

INSTITUTO FEDERAL DO ESPÍRITO SANTO
CURSO SUPERIOR DE ENGENHARIA DE CONTROLE E AUTOMAÇÃO

RENAN MAINARDI DE OLIVEIRA

**UM SISTEMA DE CONTROLE DE VAGAS E IDENTIFICAÇÃO DO TIPO DE
VEÍCULO EM UM ESTACIONAMENTO USANDO CLP**

RENAN MAINARDI DE OLIVEIRA

**UM SISTEMA DE CONTROLE DE VAGAS E IDENTIFICAÇÃO DO TIPO DE
VEÍCULO EM UM ESTACIONAMENTO USANDO CLP**

Trabalho de Conclusão de Curso apresentado à Co-ordenadoria do Curso de Engenharia de Controle e Automação do Instituto Federal do Espírito Santo, Campus Serra, como requisito parcial para a obtenção do título de Bacharel em Engenharia de Controle e Automação.

Orientador: Prof. Dr. Richard Junior Manuel Godinez Tello

Serra
2022

Dados Internacionais de Catalogação na Publicação (CIP)

O48s

2022

Oliveira, Renan Mainardi de

Um sistema de controle de vagas e identificação do tipo de veículo
em um estacionamento usando CLP / Renan Mainardi de Oliveira. -
2022.

72 f.; il.; 30 cm

Orientador: Prof. Dr. Richard Junior Manuel Godinez Tello.

Monografia (graduação) - Instituto Federal do Espírito Santo,
Coordenadoria de Automação, Curso de Graduação em Engenharia
de Controle de Automação, 2022.

1. Controladores programáveis. 2. Estacionamento de automóveis.
3. Detectores. 4. Ladder (Linguagem de programação de
computador). I. Tello, Richard Junior Manuel Godinez. II. Instituto
Federal do Espírito Santo. III. Título.

CDD 629.8

Bibliotecária Rogeria Gomes Belchior - CRB6/ES 417

RENAN MAINARDI DE OLIVEIRA

**UM SISTEMA DE CONTROLE DE VAGAS E IDENTIFICAÇÃO DO TIPO DE
VEÍCULO EM UM ESTACIONAMENTO USANDO CLP**

Trabalho de Conclusão de Curso apresentado
como parte das atividades para obtenção do
título de Bacharel em Engenharia de Controle
e Automação, do curso de Engenharia de
Controle e Automação do Instituto Federal do
Espírito Santo.

Aprovado em 22 de junho de 2022.

COMISSÃO EXAMINADORA

Prof. Dr. Richard Junior Manuel Godinez (Orientador)
Instituto Federal do Espírito Santo - Campus Serra

Prof. Dr. Flavio Barcelos Braz da Silva
Instituto Federal do Espírito Santo - Campus Serra

Prof. Dr. Wagner Teixeira da Costa
Instituto Federal do Espírito Santo - Campus Serra



Emitido em 22/06/2022

FOLHA DE APROVAÇÃO-TCC Nº 3/2022 - SER-CCECA (11.02.32.01.08.02.03)

(Nº do Protocolo: NÃO PROTOCOLADO)

(Assinado digitalmente em 23/06/2022 15:24)

FLAVIO BARCELOS BRAZ DA SILVA

PROFESSOR DO ENSINO BASICO TECNICO E TECNOLOGICO

SER-CGEN (11.02.32.01.08.02)

Matrícula: 1436508

(Assinado digitalmente em 22/06/2022 15:25)

RICHARD JUNIOR MANUEL GODINEZ TELLO

PROFESSOR DO ENSINO BASICO TECNICO E TECNOLOGICO

SER-CGEN (11.02.32.01.08.02)

Matrícula: 2280760

(Assinado digitalmente em 24/06/2022 23:36)

WAGNER TEIXEIRA DA COSTA

PROFESSOR DO ENSINO BASICO TECNICO E TECNOLOGICO

SER-CGEN (11.02.32.01.08.02)

Matrícula: 1445454

Para verificar a autenticidade deste documento entre em <https://sipac.ifes.edu.br/documentos/> informando seu número: **3**, ano: **2022**, tipo: **FOLHA DE APROVAÇÃO-TCC**, data de emissão: **22/06/2022** e o código de verificação: **50744aef4**

AGRADECIMENTOS

Agradeço primeiramente e acima de tudo a Deus, por estar sempre comigo e me permitir chegar a este momento.

À minha amada esposa Giseli, por todo apoio e paciência durante esta caminhada. Meu amor, sem você não teria sido possível!

Aos meus pais, Eduardo e Thereza, e à minha irmã, Cibele, pelo suporte e incentivo desde sempre. Vocês são essenciais na minha vida!

Ao professor Richard Tello, meu orientador, pelas valiosas dicas e correções, fundamentais para o sucesso deste trabalho.

Aos colegas da Transpetro, em especial o gerente setorial Marco Antônio Vieira e os supervisores Ricardo Galdino, Patrick Alvim e Giuliano Casteluber, pela importante participação no meu desenvolvimento técnico, que foi de grande valia para a execução deste TCC.

Por fim, agradeço aos amigos que sempre me incentivaram, apesar de todas as dificuldades, a concluir esta missão. Muito obrigado a todos!

RESUMO

A dificuldade de se encontrar vagas em estacionamentos é um dos principais problemas das grandes cidades, e é atenuada com o uso de parquímetros e estacionamentos rotativos. O valor cobrado, porém, não é o mesmo para todos os tipos de veículos. Uma pesquisa sobre trânsito veicular realizada em diferentes países foi publicada em 2017, indicando que no Reino Unido foram perdidas 1,4 bilhão de horas na procura por uma vaga para estacionar, na Alemanha foram 1,9 bilhão de horas, enquanto nos EUA foram 3,6 bilhões. Segundo a pesquisa, estas horas perdidas trouxeram números expressivos em desperdício de combustível no ano, da ordem de bilhões de galões (1 galão equivale a aproximadamente 3,8 litros). Este trabalho propõe uma solução automatizada para estes dois problemas: gerenciamento da lotação de um estacionamento e identificação dos tipos de veículos que o acessam. A solução proposta utiliza um Controlador Lógico Programável (CLP), sensores para identificação/contagem dos veículos, e sinalizadores para indicar lotação máxima atingida ou possível falha no sistema. Este processo de automação foi desenvolvido através da linguagem *Ladder*. Primeiramente foram realizadas simulações no software Codesys e, em seguida, testes em bancada, com uso de um CLP comercial da fabricante Siemens modelo S7-1200. Apresenta-se também sugestões de equipamentos e instrumentos para aplicação real. Finalmente, após a análise dos resultados, são apresentadas possíveis melhorias, que podem trazer vantagens à solução desenvolvida.

Palavras-chave: Automação. Estacionamento. Vagas. Veículos. Identificação. CLP. Controlador. Ladder. Sensores.

ABSTRACT

The difficulty of finding parking spaces is one of the main problems in big cities, which is mitigated by using parking meters and rotating parking lots. The amount charged, however, is not the same for all types of vehicles. A research about vehicular traffic carried out in different countries was published in 2017, indicating that in the UK 1.4 billion hours were lost in the search for a parking space, in Germany it was 1.9 billion hours, while in the USA were 3.6 billion. According to the research, these lost hours resulted in expressive numbers in fuel wastage per year, on the order of billions of gallons (1 gallon is equivalent to approximately 3.8 liters). This work proposes an automated solution for these two problems: managing the capacity of a parking lot and identifying the types of vehicles that access it. This can be done with use of a Programmable Logic Controller (PLC), sensors for identification and counting of vehicles, and signaling lamps to indicate maximum capacity reached or possible failure in the system. This automation process was developed through the Ladder language. First, simulations were performed in the Codesys software and, then, bench tests were performed, using a commercial PLC from Siemens, model S7-1200. Suggestions of equipment and instruments for real application are also presented. Finally, after analyzing the results, possible improvements are presented, which can bring advantages to the developed solution.

Keywords: Automation. Parking. Parking Spaces. Vehicles. Identification. PLC. Controller. Ladder. Sensors.

LISTA DE FIGURAS

Figura 1 – Gráfico comparativo de aspectos relacionados à mobilidade em cidades versus Estimativa de economia em número de barris de petróleo por ano.	13
Figura 2 – Primeiro CLP - Modicon 084	17
Figura 3 – Partes que compõem um sistema de controle baseado em CLP	17
Figura 4 – Linguagens de programação de CLP segundo a norma IEC 61131-3	19
Figura 5 – Exemplo de programação usando linguagem <i>ladder</i>	20
Figura 6 – Exemplo de programação usando linguagem de blocos funcionais (FBD)	20
Figura 7 – Exemplo de programação usando SFC/Grafcet	21
Figura 8 – Exemplo de programação usando Lista de Instrução (IL)	21
Figura 9 – Exemplo de programação usando Texto Estruturado (ST)	21
Figura 10 – Representação de um temporizador TON no (a) Codesys e no (b) TIA Portal	23
Figura 11 – Gráfico de funcionamento de um temporizador TON	23
Figura 12 – Representação de um contador CTUD no (a) Codesys e no (b) TIA Portal	24
Figura 13 – Representação da instrução MOVE no (a) Codesys e no (b) TIA Portal	24
Figura 14 – Representação de uma instrução SET/RESET (a) em degraus distintos e (b) em formato de bloco SR	25
Figura 15 – Representação de um detector de borda de subida: (a) R_TRIG no Codesys e (b) P_TRIG para o TIA Portal	25
Figura 16 – Linguagens habilitadas para programação no Codesys	27
Figura 17 – Diagrama dos elementos presentes na entrada do estacionamento indicando as variáveis (<i>tags</i>) utilizadas	31
Figura 18 – Diagrama dos elementos presentes na saída do estacionamento indicando as variáveis (<i>tags</i>) utilizadas	32
Figura 19 – Fluxograma Geral do Sistema	32
Figura 20 – Posição dos sensores fim de cursos (indicando os respectivos <i>tags</i>) nas cancelas CE e CD	33
Figura 21 – Fluxograma de detecção de falhas	34
Figura 22 – Espaço de trabalho para programação em <i>Ladder</i> no software Codesys V3.5 SP17	35
Figura 23 – Controlador Lógico Programável modelo S7-1200 da fabricante Siemens	35
Figura 24 – Placa simuladora de estados lógicos das entradas modelo SIM 1274	36
Figura 25 – Localização dos sensores de laço indutivo	37
Figura 26 – Sensor de anti-esmagamento da fabricante DECIBEL modelo D60-1	37
Figura 27 – Exemplo de cancela eletrônica da fabricante TAU de 7 metros de comprimento e motores de torque de 300 Nm	38
Figura 28 – Alguns modelos de cancela eletrônica possuem um complexo sistema de molas no seu interior	38
Figura 29 – Modelos de máquina expedidora de ticket da fabricante Amano	39

Figura 30 – Sinaleiro vermelho do fabricante Metaltex modelo L20-AR9-R	39
Figura 31 – Inicialização do sistema.	43
Figura 32 – Solicitação de ticket e confirmação de impressão.	43
Figura 33 – Abertura das cancelas.	44
Figura 34 – Fechamento das cancelas.	44
Figura 35 – Desativação da variável Ticket_impresso.	45
Figura 36 – Contagem de veículos e indicação de lotação	45
Figura 37 – Ativação do sensor antiesmagamento durante o fechamento das cancelas	46
Figura 38 – Indicação de falha	47
Figura 39 – Parada de Emergência e Reset de diversas variáveis	47
Figura 40 – Simulação da impressão do ticket	48
Figura 41 – Simulação da entrada de veículo de quatro rodas	48
Figura 42 – Simulação da entrada de veículo de duas rodas	48
Figura 43 – Simulação do estacionamento totalmente lotado	49
Figura 44 – Simulação do fechamento das cancelas.	50
Figura 45 – Simulação de falha quando a cancela não abre em 40 segundos	50
Figura 46 – TIA Portal: Tela Inicial	51
Figura 47 – TIA Portal: Seleção da CPU	51
Figura 48 – TIA Portal: Conexão Ethernet	52
Figura 49 – Equipamentos utilizados nos testes: CLP Siemens S7-1200, Simulador SIM 1274, fonte de alimentação de 24Vcc, Multimetro, cabo Ethernet e jumpers conectores	52
Figura 50 – Listagem dos <i>tags</i> e respectivos endereços no Tia Portal	53
Figura 51 – Teste no CLP: Inicialização do Sistema (Botoeira START)	54
Figura 52 – Teste no CLP: Solicitação de ticket	54
Figura 53 – Teste no CLP: Confirmação de impressão de ticket	54
Figura 54 – Teste no CLP: Abertura das cancelas para entrada de veículo de quatro rodas	55
Figura 55 – Teste no CLP: Fechamento das cancelas	55
Figura 56 – Teste no CLP: Estacionamento lotado	56
Figura 57 – Teste no CLP: Sensor anti-esmagamento atuado	57
Figura 58 – Teste no CLP: Indicação de Falha	57
Figura 59 – Teste no CLP: Desligamento do sistema (Botoeira STOP)	58

LISTA DE TABELAS

Tabela 1 – Comparativo de prejuízos associados à busca de vagas de estacionamento (valores por ano)	12
Tabela 2 – Posição dos Maiores Produtores Mundiais de Veículos em 2017 divulgado pela Organização Internacional de Fabricantes de Veículos Automotores (OICA).	14
Tabela 3 – Tabela comparativa de trabalhos correlatos entre os anos 2011-2022. . .	29
Tabela 4 – <i>Tags</i> das Saídas Digitais	40
Tabela 5 – <i>Tags</i> das Entradas Digitais	41
Tabela 6 – Variáveis internas do CLP	41
Tabela 7 – Funções utilizadas	42

SUMÁRIO

1	INTRODUÇÃO	12
1.1	OBJETIVOS	14
1.1.1	Objetivo Geral	14
1.1.2	Objetivos Específicos	14
1.1.3	Limitações do Trabalho	15
1.2	ESTRUTURA DO TRABALHO	15
2	REFERENCIAL TEÓRICO	16
2.1	O CONTROLADOR LÓGICO PROGRAMÁVEL (CLP)	16
2.1.1	Fabricantes de CLP	18
2.1.2	Linguagens de Programação de CLP e a norma IEC 61131-3	18
2.1.3	Instruções básicas e avançadas de programação de CLPs	21
2.1.3.1	Temporizadores	22
2.1.3.2	Contadores	23
2.1.3.3	Instrução MOVE	24
2.1.3.4	Instruções SET/RESET	24
2.1.3.5	Instruções R_TRIG/F_TRIG ou P_TRIG/N_TRIG	25
2.2	ESPECIFICAÇÃO DE SENsoRES E ATUADORES PARA SOLUÇÕES BASEADAS EM CLP	25
2.3	SIMULADORES PARA O CONTROLE DE CLP	26
2.3.1	O software de simulação CODESYS	26
2.4	TRABALHOS CORRELATOS	27
3	MATERIAIS E MÉTODOS	30
3.1	O PROBLEMA	30
3.2	METODOLOGIA ADOTADA NA SOLUÇÃO AO PROBLEMA	30
3.3	MATERIAIS UTILIZADOS	34
3.3.1	Software de Simulação CODESYS V3.5 SP17	34
3.3.2	CLP Siemens S7-1200 e Módulo SIM 1274	34
3.4	SENOsoRES INDICADOS PARA IMPLEMENTAÇÃO REAL	36
3.4.1	Sensor de laço indutivo para detecção de veículos	36
3.4.2	Sensor Fim de Curso para confirmação de posição das cancelas	36
3.4.3	Sensor Anti-esmagamento	37
3.5	ATUADORES E SINALIZADORES INDICADOS PARA IMPLIMENTAÇÃO REAL	38
3.5.1	Cancela eletrônica	38
3.5.2	Máquina Expedidora de Ticket	39
3.5.3	Sinalizadores visuais de Lotação e Falha	39
4	EXPERIMENTOS, RESULTADOS E DISCUSSÃO	40
4.1	A PROGRAMAÇÃO	40
4.2	SIMULAÇÃO EM CODESYS	47

4.2.1	Simulações com o Sistema em funcionamento	47
4.2.1.1	Veículo se aproxima e há vaga disponível	47
4.2.1.2	Veículo se aproxima e não há vaga disponível	48
4.2.1.3	Fechamento das cancelas	49
4.2.2	Simulações de falhas	49
4.3	TESTES NO CLP SIEMENS S7-1200	50
4.4	ANÁLISE E DISCUSSÃO	58
5	CONCLUSÃO	60
5.1	TRABALHOS FUTUROS	60
	REFERÊNCIAS	61
	APÊNDICE A – Programação completa da simulação em Codesys	64
	APÊNDICE B – Programação completa dos testes no CLP Siemens S7-1200	68

1 INTRODUÇÃO

O congestionamento veicular é um problema alarmante em escala global que vem crescendo a cada ano. Um fator que contribui para estes índices é o problema do estacionamento, cujas causas incluem a produção elevada de veículos, o aumento do tamanho dos atuais veículos e o número reduzido de vagas de estacionamento nas áreas urbanas. Procurar uma vaga para estacionar é uma atividade rotineira para muitas pessoas em cidades ao redor do mundo. Esta atividade, que pode chegar a ser frustrante, ocasiona também perdas de tempo e combustível.

Em 2017, a INRIX (uma agência norte-americana especializada em análise de dados para gerenciamento de tráfego) publicou os resultados de um estudo abrangente relacionado a estacionamentos, uma pesquisa realizada com 17.968 motoristas em 30 das maiores cidades dos EUA, Reino Unido e Alemanha. Seus dados mostram que nos EUA, a cada ano as pessoas perdem 3,6 bilhões de horas tentando encontrar uma vaga para estacionar, acarretando um desperdício de cerca de 1,7 bilhão de galões de combustível por ano (1 galão corresponde a aproximadamente 3,8 litros), o equivalente a cerca de US \$ 72,7 bilhões em perdas econômicas (COOKSON; PISHUE, 2017). Veja na Tabela 1 uma comparação com dados do Reino Unido e da Alemanha, segundo a INRIX.

Tabela 1 – Comparativo de prejuízos associados à busca de vagas de estacionamento (valores por ano)

País	Horas gastas	Galões de combustível	Perdas econômicas (Moeda)
Estados Unidos	3,6 Bilhões	1,7 Bilhão	72,7 Bilhões (USD - Dólar)
Reino Unido	1,4 Bilhão	550 Milhões	23,3 Bilhões (GBP - Libra)
Alemanha	1,9 Bilhão	3,2 Bilhões	40,4 Bilhões (EUR - Euro)

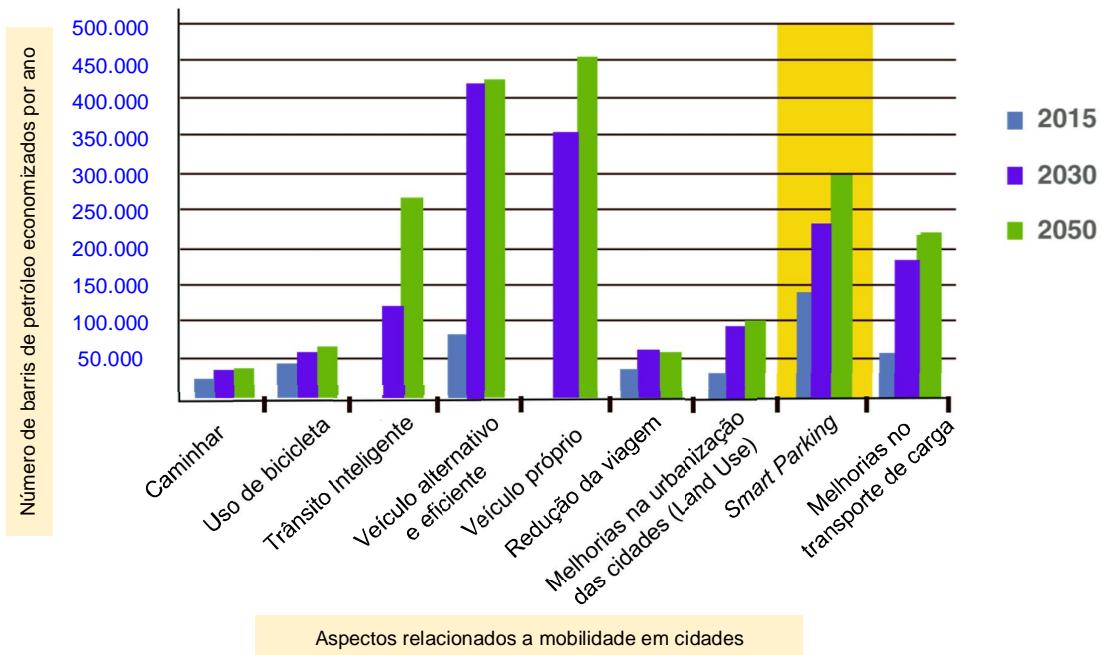
Fonte: Informações extraídas de (COOKSON; PISHUE, 2017)

Um relatório relacionado a estimativas de economia de combustível por ano (ver Figura 1) foi apresentado por Basu (2014). Da Figura 1 infere-se que a expansão de Estacionamentos Inteligentes - *Smart Parking* - pode oferecer uma economia de mais de 220.000 barris de petróleo até 2030 e de aproximadamente 300.000 barris até 2050.

Em 2021, a INRIX publicou uma informação mais recente indicando que 550 milhões de Euros é o impacto econômico anual na procura de um estacionamento em Berlim (Alemanha) (INRIX, 2021). Adicionalmente, a publicação indica que a cada ano o motorista perde, em média, 55 horas procurando estacionamento (INRIX, 2021).

À medida que as cidades crescem sem o devido planejamento ou alguma forma de gestão dos estacionamentos, os problemas devem aumentar. Um outro aspecto a considerar é o número crescente de carros em uso, tornando mais difícil encontrar uma vaga de estacionamento. Uma pergunta muito pertinente poderia ser: quantos carros existem no

Figura 1 – Gráfico comparativo de aspectos relacionados à mobilidade em cidades versus Estimativa de economia em número de barris de petróleo por ano.



Fonte: Extraída, adaptada e traduzida de Basu (2014)

mundo? A quantidade específica pode significar um verdadeiro desafio considerando os diversos órgãos e instituições responsáveis por quantificá-los. Em 2017, uma publicação da WardsIntelligence (WARDSINTELLIGENCE, 2017) - uma divisão do grupo norte-americano Informa Tech Automotive dedicado à divulgação de informações, dados e notícias referentes à indústria automobilística - fez uma estimativa do número de carros, caminhões e ônibus, indicando uma quantia em torno de 1,32 bilhão. A estimativa não incluía veículos off-road ou maquinário pesado.

Dados divulgados pela Organização Internacional de Fabricantes de Veículos Automotores (OICA - International Organization of Motor Vehicle Manufacturers) referentes à produção de veículos em 2017 (OICA, 2021) são mostrados na Tabela 2. Segundo a listagem, naquele ano o Brasil encontrava-se na nona posição entre os maiores produtores mundiais.

Uma das dificuldades nos estacionamentos públicos e privados é a capacidade de conhecer o número de vagas disponíveis dentro de um estabelecimento, o que poderia ajudar na gestão logística e de tempo dos potenciais clientes/usuários. Atualmente existem equipamentos/dispositivos para automação de entrada e saída de veículos bastante consolidados, porém, o custo é um fator a ser considerado. Uma das premissas na elaboração deste projeto é a busca pela simplicidade na programação, robustez na escolha do dispositivo controlador, menos envolvimento humano no controle e baixo custo de implementação na identificação do tipo de veículo aplicado a estacionamentos de pequeno porte. Assim, este trabalho pretende contribuir para a melhoria do gerenciamento das vagas de estacionamentos,

Tabela 2 – Posição dos Maiores Produtores Mundiais de Veículos em 2017 divulgado pela Organização Internacional de Fabricantes de Veículos Automotores (OICA).

Posição	País	Carros	Veículos Comerciais	Total
1	China	24.806.687	4.208.747	29.015.434
2	Estados Unidos	3.033.216	8.156.769	11.189.985
3	Japão	8.347.836	1.345.910	9.693.746
4	Alemanha	5.645.581	0	5.645.581
5	Índia	3.952.550	830.346	4.782.896
6	Coreia do Sul	3.735.399	379.514	4.114.913
7	México	1.900.029	2.168.386	4.068.415
8	Espanha	2.291.492	556.843	2.848.335
9	Brasil	2.269.468	430.204	2.699.672

Fonte: Informação extraída e traduzida de OICA (2021)

especialmente no que tange ao controle da ocupação e à distinção do tipo de veículo, possibilitando também a tarifação correta para cada um.

1.1 OBJETIVOS

1.1.1 Objetivo Geral

O objetivo geral deste trabalho é desenvolver um sistema de automação inteligente para a gestão da ocupação das vagas em um estacionamento através da contagem dos veículos que entram e saem deste e identificar o tipo de veículo para uma tarifação diferenciada. Esta identificação começará na entrada do estacionamento, que conta com um par de cancelas. A automação desenvolvida pretende minimizar o tempo operacional relacionado à gestão de estacionamentos, bem como garantir a segurança durante o funcionamento do sistema.

1.1.2 Objetivos Específicos

Os objetivos específicos identificados para se atingir o objetivo geral proposto são:

- Desenvolver uma lógica baseada em Controlador Lógico Programável (CLP), dadas as características de robustez e flexibilidade;
- Realizar uma revisão bibliográfica de trabalhos correlatos ou similares a este;
- Identificar as entradas, saídas e variáveis internas a serem usadas para esta automação;
- Desenvolver um programa baseado em uma das linguagens da norma IEC 61131-3 e testar a solução em uma plataforma de simulação;

- Com base nos testes de simulação, adaptar a programação, transferir a lógica programada para um CLP real e realizar testes em bancada;
- Avaliar a eficácia e a viabilidade da solução proposta com base nos testes realizados.

1.1.3 Limitações do Trabalho

A solução proposta limita-se ao desenvolvimento de um algoritmo de automação, com realização de testes simulados e uma adaptação desta lógica para um CLP comercial. Entretanto, não foi realizada a instalação em ambiente real nem realizados testes com sensores. As mudanças do estado lógico das variáveis relacionadas a sensores foram simuladas diretamente via software e, depois, através de uma placa de testes de um CLP comercial (Módulo SIM 1274).

1.2 ESTRUTURA DO TRABALHO

Este trabalho está dividido em cinco capítulos. O presente capítulo traz uma introdução ao tema, a justificativa para a sua realização e os objetivos que se pretende alcançar.

O capítulo 2 apresenta o referencial teórico. No final do capítulo são apresentados trabalhos correlatos e um quadro comparativo, incluindo este trabalho.

O capítulo 3 descreve os materiais e métodos utilizados para o desenvolvimento da automação. O capítulo inicia descrevendo o problema e a metodologia adotada na solução. Serão mostrados diversos diagramas para um melhor entendimento. Aqui serão elencados o software e hardware (CLP) para o desenvolvimento. No final do capítulo são apresentadas sugestões de sensores e atuadores para uma implementação real.

No capítulo 4 descreve-se o desenvolvimento, desde a programação até a simulação das diversas situações que podem ocorrer numa situação real. São também mostrados os testes realizados no CLP Siemens S7-1200, finalizando com uma análise e discussão dos resultados obtidos.

A conclusão do trabalho é apresentada no capítulo 5, seguida pela indicação de possíveis melhorias a serem implementadas em trabalhos futuros.

2 REFERENCIAL TEÓRICO

Neste capítulo serão abordados os conceitos relacionados ao tema que é o foco deste trabalho: Controladores Lógicos Programáveis, Linguagens de programação baseados na norma IEC 61131, Instruções básicas e avançadas de programação de CLPs, Especificação de sensores e atuadores para soluções baseadas em CLPs. No final do capítulo serão descritos trabalhos correlatos, os quais serão apresentados numa tabela comparativa.

2.1 O CONTROLADOR LÓGICO PROGRAMÁVEL (CLP)

O desenvolvimento dos Controladores Lógicos Programáveis, ou simplesmente CLPs (o termo usado no inglês é PLC - Programmable Logic Controller), teve inicio em 1969 dada uma necessidade da indústria da época (LAUGHTON, 2003). Até então o controle era feito basicamente com relés e contatores. Assim, para se desenvolver um novo modelo de fabricação era necessária uma completa modificação da estrutura e componentes da fábrica.

Com o auge da lógica baseada em semicondutores, os primeiros computadores comerciais começaram a se popularizar e grandes avanços na miniaturização dos componentes/dispositivos foram evidentes. Neste auge da eletrônica e da computação, o CLP teve sua expansão e difusão a partir das necessidades da indústria automobilística norte-americana, especificamente na Hydramatic Division da GM (FRANCHI; CAMARGO, 2009).

Com o apoio do engenheiro Richard Morley, foi desenvolvida uma especificação que podia atender as necessidades da indústria, não só automobilística, mas em geral (FRANCHI; CAMARGO, 2009).

Entre as necessidades de desenvolvimento deste controlador programável, destaca-se que este dispositivo deve trazer vantagens como (FRANCHI; CAMARGO, 2009):

- Facilidade de programação e reprogramação, preferivelmente na planta, para ser possível alterar a sequência de operações na linha de montagem;
- Possibilidade de manutenção e reparo, com blocos de entrada e saída modulares;
- Confiabilidade para seu uso em um ambiente industrial;
- Reduzir o tamanho da instalação comparado ao sistema tradicional de relés;

A companhia americana Bedford Associated lançou um dispositivo de computação denominado MODICON - Modular Digital Controller (FRANCHI; CAMARGO, 2009). O

primeiro CLP foi desenvolvido em 1969 (JONES, 1996; BOLTON, 2015), na Figura 2 mostra-se este desenvolvimento, trata-se do Modicon 084. Atualmente, versões modernas a este dispositivo pertencem à companhia Schneider Electric.

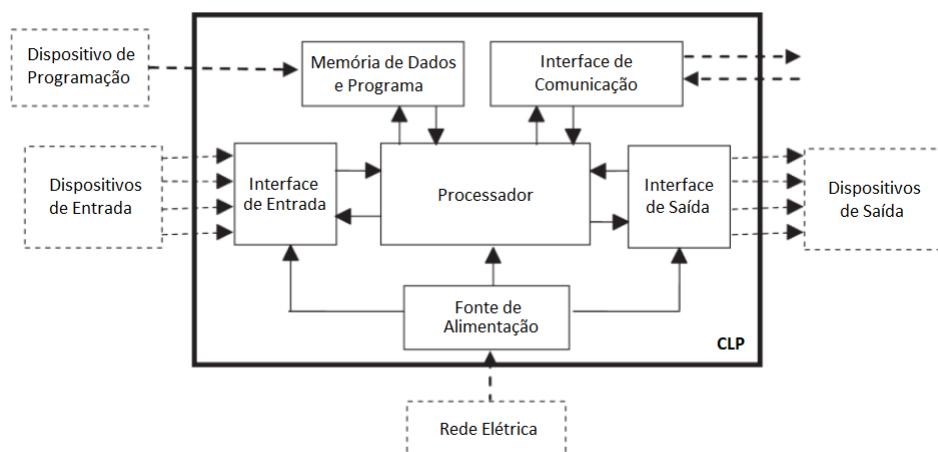
Figura 2 – Primeiro CLP - Modicon 084



Fonte: Figura extraída de: <https://blog.se.com/br/automacao-industrial/2018/11/30/modicon-50-anos-de-pioneerismo-e-inovacoes/> (2022)

Com o passar do tempo, o CLP tornou-se um dispositivo que se renova com o avanço da tecnologia, tanto no aumento da velocidade de processamento quanto na capacidade de integração em termos de novos padrões de comunicação, integração a diferentes tecnologias de Interfaces-Homem-Máquina (IHM ou HMI - Human-Machine Interface), adaptação do software de programação a novos sistemas operacionais, comunicação em rede e integração a sistemas SCADA (Supervisory Control and Data Aquisition). Na Figura 3 é mostrada uma representação em blocos das partes que compõem um CLP.

Figura 3 – Partes que compõem um sistema de controle baseado em CLP



Fonte: Extraído e traduzido de Bolton (2015)

2.1.1 Fabricantes de CLP

Atualmente existem diversos fabricantes de CLP. Entre os principais podemos mencionar os seguintes, com suas respectivas sedes entre parênteses (elencadas em ordem alfabética):

1. ABB (Suíça)
2. Festo (Alemanha)
3. General Electric (USA)
4. Mitsubishi (Japão)
5. Omron (Japão)
6. Rockwell Automation (Allen Bradley) (USA)
7. Schneider Electric (Telemechanique) (França)
8. Siemens (Alemanha)
9. WEG (Brasil)

Cada fabricante possui um modelo, um software de programação para seus dispositivos, linguagens de programação associadas a seus CLPs e diferentes estilos de programação baseadas na norma IEC 61131.

2.1.2 Linguagens de Programação de CLP e a norma IEC 61131-3

Com o aumento do número de fabricantes foi necessário também um ordenamento em questões de padronização. Este processo foi a partir das diretrizes apresentadas na norma IEC 61131. Assim, uma definição atribuída ao Controlador Lógico Programável segundo a Norma IEC 61131-1 é a seguinte (PAREDE; GOMES, 2011):

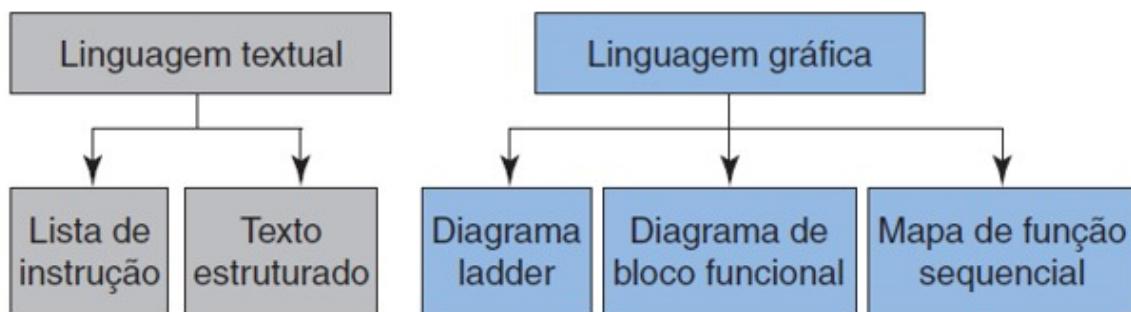
"Sistema Eletrônico digital, desenvolvido para uso em ambiente industrial, que usa uma memória programável para armazenamento interno de instruções do usuário, para implementação de funções específicas, como lógica, sequenciamento, temporização, contagem e aritmética, para controlar, por meio de entradas e saídas, vários tipos de máquinas e processos."

A norma IEC 61131-3 também recomenda uma padronização dos tipos de linguagens de programação associadas ao CLP, definindo cinco diferentes linguagens:

1. Diagrama *ladder*
2. Diagrama de Blocos de Função (FBD)
3. Gráfico de Funções Sequenciais (Grafset/SFC)
4. Lista de Instruções
5. Texto Estruturado

Estas linguagens foram categorizadas como gráficas e textuais. A Figura 4 mostra a divisão dos tipos de linguagens de acordo a sua forma de programação.

Figura 4 – Linguagens de programação de CLP segundo a norma IEC 61131-3

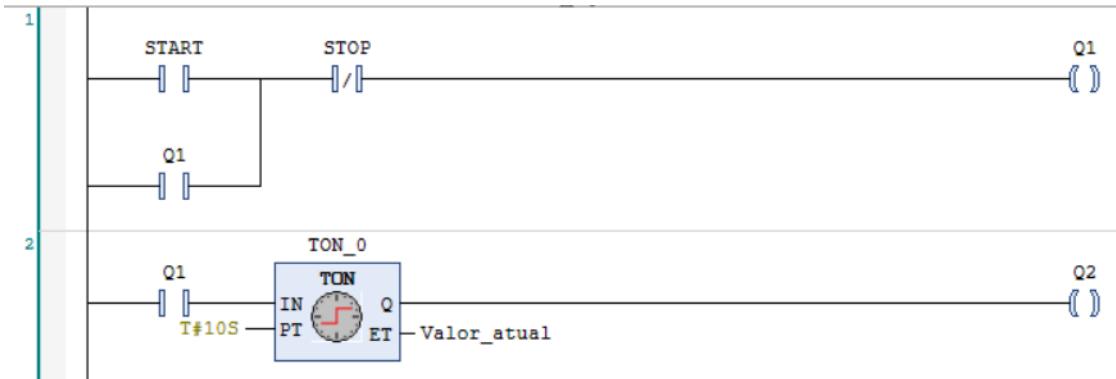


Fonte: Extraído e adaptado de Petruzella (2014)

A linguagem *ladder* ou linguagem de tipo escada é a linguagem mais tradicional, conhecida amplamente e bastante utilizada entre os programadores de CLP. Guarda relação com os sistemas de acionamento elétrico, onde existe uma rede de contatos, os quais podem estar energizados ou desenergizados, e à direita, uma bobina que representa uma saída. Existem também diversas funções adicionais, como temporizadores, contadores, multiplexadores e uma grande quantidade de recursos que tornam possível a manipulação dos dados e o controle automático dos processos. Na Figura 5 mostra-se um exemplo da lógica de controle *ladder*, onde cada linha de programação denomina-se de degrau ou *rung*.

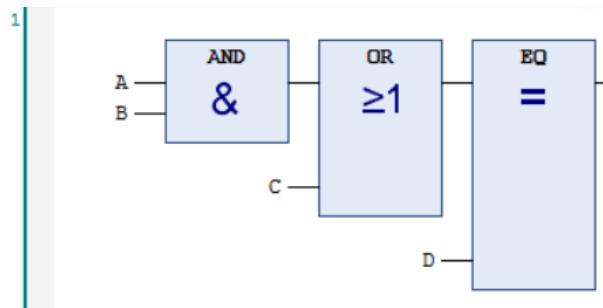
A linguagem FBD (Diagrama de Blocos de Função) – do inglês *Function Block Diagram* – é muito semelhante à lógica booleana da Eletrônica Digital, na medida em que cada bloco equivale a uma porta lógica (AND, OR, NOT, etc.). Na Figura 6 mostra-se um exemplo de programação em FBD.

Figura 5 – Exemplo de programação usando linguagem *ladder*



Fonte: Elaborado pelo autor (2022)

Figura 6 – Exemplo de programação usando linguagem de blocos funcionais (FBD)



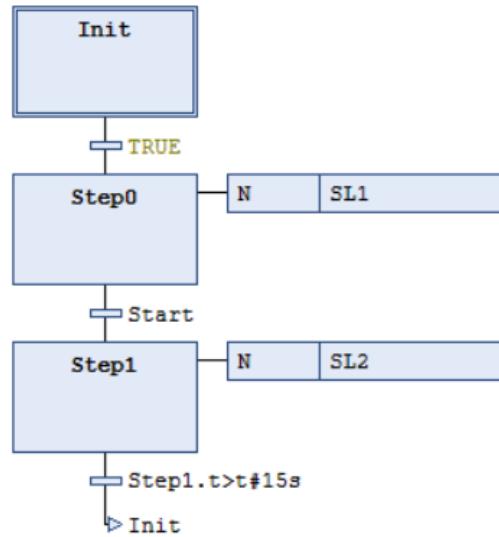
Fonte: Elaborado pelo autor (2022)

O Gráfico de Funções Sequenciais (SFC) - do inglês *Sequential Function Chart* - teve sua origem na França e era chamado inicialmente de Grafcet. É uma linguagem bastante versátil devido à sua simplicidade e similaridade a um diagrama de fluxo, além da possibilidade de incluir dentro da programação as outras linguagens descritas anteriormente. Trata-se de um diagrama que descreve o funcionamento de um processo sequencial, com ativação/desativação de etapas e transições (receptividade). Na Figura 7, mostra-se um exemplo de programação em SFC/Grafcet.

A linguagem IL (Lista de Instruções) - do inglês *Instruction List* - é uma linguagem de mais baixo nível, que utiliza mnemônicos, de maneira semelhante ao Assembly. Um exemplo desta linguagem é mostrada na Figura 8.

A linguagem de Texto Estruturado (ou, em inglês, Structured Text - ST) é similar àquelas utilizadas para programação orientada a objetos. Possui muita similaridade com a linguagem C. Com o apoio das estruturas de programação conhecidas é possível realizar a manipulação dos dados para diferentes aplicações. Na Figura 9, mostra-se um exemplo de programação em Texto Estruturado.

Figura 7 – Exemplo de programação usando SFC/Grafcet



Fonte: Elaborado pelo autor (2022)

Figura 8 – Exemplo de programação usando Lista de Instrução (IL)

```

LD   I0
OR   I1
AND  I2
ST   Q1
  
```

Fonte: Elaborado pelo autor (2022)

Figura 9 – Exemplo de programação usando Texto Estruturado (ST)

```

IF temperatura >30 THEN
  climatizador :=1;
ELSE
  climatizador :=0;
END_IF;
  
```

Fonte: Elaborado pelo autor (2022)

2.1.3 Instruções básicas e avançadas de programação de CLPs

A programação de CLPs não se resume ao uso de funções booleanas (AND, OR, XOR ou NOT) entre as entradas para uma determinada saída. Existem instruções básicas e avançadas para as diferentes linguagens de programação. Entre as instruções de seleção podemos elencar: SEL, MAX, MIN, LIMIT e MUX. Entre as instruções comparativas temos: GT (>), LT (<), GE (>=), EQ (=), LE (<=) e NE (<>). Existem também

instruções de habilitação, como SET e RESET e instruções de detecção de bordas, como R_TRIG (para detecção de bordas de subida) e L_TRIG (para detecção de bordas de descida). Há ainda instruções baseadas em sistemas sequencias, como os contadores: CTU (ascendente), CTD (descendente) e CTUD (ascendente/descendente), além de temporizadores: TP (temporizador de pulso), TON (atraso na ativação) e TOFF (atraso na desativação). Finalmente, existem instruções de programação avançada, como JUMP (para pulo), MOVE (para movimentação), BCD (para codificação binária decimal) e PID (para o controle Proporcional, Integral e Derivativo).

Algumas destas instruções foram utilizadas neste trabalho, entre elas:

- Temporizadores;
- Contadores;
- MOVE;
- Set/Reset;
- R_TRIG

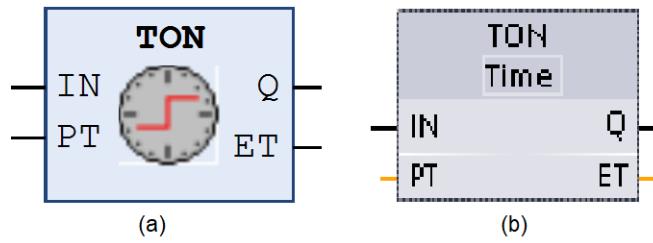
A seguir, uma breve descrição de cada uma delas:

2.1.3.1 Temporizadores

Em alguns casos, há situações em que determinado sinal de entrada deve permanecer na mesma condição por um certo tempo antes que uma ação seja tomada. Pode acontecer também de algum ruído ocasionar uma indicação falsa e acabar provocando uma reação inadequada. Há ainda os casos em que uma saída deve permanecer ativa por um determinado tempo independentemente de mudanças nas condições de entrada. Para estes casos, podemos fazer uso de temporizadores, que podem retardar o acionamento de uma variável/saída ou estabelecer um período em que esta permanecerá ativa.

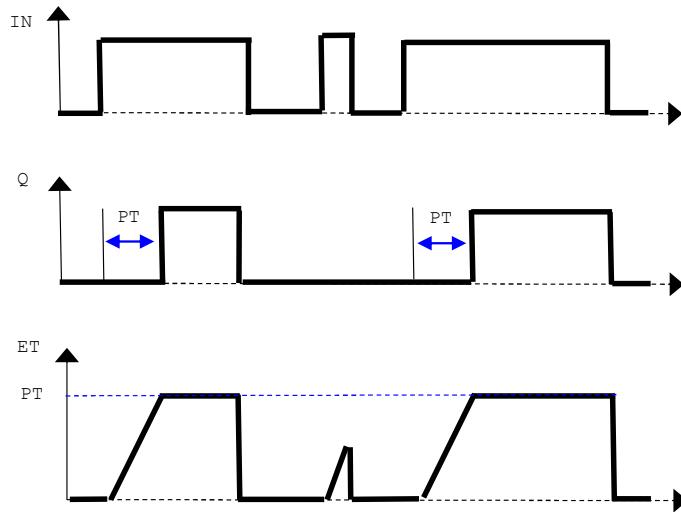
A Figura 10 ilustra a representação de um temporizador de tipo TON utilizado como tempo padrão para o fechamento das cancelas e para indicação de falha do sistema. Isto poderá ser visto em detalhes no capítulo 4. A saída (Q) é acionada depois que a entrada (IN) estiver ativa por um tempo pré-determinado (PT), conforme ilustrado na figura 11.

Figura 10 – Representação de um temporizador TON no (a) Codesys e no (b) TIA Portal



Fonte: Adaptado de Codesys e TIA Portal (2022)

Figura 11 – Gráfico de funcionamento de um temporizador TON



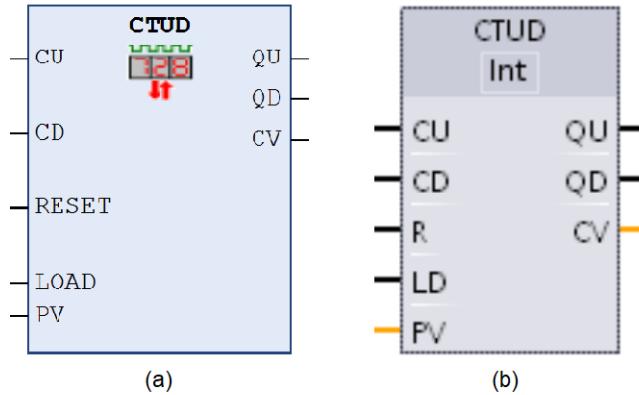
Fonte: Adaptado do Help do Codesys (2022)

2.1.3.2 Contadores

O acionamento de uma saída ou atuador pode estar condicionado ao número de repetições de uma determinada variável. Para diversas aplicações, um contador pode ser incrementado ou decrementado. Assim, existem contadores crescentes ou ascendentes chamados de CTU e contadores decrescentes ou descendentes chamados de CTD. Existe, ainda, um bloco especial - Instrução CTUD - que executa ambas as funções (incrementar e decrementar) e está representado na Figura 12.

Para este trabalho foi utilizada a instrução CTUD, onde o contador é incrementado a cada vez que um carro entra e decrementado sempre que um carro sai. O valor CV é incrementado sempre que a entrada CU é acionada e decrementado a cada atuação da entrada CD. Quando CV=PV, a saída QU é acionada. A entrada RESET (da Figura 12(a)) ou R (da Figura 12(b)) sempre que ficar ativa zera a contagem de CV, enquanto LOAD (da Figura 12(a)) / LD (da Figura 12(b)) força a igualdade de valores entre CV e PV.

Figura 12 – Representação de um contador CTUD no (a) Codesys e no (b) TIA Portal



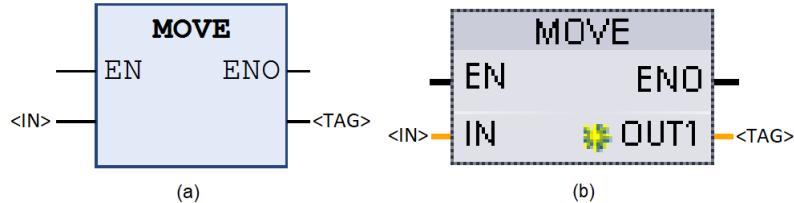
Fonte: Adaptado de Codesys e TIA Portal (2022)

2.1.3.3 Instrução MOVE

Outra funcionalidade importante na programação de CLPs é o uso da instrução Move. Sua função é movimentar um valor de memória/registro a outro, ou de um *tag* (etiqueta) a outro, dentro da programação. Uma ilustração de sua representação é mostrada na Figura 13. Quando a entrada (EN) é acionada, o valor de <IN> é movimentado para <TAG>, que pode ser uma memória ou uma saída física.

Esta instrução é frequentemente utilizada para movimentar valores lidos dos sensores de campo para memórias internas ou destas para ativação de alguma saída. Neste trabalho, por exemplo, a função é utilizada para movimentar um valor numérico para *tags* que representam o número total de vagas dos veículos de duas e quatro rodas. Este valor pode ser inserido diretamente via teclado ou até mesmo fazendo uso de um sistema supervisório.

Figura 13 – Representação da instrução MOVE no (a) Codesys e no (b) TIA Portal

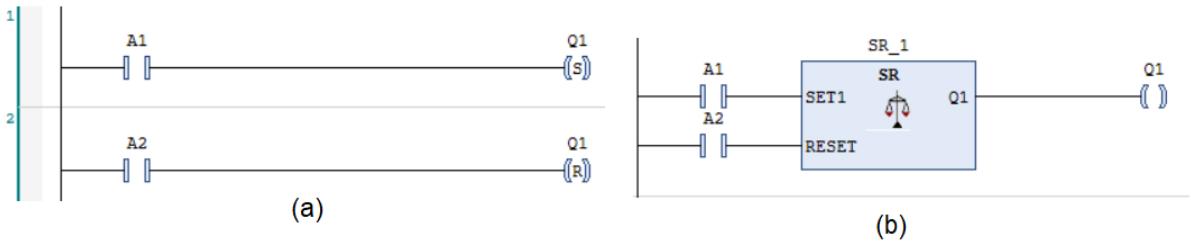


Fonte: Adaptado de Codesys e TIA Portal (2022)

2.1.3.4 Instruções SET/RESET

São instruções utilizadas para ativação e desativação de estados (memórias/saídas). Para ativação é usada a instrução SET. A representação destas instruções está ilustrada na figura 14. Assim, precisa-se apenas de um pulso no contato A1 para que o estado da saída Q1 (SET) fique em estado lógico alto e a saída Q1 só voltará a ficar desativada (ou estado lógico baixo) quando um pulso no contato A2 ativar o RESET.

Figura 14 – Representação de uma instrução SET/RESET (a) em degraus distinos e (b) em formato de bloco SR

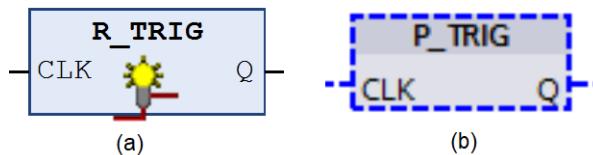


Fonte: Elaborado pelo autor (2022)

2.1.3.5 Instruções R_TRIG/F_TRIG ou P_TRIG/N_TRIG

As instruções detectoras de borda: R_TRIG(subida)/F_TRIG(descida) (Codesys) ou P_TRIG(subida)/N_TRIG(descida) (Tia Portal), ficam em estado ativo (estado lógico alto "1") quando detectam uma borda de subida ou descida. Estas instruções são úteis nos casos em que um pulso é suficiente para acionar uma saída. Na Figura 15 mostra-se a representação de detectores de borda de subida: quando há uma transição de "0" para "1" na entrada CLK, a saída Q vai para o estado lógico alto ("1").

Figura 15 – Representação de um detector de borda de subida: (a) R_TRIG no Codesys e (b) P_TRIG para o TIA Portal



Fonte: Adaptado de Codesys e TIA Portal (2022)

2.2 ESPECIFICAÇÃO DE SENsoRES E ATUADORES PARA SOLUÇÕES BASEADAS EM CLP

Um sistema baseado em CLP possui entradas analógicas ou digitais. Existe uma grande diversidade de sensores que podem ser utilizados em conjunto com estes controladores. Nível, pressão e vazão são exemplos de grandezas medidas. Entradas analógicas podem admitir valores de corrente de 4 a 20mA ou de tensão de 0 a 10V / 1 a 5V. Assim, os sensores transformam a grandeza medida (sinal de ultrassom, por exemplo) naquela que o CLP admite (corrente ou tensão).

No caso das entradas digitais, costuma-se utilizar tensão, sendo que 0V indica um estado lógico baixo ("0"), enquanto 24V indica um estado lógico alto ("1"). Para este caso, entradas digitais podem ser usadas em aplicações com sensores de presença, posição de uma válvula, pressostato ou uma chave de nível de um tanque, por exemplo.

Para as saídas digitais ou analógicas, o CLP realizará a conversão ou amplificação para ativação/desativação de atuadores. Entre os atuadores mais usados pode-se citar: bombas,

válvulas, cancelas de estacionamento simples/duplas, lâmpada, entre outros.

2.3 SIMULADORES PARA O CONTROLE DE CLP

Os CLPs são utilizados muitas das vezes em ambientes industriais, onde qualquer falha pode ocasionar acidentes com consequências graves. Por esta razão, é recomendável que os programas desenvolvidos sejam previamente testados em ambientes de simulação.

Mesmo para aplicações de pequeno porte, é importante realizar todos os testes possíveis antes de colocar o sistema em operação, pois, ainda que não haja risco de acidentes, podem ocorrer perdas financeiras.

Diante disto, vários fabricantes de CLPs disponibilizam softwares para simulação de seus controladores. Comumente estes softwares possuem diversos recursos, mas que são restritos aos dispositivos de cada fabricante e geralmente requerem a aquisição de licença. Já as versões gratuitas, costumam possuir recursos ou funcionalidades limitadas. Dentre os principais softwares podemos citar:

1. RSLogix Emulate (Rockwell)
2. S7 PLC-SIM (Siemens)
3. EcoStruxure™ Control Expert (Schneider)
4. Clic02 Edit (WEG)

Atualmente, com os avanços na virtualização e desenvolvimento de sistemas de simulação, muitos softwares estão sendo desenvolvidos e disponibilizados de forma online. Entre estes, pode-se citar: o PLC Simulator (<https://app.plcsimulator.online>) e o PLC Fiddle (<https://www.plcfiddle.com>). Uma das vantagens destes simuladores é a disponibilidade de forma gratuita e o fato de não serem dedicados a um fabricante específico, porém a desvantagem é a limitação em termos de funcionalidades.

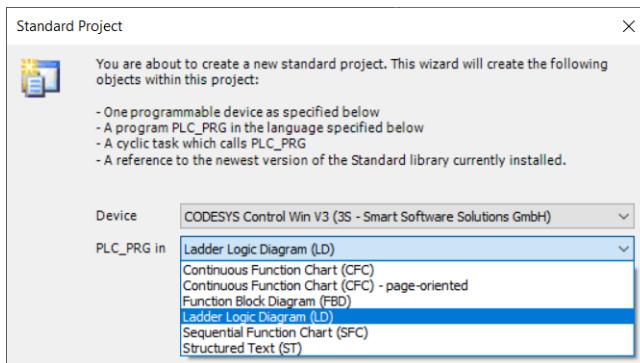
Um dos simuladores *offline* que se destaca por suas constantes atualizações e versatilidade é o software Codesys. Uma breve descrição deste software é feita a seguir.

2.3.1 O software de simulação CODESYS

O Codesys é uma plataforma de desenvolvimento de sistemas de automação que possui diversas funcionalidades para programação e simulação. É possível programar em diversas linguagens baseadas na norma IEC 61131-3, conforme mostra o próprio software (ver

Figura 16). A plataforma de programação é gratuita e possui diversos recursos gráficos para o desenvolvimento de testes. Existem extensões da plataforma que permitem uma fácil tradução para os controladores mais comuns existentes no mercado (softwares adicionais pagos).

Figura 16 – Linguagens habilitadas para programação no Codesys



Fonte: Codesys (2022)

Dadas as diversas vantagens, flexibilidade e recursos disponíveis, o Codesys foi o software escolhido para desenvolvimento e simulação do programa.

2.4 TRABALHOS CORRELATOS

Existem diversas soluções para automatizar o controle de acesso de veículos e a gestão das vagas de um estacionamento. Desde os sensores até os atuadores, passando pelo controlador e pela forma como se dará a comunicação entre eles, existe uma gama de parâmetros disponíveis para o projetista avaliar. É possível automatizar todo o processo – controle da entrada, saída, gestão do número de vagas e até sua localização – ou apenas parte dele, de acordo com a demanda e os recursos que estiverem à disposição.

Foi realizada uma revisão sistemática de trabalhos relacionados nas plataformas de artigos científicos IEEE Xplore e Semantic Scholar. A busca de referências nestas plataformas foi feita digitando as palavras-chave VEHICLE+PARKING+PLC e GARAGE+PLC. Desta forma, foi possível localizar as referências a seguir, que têm correlação com este trabalho.

No trabalho de Zhang, Mao e Wan (2011) é apresentada uma estrutura, procedimentos de montagem e o projeto do sistema elétrico de um sistema de garagem veicular de 6 camadas (andares) do tipo "estéreo". O trabalho faz uso de hardware e menciona ideias de design de garagem estéreo com elevação, permitindo assim estacionar mais carros, dada a limitação do espaço de grandes cidades. O autor explica o conceito de garagem do tipo estéreo, a qual inclui plataformas logísticas de elevação dos veículos, empilhamento destes, circulação vertical e circulação em formato circular. O trabalho apresenta uma maneira de projetar garagem do tipo estéreo com elevação fazendo uso de um CLP da

marca Omron modelo CP1H e CPU CP1HX40DR-A. O autor usou a programação clássica *Ladder*, desenvolvendo-a no software Omron CX-ONE 3.0.

No trabalho de Sheng-Wei et al. (2013) foi desenvolvido um sistema de controle de garagem inteligente fazendo uso de sensores infravermelhos, de temperatura e CLP Siemens S7-300 com o apoio de um painel de toque (*touch HMI*). Esta *HMI* coleta as informações de localização ao estacionar ou retirar um carro. Depois de ser calculado pelo CLP, um sinal de controle é transmitido para um relé que irá movimentar uma plataforma de estacionamento para o local designado, alcançando a automação do gerenciamento de entrada e saída do carro. Para este trabalho foi usado o software Siemens STEP7 e parte da programação é mostrada na linguagem SFC (Grafcet).

No trabalho de Ahamed et al. (2015) foi desenvolvido e projetado o processo de automação de um sistema de estacionamento automático baseado no CLP Siemens LOGO! 12/24 RC e lógica *Ladder*. Sensores infravermelhos (IR) foram instalados em portões de entrada e saída para detectar a presença de algum carro nestes locais. O CLP é capaz de realizar a contagem do número de veículos que entram e saem do estacionamento e, assim, o sistema desenvolvido pode monitorar o número de vagas disponíveis. Se não houver espaço disponível, o CLP enviará um sinal para o portão de entrada para mantê-lo fechado e acender a indicação “Estacionamento Cheio”. Se houver vaga, o portão de entrada será aberto para permitir que o carro entre. Da mesma forma, no momento da saída, o CLP enviará sinal ao portão de saída para abrir e permitir que o carro saia do estacionamento após o pagamento da tarifa devida. O sistema foi testado em escala menor através de uma maquete.

No trabalho de Aphiratsakun, Duy e Kongthon (2018) é apresentado um sistema Automático de Modelagem de Estacionamento (chamado de AU-CPS). O AU-CPS é um sistema de estacionamento automático controlado por CLP, que interage com o usuário via *HMI* e é acionado por três motores CC, fazendo uso de sensores fotoelétricos. Um guindaste é utilizado para levantar e levar o carro do local inicial para a vaga e desta para a saída. O sistema indica se o estacionamento está lotado ou não (acende um LED), e usa sensores para detectar o carro na vaga com o intuito de evitar sobreposição de carros numa mesma vaga.

No trabalho de Dong e Wei (2019) é abordada a construção de uma garagem tridimensional como solução para um problema de disponibilidade de vagas. A solução contou com um CLP Siemens S7-200 e uma série de dispositivos de proteção, entre eles dispositivos anti-queda, limit switch, dispositivos de detecção fotoelétrica, etc. O trabalho apresenta ilustrações do desenho desta garagem tridimensional, a listagem das entradas e saídas como variáveis e programação em *ladder*, explicando detalhes do funcionamento.

No trabalho de Selvakumar et al. (2020) é proposto um sistema automático para controle de veículos em um estacionamento, registrando a placa e determinando uma vaga específica de acordo com a disponibilidade. O sistema calcula o valor a ser pago na saída conforme o tempo de permanência. A abertura dos portões de entrada e saída é realizada através de motores de passo. O sistema é composto por leitores RFID, um CLP Siemens S7-1200 e programação desenvolvida em linguagem *ladder*. Adicionalmente, este sistema conta também com sensores de fumaça (para detecção de incêndio) e outros que visam à economia de energia, acendendo as lâmpadas apenas enquanto os carros percorrem o caminho.

Na tabela 3 é apresentada uma comparação entre este trabalho e as referências descritas.

Tabela 3 – Tabela comparativa de trabalhos correlatos entre os anos 2011-2022.

Referências	CLP utilizado	Programação	Sensores	Atuadores
Zhang, Mao e Wan (2011)	Omron CP1H	<i>Ladder</i>	Sensores de posição de gancho, sensores fotoelétricos, limit switch	Motores
Sheng-Wei et al. (2013)	Siemens S7-300	SFC/Grafcet (STEP7)	Sensores infravermelhos e de temperatura	Motor
Ahamed et al. (2015)	Siemens LOGO! 12/24 RC	<i>Ladder</i>	Sensores infravermelhos	Motor, Limit Switch e display LED
Aphiratsakun, Duy e Kongthon (2018)	ABB PM564-T-ETH	SFC	4 sensores de posição, 4 sensores (fotoelétricos), 4 switchs	3 motores
Dong e Wei (2019)	Siemens S7-200	<i>Ladder</i>	Sensor fotoelétrico, limit switch	Motor
Selvakumar et al. (2020)	Siemens S7-1200	<i>Ladder</i>	Sensores infravermelho, RFID	Lâmpadas LED e motores de passo
Este trabalho (2022)	Siemens S7-1200	<i>Ladder</i>	Sensores de laço induutivo, sensores de fim de curso, sensor de verificação de ticket impresso e sensores antiestmagamento (infravermelho)	Sinalizadores visuais, motores, impressora de ticket

Fonte: Elaborado pelo autor (2022)

3 MATERIAIS E MÉTODOS

Neste capítulo serão apresentados os materiais e a metodologia utilizados no desenvolvimento de uma solução para o problema proposto.

3.1 O PROBLEMA

A gestão da ocupação das vagas de um estacionamento e a identificação automatizada do tipo de veículo para tarifação são problemas reais encontrados nas grandes cidades e que geram grandes perdas econômicas e de tempo. Espera-se que à medida que a quantidade de veículos em circulação aumente, cresçam também os problemas relacionados ao seu uso, dentre os quais destaca-se o tempo gasto para localizar uma vaga de estacionamento, independente do tipo de vaga. Uma forma de mitigar estes problemas está relacionada à otimização da gestão dos estacionamentos, automatizando a contagem das vagas ocupadas, verificação das vagas disponíveis e identificando previamente o tipo de veículo que acessa. Uma solução para alcançar estes objetivos foi desenvolvida num sistema baseado em Controlador Lógico Programável e cuja metodologia é descrita na subseção seguinte.

3.2 METODOLOGIA ADOTADA NA SOLUÇÃO AO PROBLEMA

Para efeitos de exemplificação, iremos considerar uma solução para um estacionamento com 23 vagas, onde os usuários são clientes que possuem veículos de duas e/ou de quatro rodas. Será desenvolvido um programa na linguagem *ladder*, que é a linguagem mais comum e disponível para programação na maioria dos modelos de CLP. Este programa deverá gerenciar o acesso ao estacionamento, abrindo e fechando as cancelas conforme o tipo de veículo e controlando a quantidade de vagas.

Serão definidas e utilizadas variáveis ou *tags* ao longo do texto. Na elaboração do programa, algumas premissas devem ser atendidas, como simplicidade na implementação e flexibilidade. O programa deverá garantir o ajuste do número de vagas com facilidade e o tempo que as cancelas devem permanecer abertas. Isto é importante pois permite uma fácil adequação em casos de redimensionamento do espaço e/ou restrição do uso do número de vagas em casos de manutenção ou por qualquer outro motivo. Para este desenvolvimento, foi considerado um estacionamento cuja entrada possui duas cancelas (CE e CD) motorizadas.

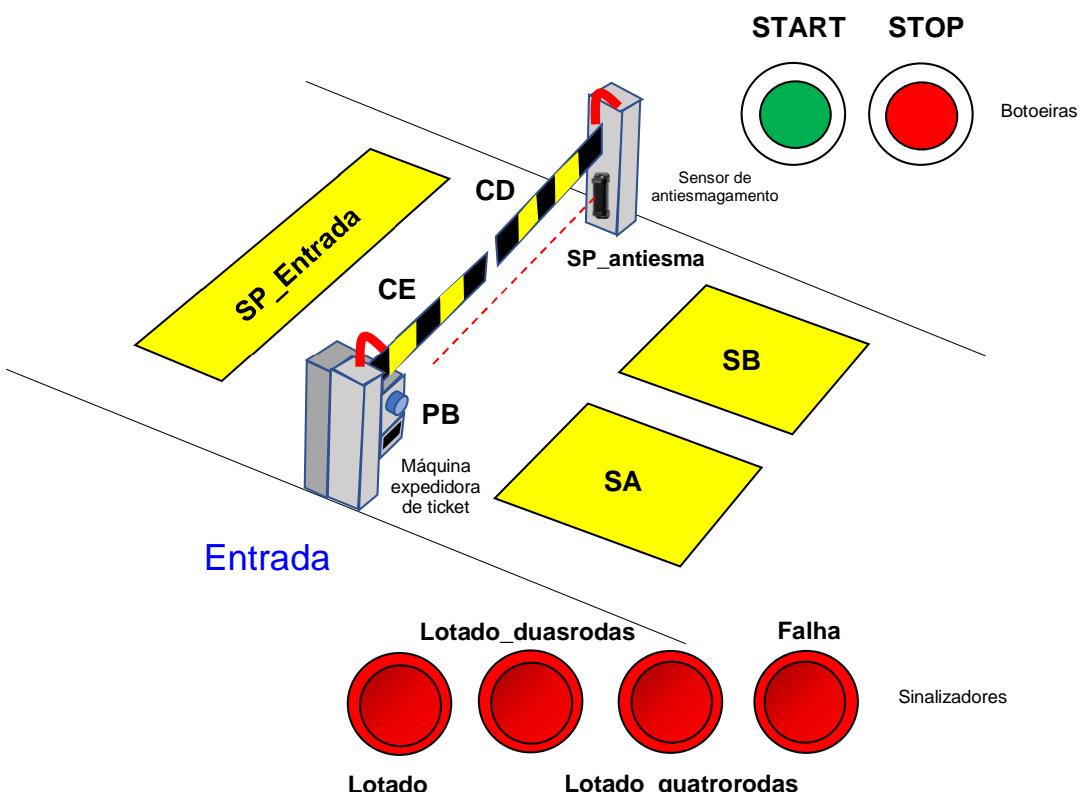
O sistema é constituído de diversos sensores e atuadores. A rotina do CLP inicia-se ao pressionar a botoeira de ativação (*tag START*) do sistema. Uma vez que o usuário realiza a solicitação do ticket através da ativação da botoeira de *tag PB*, será ativada a máquina expedidora de ticket (impressora do ticket), que irá registrar o tipo de veículo, a hora e a data de ingresso. A verificação da impressão do ticket será feita através do sensor de *tag*

Verifica_impressao. A rotina então prossegue com a abertura e fechamento das cancelas.

Este controlador deve registrar a quantidade de veículos que entram e saem do estacionamento, mantendo sempre atualizada a quantidade de vagas disponíveis e sinalizando nos casos de lotação de veículos de duas ou de quatro rodas, lotação máxima e em caso de falha no sistema. Os sensores que identificam o tipo de veículo na entrada do estacionamento foram nomeados com os *tags* SA e SB. Caso seja ativado unicamente o sensor SA, o sistema identificará que se trata de um veículo de duas rodas (moto), já no caso da ativação dos sensores SA e SB simultaneamente, o sistema identificará que se trata de um veículo de quatro rodas (carro). O sensor SP_Entrada realiza a contagem incremental da entrada do veículo e os sensores SP_Saida_A e/ou SP_Saida_B realizam o decremento da contagem na saída por tipo de veículo.

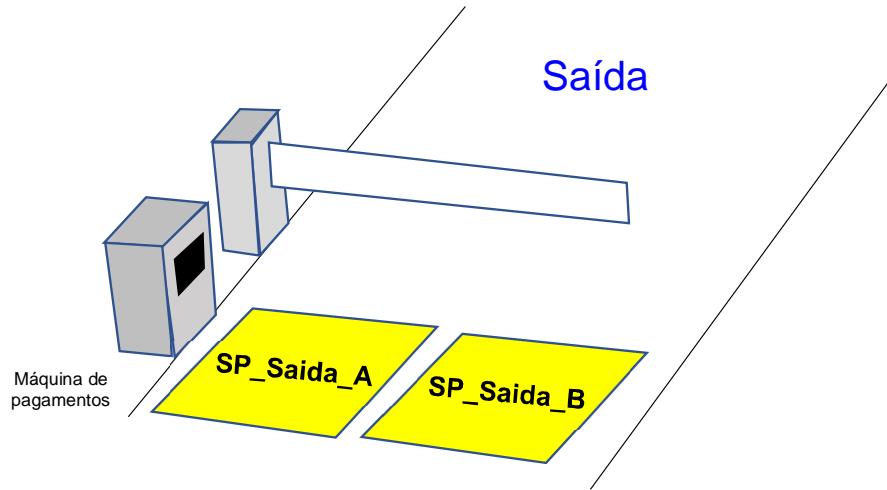
O sistema prevê a ativação de sinalizadores para os casos de lotação máxima de veículos de duas rodas (*tag* Lotado_duasrodas), de quatro rodas (*tag* Lotado_quatrorodas) e de todo o estacionamento (*tag* Lotado), além de falha no sistema (*tag* Falha). A Figura 17 ilustra os elementos presentes na entrada do estacionamento e a Figura 18, os da saída.

Figura 17 – Diagrama dos elementos presentes na entrada do estacionamento indicando as variáveis (*tags*) utilizadas



Fonte: Elaborado pelo autor (2022).

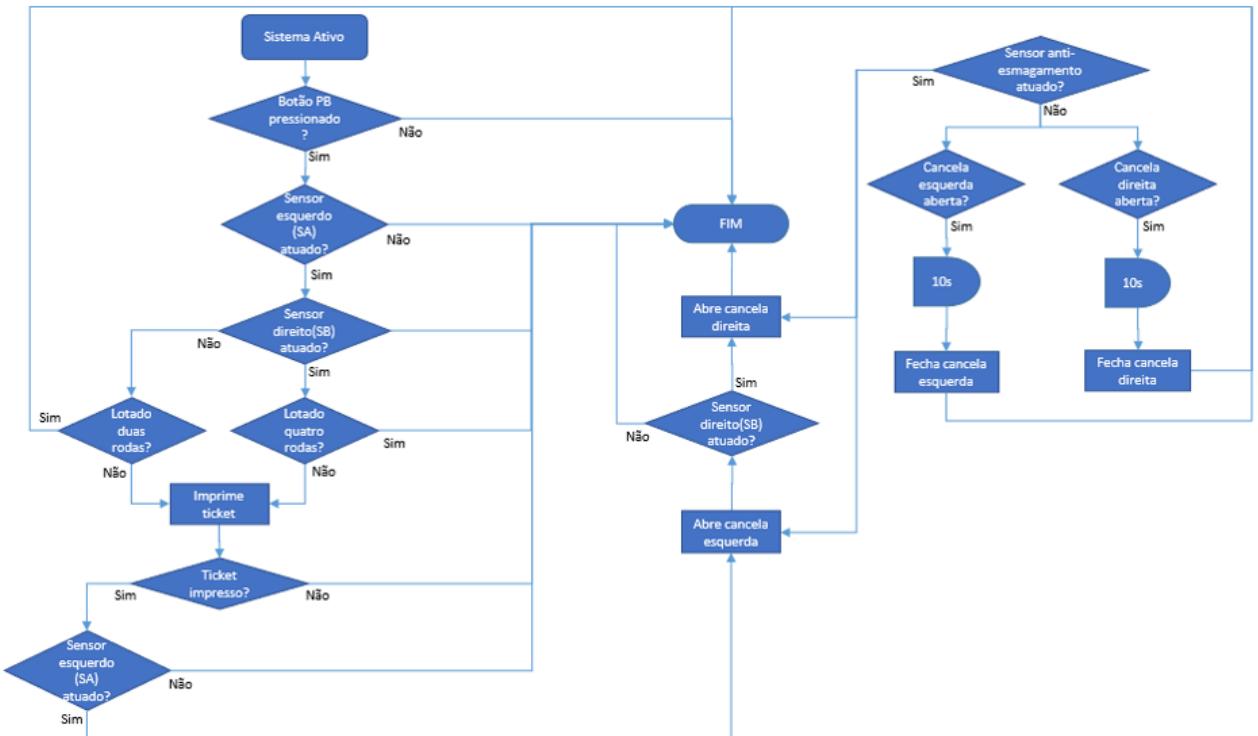
Figura 18 – Diagrama dos elementos presentes na saída do estacionamento indicando as variáveis (*tags*) utilizadas



Fonte: Elaborado pelo autor (2022).

As cancelas CE e CD são motorizadas e abrem conforme o tipo de veículo. Serão abertas em simultâneo a CE e CD para veículos de quatro rodas e apenas a cancela CE para veículos de duas rodas. Conforme a metodologia adotada, a Figura 19 mostra um diagrama de fluxo geral do sistema.

Figura 19 – Fluxograma Geral do Sistema

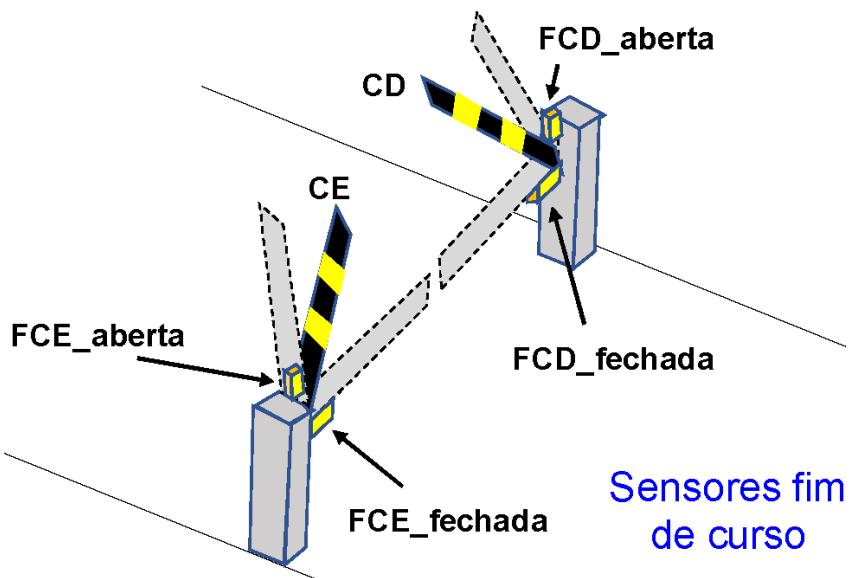


Fonte: Elaborado pelo autor (2022).

De acordo com a Figura 19, o ticket só será emitido, após a solicitação, se houver vaga disponível. Uma funcionalidade interessante deste desenvolvimento é a capacidade de identificar falhas, que podem ser de tipo mecânico ou operacional: no funcionamento dos

motores das cancelas e/ou na demora da impressão dos tickets. Para o caso das cancelas CE e CD, foram instalados 4 sensores fim de curso com os seguintes *tags*: FCE_aberta e FCE_fechada para a cancela CE e FCD_aberta e FCD_fechada para a cancela CD. Observe estes detalhes na Figura 20.

Figura 20 – Posição dos sensores fim de cursos (indicando os respectivos *tags*) nas cancelas CE e CD

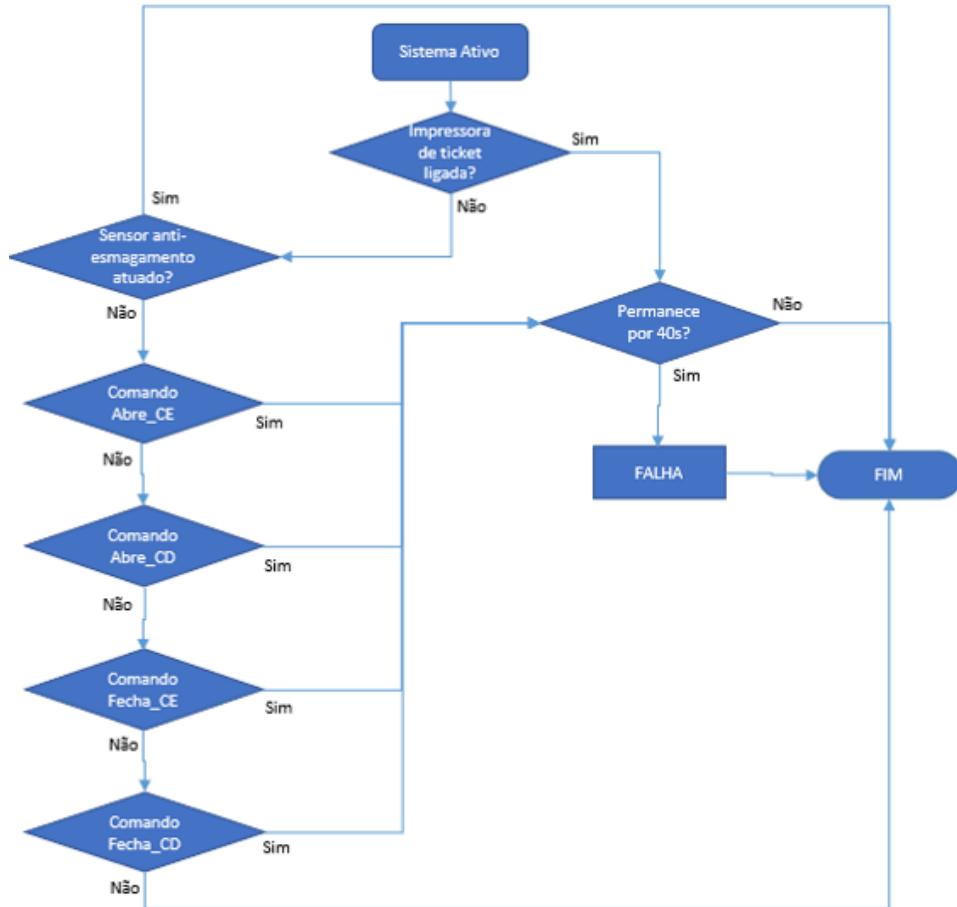


Fonte: Elaborado pelo autor (2022).

O sistema contará também com uma botoeira de emergência/parada (*tag* STOP), que irá deter o funcionamento dos motores e reiniciar a contagem das vagas. Um sensor anti-esmagamento (*tag* SP_antiesma) evitará qualquer fechamento indevido caso haja algum obstáculo na linha de detecção do sensor. A Figura 21 mostra um diagrama de fluxo de detecção de falhas.

Mais detalhes e referências em relação aos *tags* indicados serão mostrados em tabelas no capítulo seguinte.

Figura 21 – Fluxograma de detecção de falhas



Fonte: Elaborado pelo autor (2022).

3.3 MATERIAIS UTILIZADOS

Para este trabalho foram usados diversos materiais, entre hardware e software. Segue uma descrição de cada um deles:

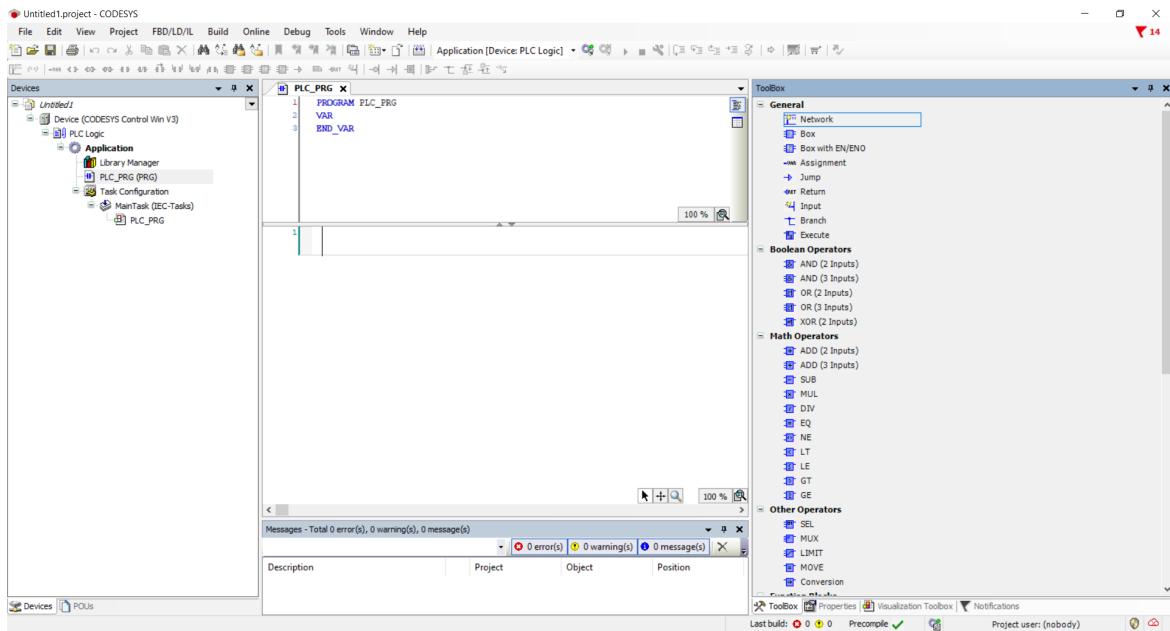
3.3.1 Software de Simulação CODESYS V3.5 SP17

A ferramenta computacional de simulação utilizada neste trabalho foi o CODESYS V3.5 SP17. Trata-se de uma plataforma que oferece programação e simulação de diversas linguagens com a possibilidade de integração a CLPs comerciais, tendo um espaço de trabalho completo e intuitivo. Na Figura 22 é mostrado o espaço de trabalho para programação em *Ladder*.

3.3.2 CLP Siemens S7-1200 e Módulo SIM 1274

Para o presente trabalho iremos realizar testes reais de funcionamento num CLP comercial, modelo S7-1200, da fabricante Siemens (ver Figura 23). Este CLP é um controlador compacto, com entradas e saídas integradas, que possui diversas funcionalidades técnicas

Figura 22 – Espaço de trabalho para programação em *Ladder* no software Codesys V3.5 SP17



Fonte: Elaborado pelo autor (2022).

que atendem à solução desta automação. Após realizar as simulações no software Codesys, o programa será traduzido e adaptado a este CLP para realização de testes em bancada.

Figura 23 – Controlador Lógico Programável modelo S7-1200 da fabricante Siemens



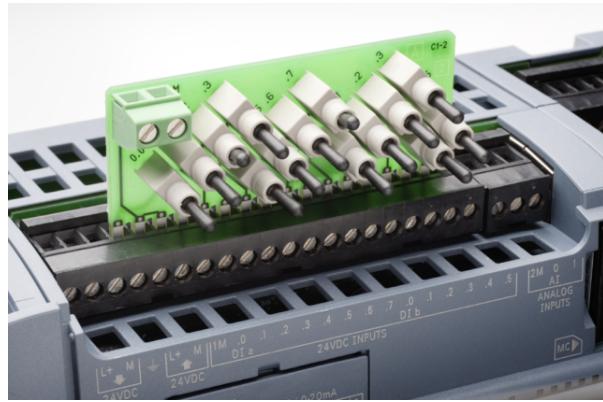
Fonte: Extraído de: <https://new.siemens.com/br/pt/produtos/automacao/controladores/s7-1200.html> (2022).

A CPU deste CLP é a S7-1214C DC/DC/DC, conta com 14 entradas digitais, 2 entradas analógicas, 10 saídas digitais a transistor e funcionamento a 24V DC. A programação deste CLP é através do software TIA Portal (utilizaremos a versão 13) com o uso de um cabo de rede Ethernet para download do programa.

Para testes neste CLP será utilizado o módulo SIM 1274 (ver Figura 24). Trata-se de

uma placa eletrônica que simula estados lógicos nas entradas digitais através de chaves de alavanca permitindo a mudança de estados nas entradas. O acionamento das saídas pode ser verificado no próprio CLP através de LEDs.

Figura 24 – Placa simuladora de estados lógicos das entradas modelo SIM 1274



Fonte: Extraído de: <https://new.siemens.com/br/pt/produtos/automacao/controladores/s7-1200.html> (2022).

3.4 SENsores INDICADOS PARA IMPLEMENTAÇÃO REAL

3.4.1 Sensor de laço indutivo para detecção de veículos

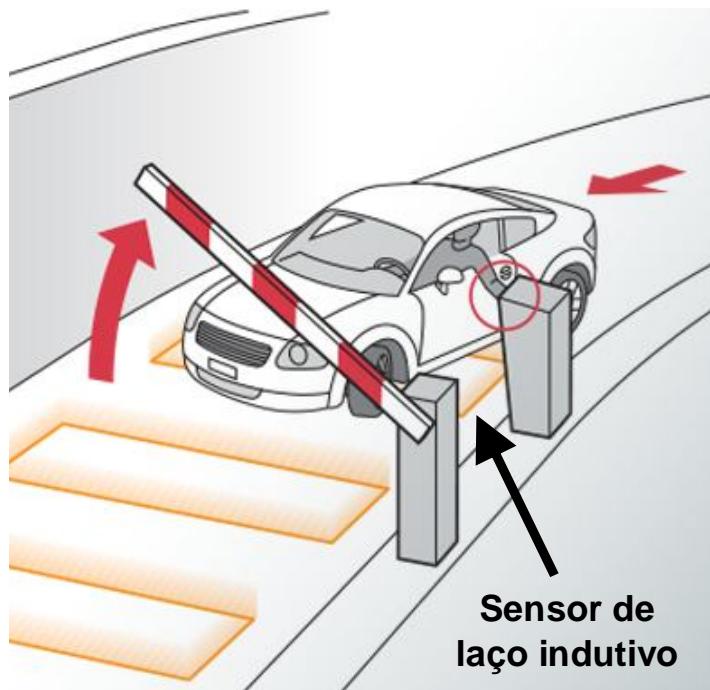
Para identificação da presença dos veículos na entrada do estacionamento, uma sugestão interessante e robusta é a utilização de sensores de laço indutivo, conforme ilustrado na Figura 25. Estes sensores devem ser instalados no solo, um ao lado esquerdo e outro ao lado direito na entrada, antes das cancelas, e ao serem ativados permitirão distinguir o tipo de veículo e, assim, determinar quais cancelas (CE e/ou CD) devem ser abertas.

Outro sensor do mesmo tipo pode ser instalado após as cancelas para confirmar a passagem do veículo e atualizar a contagem das vagas, além de acionar o fechamento das cancelas. Da mesma forma, dois sensores com estas características podem ser instalados na saída (*tags*: SP_Saida_A e SP_Saida_B), com intuito de decrementar a contagem do número de vagas ocupadas.

3.4.2 Sensor Fim de Curso para confirmação de posição das cancelas

A posição das cancelas precisa ser verificada constantemente para garantir o correto funcionamento do sistema. Essa verificação pode ser feita por meio de sensores fim de curso, tanto na abertura como no fechamento de cada cancela (ver Figura 20). A maioria das cancelas eletrônicas já possuem esta funcionalidade e um sensor indutivo seria uma sugestão adequada para esta aplicação considerando cancelas de hastes metálicas.

Figura 25 – Localização dos sensores de laço indutivo



Fonte: Extraído e adaptado de: <https://fortop.co.uk/solutions/products/loop-detectors-detection-loops/> (2022).

3.4.3 Sensor Anti-esmagamento

O fechamento das cancelas só pode ocorrer após a passagem completa do veículo, de modo a evitar acidentes. Para este e outros casos em que possa haver obstrução abaixo das cancelas, será utilizado um sensor anti-esmagamento. Neste trabalho, este sensor foi identificado com o *tag SP_antiesma* e é basicamente um sensor infravermelho instalado na direção das cancelas, abaixo das hastas. Caso este sensor fique ativo (atuado), isto é, se houver algum obstáculo na direção da cancela, o fechamento não será permitido. Um sensor deste tipo é apresentado no exemplo da Figura 26.

Figura 26 – Sensor de anti-esmagamento da fabricante DECIBEL modelo D60-1



Fonte: Extraído de: <https://www.decibel.com.br/index.php?route=product/product&path=2_24&product_id=4(2022).>

3.5 ATUADORES E SINALIZADORES INDICADOS PARA IMPLEMENTAÇÃO REAL

3.5.1 Cancela eletrônica

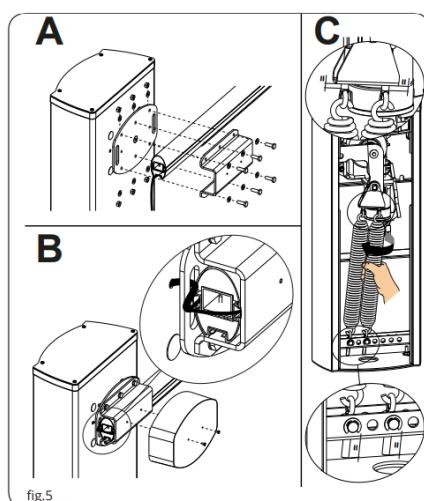
O controle físico do acesso ao estacionamento será feito por duas cancelas (CE e CD) instaladas lado a lado, que serão abertas conforme o tipo de veículo que solicitar a entrada. Para este trabalho, consideramos quatro comandos para movimentação das cancelas, os quais foram identificados com os seguintes *tags*: Abre_CE, Abre_CD, Fecha_CE, Fecha_CD. Para um carro ou caminhonete, ambas serão abertas; no caso de moto, apenas a do lado esquerdo. Um exemplo de cancela eletrônica da fabricante TAU é mostrada na Figura 27. Cancelas comerciais são compostas de uma série de componentes, como motores, peças metálicas e molas. Observe que alguns modelos de cancela eletrônica possuem um complexo sistema de molas no seu interior (ver Figura 28), por tal motivo a manutenção é importante para evitar falhas.

Figura 27 – Exemplo de cancela eletrônica da fabricante TAU de 7 metros de comprimento e motores de torque de 300 Nm



Fonte: Extraída e adaptada de: <https://tauitalia.com/pt/produtos/automacao-para-barreiras-automaticas/> (2022).

Figura 28 – Alguns modelos de cancela eletrônica possuem um complexo sistema de molas no seu interior



Fonte: Extraída de: <https://tauitalia.com/wp-content/uploads/wpallimport/files/D_MNL0LUXE.pdf> (2022).

3.5.2 Máquina Expedidora de Ticket

A impressão de um ticket contendo o registro de ingresso é realizada através de uma máquina expedidora de tickets. A abertura das cancelas é realizada após a impressão do ticket. Atualmente existem no mercado diversos modelos deste dispositivo e com diversas tecnologias integradas, como: câmeras CCTV IP integrada, comunicação por interfone, leitores de códigos QR e sistemas de reconhecimento com Inteligência Artificial. Dois modelos comerciais são mostrados na Figura 29.

Figura 29 – Modelos de máquina expedidora de ticket da fabricante Amano



Fonte: Extraído de: <https://www.amano.eu/en/parking/solutions/field-devices/> (2022).

3.5.3 Sinalizadores visuais de Lotação e Falha

Nos casos do sistema apresentar falha ou lotação máxima atingida, diversos sinalizadores luminosos estarão localizados num painel de controle numa sala de monitoramento e também na entrada do estacionamento, de modo a sinalizar a quem deseje entrar caso não seja possível.

Segundo as normas IEC 73 e VDE 0199, as sinaleiras possuem cores específicas para utilização (SENAI, 2009). A cor vermelho possui a seguinte referência: condições anormais, perigo ou alarme. Por esta razão, sugere-se a utilização de sinaleiros da cor vermelha. Um exemplo deste sinaleiro é mostrado na Figura 30.

Figura 30 – Sinaleiro vermelho do fabricante Metaltex modelo L20-AR9-R



Fonte: Extraído de: <https://www.metaltex.com.br/produtos/automacao/sinalizadores/l20-ar-sinalizador-led-22mm-com-protectao-ip65> (2021).

4 EXPERIMENTOS, RESULTADOS E DISCUSSÃO

Neste capítulo serão apresentados e discutidos os experimentos e os resultados obtidos após a aplicação da metodologia apresentada no capítulo anterior.

4.1 A PROGRAMAÇÃO

Nesta seção será apresentado o programa que automatiza o controle de entrada e saída de veículos no estacionamento.

No desenvolvimento foram utilizadas entradas e saídas digitais, bem como memórias internas e funções do CLP, conforme tabelas 4, 5, 6 e 7. Os tipos das variáveis utilizadas são os seguintes:

- BOOL: valor binário, que pode ser verdadeiro (=1) ou falso (=0);
- CTU: Contador que é incrementado a cada vez que a linha é verdadeira;
- TON: Timer que incrementa o tempo enquanto algumas condições são atendidas;
- TIME: Registro de tempo;
- R_TRIG: Pulso que é acionado no degrau de subida;
- WORD: Memória de 16 bits;
- UINT: Memória de 16 bits para armazenar valor numérico inteiro positivo.

Tabela 4 – *Tags* das Saídas Digitais

Tag	Tipo	Descrição
Maquina_ticket	BOOL	Ativação da impressora do ticket
Abre_CE	BOOL	Motor girando e abrindo a cancela esquerda
Abre_CD	BOOL	Motor girando e abrindo a cancela direita
Fecha_CE	BOOL	Motor girando e fechando a cancela esquerda
Fecha_CD	BOOL	Motor girando e fechando a cancela direita
Lotado	BOOL	Sinalizador de lotação máxima de veículos
Lotado_duasrodas	BOOL	Sinalizador de lotação máxima de veículos de duas rodas
Lotado_quatrorodadas	BOOL	Sinalizador de lotação máxima de veículos de quatro rodas
Falha	BOOL	Sinalizador visual que indica falha no sistema

Fonte: Elaborada pelo autor (2022)

Tabela 5 – Tags das Entradas Digitais

Tag	Tipo	Descrição
START	BOOL	Botoeira de inicialização do sistema
STOP	BOOL	Botoeira de parada/Emergência
PB	BOOL	Botoeira de solicitação de ticket na entrada
SA	BOOL	Sensor de laço indutivo esquerdo
SB	BOOL	Sensor de laço indutivo direito
SP_Estrada	BOOL	Sensor de laço indutivo após as cancelas da estrada
SP_Saida_A	BOOL	Sensor de laço indutivo na saída
SP_Saida_B	BOOL	Sensor de laço indutivo na saída
SP_antiesma	BOOL	Sensor antiesmagamento
FCE_aberta	BOOL	Sensor fim de curso da cancela esquerda aberta
FCE_fechada	BOOL	Sensor fim de curso da cancela esquerda fechada
FCD_aberta	BOOL	Sensor fim de curso da cancela direita aberta
FCD_fechada	BOOL	Sensor fim de curso da cancela direita fechada
Verifica_impressao	BOOL	Sensor de verificação do ticket expedido

Fonte: Elaborada pelo autor (2022)

Tabela 6 – Variáveis internas do CLP

Tag	Tipo	Descrição
Sistema_ativo	BOOL	Indica que o sistema está ativo
Ticket_impresso	BOOL	Confirmação do ticket impresso
valor_atual_1	TIME	Valor atual do temporizador
valor_atual_2	TIME	Valor atual do temporizador
valor_atual_3	TIME	Valor atual do temporizador
Reset_vagas	BOOL	Reinicia o número de vagas
vazio_1	BOOL	Indica que não ingressou veículo de duas rodas
vazio_2	BOOL	Indica que não ingressou veículo de quatro rodas
vagas_duasrodas	WORD	Número total de vagas de veículos de duas rodas
vagas_quatrorodas	WORD	Número total de vagas de veículos de quatro rodas
vagas_ocupadas	WORD	Indica o número total de vagas ocupadas
vagas_ocup_duasrodas	WORD	Vagas ocupadas por veículos de 2 rodas
vagas_ocup_quatrorodas	WORD	Vagas ocupadas por veículos de 4 rodas

Fonte: Elaborada pelo autor (2022)

Tabela 7 – Funções utilizadas

Tag	Tipo	Descrição
MOVE_0	MOVE	Número de carros que saíram
MOVE_1	MOVE	Número de carros que saíram
ADD	ADD	Bloco de adição
EQ	EQ	Bloco comparativo de igualdade
F_TRIG_0	F_TRIG	Detector de borda de descida
F_TRIG_1	F_TRIG	Detector de borda de descida
R_TRIG_0	R_TRIG	Detector de borda de subida
R_TRIG_1	R_TRIG	Detector de borda de subida
R_TRIG_2	R_TRIG	Detector de borda de subida
R_TRIG_3	R_TRIG	Detector de borda de subida
R_TRIG_4	R_TRIG	Detector de borda de subida
R_TRIG_5	R_TRIG	Detector de borda de subida
R_TRIG_6	R_TRIG	Detector de borda de subida
R_TRIG_7	R_TRIG	Detector de borda de subida
R_TRIG_8	R_TRIG	Detector de borda de subida
TON_0	TON	Temporizador com atraso na ativação
TON_1	TON	Temporizador com atraso na ativação
TON_2	TON	Temporizador com atraso na ativação
CTUD_0	CTUD	Contador Incremental/Decremental
CTUD_1	CTUD	Contador Incremental/Decremental

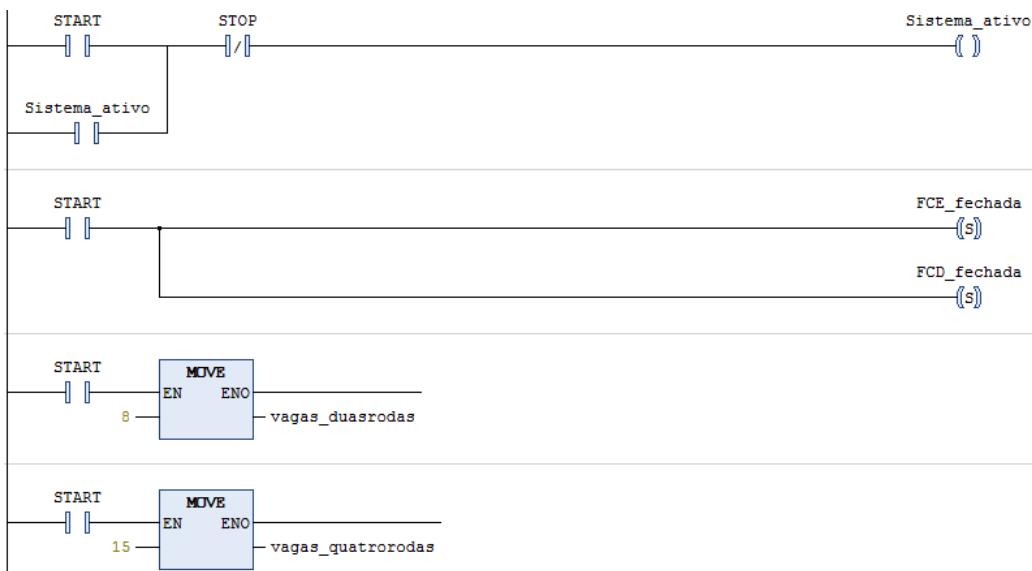
Fonte: Elaborada pelo autor (2022)

Da Figura 31 até a Figura 41 são mostradas as etapas do programa. A rotina é executada na mesma sequência em que são apresentadas as figuras.

As primeiras linhas (ver Figura 31) tratam da ativação do sistema e definição do número de vagas do estacionamento. Isto pode permitir a atualização do programa, caso exista mudança no dimensionamento do estacionamento, e a possibilidade de integração com um sistema supervisório, permitindo, por exemplo, restringir algumas vagas, caso seja necessário.

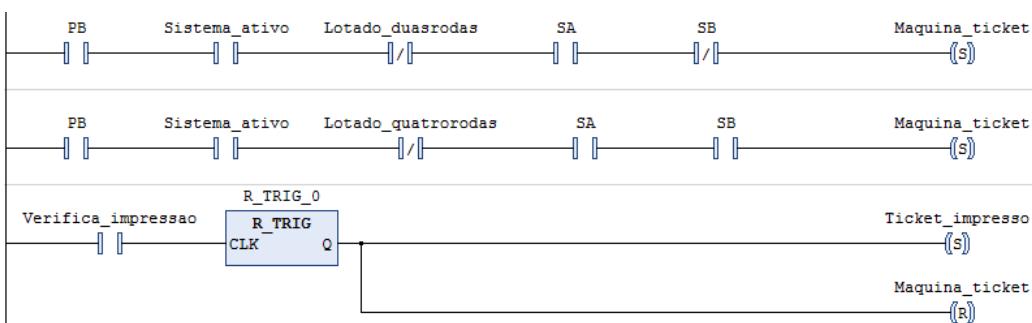
A seguir, tem-se a sequência de requisição do ticket e confirmação da impressão (ver Figura 32), de modo a permitir a abertura das cancelas. A máquina de impressão de Ticket (Maquina_ticket) será acionada desde que o usuário acione a botoeira PB e sempre que o estacionamento não esteja lotado e ao menos o sensor de presença do lado esquerdo (SA) seja atuado. Observe que o sistema aguarda a confirmação da impressão do ticket (contato Verifica_impressao) para desativar a impressora (Maquina_ticket). Adicionalmente, o contato Ticket_impresso habilita a abertura das cancelas (Figura 33).

Figura 31 – Inicialização do sistema.



Fonte: Elaborado pelo autor (2022).

Figura 32 – Solicitação de ticket e confirmação de impressão.

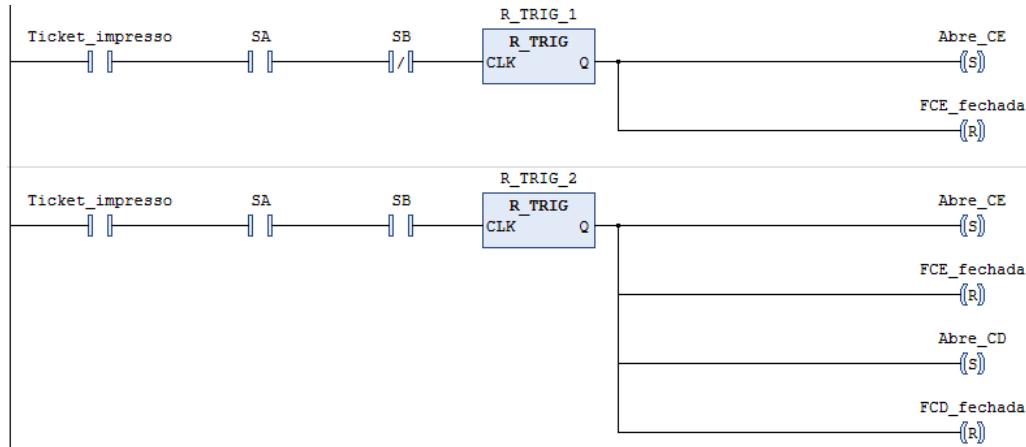


Fonte: Elaborado pelo autor (2022).

As linhas seguintes (ver Figura 33) são destinadas à abertura das cancelas. Cada cancela será aberta após a confirmação de emissão do ticket e ao instante que é iniciada a abertura das mesmas, os sensores de fim de curso que detectam cancelas fechadas são desativados (resetados). Cada cancela abrirá de acordo com a atuação dos sensores instalados no solo, SA para o lado esquerdo e SB para o lado direito.

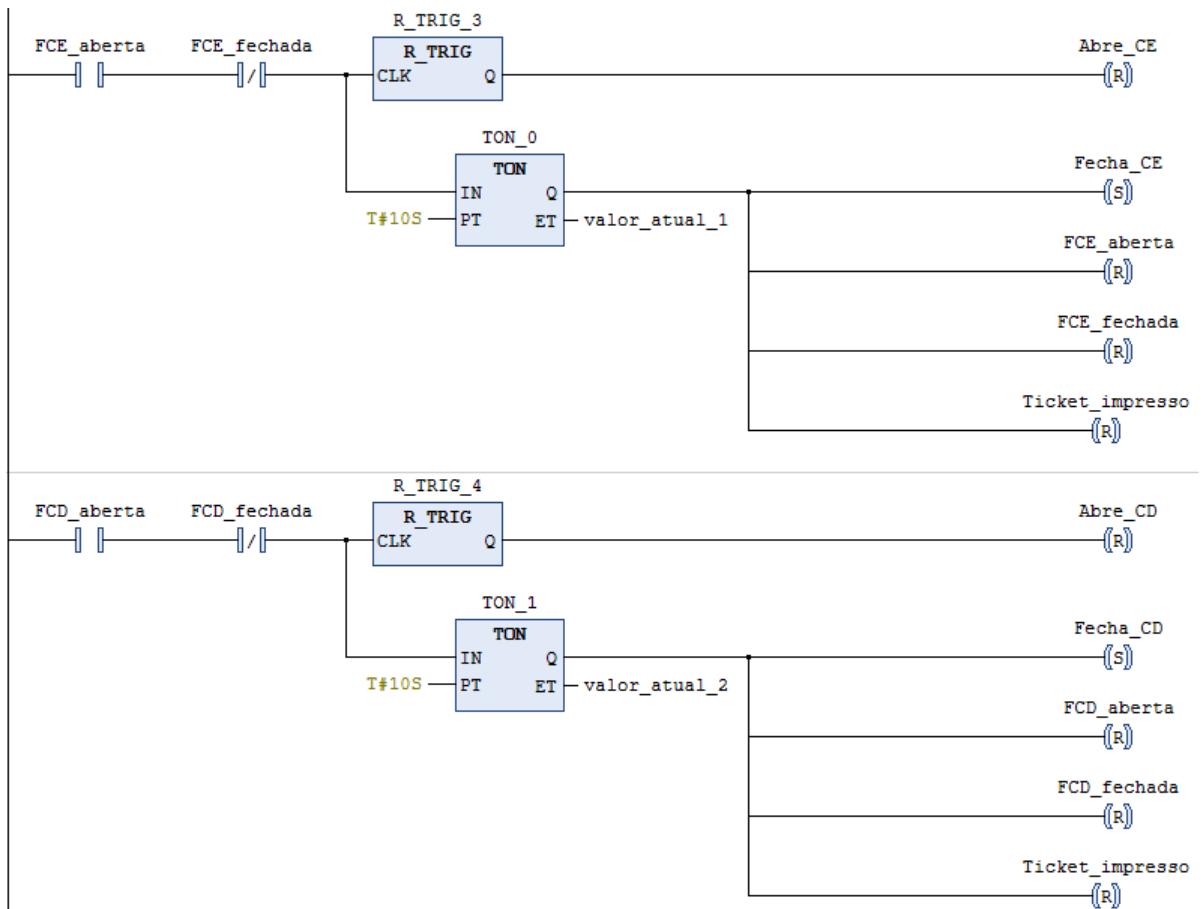
As linhas seguintes são dedicadas ao fechamento das cancelas (ver Figura 34), que só ocorre se os sensores de fim de curso estiverem nos estados previstos. Para iniciar o fechamento, o sensor de fim de curso de cancela aberta deve estar acionado e o de cancela fechada, não. Após a abertura total das cancelas (sensores de fim de curso atuados), um temporizador de 10 segundos é ativado para começar o processo de fechamento. Este parâmetro de 10 segundos é apenas um tempo referencial e pode ser modificado de acordo com o interesse do programador e o tipo de estacionamento.

Figura 33 – Abertura das cancelas.



Fonte: Elaborado pelo autor (2022).

Figura 34 – Fechamento das cancelas.



Fonte: Elaborado pelo autor (2022).

De acordo com a Figura 35, o veículo, ao passar pelo sensor SP_Estrada, desativa a variável (contato) Ticket_impresso. Esta variável impede que novos registros sejam feitos durante o processo de abertura e fechamento na entrada de um veículo no estacionamento.

Figura 35 – Desativação da variável Ticket_impresso.

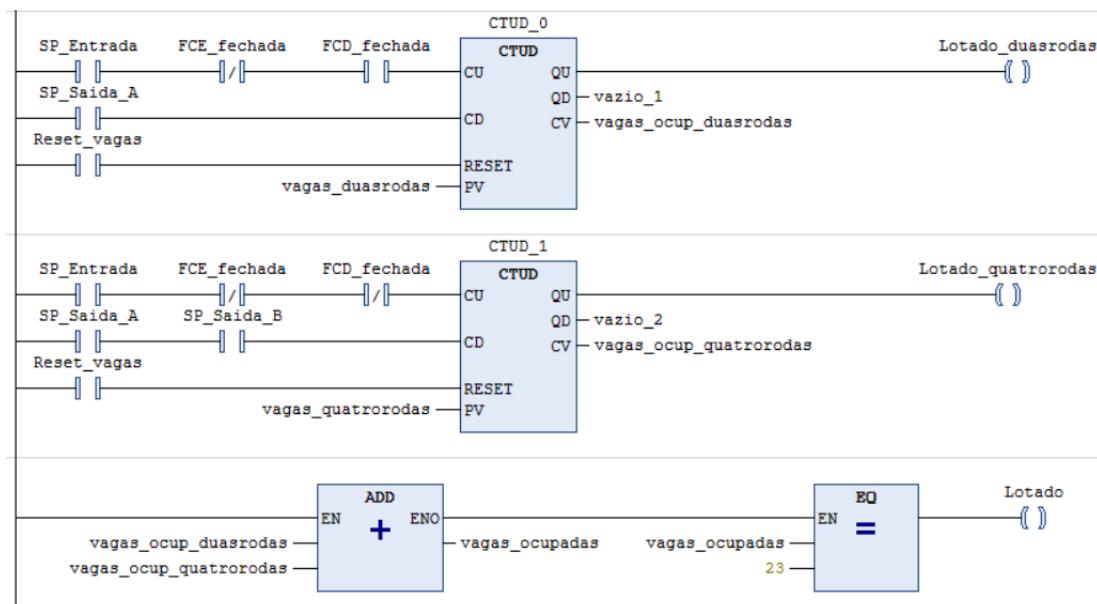


Fonte: Elaborado pelo autor (2022).

Na Figura 36, mostra-se os contadores do tipo CTUD, sendo o primeiro (CTUD_0) para os veículos de duas rodas e o segundo (CTUD_1) para os de quatro rodas. O incremento vai depender da ativação do sensor de entrada e que a cancela respectiva (esquerda ou direita) não esteja fechada. Assim, a contagem é diferenciada pelo tipo de veículo, já que apenas a cancela CE é aberta se o veículo é de duas rodas e ambas (CE e CD) são abertas para o caso dos veículos de quatro rodas.

Já o decremento vai depender do acionamento dos sensores de saída. Cada vez que um veículo retira-se do estacionamento, os contadores CTUD decrementam a contagem das vagas de forma independente com o apoio dos sensores SP_Saida_A e SP_Saida_B, que identificam o tipo de veículo. As saídas (QU) destes contadores estão representadas pelas bobinas Lotado_duasrodas e Lotado_quatrorodas e ficam estado lógico alto quando CV=PV, e, nesta condição, fica impedida a geração de novos tickets na entrada do estacionamento. Observe que foi adicionado um bloco lógico de adição (ADD) e um bloco lógico comparativo (EQ), que ao atingir 23 vagas (somação das vagas disponíveis para veículos de duas e quatro rodas), ativa a bobina denominada Lotado (capacidade máxima do estacionamento).

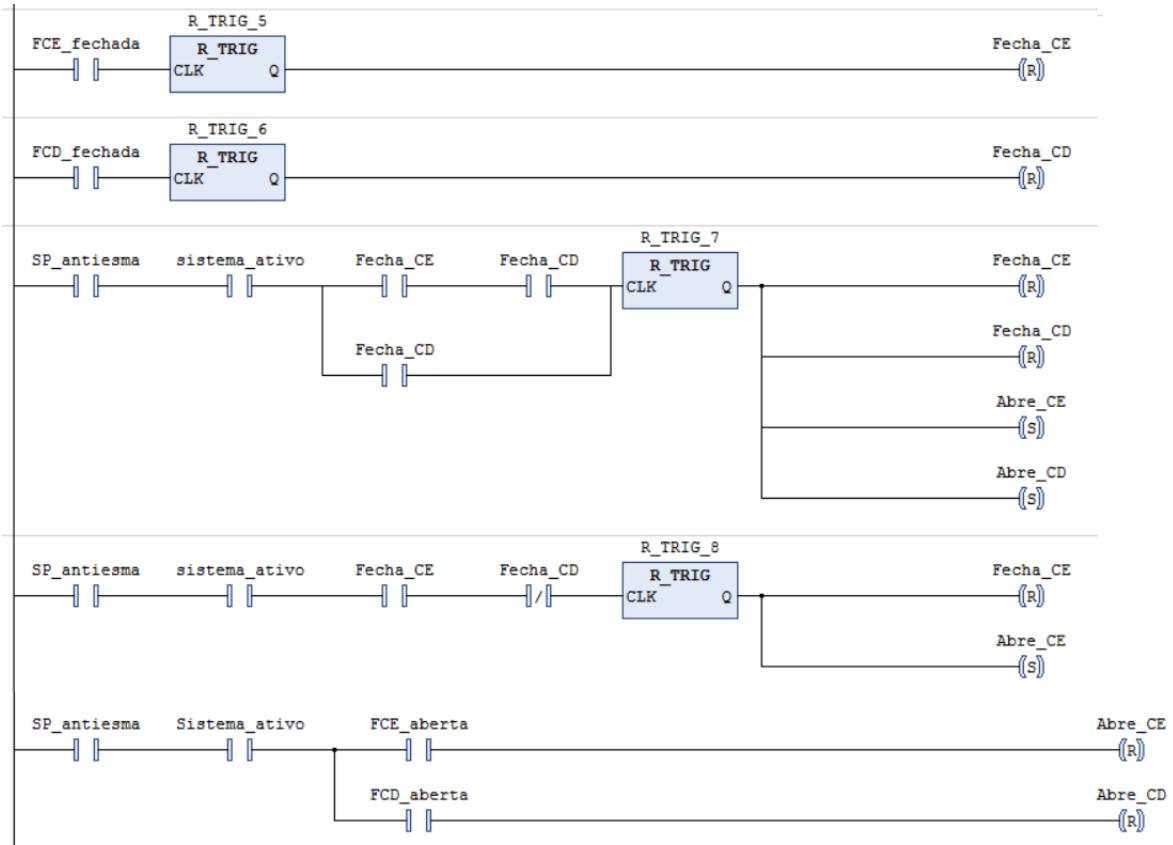
Figura 36 – Contagem de veículos e indicação de lotação



Fonte: Elaborado pelo autor (2022).

A Figura 37 mostra a ativação do sensor antiesmagamento, cuja função é evitar colisões com as cancelas, impedindo o fechamento das mesmas. A figura ilustrada também o mecanismo de abertura das cancelas quando o sensor antiesmagamento estiver ativo durante o processo rotineiro do fechamento das cancelas. Entende-se que a ativação do sensor antiesmagamento é um processo atípico e está relacionado a uma situação de alerta.

Figura 37 – Ativação do sensor antiesmagamento durante o fechamento das cancelas

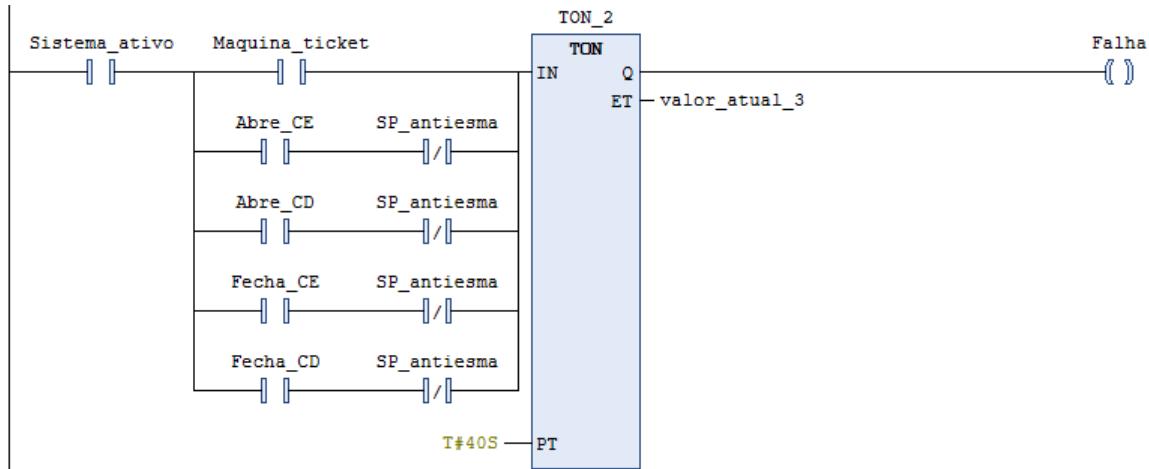


Fonte: Elaborado pelo autor (2022).

A Figura 38 mostra as condições de falha do sistema, as quais podem ocorrer pelos seguintes motivos: a) Caso o contato Maquina_ticket permaneça ativa por 40 segundos, indica não foi gerado ticket e pode estar relacionado a diversas causas; b) caso os motores das cancelas estejam ligados por mais de 40 segundos e o sensor antiesmagamento não esteja ativo.

Por fim, uma botoeira de emergência realiza o processo de parada do sistema. Desta forma, os motores são detidos, os contadores são reiniciados e diversas variáveis são reiniciadas/-resetadas (ver Figura 39).

Figura 38 – Indicação de falha



Fonte: Elaborado pelo autor (2022).

Figura 39 – Parada de Emergência e Reset de diversas variáveis



Fonte: Elaborado pelo autor (2022).

4.2 SIMULAÇÃO EM CODESYS

Após o desenvolvimento da programação, diversas situações foram testadas na simulação, com vistas a esgotar as possibilidades de ocorrências na aplicação real. O objetivo desta seção é mostrar as diversas simulações realizadas. Os primeiros testes mostraram o sistema operacionalmente funcional. Em seguida foram simuladas situações de falhas.

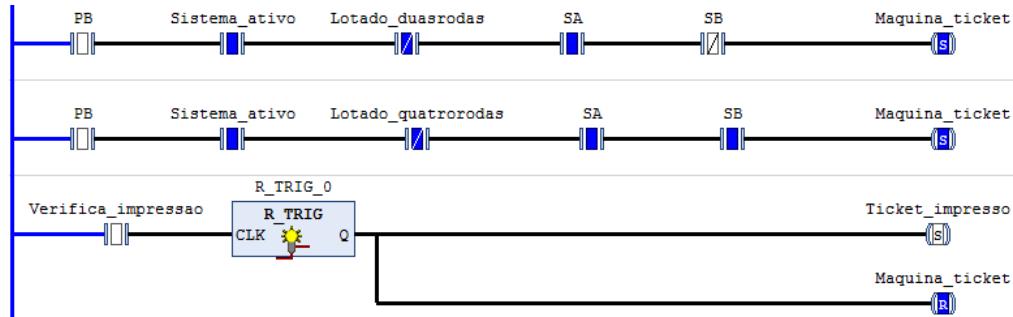
4.2.1 Simulações com o Sistema em funcionamento

4.2.1.1 Veículo se aproxima e há vaga disponível

Neste caso basta pressionar o botão PB e o comando de emissão do ticket será enviado para a impressora (ver Figura 40). Após a confirmação de expedição do ticket, ambas as

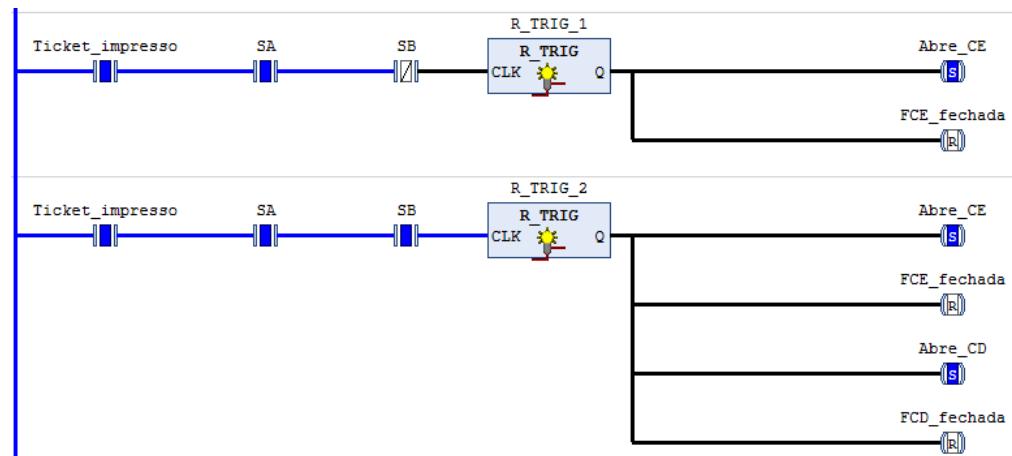
cancelas são abertas quando se tratar de veículo de quatro rodas (ver Figura 41). Se for um veículo de duas rodas, apenas a cancela esquerda será aberta (ver Figura 42).

Figura 40 – Simulação da impressão do ticket



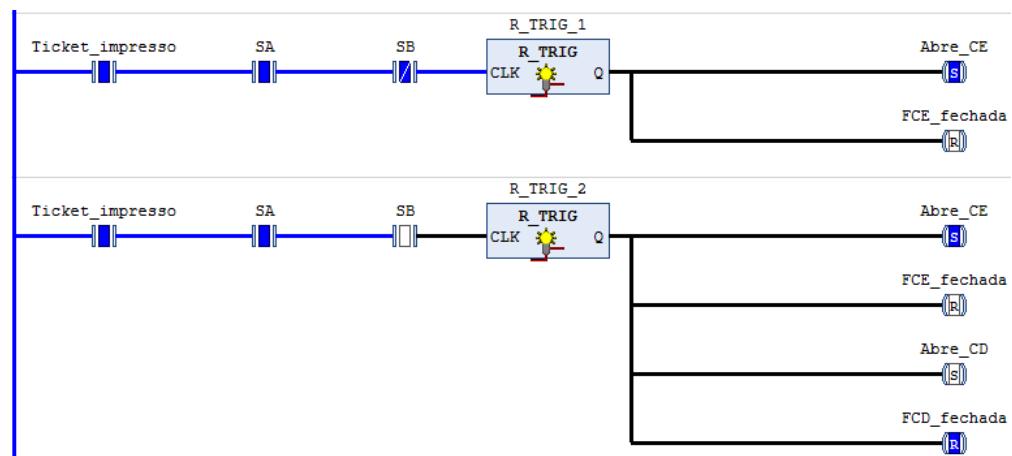
Fonte: Elaborado pelo autor (2022).

Figura 41 – Simulação da entrada de veículo de quatro rodas



Fonte: Elaborado pelo autor (2022).

Figura 42 – Simulação da entrada de veículo de duas rodas



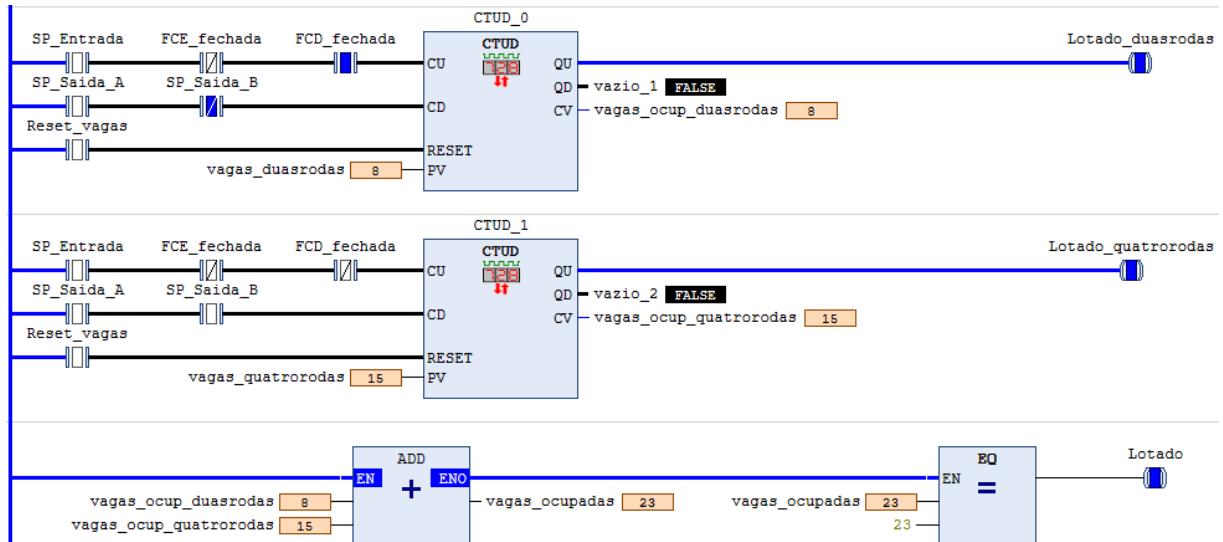
Fonte: Elaborado pelo autor (2022).

4.2.1.2 Veículo se aproxima e não há vaga disponível

Neste caso, um sinalizador luminoso alertará que o estacionamento está lotado. Nesta condição, ao tentar pressionar a botoeira PB, a impressora não deve emitir ticket e,

consequentemente, as cancelas não se abrirão. A Figura 43 ilustra o caso do estacionamento completamente lotado, isto é, não há vagas para veículos de duas ou quatro rodas. Entretanto, o sistema ao fazer distinção do tipo de veículo é capaz de ser específico em certos casos. Analise o seguinte exemplo: se as vagas para veículos de duas rodas estiverem totalmente lotadas e um veículo de quatro rodas se aproximar, existindo vagas para este tipo de veículo, o ticket deve ser emitido.

Figura 43 – Simulação do estacionamento totalmente lotado



Fonte: Elaborado pelo autor (2022).

4.2.1.3 Fechamento das cancelas

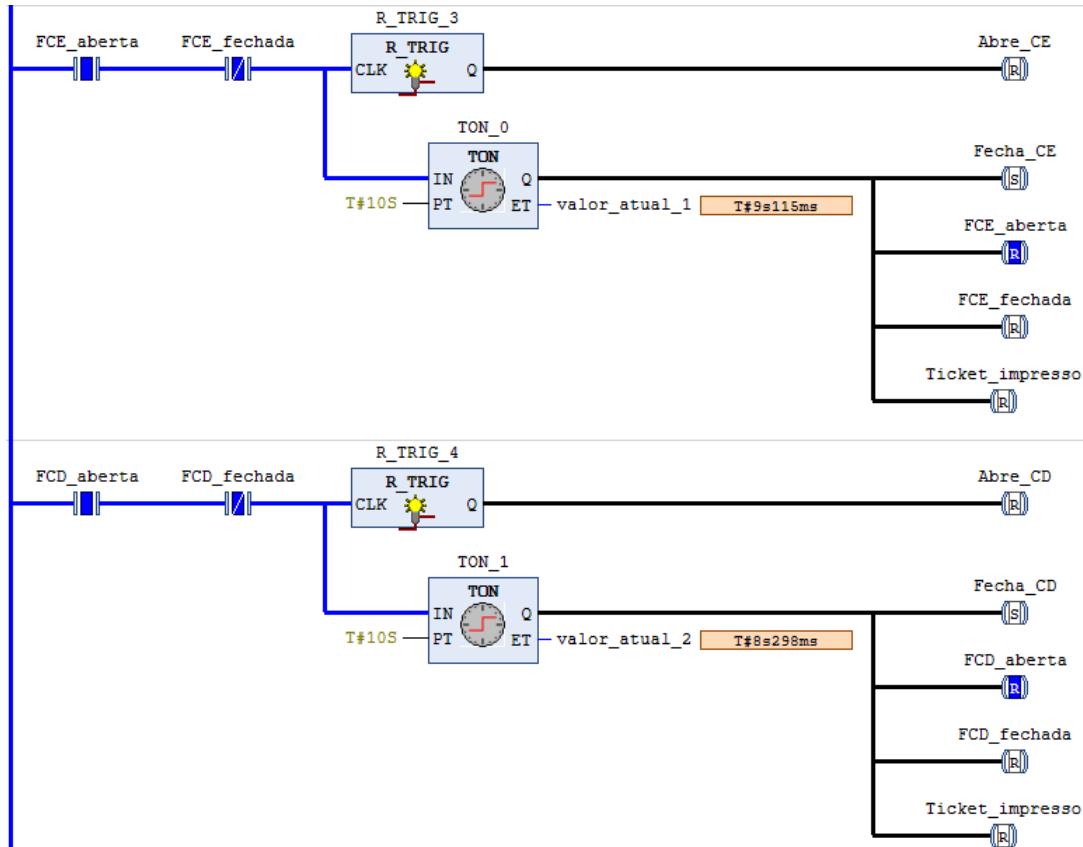
No caso de haver vaga disponível, as cancelas são abertas, permitindo a entrada do veículo. Quando o veículo acessa ao estacionamento, o sensor (SP_Estrada) registra sua entrada, incrementando o valor do contador. A Figura 44 ilustra o procedimento de fechamento das cancelas. Assim que a cancela atinge a abertura total (foram adicionados detectores de borda de subida para esta identificação), o motor é desligado e um temporizador com atraso de 10 segundos na ativação inicia o processo de fechamento das cancelas.

4.2.2 Simulações de falhas

Um sistema real funciona com diversos componentes eletrônicos e mecânicos e, portanto, está sujeito a falhas de diversos tipos.

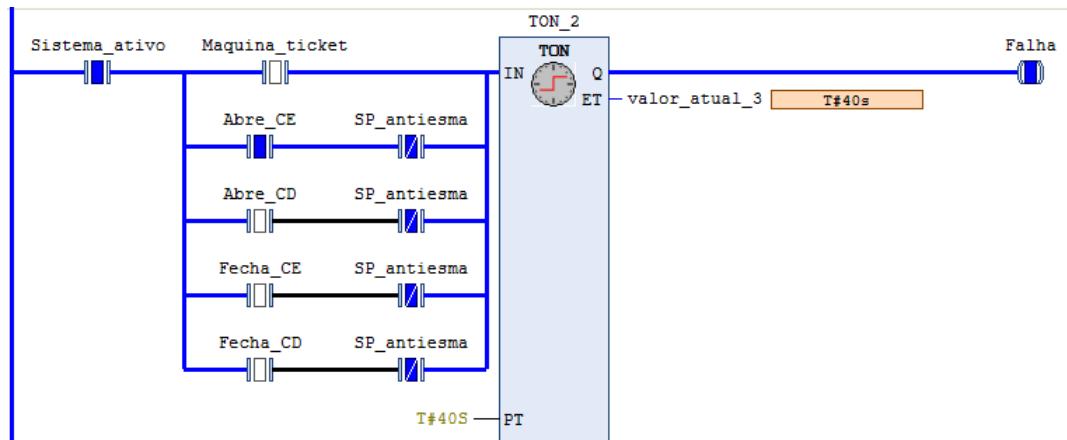
Uma das situações que está prevista é a falha com a demora na abertura ou fechamento das cancelas. Se durante 40 segundos alguma das cancelas não atingir o fim de curso respectivo, o sistema apresentará uma alerta de falha. Isto pode significar uma falha mecânica da cancela, um defeito no sensor de fim de curso ou algum outro problema. A Figura 45 exemplifica a simulação desta condição com a não abertura da cancela esquerda.

Figura 44 – Simulação do fechamento das cancelas.



Fonte: Elaborado pelo autor (2022).

Figura 45 – Simulação de falha quando a cancela não abre em 40 segundos

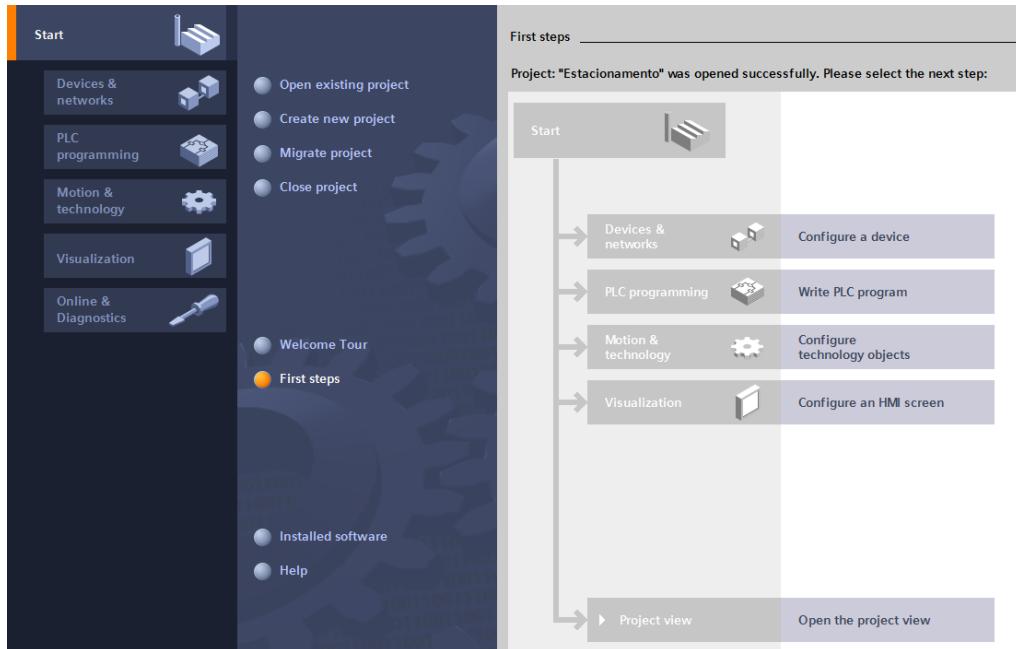


Fonte: Elaborado pelo autor (2022).

4.3 TESTES NO CLP SIEMENS S7-1200

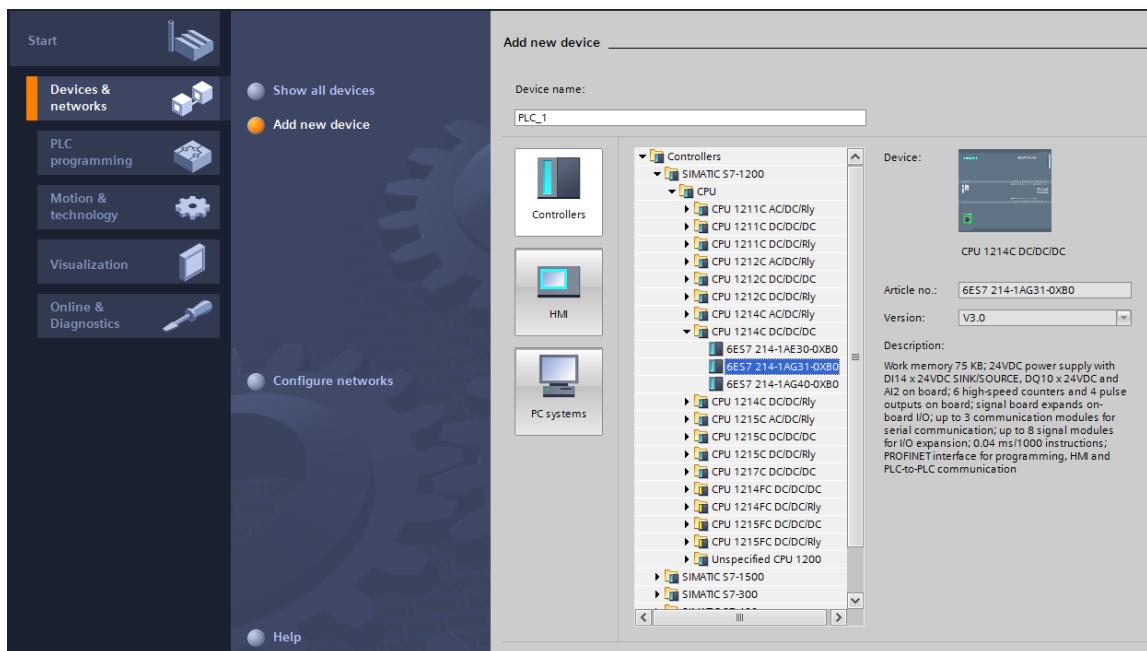
A programação desenvolvida no Codesys e descrita na seção 4.2 foi adaptada e testada num CLP Siemens modelo S7-1200. O programa foi implementado no TIA Portal (plataforma da Siemens para programação e configuração de sistemas) e carregado no CLP via Ethernet. As configuração iniciais realizadas são mostradas nas Figuras 46, 47 e 48.

Figura 46 – TIA Portal: Tela Inicial



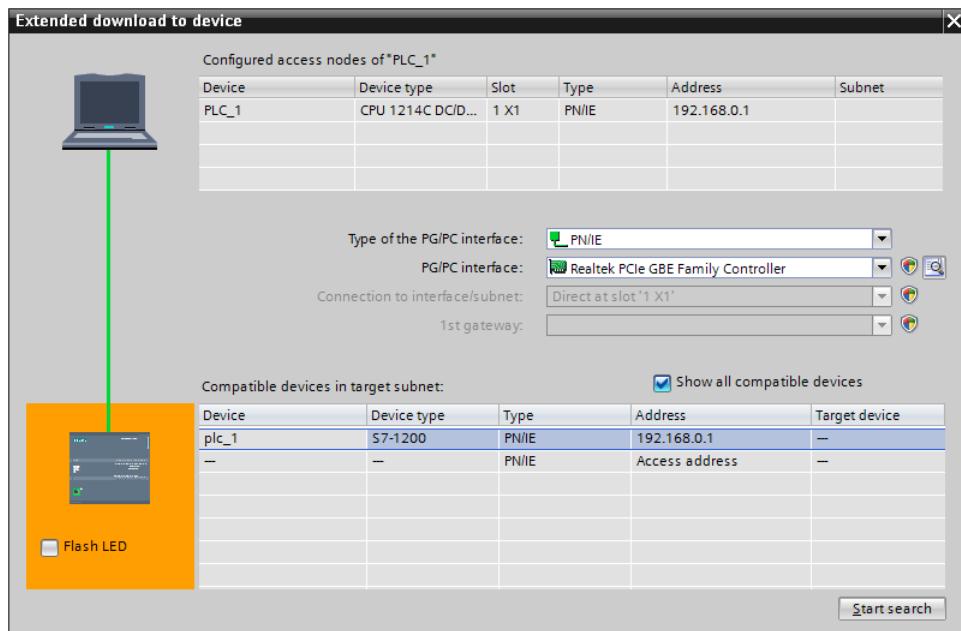
Fonte: Elaborado pelo autor (2022).

Figura 47 – TIA Portal: Seleção da CPU



Fonte: Elaborado pelo autor (2022).

Figura 48 – TIA Portal: Conexão Ethernet



Fonte: Elaborado pelo autor (2022).

Para a simulação dos estados foi acoplado o módulo SIM 1274 no CLP (Figura 49). Importante destacar que este módulo consegue controlar um número limitado de entradas comparado ao número de variáveis estabelecidas na programação desenvolvida (variáveis citadas anteriormente nas Tabelas 5 e 6). Uma solução encontrada foi alterar os estados via software e verificar as saídas diretamente no controlador. A Figura 50 mostra a listagem dos *tags* e seus respectivos endereços no Tia Portal. Diversos testes são mostrados a partir da Figura 51 até a Figura 59.

Figura 49 – Equipamentos utilizados nos testes: CLP Siemens S7-1200, Simulador SIM 1274, fonte de alimentação de 24Vcc, Multimetro, cabo Ethernet e jumpers conectores



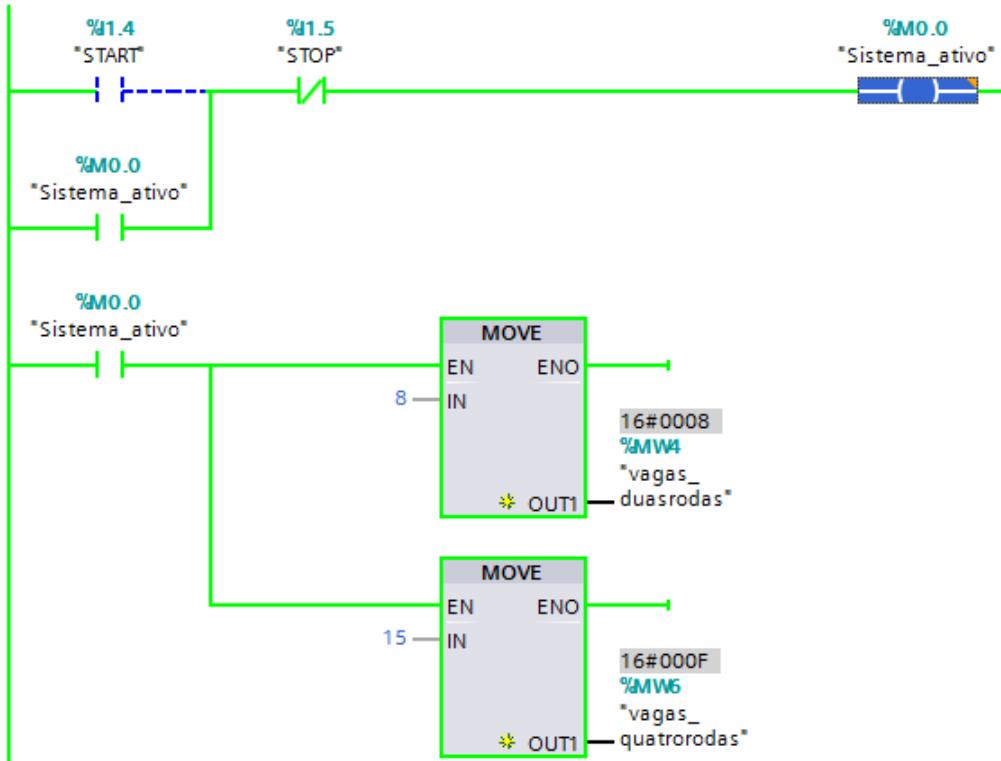
Fonte: Elaborado pelo autor (2022).

Figura 50 – Listagem dos *tags* e respectivos endereços no Tia Portal

Default tag table			
	Name	Data type	Address
1	DI PB	Bool	%I0.0
2	DI SA	Bool	%I0.1
3	DI SB	Bool	%I0.2
4	DI SP_Entrada	Bool	%I0.3
5	DI FCE_aberta	Bool	%I0.4
6	DI FCE_fechada	Bool	%I0.5
7	DI FCD_aberta	Bool	%I0.6
8	DI FCD_fechada	Bool	%I0.7
9	DI Verifica_impressao	Bool	%I1.0
10	DI SP_antiesma	Bool	%I1.1
11	DI SP_Saida_A	Bool	%I1.2
12	DI SP_Saida_B	Bool	%I1.3
13	DI START	Bool	%I1.4
14	DI STOP	Bool	%I1.5
15	DI Maquina_ticket	Bool	%Q0.0
16	DI Abre_CE	Bool	%Q0.1
17	DI Abre_CD	Bool	%Q0.2
18	DI Fecha_CE	Bool	%Q0.3
19	DI Fecha_CD	Bool	%Q0.4
20	DI Lotado	Bool	%Q0.5
21	DI Lotado_duasrodas	Bool	%Q0.6
22	DI Lotado_quatrorodas	Bool	%Q0.7
23	DI Falha	Bool	%Q1.0
24	DI Sistema_ativo	Bool	%M0.0
25	DI Ticket_impresso	Bool	%M0.1
26	DI Reset_vagas	Bool	%M0.2
27	DI vazio_1	Bool	%M0.3
28	DI vazio_2	Bool	%M0.4
29	DI R_TRIG_0	Bool	%M0.5
30	DI R_TRIG_1	Bool	%M0.6
31	DI R_TRIG_2	Bool	%M0.7
32	DI R_TRIG_3	Bool	%M1.0
33	DI R_TRIG_4	Bool	%M1.1
34	DI R_TRIG_5	Bool	%M1.2
35	DI R_TRIG_6	Bool	%M1.3
36	DI R_TRIG_7	Bool	%M1.4
37	DI R_TRIG_8	Bool	%M1.5
38	Word Vagas	Word	%MW2
39	Word vagas_duasrodas	Word	%MW4
40	Word vagas_quatrorodas	Word	%MW6
41	Word vagas_ocupadas	Word	%MW8
42	Word vagas_ocup_duasrodas	Word	%MW10
43	Word vagas_ocup_quatrorodas	Word	%MW12
44	Time valor_atual_1	Time	%MD8
45	Time valor_atual_2	Time	%MD12
46	Time valor_atual_3	Time	%MD16

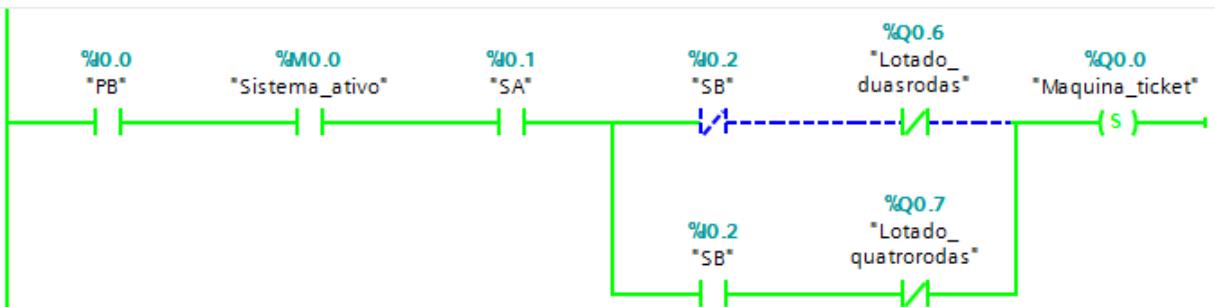
Fonte: Elaborado pelo autor (2022).

Figura 51 – Teste no CLP: Inicialização do Sistema (Botoeira START)



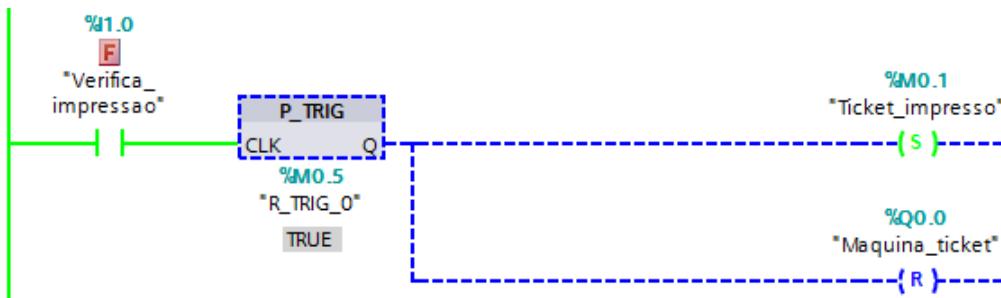
Fonte: Elaborado pelo autor (2022).

Figura 52 – Teste no CLP: Solicitação de ticket



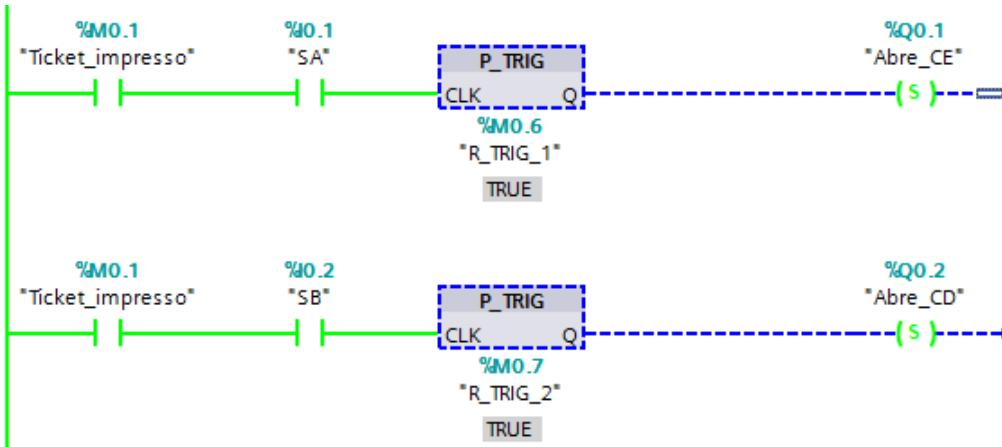
Fonte: Elaborado pelo autor (2022).

Figura 53 – Teste no CLP: Confirmação de impressão de ticket



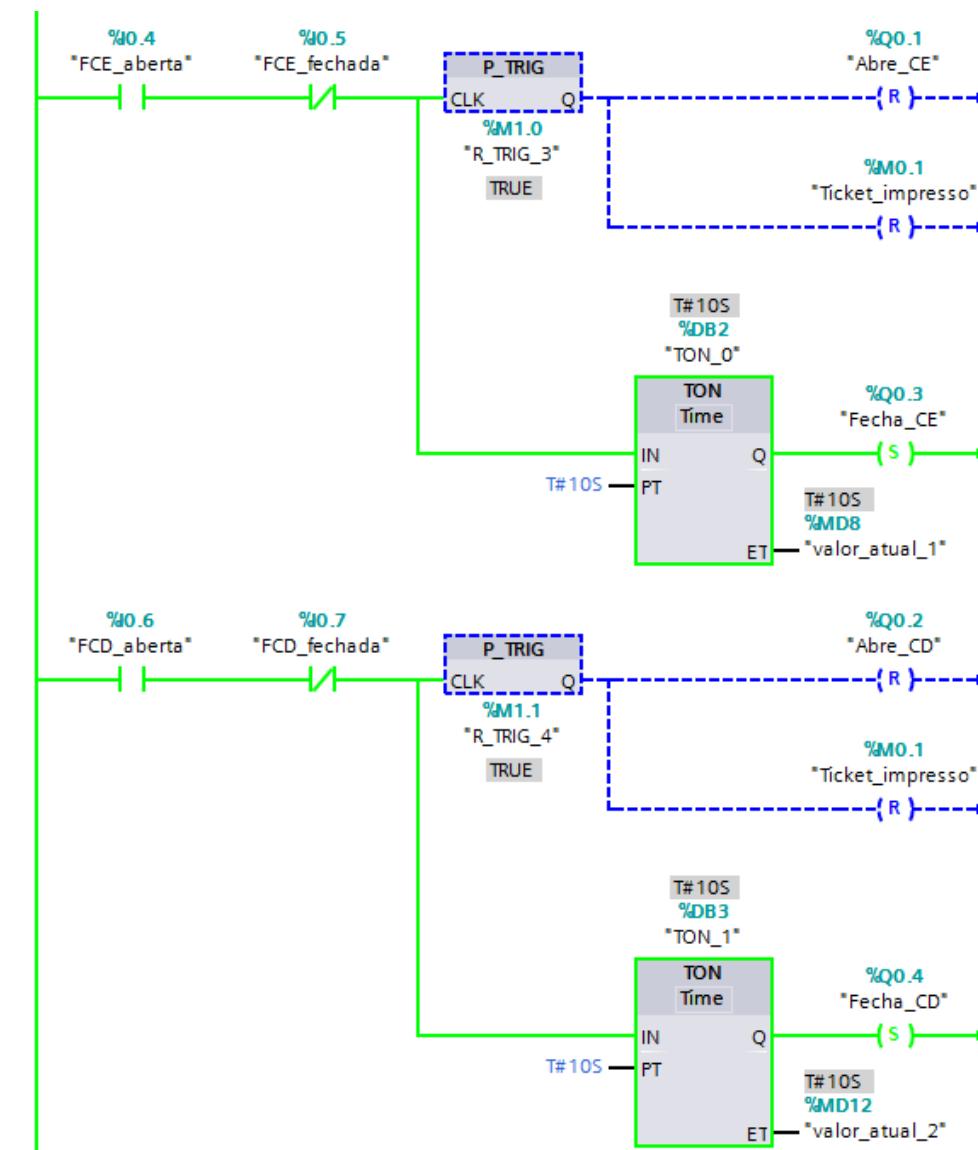
Fonte: Elaborado pelo autor (2022).

Figura 54 – Teste no CLP: Abertura das cancelas para entrada de veículo de quatro rodas



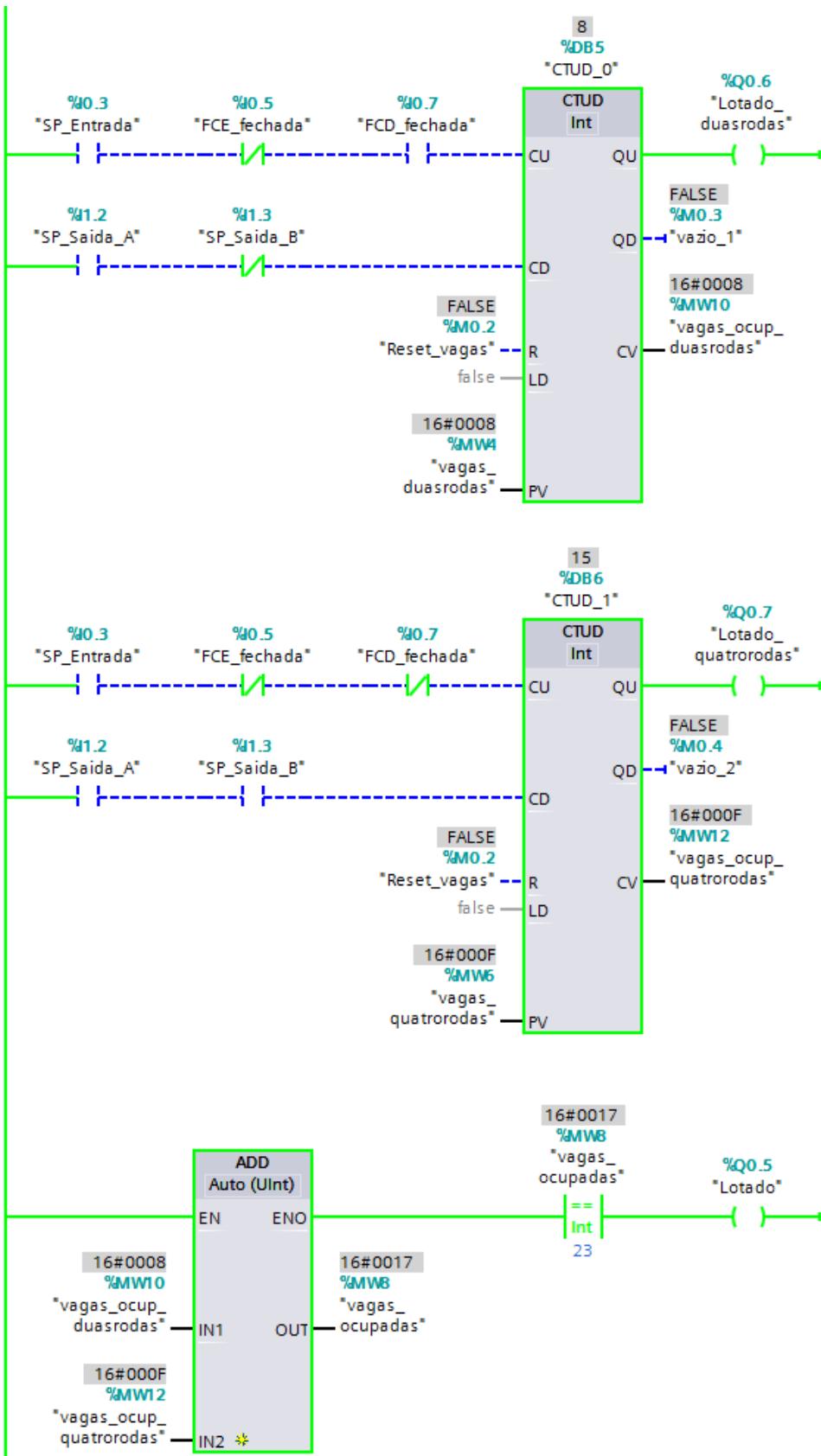
Fonte: Elaborado pelo autor (2022).

Figura 55 – Teste no CLP: Fechamento das cancelas



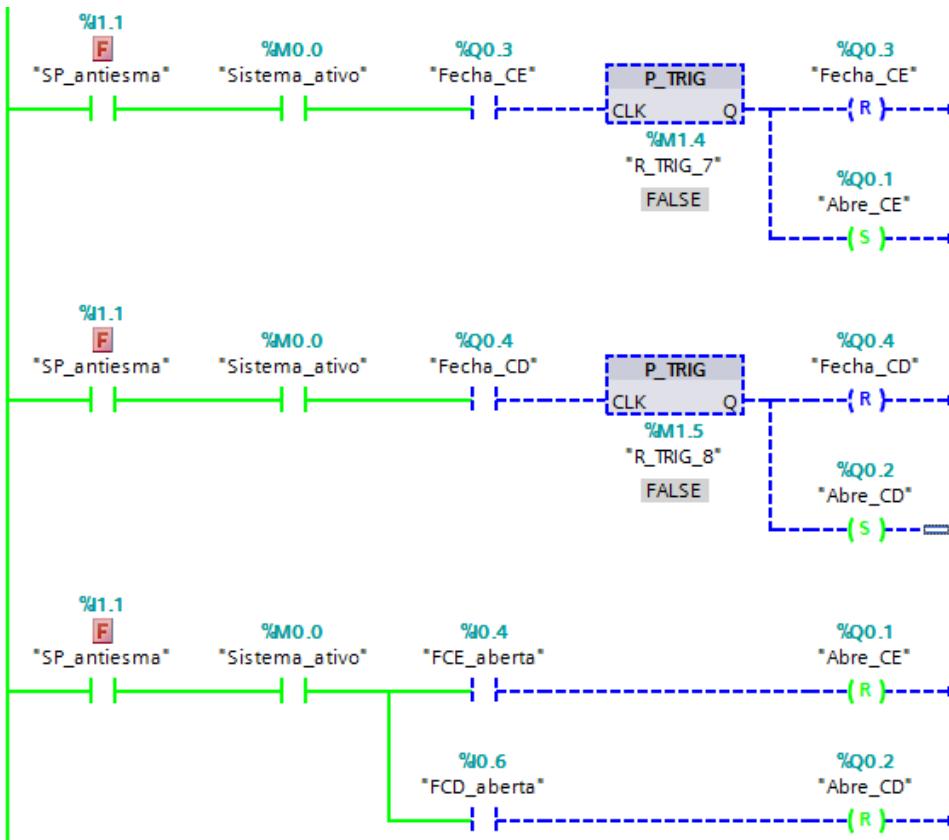
Fonte: Elaborado pelo autor (2022).

Figura 56 – Teste no CLP: Estacionamento lotado



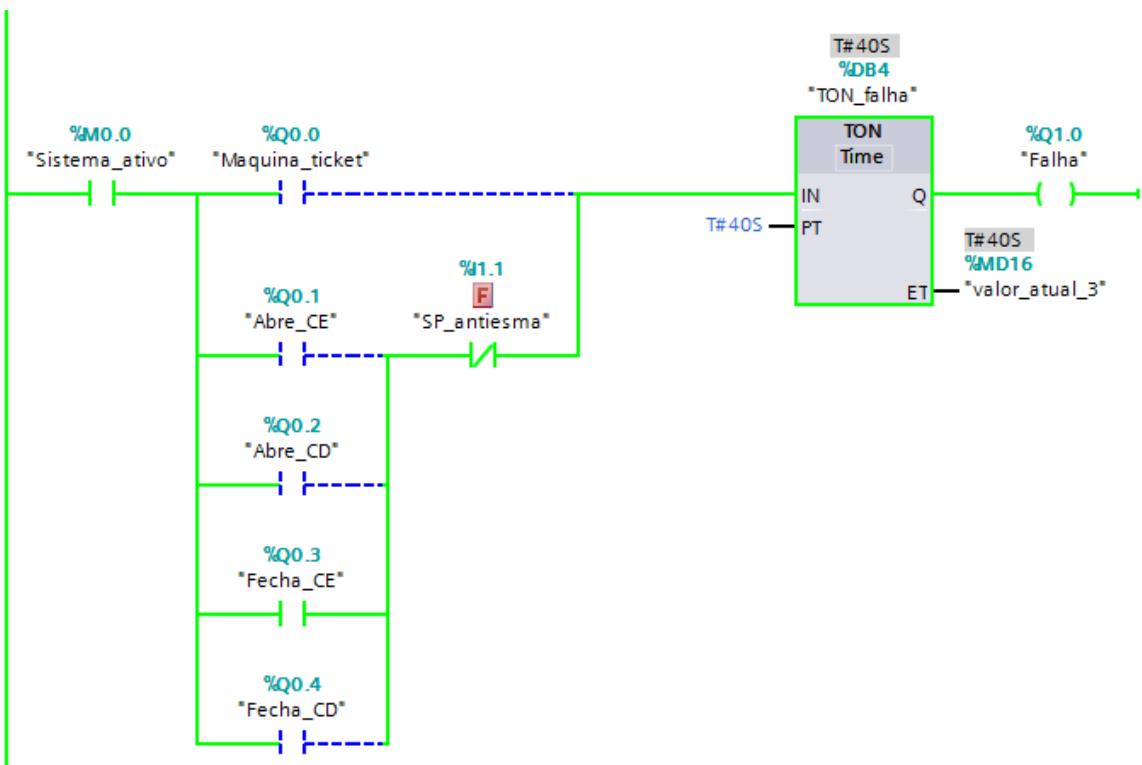
Fonte: Elaborado pelo autor (2022).

Figura 57 – Teste no CLP: Sensor anti-esmagamento atuado



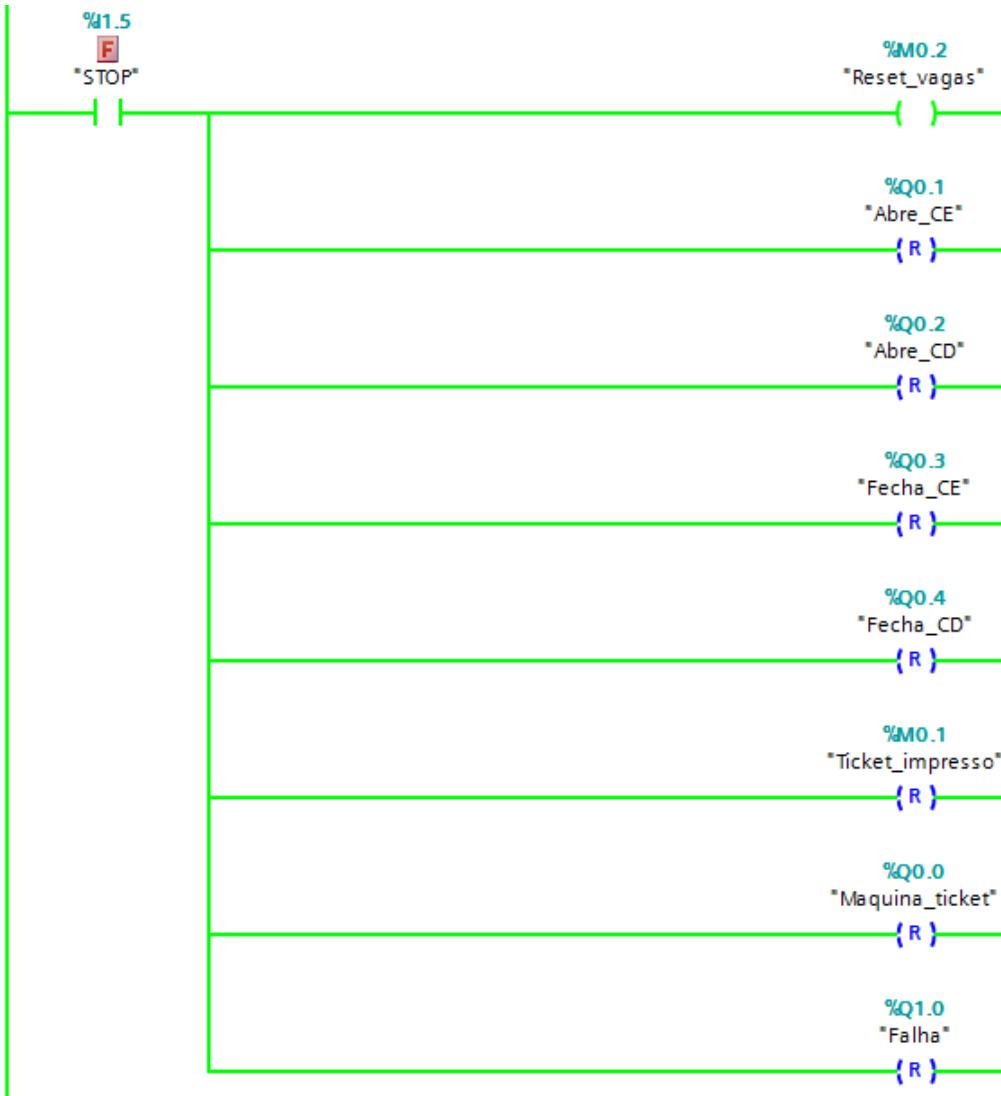
Fonte: Elaborado pelo autor (2022).

Figura 58 – Teste no CLP: Indicação de Falha



Fonte: Elaborado pelo autor (2022).

Figura 59 – Teste no CLP: Desligamento do sistema (Botoeira STOP)



Fonte: Elaborado pelo autor (2022).

4.4 ANÁLISE E DISCUSSÃO

Esta seção visa a fazer uma síntese dos resultados obtidos nas seções anteriores. Após simulações e testes realizados, o sistema mostra resultados satisfatórios e apresenta ser funcional no controle automático do acesso de veículos de duas ou quatro rodas para tarifação. Diversas situações foram simuladas e testadas e foi possível verificar as vantagens deste sistema comparado a controles manuais que dependem em grande parte de operadores e supervisores.

A programação desenvolvida permite a flexibilização para readequação a estacionamentos de diferentes portes. A instalação da parte física requer uma maior atenção e tempo, considerando a motorização das cancelas, instalação da máquina expedidora de ticket, dos sensores, sinalizadores e o cabeamento para o CLP. Uma vez que o sistema esteja instalado, novas funcionalidades podem ser incorporadas tendo como base a programação

desenvolvida. Da mesma forma, a adequação a novas tendências do mercado e tecnologias de sensores podem ser incorporadas.

5 CONCLUSÃO

O desenvolvimento deste trabalho foi motivado por dois problemas: a dificuldade de se quantificar as vagas disponíveis em estacionamentos e a definição da tarifação de acordo com os tipos de veículos que os acessam.

Tendo isto em conta, os resultados mostraram que os objetivos foram atingidos, desde a simulação, passando pela verificação de situações diversas, até os testes reais num CLP comercial.

A quantificação atualizada dos veículos foi realizada com o uso de dois contadores do tipo CTUD (incremental/decremental) e o sistema foi capaz de identificar o tipo de veículo para abertura da respectiva cancela e também para determinação da correta tarificação.

As simulações realizadas no software Codesys e os testes em bancada demonstram que o sistema é plenamente funcional e atende aos objetivos propostos, podendo ser aplicável em estacionamentos dos mais diversos tipos e dimensões, bem como se adaptar a diversas tecnologias de sensores digitais.

5.1 TRABALHOS FUTUROS

Dentre as melhorias que podem aprimorar e/ou otimizar a automação desenvolvida, podemos citar a inclusão de um painel que exibe o número de vagas disponíveis para uma fácil orientação dos usuários e que poderia também mostrar algumas informações adicionais, como, por exemplo, sistema em manutenção. Além disso, é possível desenvolver um sistema supervisório onde todas as informações estejam disponíveis, inclusive com animações de abertura e fechamento das cancelas. Uma das sugestões para aprimorar o funcionamento adequado do sistema está relacionada com a indicação ao usuário de que existem marcações onde deve posicionar seu veículo.

Entre os trabalhos futuros, é interessante mencionar a inclusão de sistemas de reconhecimento por visão computacional, reconhecimento de placas de veículos ou sistemas RFID, os quais poderiam atualizar e aprimorar o trabalho desenvolvido.

REFERÊNCIAS

- AHAMED, Nizam Uddin et al. Automatic car parking and controlling system using programmable logic controller (plc). *International journal of applied engineering research*, v. 10, p. 69–75, 2015.
- APHIRATSAKUN, Narong; DUY, Lam Thanh; KONGTHON, Jiradech. Au automatic car-parking modeling system. In: *2018 International Electrical Engineering Congress (iEECON)*. [S.l.: s.n.], 2018. p. 1–4.
- BASU, Aditya. *Whitepaper-Smart Parking.*: Happy minds publication. [S.l.], 2014. Disponível em: <<https://www.happiestminds.com/whitepapers/smart-parking.pdf>>.
- BOLTON, William. *Programmable Logic Controllers*. 6. ed. Waltham: Elsevier, 2015.
- COOKSON, Graham; PISHUE, Bob. *The Impact of Parking Pain in the US, UK and Germany*: Searching for parking costs drivers billions a year. [S.l.], 2017. Disponível em: <<https://www2.inrix.com/research-parking-2017>>.
- DONG, Bochen; WEI, Dong. Design of three-dimensional garage based on plc. In: *2019 Chinese Control And Decision Conference (CCDC)*. [S.l.: s.n.], 2019. p. 6390–6395.
- FRANCHI, Claiton Moro; CAMARGO, Valter Luís A. de. *Controladores lógicos programáveis: Sistemas discretos*. [S.l.: s.n.], 2009.
- INRIX. *INRIX ON-STREET PARKING*: Big data helps drivers park faster, smarter. [S.l.], 2021. Disponível em: <<https://inrix.com/wp-content/uploads/2021/05/INRIX-On-Street-Parking-Infographic-EU.pdf?x25380>>.
- JONES, Clarence T. *Programmable Logic Controllers: The Complete Guide to the Technology*. 1. ed. Atlanta: Patrick-Turner, 1996.
- LAUGHTON, D. F. Warne M. A. *Electrical Engeneer's Reference Book*. 16. ed. Oxford: Elsevier, 2003.
- OICA. *Production Statistics 2017. International Organization of Motor Vehicle Manufacturers (OICA)*. 2021. Disponível em: <<https://www.oica.net/category/production-statistics>>.
- PAREDE, Ismael Moura; GOMES, Luiz E. Lemes. *Eletrônica: automação industrial*. Coleção técnica interativa. série eletrônica, v. 6. [S.l.], 2011.
- SELVAKUMAR, K. et al. Control system of automatic garage using programmable logic controller. *TELKOMNIKA Telecommunication, Computing, Electronics and Control*, v. 18, p. 2654–2664, 2020.
- SENAI, Serviço Nacional de Aprendizagem Industrial. *Introdução à eletricidade*: Projeto bipbop brasil. Brasília, 2009. S491i.
- SHENG-WEI, Dai et al. The plc-based control system for intelligent garage. In: *2013 Third International Conference on Intelligent System Design and Engineering Applications*. [S.l.: s.n.], 2013. p. 1483–1486.

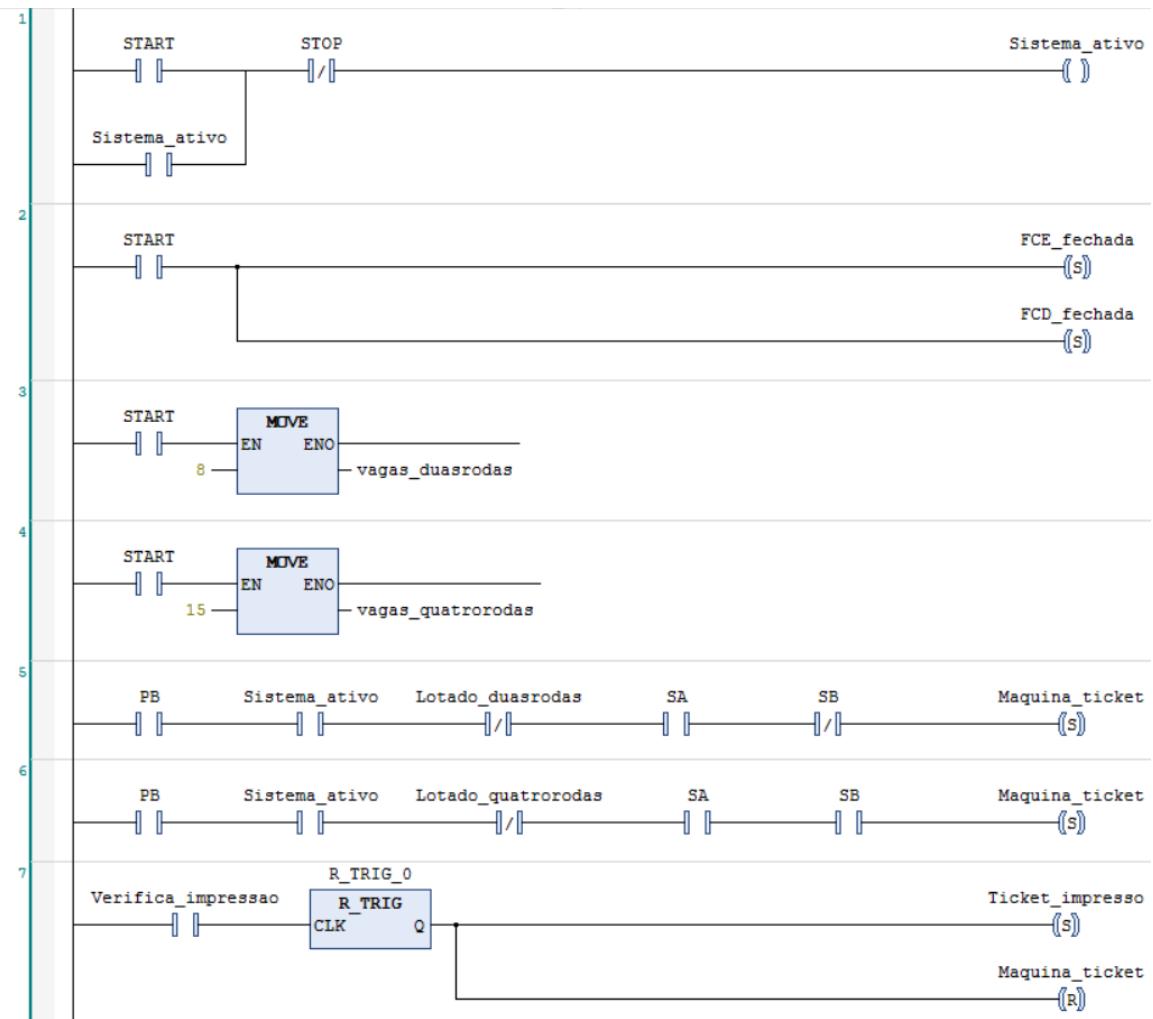
WARDSINTELLIGENCE. *World Vehicle Population Rose 4.6% in 2016.* 2017. Disponível em: <<https://wardsintelligence.informa.com/WI058630/World-Vehicle-Population-Rose-46-in-2016>>.

ZHANG, Zhuoran; MAO, Zhe; WAN, Hailing. The 6 layers stereo garage design based on omron plc. In: *Proceedings of 2011 International Conference on Electronic Mechanical Engineering and Information Technology*. [S.l.: s.n.], 2011. v. 7, p. 2682–2685.

APÊNDICES

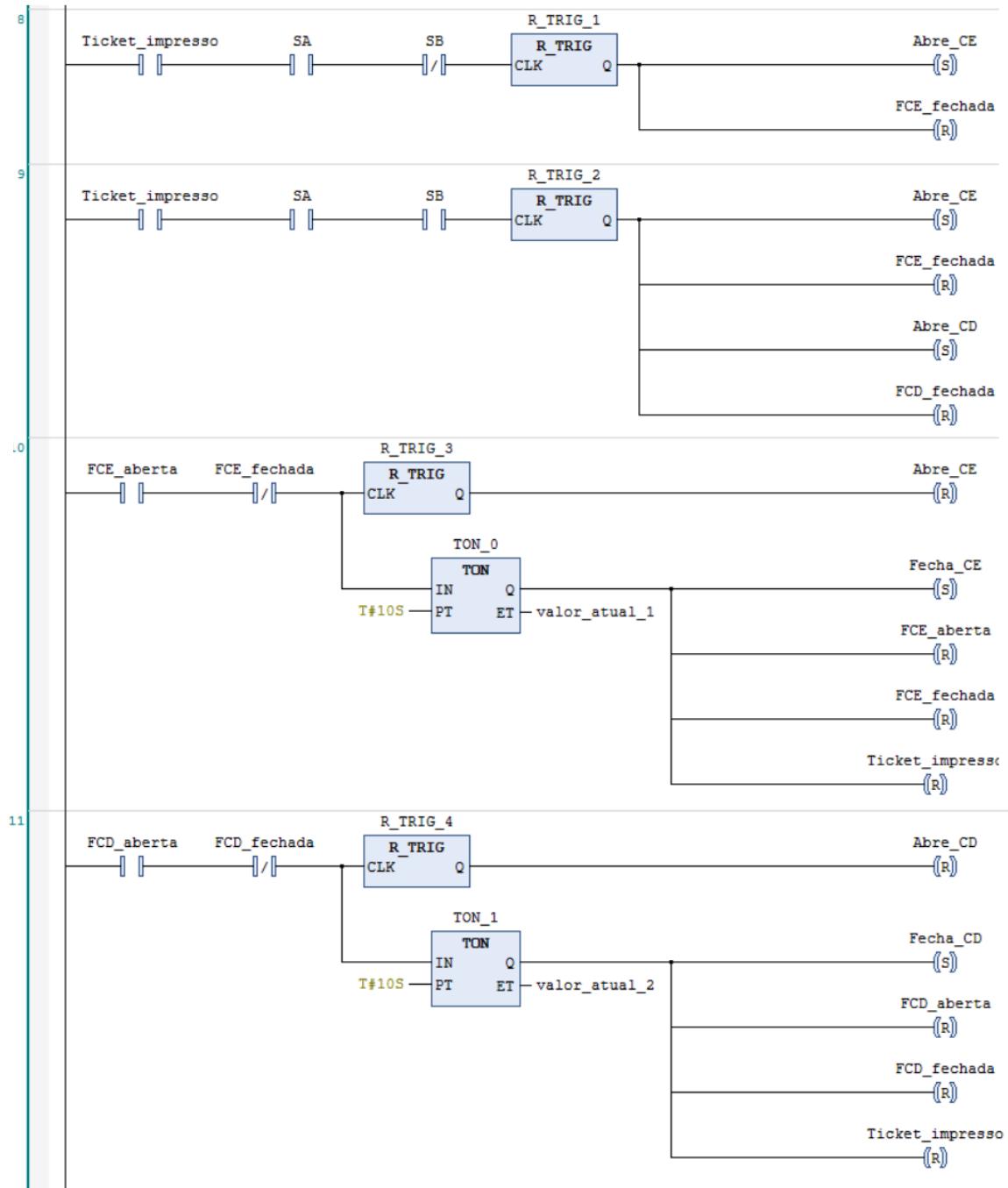
APÊNDICE A – PROGRAMAÇÃO COMPLETA DA SIMULAÇÃO EM CODESYS

Figura A.0.1 – Programação em *Ladder* referente à simulação no Codesys



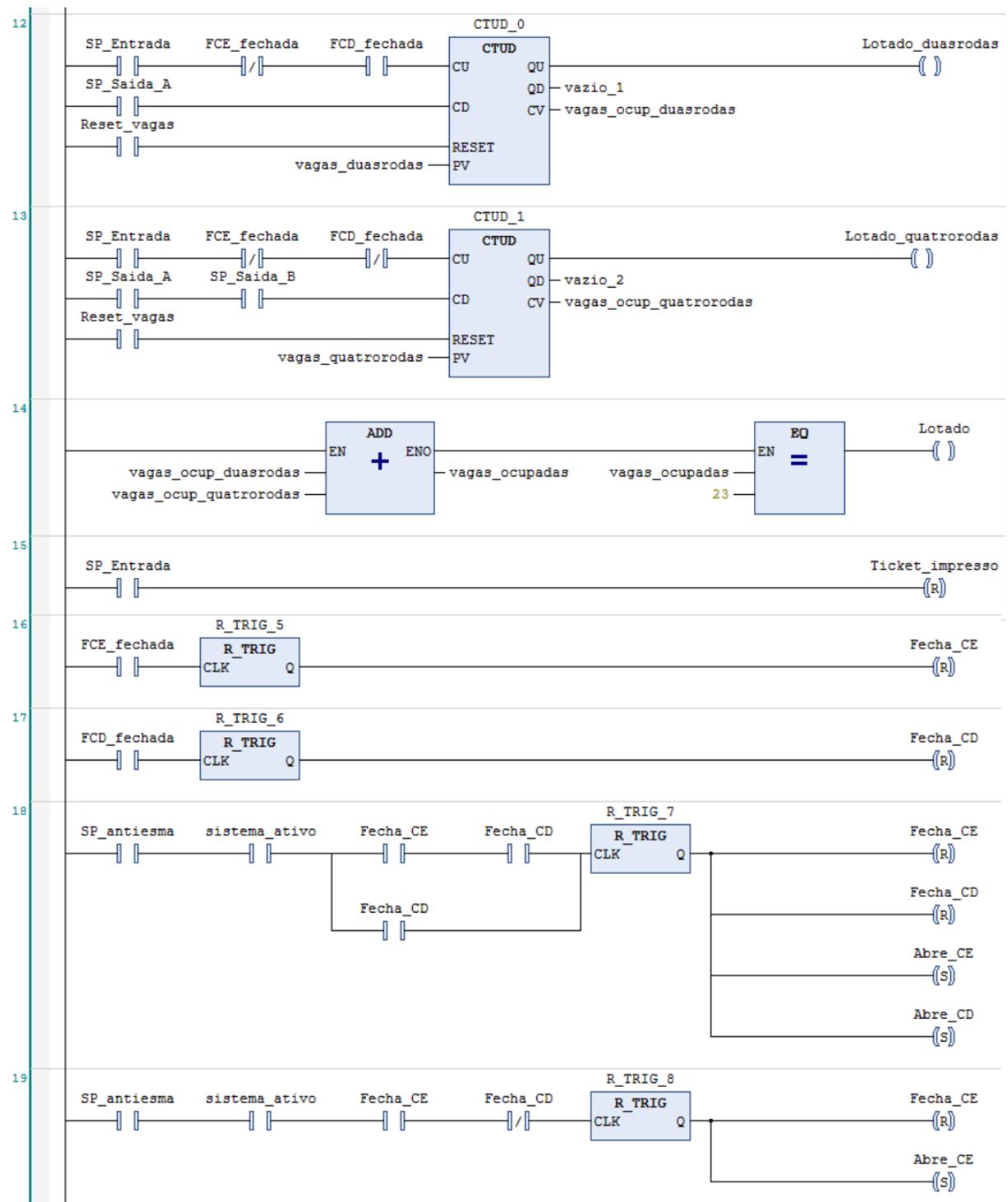
Fonte: Elaborado pelo autor (2022).

Figura A.0.2 – Continuação da programação no Codesys



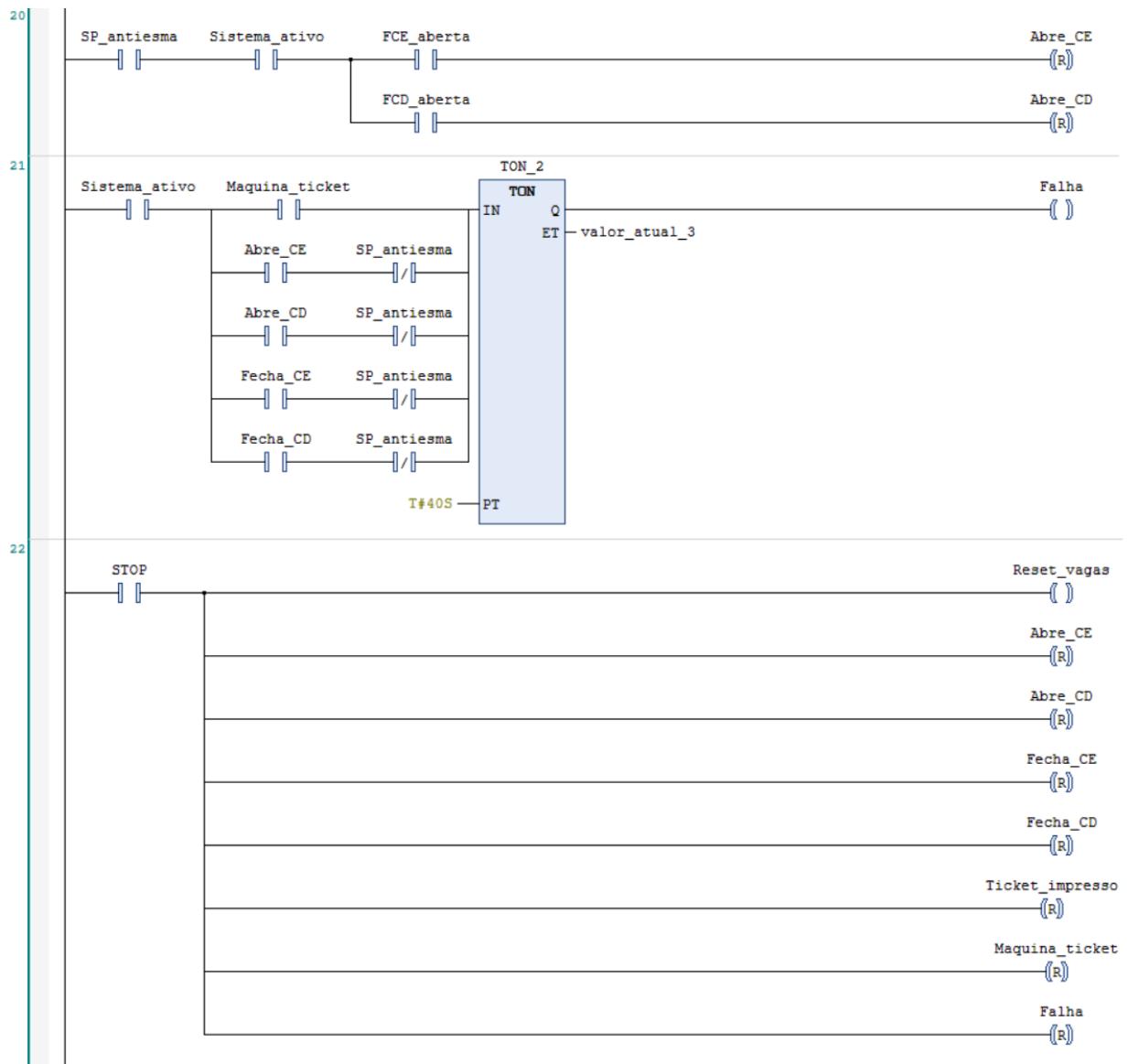
Fonte: Elaborado pelo autor (2022).

Figura A.0.3 – Continuação da programação no Codesys



Fonte: Elaborado pelo autor (2022).

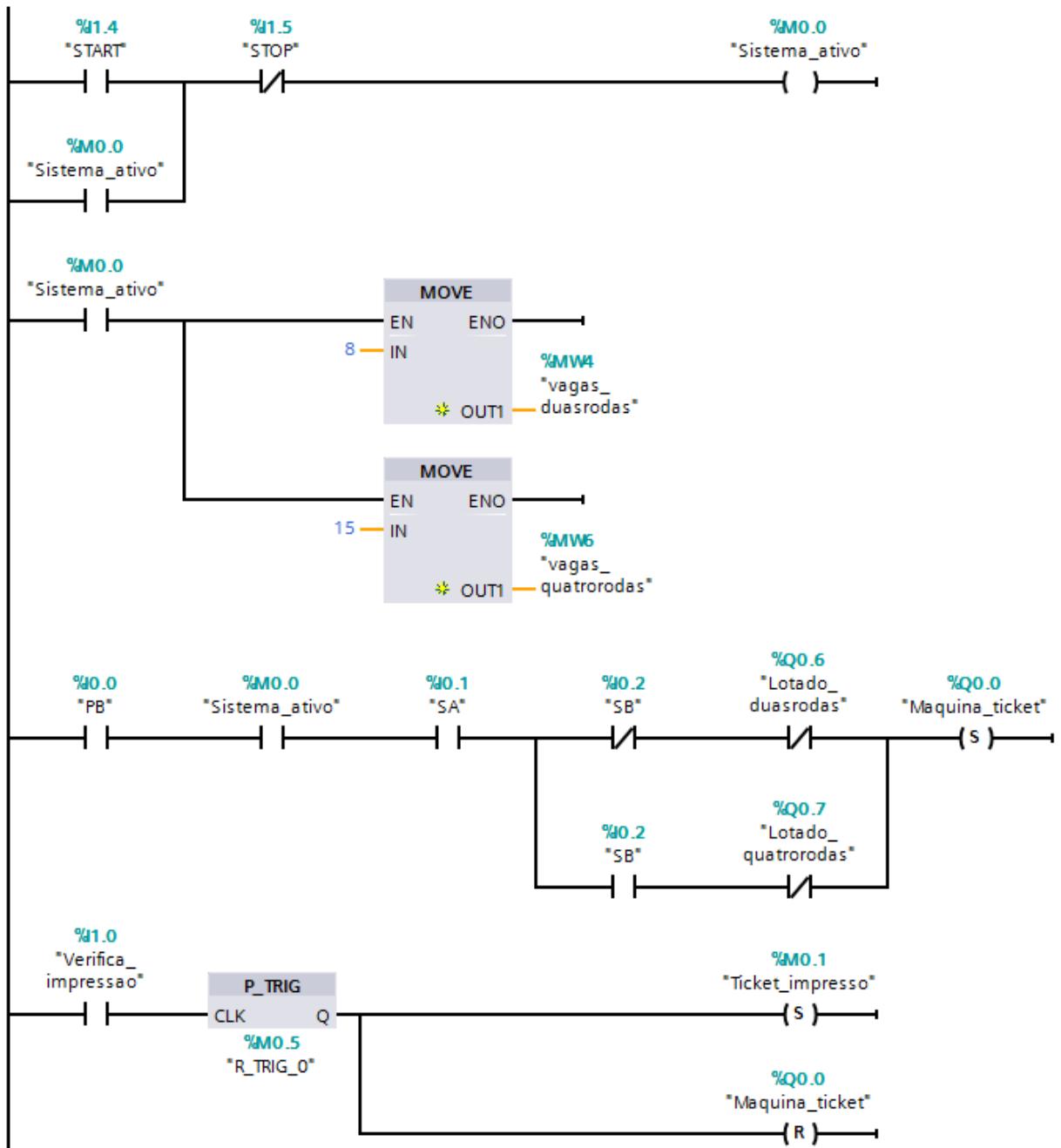
Figura A.0.4 – Finalização da programação no Codesys



Fonte: Elaborado pelo autor (2022).

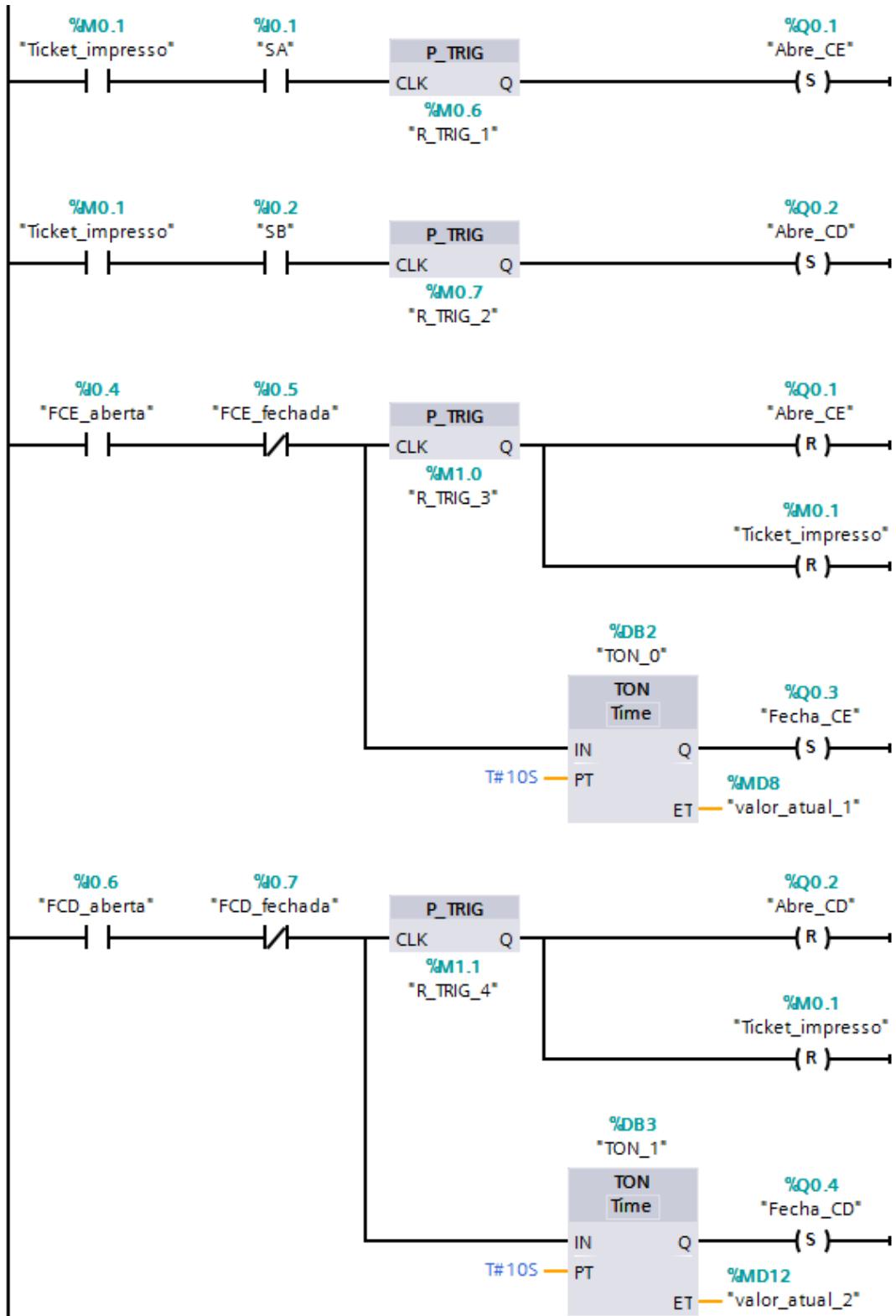
APÊNDICE B – PROGRAMAÇÃO COMPLETA DOS TESTES NO CLP SIEMENS S7-1200

Figura B.0.1 – Programação em *Ladder* referente aos testes no CLP



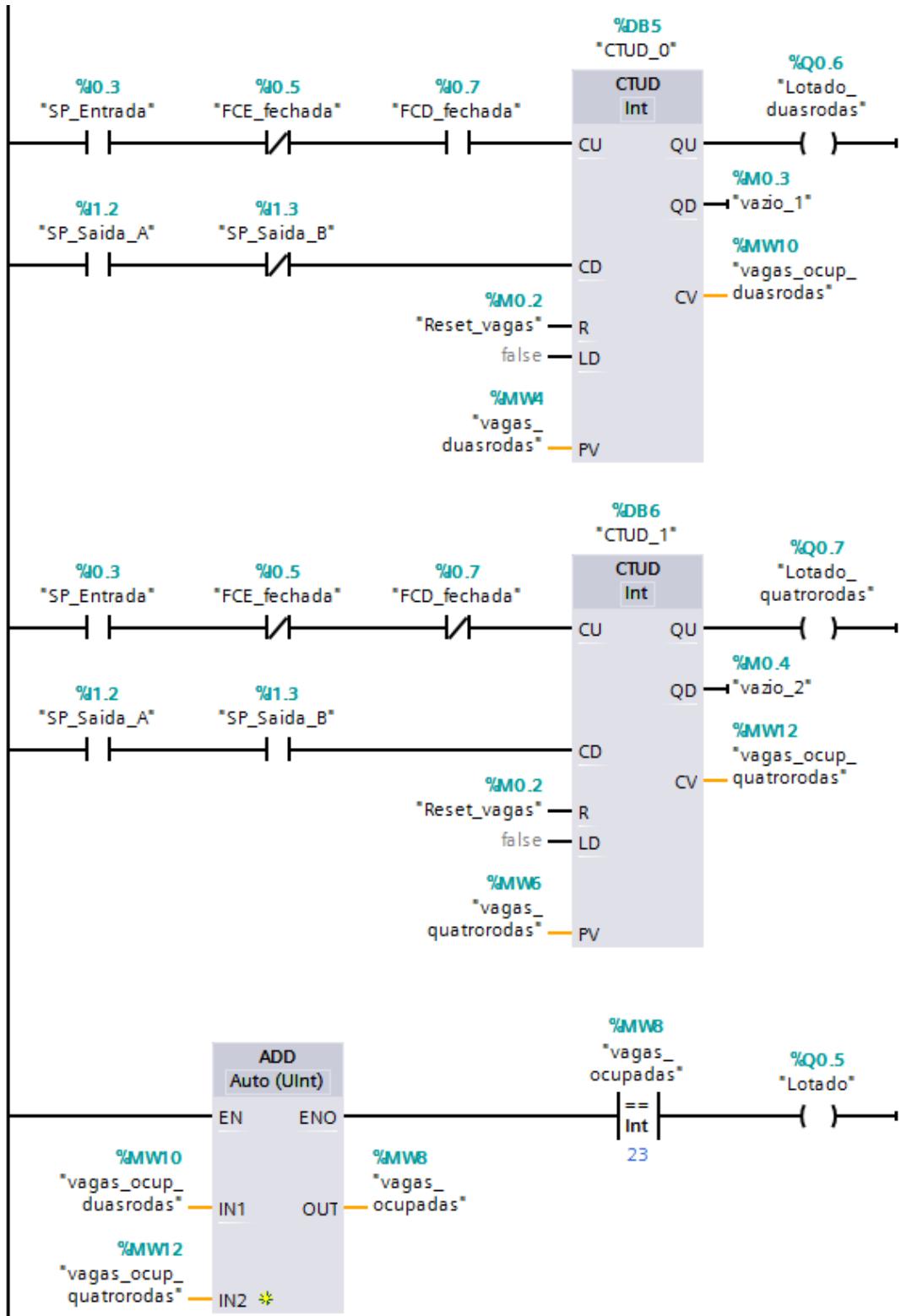
Fonte: Elaborado pelo autor (2022).

Figura B.0.2 – Continuação da programação no CLP



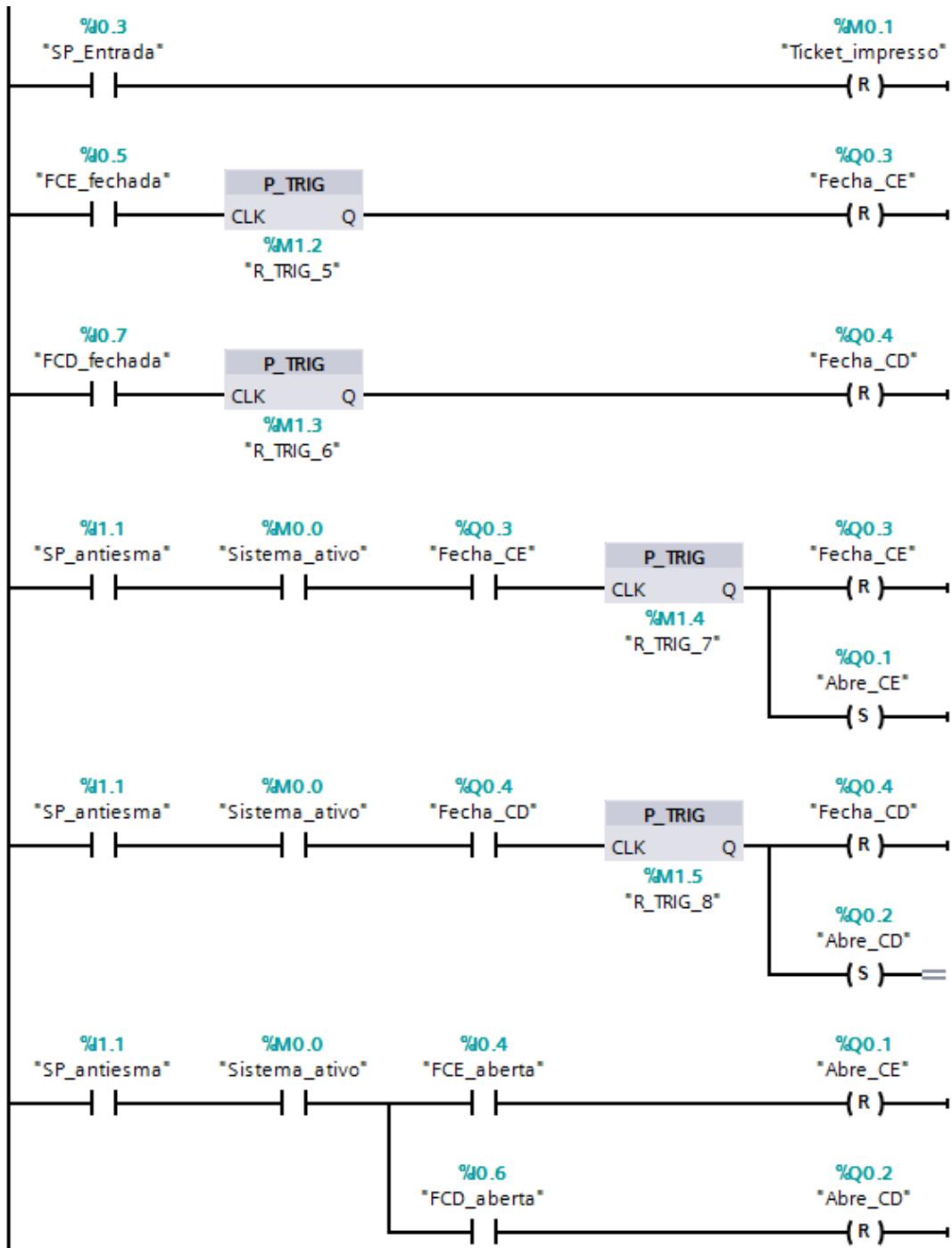
Fonte: Elaborado pelo autor (2022).

Figura B.0.3 – Continuação da programação no CLP



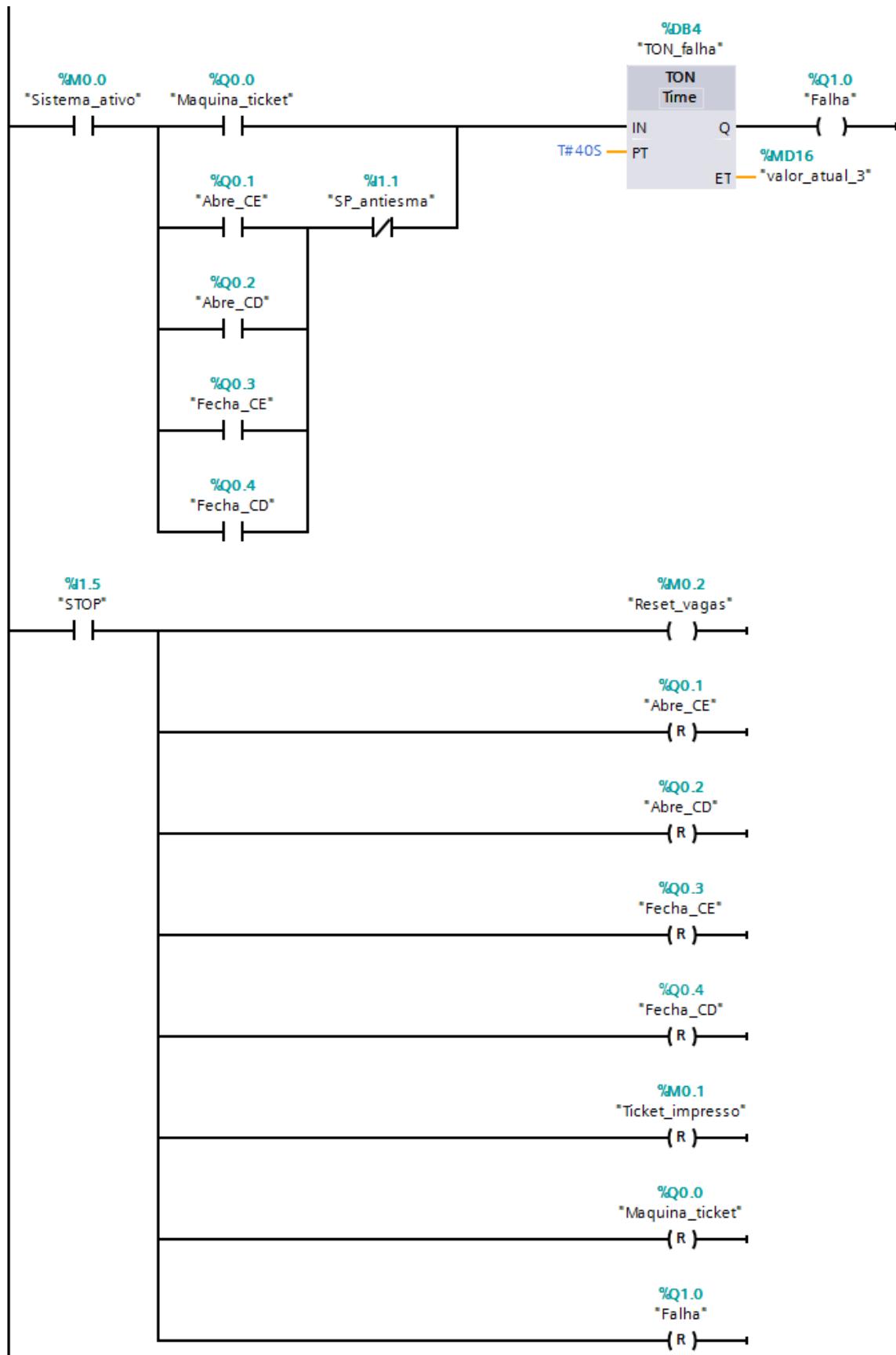
Fonte: Elaborado pelo autor (2022).

Figura B.0.4 – Continuação da programação no CLP



Fonte: Elaborado pelo autor (2022).

Figura B.0.5 – Finalização da programação no CLP



Fonte: Elaborado pelo autor (2022).