada na Figura 23.

Figura 23 – Lógica Ladder para registro de dados RFID



Fonte: O Autor (2025)

Exemplo: Se é a 1^a caixa, o contador vale 1. O código entende Lista_IDs[1] e grava o ID na primeira linha.

Exemplo: Se é a 5^a caixa, o contador vale 5. O código entende Lista_IDs[5] e grava na quinta linha.

Dessa forma, o sistema preenche o vetor sequencialmente (de 1 a 12), criando uma lista digital completa e ordenada das caixas contidas no palete, sem risco de sobreposição de dados.

3.4.4.3.4. Transferência de Dados para a Etapa 3

No final do ciclo, quando a memória %M11.3 ("Palete_Pronto") é ativada, a lógica faz a troca de dados principal entre a Etapa 2 e a Etapa 3. Uma instrução MOVE_BLK (Mover Bloco) é usada para copiar o *array* completo Paletizadora.Lista_IDs (agora com 12 IDs) para o *array* Lista_Transferencia no DB_Interface. Logo depois da cópia, a Lista_IDs local é zerada (movendo Lista_Vazia para ela) para se preparar para o próximo palete.

Isso transfere a lista de dados do palete para uma área comum, onde o FB da Etapa 3 (AS/RS) pode ler esse dado quando estiver pronto para armazenar o palete.

3.4.5. Lógica de Controle: Etapa 3 – Sistema AS/RS e Máquina de Estados (FSM)

O controle do Sistema de Armazenagem e Recuperação Automatizado (AS/RS) é a etapa mais difícil do projeto, pois controla vários eixos do transelevador e três modos de operação diferentes. Como explicado na Seção 2.9, uma lógica Ladder comum não seria uma boa ideia para esta tarefa. Por isso, a solução usada foi uma Máquina de Estados Finitos (FSM).

A FSM força o transelevador a estar sempre em um único estado operacional definido (ex: AGUARDANDO , MOVENDO , COLETANDO). As mudanças de um estado para outro são controladas por condições lógicas exatas (sensores, timers, comandos), garantindo um processo forte. A lógica da FSM está separada em três FBs diferentes: Armazenar, Recuperação e Recuperação_Especificada.

3.4.5.1. Estrutura de Dados de Gerenciamento

O centro da lógica do AS/RS é o Bloco de Dados Global DB "Mapa". Este DB funciona como o WMS (Sistema de Gerenciamento de Armazém) do projeto. Ele é organizado como um *array* (uma lista) de 54 posições, onde cada posição da lista corresponde a uma vaga na estante. Cada item dessa lista é um Tipo de Dado de Usuário (UDT), uma estrutura de dados personalizada que guarda todas as informações daquela vaga específica. A estrutura completa do tipo de dado de usuário (UDT) criado para gerenciar cada vaga do armazém, conforme implementado no TIA Portal, é detalhada na Figura 24. Complementarmente, a Tabela 5 descreve a função técnica de cada variável (*Tags*) que compõe essa estrutura UDT.

Figura 24 - Estrutura do Bloco de Dados "Mapa" (WMS) configurada no TIA Portal

	Name	Data type	Offset	Start value
	Static			
	posições	Array[1..6, 1...]	0.0	
	posições[1,1]	"Coordenadas"	0.0	
	Coordenada_X	Real	0.0	1.45
	Altura_Navega...	Real	4.0	0.8
	Altura_Deposit...	Real	8.0	0.5
	Ocupado	Bool	12.0	false
	Vaga	Int	14.0	1
	Conteudo	"UDT_ConteudoPal..."	16.0	
0	IDs_Caixas	Array[1..12] of DInt	16.0	
1	IDs_Caix...	DInt	16.0	0
2	IDs_Caix...	DInt	20.0	0
3	IDs_Caix...	DInt	24.0	0
4	IDs_Caix...	DInt	28.0	0
5	IDs_Caix...	DInt	32.0	0
6	IDs_Caix...	DInt	36.0	0
7	IDs_Caix...	DInt	40.0	0
8	IDs_Caix...	DInt	44.0	0
9	IDs_Caix...	DInt	48.0	0
10	IDs_Caix...	DInt	52.0	0
11	IDs_Caix...	DInt	56.0	0
12	IDs_Caix...	DInt	60.0	0
13	posições[1,2]	"Coordenadas"	64.0	

Fonte: O autor (2025).

Tabela 5 - Estrutura do UDT (User-Defined Type) "ty_Coordenada" no DB "Mapa"

Variável	Tipo de Dado	Descrição
Coordenada_X	REAL	Armazena o valor analógico (tensão 0-10V) do setpoint do eixo X para esta vaga.
Altura_Navegacao	REAL	Armazena o setpoint superior do eixo Y (altura de folga). Define a posição ligeiramente acima da estante, utilizada para entrar com o palete carregado (no armazenamento) ou para sair com ele suspenso (na recuperação).
Altura_Deposito	REAL	Armazena o setpoint inferior do eixo Y (altura de apoio). Define a posição levemente abaixo da estante, sendo o nível para onde o garfo deve descer para apoiar o palete (armazenar) ou onde deve entrar vazio para buscá-lo (recuperação).
Ocupado	BOOL	Flag de status da vaga (True = Ocupada, False = Livre). É a variável principal para o controle de inventário.
Vaga	INT	Identificador numérico único da vaga (1 a 54).
IDs_Caixas_Armazenadas	ARRAY[1..2] OF DINT	Array que armazena os 12 IDs das caixas (a "lista") do palete que ocupa esta vaga.

Fonte: O Autor (2025)

3.4.5.2. Definição dos Estados da FSM

O sistema utiliza coordenadas X e Y baseadas em tensões analógicas para posicionar o transelevador. O inventário é mantido no DB "Mapa", que contém as posições (6 linhas x 9 colunas = 54 vagas), o status (Ocupado), e os IDs das caixas. Para controlar a sequência lógica do processo, foram definidos estados operacionais específicos, cujas ações e finalidades são detalhadas na Tabela 6.

Tabela 6 – Função do estados na AR/RS

Estado (INT)	Nome do Estado	Ações e Finalidade
0	Ocioso / Posição Home	Estado Base. Move os eixos X (%QD54) e Y (%QD58) para 0.0 V (Posição de segurança). Aguarda a seleção de um modo (Armazenar/Recuperação).
1	Verificação de Posição Inicial	Confirma, via lógica de subtração e valor absoluto (SUB/ABS), que os eixos estão precisamente na Posição base.
10	Coleta / Movimentação Inicial	Armazenar: Move os eixos para a Altura_Navegação da vaga-alvo. Recuperação: Move os eixos para a Altura_Depósito da vaga-alvo.
20	Confirmação de Posição (Chegada)	Espera Ativa. Confirma que os eixos X e Y chegaram ao setpoint da vaga-alvo, usando novamente a rotina SUB/ABS.
30	Posicionamento Vertical	Ajuste Fino. Move o eixo Y da Altura_Navegação para a Altura_Depósito (Armazenar) ou vice-versa (Recuperação).
40	Operação do Garfo	Estende o garfo para depositar ou coletar o palete na estante.
45	Atualização do Inventário	Gravação no WMS. Armazenar: Define Ocupado = TRUE e move o lista de IDs para o DB "Mapa". Recuperação: Define Ocupado = FALSE e limpa os dados da vaga.
50	Retorno do Garfo	Recolhe o garfo. Aguarda o sensor %I11.7 ("no meio") para confirmar que o garfo está totalmente recuado antes de mover os eixos.
60	Retorno à Posição Inicial	Move os eixos X e Y de volta para 0.0 V (Home).
70	Preparação Próximo Ciclo	Lógica de Continuidade. Incrementa os contadores de busca (Linha_Atual/Coluna_Atual) para o próximo ciclo (principalmente no modo Recuperação) e move para o Estado 0.
900-902	Sequência de Reset	Estados de segurança. Ativados manualmente pelo botão de Reset (%I10.2). Movem os eixos para 0.0V e zeram os contadores de linha/coluna, forçando o retorno seguro ao Estado 0.

Fonte:o Autor (2025)

3.4.5.3. Lógica de Busca e Iteração

Estes estados são de decisão e envolvem a iteração (busca sequencial) sobre o DB "Mapa", utilizando variáveis Linha_Atual e Coluna_Atual. protocolo detalhado de iteração para encontrar vagas disponíveis ou ocupadas, dependendo do modo de operação (Armazenar ou Recuperar), é descrito na Tabela 7.

Tabela 7 – Protocolo de busca para cada modo

Estado	Nome do Estado	Ação Detalhada (Busca e Tratamento de Falhas)
5	Busca de Vaga (Armazenar)	<p>Função: Encontrar a primeira vaga onde Ocupado == FALSE (livre).</p> <p>Iteração: O sistema faz um loop (Network 4)sobre o DB "Mapa", checando vaga por vaga, aumentando a coluna e, se necessário, a linha, até encontrar o status FALSE.</p> <p>Sucesso: Salva a vaga-alvo e move para o Estado 10.</p> <p>Falha: Se a vaga estiver TRUE (cheias), o sistema checa a próxima vaga, até achar uma livre.</p>
5	Busca de Vaga (Recuperação)	<p>Função: Encontrar a primeira vaga onde Ocupado == TRUE (para Recuperação automática).</p> <p>Iteração: O sistema itera sobre o DB "Mapa", procurando por TRUE.</p> <p>Sucesso: Salva a vaga-alvo e move para o Estado 10.</p> <p>Falha: Se a vaga estiver FALSE (livre), o sistema checa a próxima vaga, até achar uma ocupada.</p>
5	Busca de Vaga (Específica)	<p>Função: Encontrar uma vaga específica selecionada via IHM (DB_Interface.Vaga_Recuperação)</p> <p>Lógica: O sistema itera até que Mapa[i].Vaga == Vaga_Recuperação E Ocupado seja TRUE.</p>

Fonte: O Autor(2025)

A busca por vagas utiliza as variáveis de controle Linha_Atual (1 a 6) e Coluna_Atual (1 a 9) para atuar como indexadores dentro da matriz de posições do DB "Mapa". O algoritmo

inicia a varredura sempre pela primeira posição (Linha 1, Coluna 1) e segue a lógica de iteração descrita abaixo:

1. Busca por Coluna (Iteração Horizontal): O sistema inicia com a Coluna em 1. Após verificar o *status* da vaga, uma instrução ADD incrementa a variável Coluna_Atual em uma unidade. A busca desloca-se horizontalmente ($1,1 \rightarrow 1,2 \rightarrow 1,3\dots$) até cobrir toda a linha.
2. Mudança de Linha (Loop Aninhado): Quando a variável Coluna_Atual ultrapassa o valor 9, um comparador de limite detecta o fim da fileira. A lógica move o valor da coluna de volta para 1 e aciona uma instrução ADD para incrementar a Linha_Atual em uma unidade. A busca passa então para a próxima fileira ($1,9 \rightarrow 2,1$).
3. Reinício do Mapeamento (Reset): Caso a Linha_Atual ultrapasse o valor 6 (fim da estante) sem encontrar uma vaga livre, o sistema reseta ambas as variáveis para 1 e reinicia o ciclo de varredura.

3.4.5.4. Rotina de Validação de Posição Analógica

Um desafio principal no controle do transelevador é saber quando um movimento terminou, pois o controle é analógico (0-10V). Não é confiável comparar direto se o *setpoint* (ex: 5.25V) é igual ao *feedback* de posição atual (ex: %ID38), por causa de erros de cálculo, ruído ou pequenas variações do sensor.

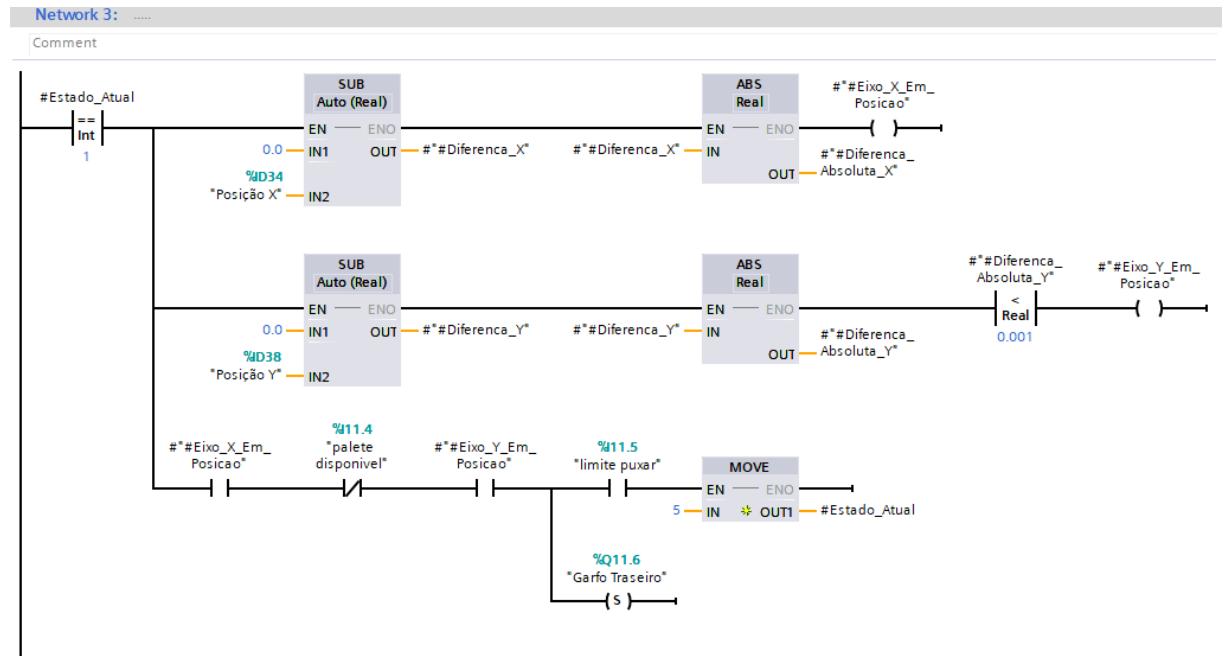
Para resolver isso, foi criada uma rotina de "faixa morta" digital, que é usada em estados de confirmação (como o Estado 1, 20 e 60). Esta lógica funciona assim:

1. Subtração (SUB): O erro entre o *setpoint* desejado (lido do DB "Mapa") e a posição atual (lida de %ID38) é calculado: $\text{Diferenca_X} = \text{Setpoint_X} - \text{Posicao_Atual_X}$.
2. Valor Absoluto (ABS): O valor absoluto do erro é obtido: $\text{Diferenca_Absoluta_X} = |\text{Diferenca_X}|$.
3. Comparação (CMP <): O sistema checa se esse erro é menor que um limite aceitável (ex: 0.001 V): $\text{Eixo_X_Em_Posicao} = (\text{Diferenca_Absoluta_X} < 0.001)$.

A FSM só muda para o próximo estado (ex: do 20 para o 30) quando as condições Eixo_X_Em_Posicao E Eixo_Y_Em_Posicao são ambas verdadeiras. Isso garante que o transelevador esteja fisicamente parado na posição correta antes de fazer a próxima ação (como estender os garfos). implementação da lógica matemática de comparação, que calcula o erro

absoluto entre o setpoint e a posição atual para validar a chegada do transelevador, é apresentada na Figura 25.

Figura 25 – Lógica Ladder para validação de posição analógica com faixa morta



Fonte: O Autor (2025).

3.4.6. Implementação e Lógica da Interface Homem-Máquina (IHM)

A IHM (SIMATIC HMI KTP700 Basic) serve como a central de controle e visualização do sistema. A comunicação entre a IHM e o CLP é totalmente separada através do DB "DB_Interface".

3.4.6.1. Lógica de Seleção de Modo

A IHM permite ao operador selecionar o modo de operação do AS/RS (Armazenar ou Recuperação). Para garantir que só um modo funcione por vez, os botões na tela da IHM usam uma lógica de *Set/Reset*:

- Evento "Press" do Botão "Armazenar":
 - SetBit (Seta Bit): DB_Interface.Armazenamento (ativa este modo na FSM).
 - ResetBit (Reseta Bit): DB_Interface.Recuperação.

- ResetBit (Reseta Bit): DB_Interface.Recuperação_Selecionada.
- Evento "Press" do Botão "Recuperação":
 - SetBit: DB_Interface.Recuperação.
 - ResetBit: DB_Interface.Armazenamento.
 - ResetBit: DB_Interface.Recuperação_Selecionada.
 - Esta lógica garante que só um modo seja ativado por vez.

3.4.6.2. Supervisão do Armazém (Mapa Visual)

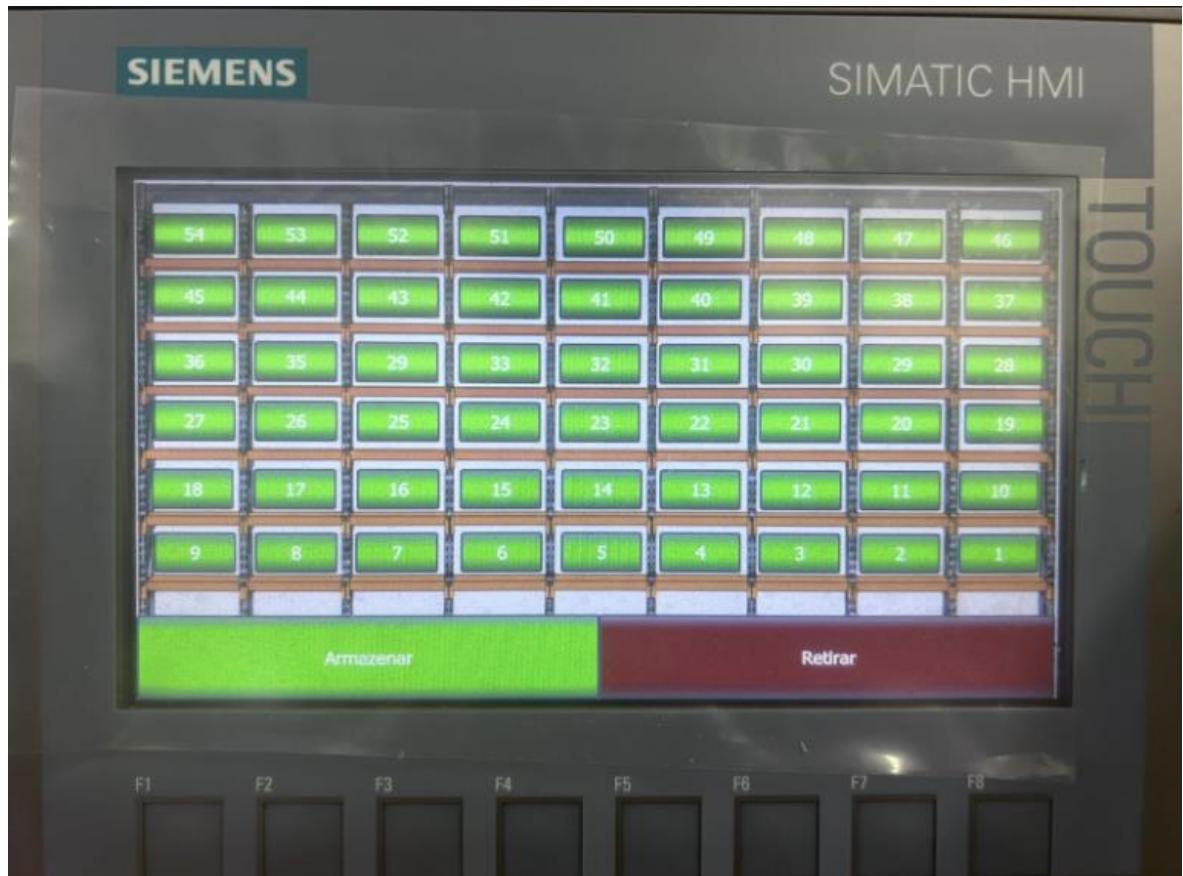
Uma das principais funções da IHM é mostrar um "mapa do armazém" em tempo real, como planejado nos objetivos. Este mapa tem 54 desenhos (botões) que representam as 54 vagas da estante.

O status visual de cada vaga (ex: cor) não é fixo; ele é ligado diretamente (taggeado) à variável Ocupado (BOOL) do seu respectivo item no DB "Mapa" no CLP. A lógica de animação da IHM foi configurada através da propriedade de Aparência (*Appearance*), vinculando a variável booleana de status à cor do objeto:

- Estado 0 (FALSE): Exibe a cor Verde, indicando vaga livre.
- Estado 1 (TRUE): Exibe a cor Vermelha, indicando vaga ocupada.

Isso permite que o operador tenha uma visão rápida e em tempo real do estado de ocupação de todo o armazém. A interface, em seu estado inicial de pronto para uso, é visualizada na Figura 26.

Figura 26 - IHM exibindo o mapa visual de ocupação do AS/RS em estado inicial



Fonte: O Autor (2025)

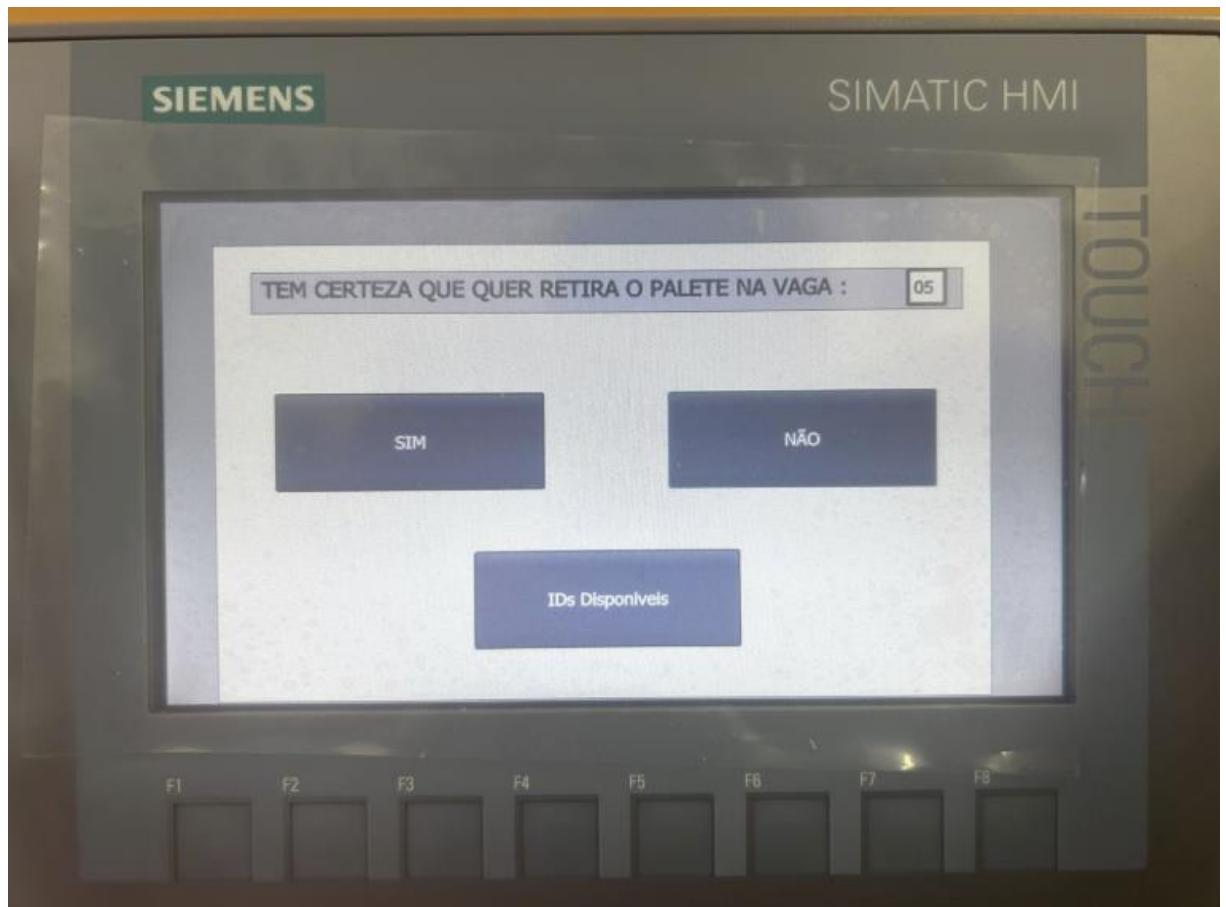
3.4.6.3. Lógica de Recuperação Específica (Interação)

O mapa visual desenvolvido possui funcionalidade interativa, permitindo a operação no modo de "Recuperação Específica". O fluxo de interação entre o operador e a IHM segue a sequência lógica descrita abaixo:

1. Seleção da Vaga: O operador visualiza o mapa e seleciona uma vaga que esteja com status *Vermelho* (Ocupada). Por exemplo, ao clicar no botão representativo da Vaga 10, o sistema inicia o processo.
2. Evento de Clique (*Press*): O acionamento do botão executa, via script da IHM, duas ações simultâneas:
 - SetTag (Definição de Variável): O sistema escreve o número da vaga selecionada (ex: 10) na tag de destino DB_Interface.Vaga_Recuperacao.
 - ActivateScreen (Navegação): O sistema redireciona o operador para uma tela de confirmação (Pop-up), exibindo a mensagem: "Tem certeza que deseja recuperar o palete da Vaga [X]?".

O sistema redireciona o operador para uma tela de confirmação (Pop-up), que é ilustrada na Figura 27, e tem como objetivo garantir que a ação de retirada não seja acionada por engano.

Figura 27 - Tela de confirmação da ordem de recuperação de palete específico



Fonte: O Autor (2025)

3. Confirmação da Ordem: Na tela de confirmação, o operador tem duas opções:

- Botão "Sim" (Confirmar): Ao pressionar, executa a função SetBit na tag DB_Interface.Recuperação_Selecionada. Este bit atua como um *trigger* (gatilho) para a FSM do CLP iniciar o ciclo de busca. Logo após, a função ActivatePreviousScreen retorna à tela principal para monitoramento.
- Botão "Não" (Cancelar): Ao pressionar, a ação é abortada e o sistema retorna à tela anterior sem alterar o status do CLP.

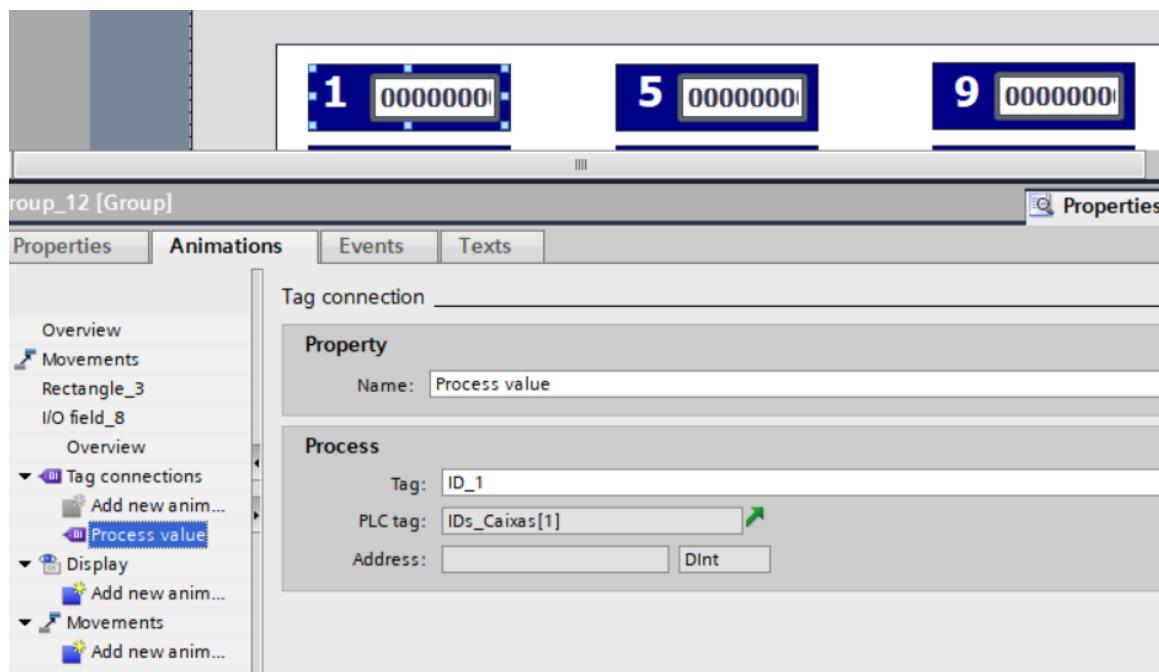
3.4.6.4. Visualização de IDs

Complementando as funcionalidades de supervisão, o sistema oferece ao operador a capacidade de consultar o inventário detalhado de qualquer palete armazenado. Esta funcionalidade é crítica para garantir a rastreabilidade total das caixas e validar o conteúdo da vaga selecionada.

A interface de consulta, exibe 12 campos que correspondem aos IDs de caixa armazenados na estrutura de dados do paleta. O acionamento desta tela, que geralmente ocorre após a seleção de uma vaga para retirada específica, exige que o CLP copie os dados da matriz de inventário (DB Mapa) para uma memória (DB_Interface) de fácil acesso à IHM.

Para que os dados sejam exibidos corretamente, cada um dos 12 campos no design da IHM é individualmente taggeado a uma posição específica nesta memória de transferência. A Figura 28 demonstra o processo de vinculação das *tags*, que permite à IHM ler e exibir o conteúdo atual do paleta requisitado.

Figura 28 - Vinculação das tags da IHM aos campos de exibição dos IDs



Fonte: O Autor (2025)

4. RESULTADOS EXPERIMENTAIS E DISCUSSÃO TÉCNICA

A validação funcional de sistemas de automação complexos, tradicionalmente realizada no chão de fábrica durante as fases finais de comissionamento, evoluiu significativamente com o advento das tecnologias da Indústria 4.0. Este capítulo detalha os resultados obtidos no desenvolvimento da célula logística automatizada, utilizando a metodologia de Comissionamento Virtual. A integração entre o controlador lógico programável (CLP) Siemens S7-1200 e o geminado digital (Digital Twin) desenvolvido no ambiente Factory I/O permitiu uma análise exaustiva do comportamento dinâmico, lógico e físico do sistema antes de qualquer implementação física.

Os resultados aqui apresentados transcendem a verificação binária de funcionalidade. A discussão aprofunda-se nas nuances da interação entre a lógica de controle discreta e a física simulada — inércia, atrito, gravidade e temporalidade — que o ambiente Factory I/O reproduz com alta fidelidade. A análise está estruturada sequencialmente, acompanhando o fluxo produtivo das caixas: desde a separação e classificação, passando pela paletização e rastreabilidade RFID, até o armazenamento e recuperação automatizados (AS/RS). A abordagem de validação focou em três pilares fundamentais:

- Estabilidade Dinâmica: A capacidade do sistema de manipular cargas físicas sem danos ou travamentos.
- Integridade de Dados: A garantia de que as informações de rastreabilidade (IDs) são processadas e armazenadas sem corrupção, duplicidade ou perda.
- Robustez Lógica: A resiliência do software de controle frente a cenários de borda, falhas de sensores e variabilidade temporal (scan cycle).

4.1. SUBSISTEMA DE SEPARAÇÃO E CLASSIFICAÇÃO

O primeiro estágio da célula, responsável pela triagem de caixas baseada em altura, apresentou desafios significativos relacionados à física do transporte. A lógica de controle, distribuída no bloco de separação, foi refinada iterativamente com base nos comportamentos observados no modelo virtual.

4.1.1. Comportamento Físico e Ajuste de Velocidade

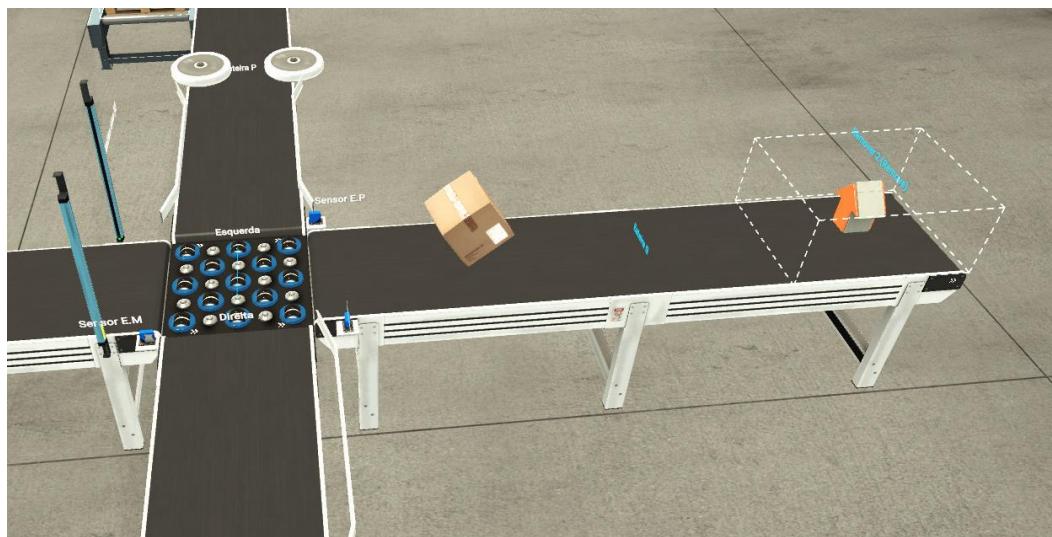
Durante a fase inicial de testes, buscou-se maximizar a produtividade da linha configurando as esteiras para operar próximo à capacidade máxima (10.0V na escala analógica). No entanto, os resultados experimentais demonstraram que a física do sistema impõe limites severos à velocidade de operação, independentemente da capacidade de processamento do CLP. Sendo o maior problema a instabilidade em Alta Velocidade

Ao operar com tensão igual a **2.0V** (em uma escala de 0-10V), pois quanto maior a tensão na esteira, maior seria a velocidade e observou-se um comportamento caótico no momento da classificação. O atuador de desvio (*Pop-up Wheel Sorter*) ao ser acionado, aplica um vetor de força lateral à caixa. Quando somado a um vetor de velocidade longitudinal elevado, a resultante de forças superava o atrito da caixa com o classificador. Os testes revelaram três modos de falha críticos induzidos pelo excesso de velocidade:

- Rotação Descontrolada: Caixas, especialmente as de base quadrada (Grandes), entravam em movimento de rotação sobre o próprio eixo vertical (spin) ao tocar as rodas de desvio.
- Colisões de Alto Impacto: A inércia excessiva fazia com que as caixas fossem arremessadas contra as guias laterais das esteiras de saída. Em um cenário real, isso resultaria em danos imediatos a produtos frágeis ou deformação das embalagens.
- Falha de Transferência: Em velocidades muito altas, o tempo de residência da caixa sobre o módulo de desvio era insuficiente para que a transferência lateral se completasse, fazendo com que as caixas fossem quase sempre para a esteira das caixas grande.

Essa instabilidade física, que resultaria em danos ao produto em um cenário real, é demonstrada na Figura 29, onde observa-se a rotação excessiva da carga.

Figura 29 – Instabilidade dinâmica observada em velocidades



Fonte: O Autor (2025).

Utilizou-se a instrução MOVE para fixar o setpoint de velocidade em 2.0V na memória %MD100. Os resultados experimentais que justificaram essa escolha, detalhando os efeitos de cada faixa de tensão na estabilidade da carga, são apresentados na Tabela 8.

Tabela 8 - Comparativo de Comportamento em Função da Velocidade da Esteira

Tensão Aplicada (V)	Estabilidade da Caixa	Comportamento no Desvio	Integridade do Produto (Simulado)
5.0 - 10.0V	Crítica	Arremesso violento, rotação excessiva, colisão com guias.	Alta probabilidade de dano/quebra.
4 - 4.9V	Baixa	Oscilação lateral significativa, correções ocasionais necessárias.	Risco moderado de amassamento.
2.0-3.0V (Ideal)	Alta	Transição suave, manutenção do alinhamento, sem impacto.	Preservação total da carga.
< 1.5V	Alta	Transição muito lenta, redução desnecessária.	Seguro, porém, ineficiente.

Fonte: O Autor (2025)

Esta decisão valida a importância do Comissionamento Virtual: a descoberta desse limite físico em um protótipo real teria custado material e tempo de ajuste mecânico. No ambiente virtual, a correção foi puramente via software, movendo um valor fixo que garante

que, mesmo se o sistema for reiniciado ou barrado (Network 1), a velocidade segura seja restaurada

4.1.2. Precisão da Instrumentação

A classificação dimensional utiliza um sensor de barreira analógico (%ID30) que mede a altura da caixa. A conversão deste sinal contínuo em decisões discretas (Pequena, Média, Grande) exigiu o tratamento de ruídos de sinal e a definição de faixas de tolerância.

Uma descoberta crucial durante os testes foi a instabilidade da leitura quando ausência de caixas. Inicialmente, a lógica considerava qualquer valor abaixo de 1.4V como "Caixa Pequena". Se a lógica fosse simplesmente $\text{Valor} < 1.4$, a ausência de caixa (0V) seria interpretada como uma caixa válida, acionando o desvio para a esquerda indevidamente. Mais grave ainda, a transição entre o fim de uma caixa e o início do vazio gerava um flanco de descida que, por milissegundos, passava pela faixa de "caixa pequena" antes de zerar, causando classificações falsas.

4.1.2.1. Implementação da “Faixa Morta”

Para resolver isso, foi implementado comparadores do tipo IN_RANGE com limites inferiores estritamente definidos, criando uma faixa morta eficaz:

- Caixa Pequena: A lógica só é ativada se o sinal estiver entre 0.1V e 1.4V. O limiar inferior de 0.1V atua como um filtro, ignorando o ruído de fundo e o estado de "vazio".
- Caixa Média: Definida estritamente entre 1.5V e 3.0V.
- Caixa Grande: Definida como qualquer valor maior que 3.0V.

4.1.2.2. Validação Empírica dos Limiares

Os valores de corte (1.4V e 3.0V) não foram teóricos, mas obtidos empiricamente. Durante os testes de passagem, monitorou-se a variável %ID30 online no TIA Portal. Observou-se que a caixa pequena gerava um sinal estável de aproximadamente 1.3V. Ao definir o corte em 1.4V, estabeleceu-se uma margem de segurança de 0.1V.

Da mesma forma, a caixa média operava na faixa de 2.8V, justificando o corte em 3.0V.

A caixa grande, única a ultrapassar 3.0V, simplificou a lógica final ($CMP > 3.0$). Essa calibração fina eliminou 100% dos erros de classificação cruzada (falsos positivos) durante os ensaios de validação.

4.1.3. Lógica de Desvio

A atuação física do desviador (*sorter*) não pode ser instantânea em relação à detecção do sensor. Existe uma distância física entre o feixe de medição e o ponto de atuação das rodas. A gestão desse atraso é crítica para o sucesso da operação.

Foi implementado a lógica de exclusão mútua. As memórias de decisão (%M0.4 Pequena, %M0.5 Média, %M0.6 Grande) foram programadas de forma que a ativação de uma bloqueia imediatamente as outras.

Por exemplo, a linha de código que ativa a memória "A.S Pequena" (%M0.4) possui contatos normalmente fechados (NF) das memórias Média e Grande. Isso previne um cenário de erro catastrófico: se duas caixas passassem muito próximas e o sensor oscilasse, o CLP poderia tentar acionar o desviador para a esquerda e para a direita simultaneamente no mesmo ciclo de scan. O intertravamento lógico impede essa condição física impossível, protegendo os atuadores.

4.1.3.1. Compensação

O resultado mais crítico nesta etapa foi a definição precisa dos tempos de atuação para o desvio, tanto para o início quanto para a duração do acionamento, sendo eles:

- Atraso de Entrada (TON - 100ms): Testes demonstraram que a sincronia entre a detecção e a ação é fundamental.
 - *Sem Atraso:* Se o *sorter* fosse acionado no exato instante da detecção, as rodas girariam antes que a caixa estivesse sobre elas. A caixa colidiria lateralmente com as rodas já em ângulo, em vez de ser transportada suavemente. Além disso, em fluxo contínuo, o acionamento prematuro poderia desviar incorretamente uma caixa anterior que ainda estivesse na zona de atuação.
 - *Atraso Excessivo:* A caixa passaria do ponto de pivô, resultando em um desvio tardio que a faria colidir com o divisor central. Através de testes iterativos para a

velocidade de 2.0V, determinou-se o atraso ideal de 100 milissegundos via temporizador TON, garantindo a entrada suave da caixa no mecanismo.

- Compensação de Duração (TOF - 500ms): Especificamente para o desvio à Direita (Caixas Médias), foi necessária uma compensação adicional devido à geometria do objeto.
 - *O Problema:* Observou-se que, devido à maior extensão longitudinal da caixa média em comparação à pequena, o tempo de sinal do sensor era insuficiente. O *sorter* tendia a desativar enquanto a parte traseira da caixa ainda estava passando, causando erros de classificação.
 - *A Solução:* Foi implementado um temporizador TOF (*Time Off-Delay*) de 500ms. Este recurso mantém o atuador ligado por meio segundo extra após a perda de sinal do sensor, compensando o comprimento da caixa e garantindo que ela transponha totalmente a zona de desvio antes que o mecanismo retorne à posição original.

4.1.3.2. Gestão de Saída

Para as esteiras de saída (ramais de descarte e alimentação da paletizadora), a lógica buscou eficiência operacional. Em vez de manter os motores ligados permanentemente, utilizou-se a lógica de detecção por demanda.

Os sensores de barreira (Sensor E.M, Sensor E.G) disparam temporizadores TOF (*Time Off-Delay*) configurados para 5 segundos.

Este tempo não é aleatório; foi calculado com base no comprimento da esteira e na velocidade de 2.0V. O teste confirmou que 5 segundos fornecem uma janela de segurança suficiente para que a caixa percorra todo o ramal e chegue ao removedor (*Remover*), mesmo que haja micro-deslizamentos. Após esse tempo, se nenhuma nova caixa for detectada, o motor desliga (MOVE 0 para velocidade), economizando “energia” e simulando uma prática real.

4.2. SUBSISTEMA DE PALETIZAÇÃO

O subsistema de paletização apresentou a maior complexidade em termos de sequenciamento de operações discretas e manipulação de dados. A validação focou na precisão mecânica da formação de cargas e na robustez da rastreabilidade RFID.

4.2.1. Precisão de Posicionamento e Controle do Elevador

A estabilidade da pilha de caixas depende inteiramente do palete estar parado no lugar exato e na altura correta antes de começar o processo. Durante os testes, percebeu-se que confiar apenas na lógica padrão da máquina não era suficiente, exigindo ajustes tanto no alinhamento horizontal quanto no vertical.

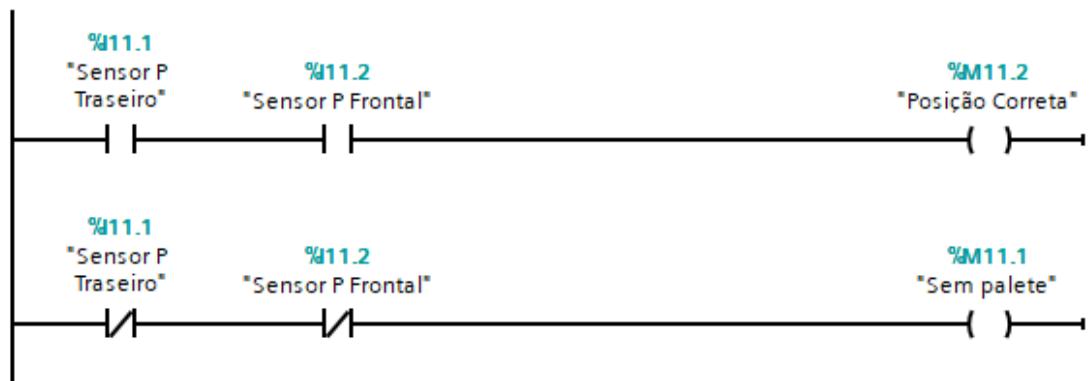
A paletizadora já possui dois sensores ópticos na esteira: um Traseiro (%I11.1) e um Frontal (%I11.2). Testes iniciais tentando usar apenas um deles como referência falharam, pois o palete parava ligeiramente torto ou fora de centro.

A solução foi criar uma condição lógica estrita de "Posição Correta" (%M11.2) que funciona como um gabarito virtual. O sistema só considera que o palete está pronto se ambos os sensores estiverem ativos ao mesmo tempo:

$$\text{Posicao_Correta} = (\text{Sensor Traseiro} == 1) \text{ and } (\text{SensorFrontal} == 1)$$

Se apenas o traseiro está ativo, o palete ainda está entrando. Se apenas o frontal está ativo, ele passou do ponto ou está saindo. Somente a ativação simultânea garante o alinhamento geométrico perfeito para receber a carga. A implementação dessa lógica de intertravamento em Ladder, que exige a presença simultânea dos dois sinais para validar a posição segura, é demonstrada na Figura 30.

Figura 30 – Lógica Ladder para validação de posicionamento do palete



Fonte: O Autor (2025).

4.2.1.1. Controle Vertical (A Adição do Sensor "P Baixo")

Além do alinhamento horizontal, houve uma complicação crítica com a altura do

elevador. O sistema original perdia a referência de quando o elevador estava totalmente abaixado (no nível do chão/esteira), o que causava travamentos ou tentativas de mover o palete com o elevador ainda erguido.

Para corrigir isso, foi necessário adicionar um sensor extra, denominado Sensor P Baixo (%I11.3). Este sensor serve como uma confirmação física absoluta de que o elevador completou sua descida. A lógica de liberação do palete foi alterada para depender obrigatoriamente deste sinal, garantindo que a troca de paletes só ocorra com o elevador na posição zero, eliminando os erros de nível

4.2.2. Gestão de Fluxo e Criação de brechas

Um desafio clássico em logística, a fusão de caixas, foi observado na entrada da paletizadora. Caixas saindo da separação tendiam a se acumular, formando um "trem" contínuo sem espaço entre as unidades. Isso impedia a contagem correta e a leitura individual das etiquetas RFID.

Para resolver isso, implementou-se um diferencial de velocidade entre a esteira de alimentação (Esteira 1) e a esteira de entrada da paletizadora (Esteira 2).

- Esteira 1: Mantida em velocidade base (2.0V).
- Esteira 2: Acelerada via instrução ADD na lógica de controle.

Fisicamente, quando a caixa transita da esteira lenta para a rápida, ela é "puxada" para frente, criando um distanciamento em relação à caixa que vem atrás. Esta brecha é essencial para que o sensor de contagem tenha tempo de voltar a zero (reset) entre uma caixa e outra, garantindo a precisão da contagem unitária.

Além disso, implementou-se uma lógica de redução de velocidade (para 0.5V) quando o empurrador está em operação. Isso atua como um "freio dinâmico", prevenindo que uma nova caixa entre na zona de perigo enquanto o braço mecânico ainda está estendido, evitando que as caixas emperrem o atuador, enquanto acionado.

4.2.3. Rastreabilidade RFID e Integridade de Dados

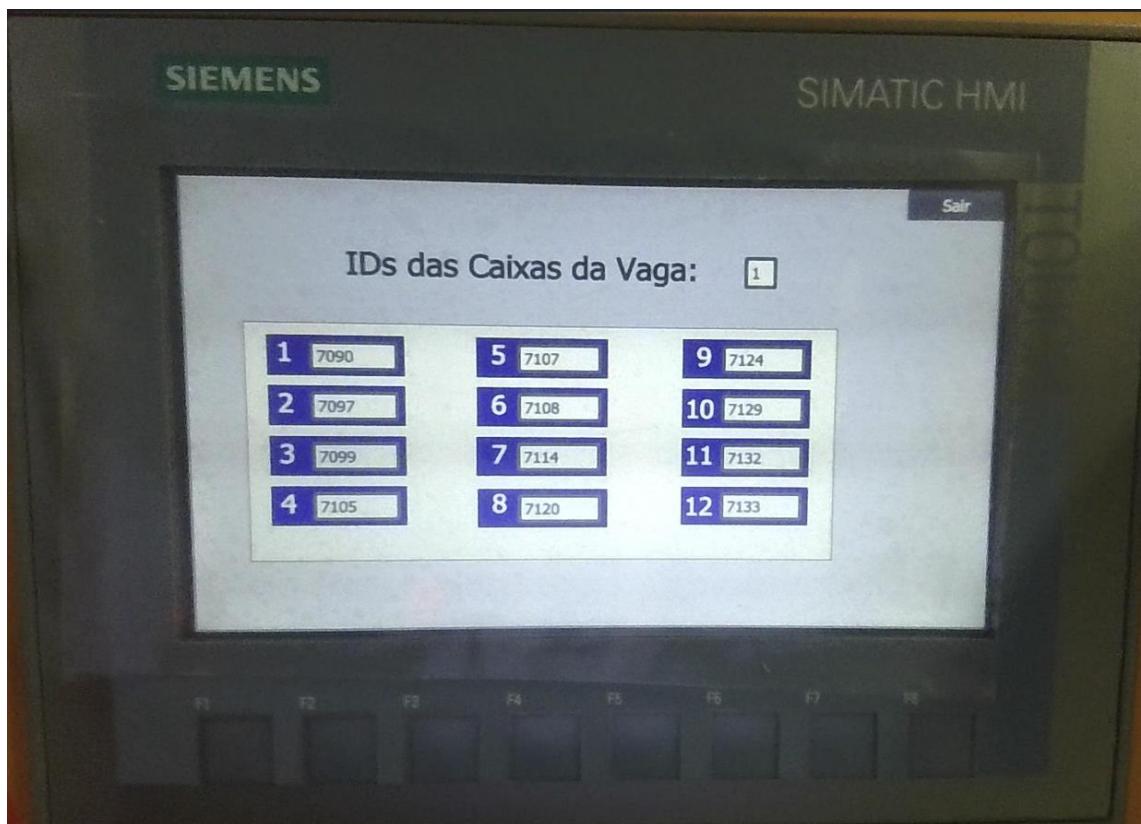
A implementação da tecnologia RFID para rastreabilidade é um diferencial alinhado à Indústria 4.0. No entanto, a integração dessa tecnologia com o ciclo de varredura do CLP exigiu

tratamento de sinal.

Durante os primeiros testes de integração, notou-se um erro crítico na gravação dos dados. O leitor RFID detecta a tag durante todo o tempo em que a caixa passa por ele (aproximadamente 500ms a 1s). Como o CLP possui um ciclo de varredura extremamente rápido (ex: 10ms), a lógica de "Gravar ID na Lista" era executada várias vezes para uma única caixa.

Isso fazia com que o contador de índice do array disparasse, preenchendo todas as 12 posições de memória do palete com o ID das primeiras caixas, corrompendo o banco de dados. A funcionalidade de rastreabilidade total do palete é validada pela interface de consulta que permite a visualização de todos os IDs de caixa contidos na matriz de dados, conforme demonstrado na Figura 31.

Figura 31 - Tela de consulta do inventário



Fonte: O Autor (2025)

4.2.3.1. Solução via P_TRIG e Indexação Indireta

A solução definitiva utilizou a instrução P_TRIG (Detecção de Borda de Subida). O P_TRIG monitora o sinal do sensor de presença (%I11.4). No exato milissegundo em que o sinal transita de 0 para 1 (a frente da caixa toca o sensor), o bloco gera um pulso único de

duração de apenas um ciclo de scan. Este pulso dispara duas ações simultâneas e únicas:

- Incremento do Ponteiro: A instrução ADD 1 atualiza o índice do contador.
- Escrita Indexada: A instrução MOVE transfere o ID lido (%ID46) para o endereço do array apontado pelo contador: *Lista_IDs[Contador]*.

Esta lógica garante uma relação de 1:1 entre o evento físico (passagem da caixa) e o evento lógico (registro de dados), assegurando a integridade absoluta da informação de rastreabilidade. Adicionalmente, implementou-se um timer de sincronização de 500ms para garantir que a leitura do RFID estivesse estabilizada antes da gravação, prevenindo a captura de IDs nulos ou parciais em caso de falha de leitura momentânea.

4.2.4. Estabilidade de Carga e Lógica de Contadores (Cascata)

A definição do layout do palete foi resultado de testes de estabilidade física simulada.

- Testes com 1 Camadas: O palete resultava muito baixo, subutilizando o volume do armazém.
- Testes com >2 Camadas: A altura excessiva tornava a pilha instável. Acelerações laterais do transelevador (AS/RS) faziam as caixas superiores deslizarem e caírem, pois o próprio programa não contempla uma embaladora para contenção.
- Configuração Otimizada (2 Camadas): Mostrou-se o equilíbrio ideal entre volume transportado e estabilidade dinâmica.

A lógica de controle para alcançar essa configuração utiliza contadores (CTU) em cascata:

- Contador de Fileira: Conta 2 caixas -> Aciona o Empurrador (%Q10.4).
- Contador de Camada: Conta 3 atuações do empurrador -> Aciona a abertura da placa (%Q10.5) para depositar a camada.
- Contador de Paleta: Conta 2 aberturas de placa -> Finaliza o paleta (%M11.3) e libera para o AS/RS.

4.2.4.1. Ajuste do Ciclo do Empurrador (Timer de 150ms)

Um ajuste fino foi necessário na lógica de acionamento do braço empurrador para evitar um "falso positivo" na leitura dos sensores. O sistema utiliza um sensor de limite para confirmar

que o braço está recolhido (posição de repouso) e encerrar o ciclo.

- O Problema (Reset Instantâneo): Como o empurrador inicia o movimento parado exatamente em cima desse sensor de limite, ocorria um erro lógico: ao receber o comando de avançar, o CLP lia o sensor ativo e interpretava que o braço já tinha completado a volta. O resultado era que o sistema resetava (desligava) o empurrador instantaneamente, fazendo com que ele mal saísse do lugar.
- A Solução (Janela de Tempo): Foi adicionado um temporizador (TON) de 150 milissegundos no início do comando de empurrar.
- Resultado: Esse tempo de 150ms funciona como uma janela de escape. Ele permite que o motor fique ligado tempo suficiente para o braço andar um pouco e desativar o sensor de limite (sair da frente dele). Somente após esse tempo a lógica de parada é habilitada novamente. Assim, o sistema ignora o sensor na partida e só volta a lê-lo quando o braço realmente retorna da viagem, garantindo o ciclo completo de ir e voltar.

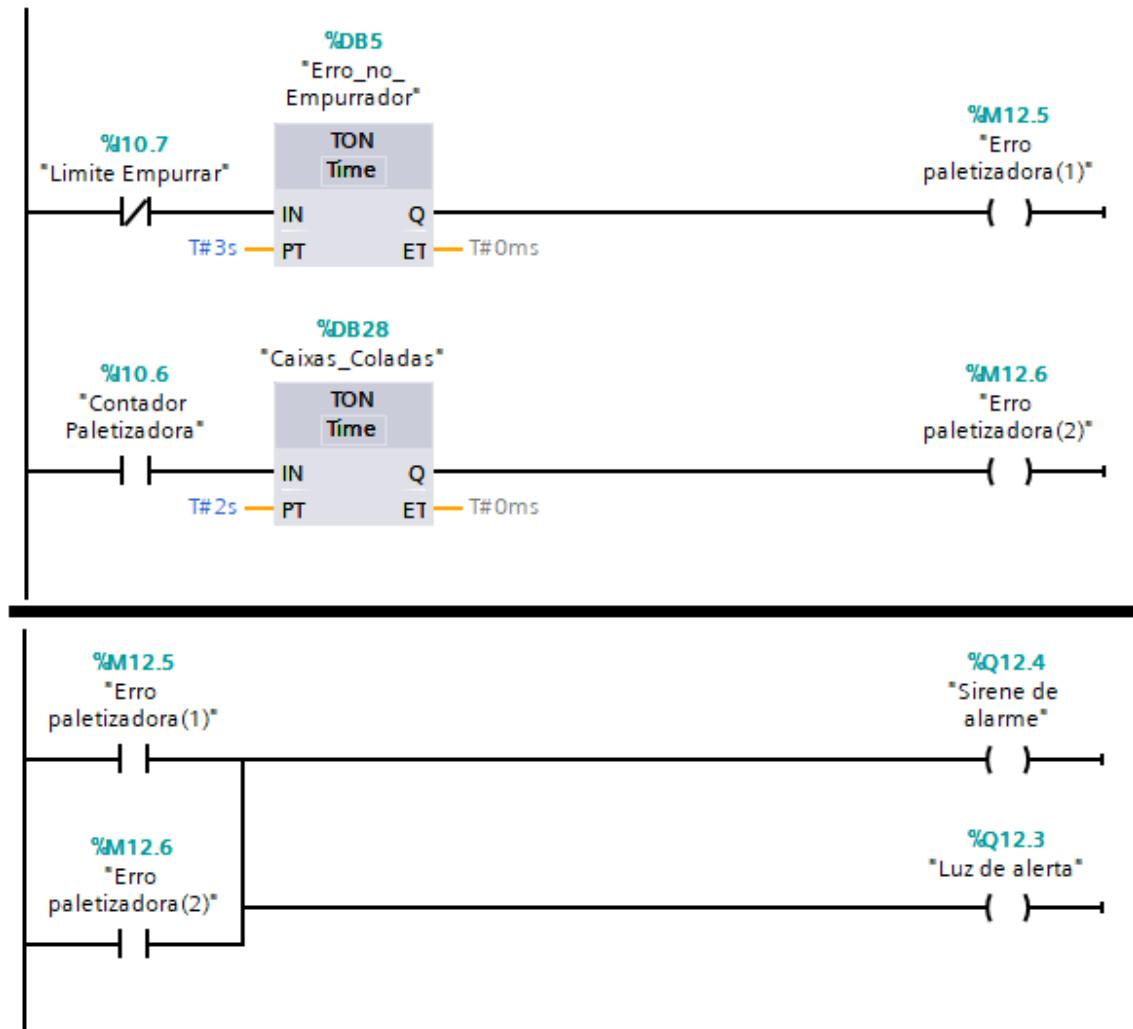
4.2.5. Sistemas de Segurança e Alarmes

Para simular a proteção dos equipamentos como em um caso físico, foram implementadas lógicas de monitoramento de tempo de execução, sendo eles:

- Alarme de Empurrador (%M12.5): Se o atuador de empurrar for acionado e não retornar à posição de repouso em 3 segundos (o ciclo normal é <1s), o sistema infere um travamento mecânico.
- Alarme de Contagem (%M12.6): Se o sensor de entrada permanecer acionado por mais de 2 segundos, infere-se um congestionamento na esteira.

Ambos os eventos disparam a sirene de alarme e a luz de alerta, interrompem a memória geral de operação e congelam o processo, exigindo intervenção (reset manual). Esta lógica valida o sistema contra falhas operacionais imprevistas. A implementação dessa lógica de monitoramento temporal (*watchdog*), que dispara alarmes caso os atuadores não completem o ciclo no tempo previsto, é apresentada na Figura 32.

Figura 32 – Lógica Ladder de segurança: Monitoramento e disparo de alarmes



Fonte: O Autor (2025).

4.3. SUBSISTEMA AS/RS

O controle do Transelevador (*Stacker Crane*) representou o ápice da complexidade lógica do projeto. A natureza do equipamento de movimento simultâneo em dois eixos (X, Y) e manipulação de garfos, exigiu uma abordagem de programação estruturada via Máquina de Estados Finitos (FSM).

4.3.1. Comparativo: Lógica Sequencial vs. Máquina de Estados (FSM)

Durante as fases preliminares, tentou-se controlar o AS/RS utilizando lógica Ladder sequencial simples (bobinas Set/Reset). O resultado foi um código instável e de difícil manutenção, onde condições de corrida frequentemente causavam movimentos erráticos, como tentar estender o garfo enquanto o transelevador ainda estava se movendo horizontalmente. A transição para a FSM (implementada nos FBs Armazenar e Retirar) trouxe resultados imediatos em termos de determinismo e segurança.

- Exclusividade de Estados: A FSM garante que o sistema esteja em apenas um estado por vez (ex: Estado 10 = Movendo, Estado 40 = Depositando). É logicamente impossível que o comando de "Estender Garfo" seja enviado enquanto o estado for "Movendo", eliminando 100% das colisões operacionais observadas anteriormente.
- Diagnóstico de Falhas: Durante os testes, a variável Estado_Atual serviu como uma ferramenta de diagnóstico poderosa. Se o sistema parava, bastava observar o número do estado (ex: travado no Estado 20) para saber exatamente qual condição de transição (ex: sensor de posição) falhou.

4.3.2. Controle de Posição Analógico e a "Faixa Morta" de 0.001V

O posicionamento do AS/RS é baseado em setpoints de tensão (0-10V) armazenados no banco de dados (DB Mapa). O desafio encontrado foi a precisão da realimentação (*feedback*) em malha fechada.

Como o sistema é digital e discreto, a comparação direta de igualdade (Posição_Atual == Setpoint) falhava sistematicamente. Devido à resolução do sinal analógico simulado e micro-ruídos, o transelevador nunca atingia o valor matemático exato (ex: 5.453V). Isso causava problemas da confirmação da posição para prosseguir para a próxima etapa.

A solução validada foi a implementação de uma lógica de comparação por faixa de tolerância, utilizando as instruções SUB (Subtração) e ABS (Valor Absoluto):

$$\text{Erro} = \text{Setpoint} - \text{Posicao_Atual}/$$

$$\text{Em_Posicao} = (\text{Erro} < 0.001)$$

A tolerância de 0.001V foi determinada experimentalmente.

- Testes com 0.01V: A tolerância era muito larga, fazendo com que o garfo parasse visivelmente desalinhado com a estante, causando atrito na entrada do palete.
- Testes com 0.0001V: O sistema voltava a oscilar.
- 0.001V: Provou ser o ponto de equilíbrio ótimo, garantindo alinhamento visual perfeito e estabilidade de controle imediata ao chegar na posição.

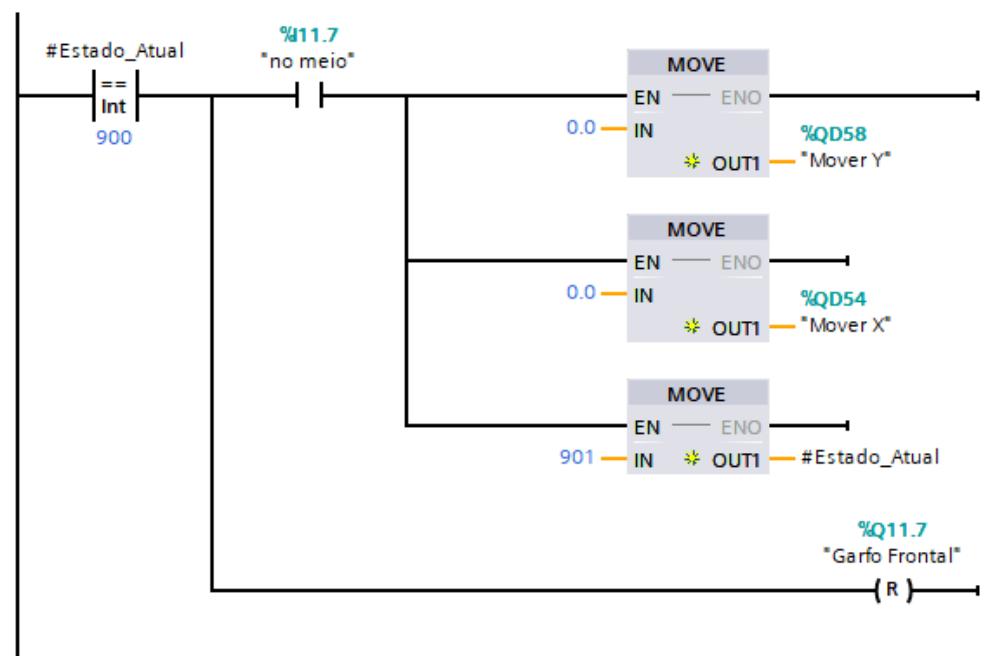
4.3.3. Algoritmo de Busca de Vagas (WMS)

A inteligência do sistema reside no algoritmo de busca de vagas livres (para armazenamento) ou ocupadas (para retirada) no DB Mapa. A lógica de iteração utiliza dois ponteiros: Linha_Atual e Coluna_Atual.

4.3.4. Recuperação de Falhas e Rotina de Reset

A segurança operacional exige que o sistema seja capaz de se recuperar de paradas abruptas. A validação da rotina de Reset (Estados 900-902) mostrou-se eficaz. Quando acionada, a rotina força o recolhimento dos garfos, em seguida, move o transelevador lentamente para a posição (0,0). Simultaneamente, zera os parâmetros de busca (Linha=1, Coluna=1). Isso garante que, independentemente de onde ou como o sistema parou, a reinicialização ocorra a partir de um estado conhecido e seguro, prevenindo movimentos imprevisíveis na retomada. A Figura 33 detalha a implementação da rotina de Reset no TIA Portal, onde o sistema força as saídas analógicas para zero e recolhe os atuadores para garantir um estado seguro de reinício.

Figura 33 – Rotina de Reset e recuperação de falhas na Máquina de Estados



Fonte: O autor (2025).

4.4. INTEGRAÇÃO COM INTERFACE HOMEM-MÁQUINA (IHM)

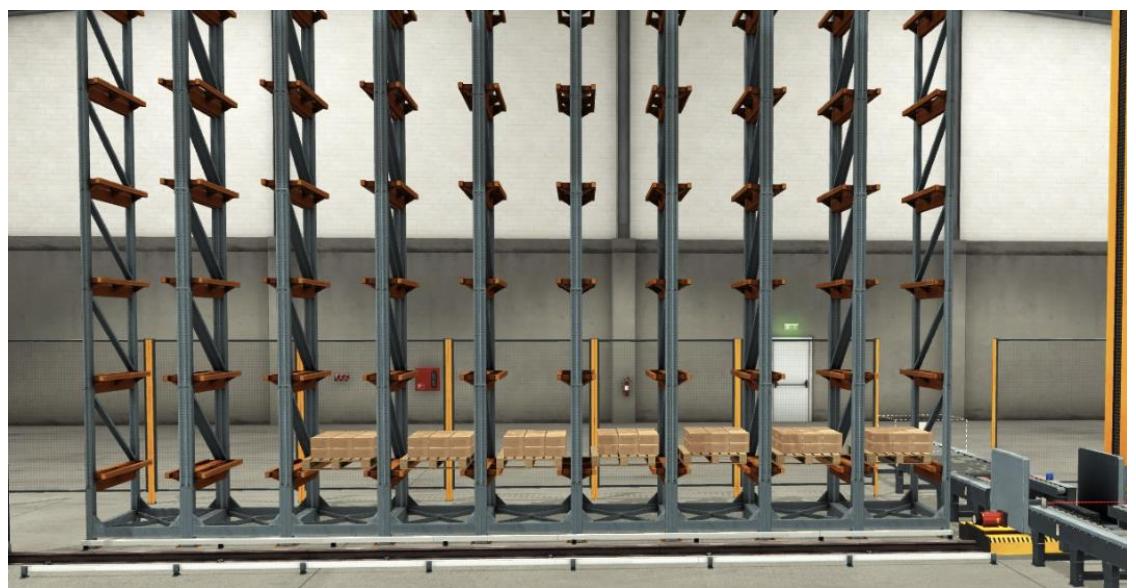
A IHM (Modelo KTP700 Basic) foi validada não apenas como painel de comando, mas como ferramenta de supervisão em tempo real. A comunicação via DB Interface mostrou-se robusta. O mapa visual do armazém, composto por 54 indicadores gráficos vinculados ao array de dados do WMS, atualizava o status das vagas (Verde/Vermelho) instantaneamente após a conclusão do ciclo de armazenamento como é demonstrada visualmente nas Figuras 34 e 35, que mostram a interface do mapa e a estrutura física do armazém rastreado, respectivamente.

Figura 34 - Interface HMI exibindo status em tempo real



Fonte: O Autor (2025)

Figura 35 - Estrutura física das 54 vagas



Fonte: O Autor (2025)

A funcionalidade de "Retirada Específica", onde o operador clica em uma posição no mapa para solicitar aquele paleta específico, validou a capacidade do sistema de interpretar comandos externos e traduzi-los em coordenadas de destino para a FSM, fechando o ciclo de automação ciber-física proposto.

4.5. ANALISE DOS RESULTADOS

A validação funcional da célula logística automatizada, executada por meio da metodologia de Comissionamento Virtual, permitiu a avaliação rigorosa do desempenho dinâmico e lógico do sistema. Os testes realizados com a arquitetura *Hardware-in-the-Loop* (HiL) confirmaram o atendimento pleno tanto do objetivo geral quanto dos objetivos específicos previamente estabelecidos no projeto.

4.5.1. Objetivos Gerais

O objetivo geral do trabalho foi desenvolver e validar a automação de uma célula logística (separação, paletização e armazenamento) com CLP Siemens S7-1200 e Comissionamento Virtual via Gêmeo Digital (Factory I/O).

- Validação: O objetivo foi integralmente alcançado. O sistema desenvolvido conecta o CLP Siemens S7-1200 real ao Gêmeo Digital (Factory I/O). A lógica de controle executada no hardware físico gerenciou com sucesso a sequência completa da célula logística: (1) classificação analógica de caixas, (2) sequenciamento do paletizador e rastreabilidade RFID, e (3) o controle do sistema AS/RS por Máquina de Estados Finitos.
- Comprovação Metodológica: Os resultados comprovaram a eficácia da metodologia de Comissionamento Virtual, que permitiu otimizar parâmetros dinâmicos (como a fixação da velocidade da esteira em 2.0V para garantir a estabilidade das caixas) e corrigir erros de lógica antes da implementação física, mitigando riscos e custos.

4.5.2. Objetivos Específicos

O cumprimento de cada objetivo específico foi comprovado com base nos resultados práticos obtidos na simulação, conforme detalhado a seguir:

Tabela 9 - Resultados da Validação do Projeto em Relação aos Objetivos Propostos

Objetivo Específico	Comprovação Técnica (Seção 4)
1. Desenvolver a lógica de controle para a separação automática de caixas por tamanho e para o processo sequencial de paletização.	Comprovado. A separação utilizou o Controle Analógico com a instrução IN_RANGE e uma lógica de "faixa morta" para definir limiares precisos de classificação (ex.: 0.1V a 1.4V para caixas pequenas). A paletização foi controlada por uma lógica de Contadores Crescentes (CTU) em cascata, garantindo a formação rigorosa de camadas (Contador de Camadas PV=2).
2. Desenvolver a lógica de controle para o posicionamento preciso do transelevador do sistema AS/RS, utilizando uma estrutura de Máquina de Estados.	Comprovado. O controle do AS/RS foi estruturado em uma Máquina de Estados Finitos (FSM), implementada nos Blocos de Função (FBs) Armazenar e Recuperação. A precisão de posicionamento (eixos X e Y) foi validada por uma rotina de software que calcula o erro absoluto (SUB e ABS) e exige que este seja inferior a 0.001V para a transição de estado.
3. Implementar um sistema de gestão de inventário em tempo real, utilizando tecnologia RFID para a rastreabilidade das caixas desde a paletização até o armazenamento.	Comprovado. O sistema de gerenciamento de armazém (WMS) foi implementado no Bloco de Dados Global DB "Mapa". A rastreabilidade foi garantida por meio da sincronização do leitor RFID com o detector de borda de subida (P_TRIG), que assegurou que cada ID de caixa fosse registrado no Array de dados (endereçamento indireto) uma única vez.
4. Projetar e integrar uma Interface Homem-Máquina (IHM) para a supervisão e o monitoramento visual do sistema.	Comprovado. A IHM SIMATIC KTP700 Basic foi configurada e validada com o mapa visual de ocupação do armazém (54 vagas). A cor de cada vaga é vinculada em tempo real à variável booleana Ocupado no DB "Mapa", fornecendo supervisão imediata. A funcionalidade de Recuperação Específica confirmou a capacidade de interação do operador com a FSM do CLP.
5. Integrar um painel de controle físico ao ambiente virtual para permitir a operação e o teste de funcionalidades.	Comprovado. A integração HiL foi estabelecida via rede PROFINET, utilizando o switch SCALANCE XB005. A lógica de controle e a interface (IHM) operaram sobre o Gêmeo Digital, com o software Factory I/O lendo as saídas físicas do CLP e escrevendo os estados dos sensores virtuais nas entradas do CLP.

Objetivo Específico	Comprovação Técnica (Seção 4)
6. Validar a funcionalidade integrada de todo o sistema no ambiente de simulação.	Comprovado. A validação demonstrou que todas as etapas funcionam de forma coesa e segura. A robustez lógica foi confirmada por rotinas de segurança (como o Alarme de Empurrador de 3 segundos) e a rotina de Reset da FSM (Estados 900-902), que força o retorno do Transelevador a uma posição segura (0.0V), prevenindo movimentos imprevisíveis durante a retomada.

Fonte: O Autor (2025)

5. CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Este trabalho comprovou que o uso de Comissionamento Virtual e Gêmeos Digitais é eficiente para desenvolver sistemas de automação complexos, seguindo os princípios da Indústria 4.0. Os objetivos determinados foram alcançados: desenvolver e validar uma célula logística que integra classificação, paletização e um sistema AS/RS, conectando um CLP físico S7-1200 ao simulador Factory I/O. O uso do software TIA Portal foi essencial para criar uma lógica de controle organizada. A mudança da programação Ladder comum para a Máquina de Estados Finitos (FSM) foi o ponto chave, principalmente para controlar o Transelevador. Isso garantiu que o sistema funcionasse de forma previsível, sem conflitos lógicos e com mais segurança, confirmado que métodos estruturados são necessários para máquinas com muitos movimentos e sequências importantes. Os testes feitos no simulador mostraram que o Comissionamento Virtual consegue prever erros que causariam prejuízo no mundo real. Um exemplo prático foi descobrir que as caixas ficavam instáveis se a tensão de controle da esteira passasse de 3.0V, pois quanto mais tensão na esteira maior seria a velocidade. Corrigir isso no software antes de montar a máquina prova que a simulação ajuda a evitar riscos. Além disso, problemas técnicos da simulação, como a imprecisão dos sensores analógicos e a leitura do RFID, foram resolvidos com ajustes no código. O uso de "faixa morta" para posicionar os eixos e detectores de borda (P-TRIG) para gravar os dados mostrou que o sucesso depende tanto do equipamento quanto de um bom algoritmo. Conclui-se que o sistema criado funciona bem, oferece rastreabilidade e serve como um modelo técnico para engenharia, estando pronto para ser montado fisicamente com muito menos chance de erros na partida.

Para a continuidade deste estudo, sugere-se, primeiramente, a transposição da lógica validada virtualmente para um protótipo físico em escala reduzida, o que permitiria comparar o desempenho e finalizar a validação do modelo em um ambiente real. Um desenvolvimento de alto valor industrial seria a modificação da lógica do paletizadora para permitir a alteração do layout de empilhamento via IHM. Isso envolveria a criação de campos na interface para o operador definir o número de caixas por fileira e o total de camadas, adaptando o ciclo do paletizadora em tempo real para formar diferentes padrões de palete. Outra melhoria pertinente seria a implementação de rotinas de Diagnóstico de Falhas no AS/RS. Isso exigiria criar lógica para identificar automaticamente a causa raiz de uma parada (ex: "tempo limite de garfo excedido") e exibir uma mensagem de solução clara na IHM.

REFERÊNCIAS

ALTUS. Conheça os nove pilares da Indústria 4.0 e sua relevância para a atividade industrial. [S.l.: s.n.], [s.d.]. Disponível em: <<https://www.altus.com.br/post/212/conheca-os-nove-pilares-da-industria-4-0-e-sua-relevancia-para-a-atividade-industrial>>. Acesso em: 22 nov. 2025.

ATM ENGENHARIA EM AUTOMAÇÃO. Qual linguagem é melhor para programação de PLC? Blog ATM Solutions. [S.d.]. Disponível em: <<https://blog.atmsolutions.com.br/qual-linguagem-e-melhor-para-programacao-de-plc>>. Acesso em: 22 nov. 2025.

AUTOMATION READY PANELS. IEC 61131 standard for industrial automation programming. [S.l.: s.n.], [s.d.]. Disponível em: <<https://www.automationreadypanels.com/plc-systems/iec-61131-standard-for-industrial-automation-programming>>. Acesso em: 13 nov. 2025.

CNI. Indústria 4.0: novo desafio para a indústria brasileira. Brasília: Confederação Nacional da Indústria, 2016. Disponível em: <https://static.portaldaindustria.com.br/media/filer_public/e0/aa/e0abd52-53ee-4fd8-82ba-9a0ffd192db8/sondespecial_industria40_abril2016.pdf>. Acesso em: 18 nov. 2025.

ENGELOGIC. Estrutura de um CLP. [S.l.: s.n.], [s.d.]. Disponível em: <https://www.engelogic.com.br/download/estrutura_clp.pdf>. Acesso em: 22 nov. 2025.

FESTO. Interface homem máquina: vantagens para a automação industrial. [S.l.]: Festo Brasil, 2024. Blog In Practice. Disponível em: <https://www.festo.com.br/pt/e/sobre-a-festo/blog/in-practice/interface-homem-mquina-vantagens-para-a-automacao-industrial-id_2107971/>. Acesso em: 20 nov. 2025.

GEORGINI, João Marcelo. Automação aplicada. 9. ed. São Paulo: Érica, 2009.

GRIEVES, M. Digital Twin: Manufacturing Excellence through Virtual Factory Replication. White Paper. Melbourne, FL: Florida Institute of Technology, 2014. Disponível em: <<https://www.researchgate.net/publication/275211047>>. Acesso em: 16 nov. 2025.

INETEC. A evolução dos sistemas de controle: Relés vs PLCs. [S.d.]. Disponível em: <<https://inetec.com.br/a-evolucao-dos-sistemas-de-controle-reles-vs-plcs>>. Acesso em: 22 nov. 2025.

KAGERMANN, H.; WAHLSTER, W.; HELBIG, J. Recommendations for implementing the strategic initiative INDUSTRIE 4.0: Final report of the Industrie 4.0 Working Group. Frankfurt: ACATECH, 2013. Disponível em: <<https://www.acatech.de/publikation/industrie-4-0-im-globalen-kontext-strategien-der-zusammenarbeit-mit-internationalen-partnern>>. Acesso em: 15 nov. 2025.

MACHADO, Pedro Henrique Ferreira; FREITAS, Elias José de Rezende; JUNIOR, Carlos Dias da Silva. Operador e programador de sistemas automatizados: práticas iniciais com CLP. 1. ed. Belo Horizonte: Instituto Federal de Minas Gerais, 2021.

SCHWAB, K. **A Quarta Revolução Industrial**. São Paulo: Edipro, 2016.

SIEMBRA. IHM Interface Homem Máquina. [S.d.]. Disponível em:
<<https://www.siembra.com.br/pecas-de-reposicao/ihm-interface-homem-maquina>>. Acesso em: 22 nov. 2025.

SIEMENS. 6ES7214-1AG40-0XB0: Technical data. Disponível em:
<<https://support.industry.siemens.com/cs/pd/255094?pti=td&lc=en-WW>>. Acesso em: 15 nov. 2025.

SIEMENS. 6GK5005-0BA00-1AB2. Siemens Industry Mall. Disponível em:
<<https://mall.industry.siemens.com/mall/en/ww/Catalog/Product/6GK5005-0BA001AB2>>. Acesso em: 22 nov. 2025.

SIEMENS. Gêmeo Digital: A Digitalização na Indústria. [S.l.]: Siemens Brasil, [s.d.]. Disponível em: <<https://www.siemens.com.br/pt/empresa/stories/industria/gemeos-digitais-negocios.html>>. Acesso em: 18 nov. 2025.