

Linha de fabricação de carrinho seguidor de linha

Dayane Pereira de Souza, Miguel Fernandes Viana dos Santos Oliveira

Resumo – Este projeto tem como objetivo o desenvolvimento e análise de um sistema de montagem de carrinhos seguidores de linha utilizando a abordagem de Sistemas a Eventos Discretos (SED). O sistema aborda três etapas principais: uma bancada de corte, onde um robô realiza o corte das peças; uma bancada de montagem, onde o operador adiciona a bateria, o Arduino e dois sensores de linha ao carrinho; e uma bancada de adição de código, na qual outro operador programa o microcontrolador. O principal problema de automação e modelagem é garantir a sincronização das etapas, eliminando bloqueios no fluxo produtivo e assegurando a eficiência operacional. O sistema apresenta como entradas o comando do robô na bancada de corte, a entrada manual de componentes na bancada de montagem e o código inserido pelo operador na bancada final. As saídas incluem os estados dos atuadores responsáveis pelo corte, montagem e finalização do carrinho. Para modelar o processo de montagem, utilizaram-se Redes de Petri, que permitem analisar o comportamento dinâmico do sistema, e Programação Funcional Sequencial (PFS), para estruturar a lógica de controle e garantir a sincronização das operações. A análise do modelo do processo de montagem possibilita um olhar de estudo para as etapas do processo que atuam, podendo estudar como é feita a modelagem do processo de montagem. Os resultados obtidos demonstram que a aplicação de metodologias SED viabilizou uma análise detalhada do processo de montagem e permitiu a implementação de um sistema confiável e eficiente. O impacto do projeto inclui a otimização do processo produtivo, a redução de tempos improdutivos e o aumento da qualidade final do produto. O desenvolvimento reforça o uso de SED em aplicações industriais, contribuindo para avanços em automação e controle.

Palavras-chave – Sistemas a Eventos Discretos, Redes de Petri, Automação, Process Flow Schema.

1 Introdução

O presente projeto aborda o desenvolvimento e análise de um sistema automatizado para montagem de carrinhos, utilizando a abordagem de Sistemas a Eventos Discretos (SED) como base para a modelagem e análise do processo. O objetivo é otimizar a eficiência, segurança e integração dos sistemas produtivos, especialmente em indústrias com processos complexos e multifásicos.

O sistema é composto por três etapas principais: na bancada de corte, um robô automatizado realiza o corte das peças necessárias; na bancada de montagem, o operador adiciona componentes como bateria, Arduino e sensores; e, na bancada de adição de código, outro operador programa o microcontrolador para a funcionalidade do carrinho. Cada etapa apresenta desafios específicos de sincronização, automação e fluxo, exigindo uma modelagem e um controle lógico bem definidos para evitar atrasos, falhas e bloqueios no processo produtivo.

Para enfrentar esses desafios, foram aplicadas Redes de Petri, uma ferramenta eficaz para representar e analisar o comportamento dinâmico de sistemas discretos, e a Programação Funcional Sequencial (PFS), que permite estruturar a lógica de controle em detalhes. A validação do modelo foi conduzida no software PIPE 4.3.0, permitindo verificar o desempenho e identificar melhorias potenciais no fluxo produtivo.

Este estudo não apenas propõe uma solução otimizada para o processo de montagem de carrinhos, mas também destaca a aplicabilidade de SED em sistemas produtivos mais amplos. Além

de alcançar maior eficiência, redução de tempos improdutivo e aumento da qualidade do produto final, o projeto reforça a importância da modelagem como ferramenta essencial para compreender e melhorar cada etapa do processo, demonstrando o impacto estratégico de abordagens sistemáticas na automação industrial.

2 Materiais e Métodos

De acordo com Miyagi (2020), a simulação de sistemas a eventos discretos é uma ferramenta poderosa para entender e prever o comportamento de sistemas complexos. O modelo SED se baseia em métodos numéricos para reproduzir uma “história artificial” do sistema, permitindo observar seu desempenho sob diferentes condições ou configurações. Isso é especialmente útil para identificar gargalos, testar novos procedimentos e otimizar processos sem interferir na operação real.

2.1 Conceito de Sistemas a Eventos Discretos (SED)

O conceito de um Sistema a Eventos Discretos (SED) refere-se a um sistema onde o estado das variáveis só muda quando ocorrem eventos. Esses estados são discretos, e as transições entre eles acontecem de forma instantânea. Assim, a descrição das ações do sistema, que provocam mudanças nos estados, não depende mais do tempo, mas sim da ocorrência de eventos. Quando o sistema detecta um evento, seja ele interno ou externo, reage imediatamente e, em seguida, estabiliza-se instantaneamente até que ocorra o próximo evento.

Os SEDs são fundamentais para a análise metódica e analítica de sistemas complexos. Através de um modelo que representa fielmente o sistema, é possível simular e verificar a eficiência operacional, identificar gargalos e falhas, e testar o desempenho de forma prática. Dessa forma, a modelagem de SEDs é amplamente utilizada atualmente. Exemplos de aplicações incluem o controle de sistemas de armazenagem, filas e trânsito, como abordado neste trabalho.

2.2 Modelo estrutural adotado

A utilização das Redes de Petri neste trabalho justifica-se por seu caráter versátil e robusto. Em um contexto de interação simultânea de diferentes entidades, com ocorrência de paralelismos e conflitos, a Rede de Petri se apresenta como uma solução eficaz para a simulação do comportamento do sistema sob diferentes condições iniciais. Além da simulação, a modelagem por Redes de Petri possibilita a análise formal de propriedades do sistema, como travamentos, segurança e alcance.

A Modelo comportamental utilizando técnicas de modelagem (Redes de Petri, metodologia PFS/RdP)

Além disso, para alcançar um modelo preciso do sistema em Redes de Petri (RdP), existem ferramentas como a metodologia Process Flow Schema (PFS). O PFS é utilizado para modelar sistemas complexos com uma abordagem orientada a atividades. Esta rede de alto nível descreve de forma conceitual e gráfica os processos associados a etapas sequenciais. Embora represente a relação de causalidade entre os eventos e a lógica dos sistemas modelados por Redes de Petri, o PFS não consegue capturar o comportamento dinâmico do sistema, tarefa que cabe às Redes de Petri.

2.3 A norma IEC 61131-3 para programação

O padrão internacional IEC 61131-3 estabelece linguagens de programação para Controladores Lógicos Programáveis (CLPs). Este padrão abrange cinco linguagens de programação: Diagrama de Ladder (LD), Diagrama de Bloco de Funções (FBD), Texto Estruturado (ST), Lista de Instruções (IL) e Diagrama de Funções Sequenciais (SFC). A IEC 61131-3 visa padronizar a programação de CLPs, promovendo a interoperabilidade entre diferentes fabricantes e

sistemas. Adicionalmente, a norma define orientações para a organização de programas, simplificando a manutenção e a expansão dos sistemas de automação industrial.

A Transcrição RdP para linguagem de programação

Ao converter Redes de Petri (RdP) para uma linguagem de programação, os componentes da rede são transformados em código executável por um controlador lógico programável (CLP). A RdP mapeia elementos como lugares, transições e arcos para estruturas de dados e instruções de controle na linguagem de programação selecionada. Este processo permite que a lógica de controle modelada pela RdP seja implementada diretamente no hardware de controle, garantindo que o comportamento dinâmico do sistema seja fielmente reproduzido. A transcrição facilita a verificação e validação do sistema, pois a lógica de controle pode ser testada em um ambiente simulado antes da implementação final. Além disso, a utilização de linguagens padronizadas, como as definidas pela norma IEC 61131-3, assegura a interoperabilidade e a manutenção do sistema ao longo do tempo.

2.4 Metodologia de Projeto

Por fim, aplicando as metodologias descritas anteriormente, a abordagem escolhida foi simplificar o problema e gradualmente solucioná-lo com mais elementos. Dessa forma, foram modeladas três etapas distintas para a linha de montagem do carrinho seguidor de linha, cada uma com uma aplicação prática. Inicialmente, a bancada de corte foi modelada para realizar o corte das peças necessárias. Em seguida, a bancada de montagem foi introduzida, onde o operador adiciona componentes como bateria, Arduino e sensores de linha ao carrinho. Por último, a bancada de adição de código foi incorporada, permitindo que outro operador programe o microcontrolador. Essa abordagem incremental permitiu a análise detalhada e a otimização de cada etapa do processo, assegurando a eficiência e a sincronização do sistema de montagem.

3 Aplicação de Automação Mecatrônica

3.1 Descrição do sistema que será automatizado (modelo esquemático)

O sistema a ser automatizado é uma linha de montagem de carrinhos seguidores de linha, composta por três etapas principais: bancada de corte, bancada de montagem e bancada de adição de código. Cada uma dessas etapas desempenha um papel crucial na construção e programação dos carrinhos, garantindo a eficiência e a precisão do processo de montagem.

A Bancada de Corte

Nesta etapa inicial, um robô realiza o corte preciso das peças necessárias para a montagem dos carrinhos. O robô é programado para cortar materiais específicos de acordo com as dimensões e formas requeridas, assegurando a uniformidade e a qualidade das peças produzidas.

B Bancada de montagem

Após o corte das peças, o carrinho é montado nesta bancada. Um operador adiciona componentes essenciais, como a bateria, o Arduino e os sensores de linha. Esta fase é fundamental para a integração dos componentes eletrônicos e mecânicos, formando a base funcional do carrinho seguidor de linha.

C Bancada de adição de código

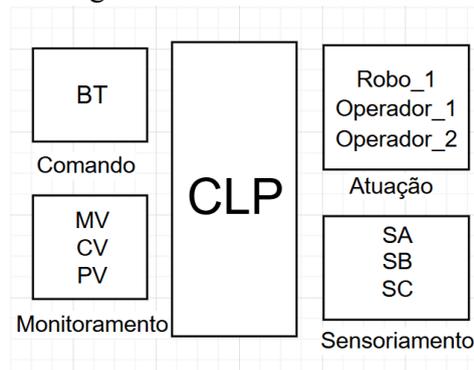
Na etapa final, outro operador programa o microcontrolador do carrinho. O código inserido permite que o carrinho siga a linha de forma autônoma, respondendo aos sinais dos sensores de linha. Esta fase garante que o carrinho esteja pronto para operar conforme o esperado.

3.2 Modelo estrutural do sistema

O sistema é estruturado em processos sequenciais e paralelos, representados pelas Redes de Petri e implementados via Ladder. O fluxo geral é dividido em: Comandos, Monitoramento, Sensores e Atuadores.

- Comandos:
 - BT: Botão de partida do sistema.
- Monitoramento:
 - SA: Sensor que indica a chegada da chapa na esteira.
 - SB: Sensor que indica a chegada do chassi para o operador 1.
 - SC: Sensor que indica a chegada do chassi para o operador 2.
- Entradas (Sensores):
 - MV: Sensor que valida a montagem.
 - CV: Sensor que valida o corte.
 - PV: Sensor que valida o upload.
- Saídas (Atuadores):
 - Robô 1: Robô que realiza o corte da chapa de MDF.
 - Operador 1: Operador responsável pela montagem do circuito eletroeletrônico no chassi.
 - Operador 2: Operador responsável por fazer o upload do programa de controle no Arduino do carrinho.

Figura 1 - Modelo Estrutural

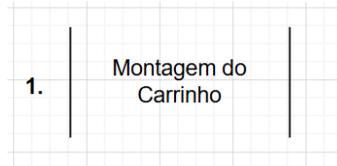


Fonte: Autoria própria

3.3 Aplicação da metodologia de modelagem

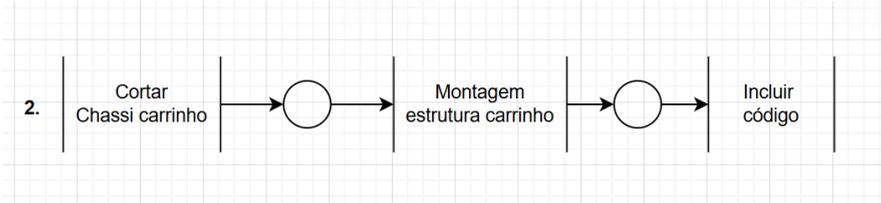
Seguindo a metodologia PFS/RdP para a linha de montagem do carrinho seguidor de linha, cada etapa do processo foi representada como um processo distinto, dado que cada uma delas envolve diferentes objetos de controle. A bancada de corte, a bancada de montagem e a bancada de adição de código foram modeladas separadamente, permitindo uma análise detalhada e específica de cada fase. Essa abordagem facilita a identificação de possíveis gargalos e a otimização do fluxo de trabalho, assegurando a eficiência e a sincronização do sistema de montagem como um todo.

Figura 2 - Representação da linha de fabricação em PFS.



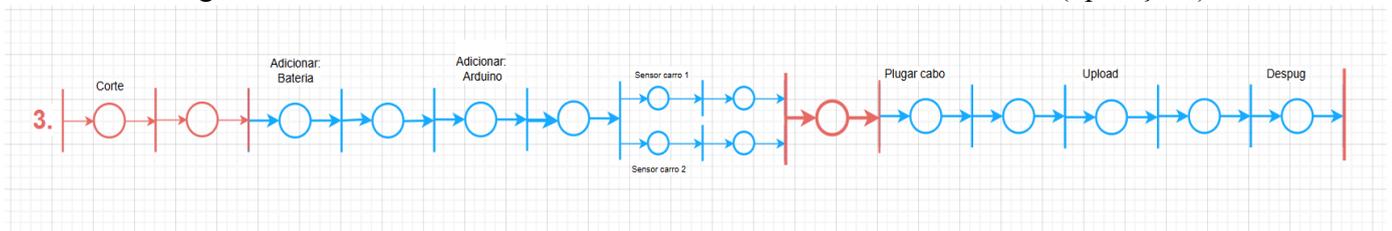
Fonte: Autoria própria

Figura 3 - Detalhamento dos processos em atividades (funções) do PFS.



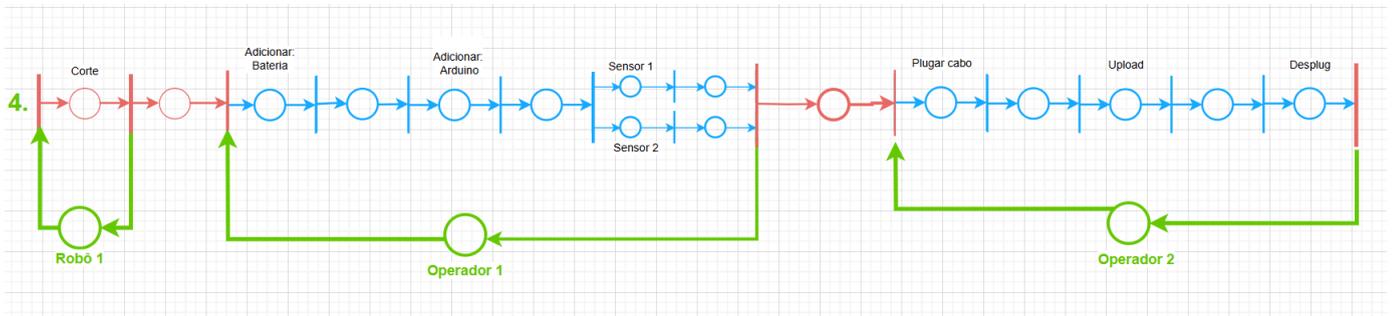
Fonte: Autoria própria

Figura 4 - Detalhamento das atividades introduzindo elementos RdP (operações).



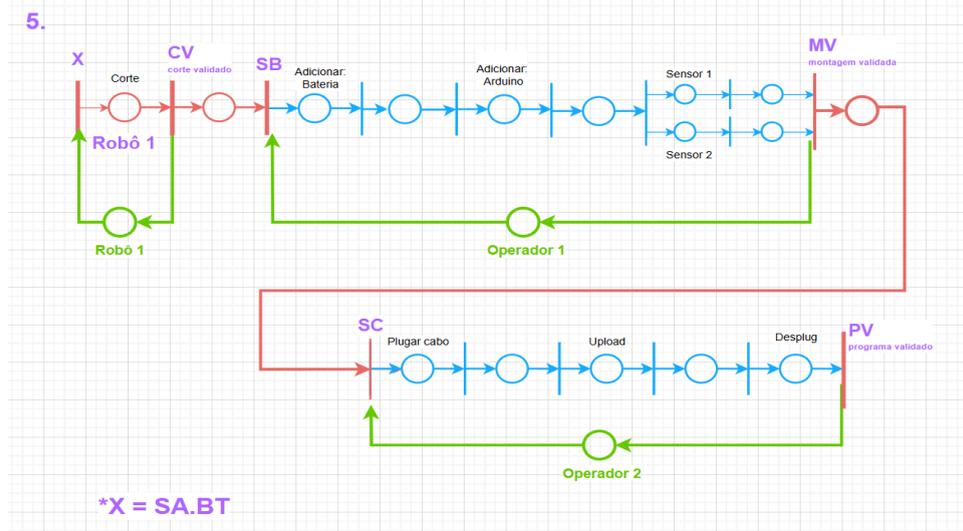
Fonte: Autoria própria

Figura 5 - Introdução dos elementos de controle de recursos para compartilhamento.



Fonte: Autoria própria

Figura 6 - Representação do fluxo de dados com o ambiente externo.



Fonte: Autoria própria

A Intertravamento de partida

O intertravamento de partida garante que o sistema só inicia o processo de montagem se todas as condições iniciais forem atendidas, como:

- Validação do estado inicial do sistema.
- Sinal do botão de partida.
- Presença da peça na bancada de corte.
- Peça detectada no sensor inicial.

B Intertravamento de processo

- Bancada de Corte: O robô de corte inicia uma nova peça assim que a anterior for validada e enviada para a próxima bancada. O sensor CV sinaliza que a peça foi cortada e validada.
- Bancada de Montagem: A montagem pode ocorrer independentemente do corte de novas peças. O operador só monta componentes se a peça estiver presente na esteira.
- Bancada de Adição de Código: O operador 2 começa o upload enquanto a bancada de montagem está processando outra peça.

C Intertravamento de funcionamento

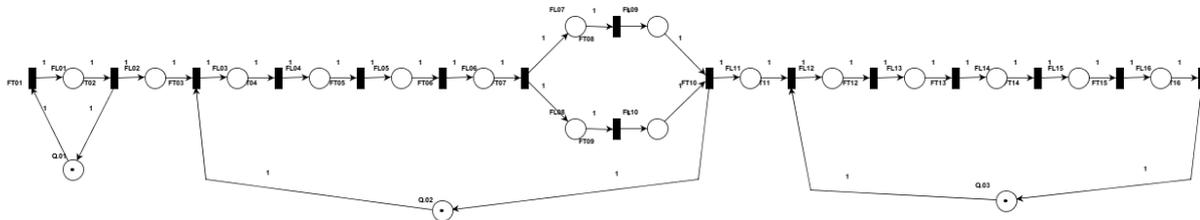
O funcionamento do sistema segue o fluxo baseado na validação de cada etapa e nos sensores. A lógica de controle do CLP deve garantir que, a próxima bancada só é liberada após a validação da bancada anterior e que os alarmes são acionados em caso de falhas nos sensores ou nas transições.

3.4 Validação do modelo

O modelo do sistema de controle da linha de montagem do carrinho seguidor de linha foi testado e apresentou o funcionamento esperado. Na ausência de sinais de entrada ativos, o sistema opera de forma contínua, alternando entre as etapas de corte, montagem e adição de código. Quando um sinal de entrada é detectado na bancada de corte, o robô inicia o corte das peças necessárias. Após a conclusão do corte, o sistema sinaliza a bancada de montagem para que o operador adicione os componentes eletrônicos e mecânicos ao carrinho. Finalmente, o sistema direciona o carrinho

para a bancada de adição de código, onde o microcontrolador é programado.

Figura 7 - Modelo Validado no PIPE.



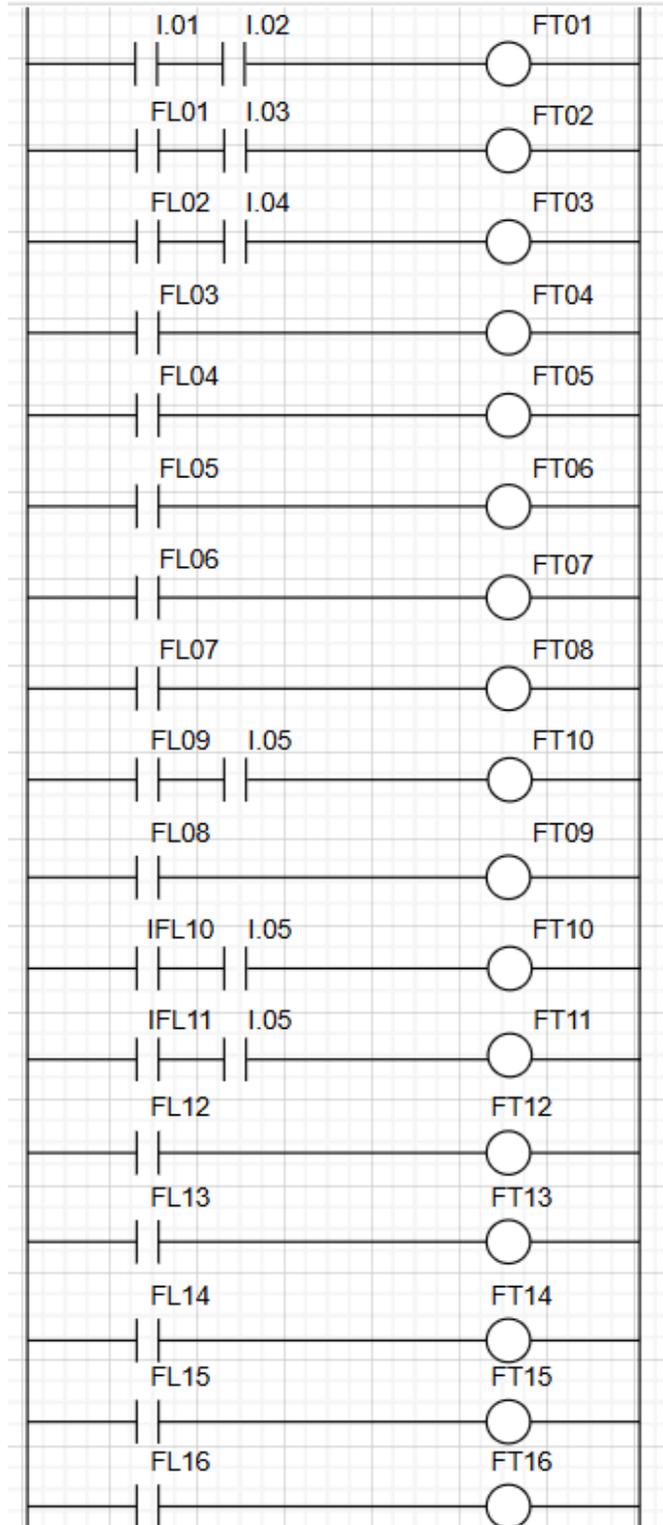
Fonte: Autoria própria

3.5 Aplicação do método para programação do controlador

Para programar o sistema de controle da linha de montagem do carrinho seguidor de linha, utilizou-se a linguagem Ladder. Foram atribuídas variáveis internas para cada transição e lugar da Rede de Petri (RdP). Os eventos internos foram associados às variáveis de saída de monitoração e atuação, enquanto os eventos externos foram associados às variáveis de entrada de comando e sensoriamento.

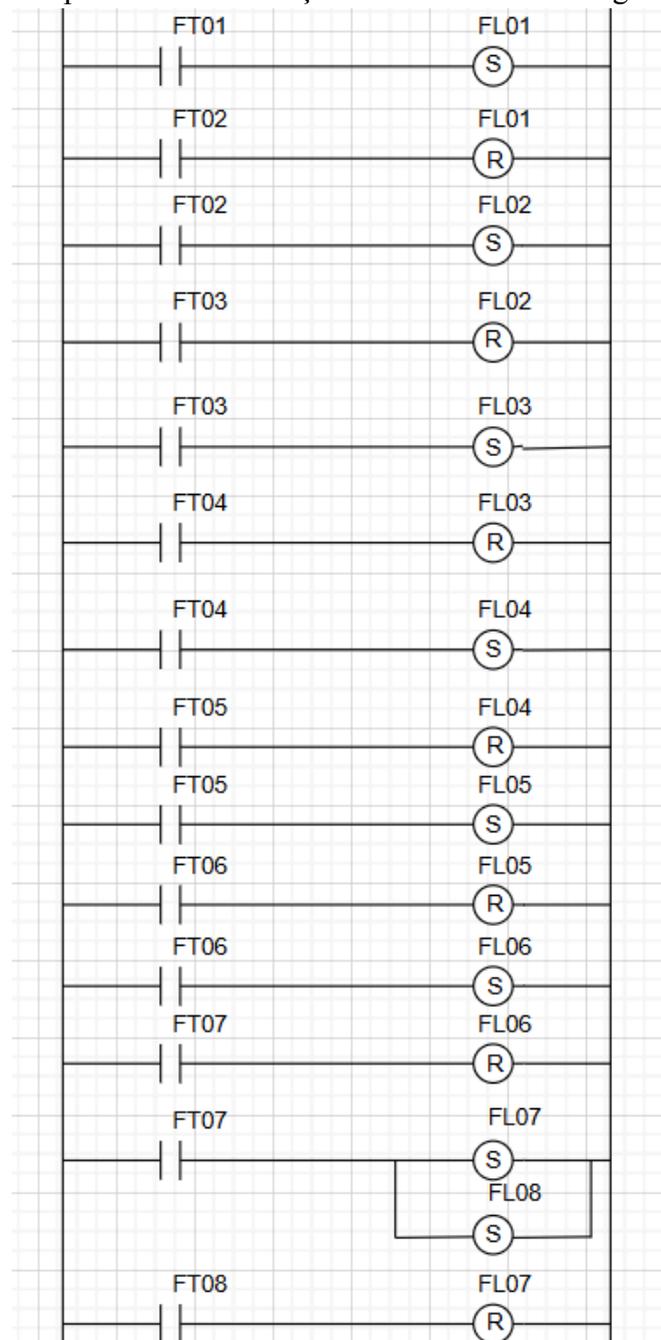
A programação em Ladder foi realizada em etapas, seguindo um processo metódico. Primeiramente, gerou-se o diagrama Ladder (LD) correspondente à habilitação de cada uma das transições. Em seguida, foi gerado o LD correspondente à mudança de estado dos lugares. Por último, foi gerado o LD correspondente aos eventos externos. Essa abordagem estruturada garantiu a precisão e a eficiência do sistema de controle da linha de montagem.

Figura 8 - LD correspondente a habilitação de cada uma das transições.



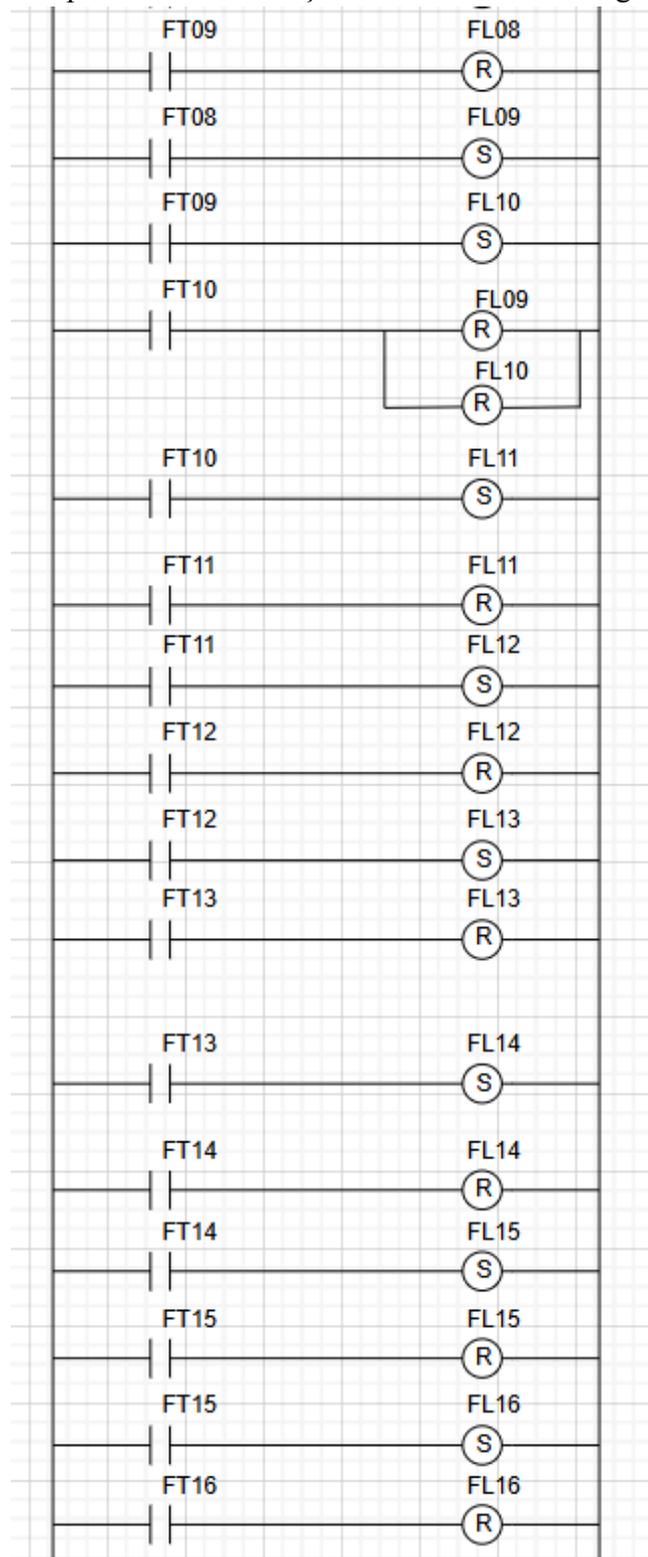
Fonte: Autoria própria

Figura 9 - LD correspondente a mudança de estado local dos lugares, primeira parte.



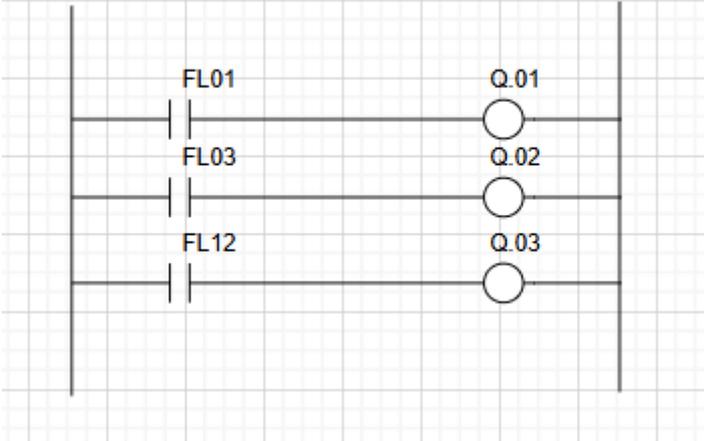
Fonte: Autoria própria

Figura 10 - LD correspondente a mudança de estado local dos lugares, segunda parte.



Fonte: Autoria própria

Figura 11 - LD correspondente aos eventos externos..



Fonte: Autoria própria

4 Observações finais

Durante o estudo do conceito de Sistemas a Eventos Discretos (SED), identificamos que um dos principais desafios está na modelagem inicial do sistema. Esta etapa demanda habilidades para simplificar o processo sem comprometer a validade e a representatividade do modelo. Tal complexidade destaca a necessidade de um aprendizado contínuo, tanto teórico quanto prático, para dominar as ferramentas, metodologias e técnicas associadas. A validação dos resultados, por sua vez, requer uma análise minuciosa, pois inconsistências podem surgir tanto da aleatoriedade dos dados quanto da falta de refinamento do modelo.

Ao longo de nossa experiência prática, enfrentamos dificuldades específicas relacionadas à análise de processos em sistemas dinâmicos, como na modelagem de linhas de montagem de carrinhos seguidores de linha. Inicialmente, erros em etapas específicas complicaram desnecessariamente a estrutura do sistema. No entanto, aprendemos a abordar esses desafios observando cada processo individualmente e desmembrando a linha de montagem em etapas claras e simples. Essa abordagem permitiu construir e validar uma metodologia PFS/RdP mais robusta, capaz de representar fielmente a dinâmica do sistema.

Portanto, os SEDs se destacam não apenas como ferramentas indispensáveis na análise de sistemas dinâmicos, mas também como métodos que promovem uma compreensão mais profunda dos modelos estudados. Através de uma abordagem estruturada e incremental, é possível identificar gargalos, refinar processos e aplicar metodologias de controle, como a PFS/RdP, para alcançar uma visão integrada e otimizada do sistema como um todo.

REFERÊNCIAS

MIYAGI, Paulo Eigi. Controle Programável: Fundamentos do Controle de Sistemas a Eventos Discretos. São Paulo, Edgard Blücher, 1996.

MIYAGI, Paulo Eigi. Introdução à Simulação Discreta. São Paulo: Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos, 2020.

CASSOLI, Beatriz Bretones. Product architecture definition and implementation in a Learning Factory focused on Industry 4.0. Trabalho de Formatura (Engenharia de Produção). Escola Politécnica da Universidade de São Paulo, 2019.

LUO, Jiliang; WAN, Yaxin; WU, Weimin; LI, Zhiwu. Optimal Petri-Net Controller for Avoiding Collisions in a Class of Automated Guided Vehicle Systems. IEEE Transactions on Intelligent Transportation Systems, v. 21, n. 11, p. 4526-4530, nov. 2020. DOI: 10.1109/TITS.2019.2937058.

Title – Line Follower Cart Manufacturing Line

Abstract – This project aims to develop and analyze a line-following cart assembly system using the Discrete Event Systems (DES) approach. The system encompasses three main stages: a cutting station, where a robot performs the cutting of parts; an assembly station, where the operator adds the battery, Arduino, and two line sensors to the cart; and a coding station, where another operator programs the microcontroller. The primary challenge in automation and modeling is ensuring the synchronization of these stages, eliminating bottlenecks in the production flow, and ensuring operational efficiency. The system inputs include the robot command at the cutting station, the manual input of components at the assembly station, and the code entered by the operator at the final station. The outputs include the states of the actuators responsible for cutting, assembling, and finalizing the cart. To model the assembly process, Petri Nets were used, allowing the analysis of the system's dynamic behavior, and Sequential Functional Programming (SFP) was employed to structure the control logic and ensure the synchronization of operations. The analysis of the assembly process model provides insights into the stages of the process, enabling the study of how the assembly process is modeled. The results obtained demonstrate that the application of DES methodologies enabled a detailed analysis of the assembly process and allowed the implementation of a reliable and efficient system. The project's impact includes optimizing the production process, reducing idle times, and improving the final product's quality. The development reinforces the use of DES in industrial applications, contributing to advances in automation and control.

Keywords – Discrete Event Systems, Petri Nets, automation, cart assembly, PFS.



Dayane Pereira de Souza. Aluna de graduação do curso de Engenharia Mecatrônica na Escola Politécnica da Universidade de São Paulo. Membro do grupo de extensão PET Mecatrônica.



Miguel Fernandes Viana dos Santos Oliveira. Aluno de graduação do curso de Engenharia Mecatrônica na Escola Politécnica da Universidade de São Paulo. Membro do grupo de extensão PET Mecatrônica.